

**DESARROLLO DE APLICACION PARA EL DISEÑO ESTOCÁSTICO DE EJES
DE TRANSMISION DE POTENCIA**

AUTORES

**ANA MARGARITA BUELVAS HERNÁNDEZ
ARTURO FABIO OLIER ARROYO
JORGE ANDRÉS CARAZO TORDECILLA**

**UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR
FACULTAD DE INGENIERÍA
DIRECCIÓN DE PROGRAMA DE INGENIERÍA MECÁNICA
CARTAGENA DE INDIAS
2016**

**DESARROLLO DE APLICACION PARA EL DISEÑO ESTOCÁSTICO DE EJES
DE TRANSMISION DE POTENCIA**

**ANA MARGARITA BUELVAS HERNÁNDEZ
ARTURO FABIO OLIER ARROYO
JORGE ANDRÉS CARAZO TORDECILLA**

Proyecto de grado

**Tutor:
Edgardo William Arrieta Ortiz
MSc. en Ingeniería Mecánica**

**UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR
FACULTAD DE INGENIERÍA
DIRECCIÓN DE PROGRAMA DE INGENIERÍA MECÁNICA
CARTAGENA DE INDIAS
2016**

Nota de Aceptación

Presidente del Jurado

Jurado

Jurado

Cartagena, 2016

Dedicamos este trabajo de grado a Dios el cual merece todo el mérito, a nuestros padres quienes siempre nos apoyaron y alentaron en el día a día, y finalmente a nuestro profesor Edgardo William Arrieta Ortiz quien encaminó nuestro proyecto.

AGRADECIMIENTOS

Primero que todo agradecimiento a Dios, porque gracias a él fue posible que pudiéramos realizar este proyecto de grado, muchas gracias a nuestros familiares, ya que nos brindaron apoyo y fueron colaboradores al momento de efectuar la tesis y finalmente a nuestro tutor Edgardo Arrieta por su labor ilustrándonos a lo largo del desarrollo de la tesis, pues nos proporcionó indicación, consejos y datos, que fueron de gran importancia para desarrollar exitosamente la tesis.

TABLA DE CONTENIDO

1	INTRODUCCIÓN.....	10
2	JUSTIFICACIÓN.....	11
3	OBJETIVOS.....	12
3.1	Objetivo general.....	12
3.2	Objetivos específicos.....	12
4	PROPUESTA INNOVADORA.....	13
5	MARCO CONCEPTUAL.....	14
5.1	Resistencia del material como variable aleatoria.....	14
5.2	Relación entre la probabilidad de falla y el factor de seguridad para diseño.....	15
5.3	Diseño de ejes sometidos a fatiga.....	17
5.4	Diseño estocástico de ejes sometidos a fatiga.....	18
5.4.1	Vida para fatiga estocástica.....	18
5.4.2	Factores de Marin estocásticos K_a, K_b, K_c, K_d	18
5.4.3	Factor de Concentracion K_f, K_t y sensibilidad a la muesca.....	18
5.4.4	Tipos de cargas y condiciones diseño para los ejes a diseñar.....	19
6	METODOLOGÍA.....	21
6.1	Elaboración de la aplicación.....	21
6.1.1	Introducción a la aplicación.....	21
6.1.2	Diagrama de Flujo.....	24
6.1.3	Selección del material.....	25
6.1.4	Cálculo del límite de resistencia a la fatiga.....	26
6.1.5	Factor de acabado superficial.....	26
6.1.6	Factor de temperatura.....	27
6.1.7	Factor de concentración de esfuerzo.....	27
6.1.8	Factor de tamaño.....	29
6.1.9	Factor de modificación por la carga.....	29
6.1.10	Cálculo de la resistencia a la fatiga.....	30
6.1.11	Análisis de fuerzas en engranes helicoidales.....	30
6.1.12	Análisis y ubicación de los momentos resultantes.....	33

6.1.13	Cálculo de esfuerzos	34
6.1.14	Esfuerzos medios	35
6.1.15	Esfuerzos alternantes	35
6.1.16	Ajustes por interferencia	36
6.1.17	Cálculo del esfuerzo debido a la presión por interferencia	37
6.1.18	Cálculo del tensor de esfuerzos.....	38
6.1.19	Selección de los criterios de falla por fatiga.....	40
6.1.20	Cálculo de la distribución de probabilidad.....	42
6.1.21	Manual para la aplicación	43
7	VERIFICACIÓN DE LA APLICACIÓN	44
7.1	Probabilidad de falla para fatiga en vida finita del eje.....	44
7.1.1	Potencia absorbida por el freno	44
7.1.2	Calculo resorte	45
7.1.3	Momento máximo en el eje	46
7.1.4	Medición de resistencia ultima del material	46
7.1.5	Calculo de esfuerzos.....	48
7.1.6	Calculo de la resistencia a la fatiga.....	48
7.1.7	Probabilidad de falla.....	49
7.2	Construcción del prototipo	49
8	RESULTADOS DE LOS ENSAYOS	50
9	CONCLUSIONES	56
10	BIBLIOGRAFÍA	57
11	ANEXOS.....	58

LISTA DE TABLAS

	Pág
Tabla 1. Dureza del material	47
Tabla 2. Resultados de ensayos de ejes	55

LISTA DE GRÁFICAS

	Pág
Grafica 1. Distribución normal	15
Grafica 2. Relación factor de seguridad - probabilidad de falla	16
Gráfica 3. Esfuerzo de flexión en eje	17
Grafica 4. Criterio de falla para fatiga	20
Grafica 5. Diagrama de momento y fuerza	46
Gráfica 6. Numero de ciclos	48

LISTA DE FIGURAS

	Pág
Figura 1. Eje de transmisión de potencia	19
Figura 2. Fuerzas en engranes helicoidales	32
Figura 3. Diagrama de cuerpo libre	44
Figura 4. Diagrama de sistema de frenado	45

LISTA DE ANEXOS

	Pág
Anexo A. Código en Scilab ensayo	58
Anexo B. Manual para el usuario	263
Anexo C. Ejercicio resuelto con aplicación	282

RESUMEN

El diseño de ejes que están sometidos a distintos tipos de cargas, requiere de un análisis minucioso, en el que hay que tener en cuenta las diferentes variables a las que va a estar sometido, lo que provoca que el cálculo se vuelva en ocasiones extenso. Es por esto que existen algunas aplicaciones computacionales que generan diseños inmediatos de ejes, facilitando el cálculo. Aplicaciones como DICAVE, entre otras. Todas las existentes trabajan con factor de seguridad el cual corresponde a diseño determinístico.

Dentro de este trabajo, se explica el desarrollo de una aplicación que facilite el diseño de ejes de transmisión de potencia, utilizando como base el análisis estocástico, lo que permite lograr más alta confiabilidad y credibilidad del diseño. Se evaluaron los análisis y resultados de la aplicación por medio de ensayos experimentales sobre varios ejes.

Con esto se verifica la habilidad de la aplicación para producir diseños con consideraciones estocásticas y se entrega de este modo una aplicación que sirve a fines didácticos e investigativos en diseño de ejes.

PALABRAS CLAVE: Diseño estocástico, fatiga, esfuerzos, momentos resultantes, fuerzas, presión por interferencia, probabilidad, tensor de esfuerzos, teoría de falla, criterio de falla.

1 INTRODUCCIÓN

Los ejes son esenciales para la mayoría de los artefactos mecánicos que requieran transmisiones de potencia. Por esto cotidianamente en las industrias y repetitivamente las instituciones donde se estudia el diseño mecánico, se requiere el cálculo preciso de estos. Dentro de este cálculo existe una gran cantidad de variables que van desde los ajustes por interferencia hasta el diámetro de cada sección del eje, lo que requiere previamente de iteraciones que normalmente son apoyadas por programas computacionales que a su vez deben ser programados previamente en cada problema en particular.

Sería un aporte a la eficiencia, generar un software que pueda resumir todos estos pasos, y que una vez tenidos en cuenta los requerimientos del diseño del eje, pueda ofrecer un prototipo inmediato escogiendo sus mejores características para la aplicación correspondiente. Han existido anteriores aplicaciones tales como DICAVE¹ cuyo modelo de diseño es determinístico.

Nuestro proyecto consiste en resumir la metodología de Diseño Estocástico de ejes, estudiada en los cursos de Introducción al diseño de la universidad tecnológica y plasmarla en una aplicación (inicialmente escrita como un código en Scilab ([1]. Scilab 5.5.1)) La aplicación se desarrolla mediante variables aleatorias las cuales dan como resultado la confiabilidad del tiempo de vida del eje.

¹ <http://e-archivo.uc3m.es/handle/10016/13132>

2 JUSTIFICACIÓN

Diseñar ejes de transmisión de potencia confiables es uno de los trabajos más estrictos y rigurosos, requeridos en Ing. Mecánica, debido a que el desempeño, desarrollo y la producción de las diferentes maquinarias existentes en los respectivos campos laborales se encuentra directamente vinculada con el buen funcionamiento de estos. Nuestro proyecto de grado contiene la creación de un software programado que permite al diseñador reducir el tiempo de trabajo empleado en la creación de un eje de transmisión de potencia, el cual puede tener de 1 a 5 engranes, es decir, 13 secciones de diferentes diámetros, además, debido al tipo de análisis realizado (Estocástico) la medida de confiabilidad del diseño no será un factor de seguridad sino a escala porcentual de 1 a 100% dada a partir del cumplimiento de los diferentes criterios de falla como lo son, el criterio de la ASME-elíptica, Goodman y Gerber. También es posible conocer los ciclos de vidas del eje debido a las cargas comunes o sobrecargas.

Son varios los factores por los cuales se realizó esta investigación:

Desde el punto de vista estudiantil apporto a nuestro conocimiento y manejo de ejes. Al llevar cada caso en particular a un software, exigió el conocimiento de cada caso al autor. Por otra parte expandió nuestro manejo de lenguajes de programación.

En el campo laboral, sería una herramienta para cualquier ingeniero que este cotidianamente en su trabajo en responsabilidad de máquinas que cuenten con ejes de este tipo. En el momento que se requiera el diseño de un eje para un maquina o la repotenciación de un eje para la misma.

El tiempo empleado en este proceso sería menor al normalmente empleado, generando eficiencia en la exactitud del software y además recortando el valioso tiempo de una empresa.

Los datos acerca del eje que el software arrojaría, sería de gran importancia para la planificación de la empresa. Datos como lo son esfuerzo máximo soportado y numero de ciclos máximo de la pieza según el esfuerzo aplicado.

3 OBJETIVOS

3.1 Objetivo general

Desarrollar una herramienta computacional para diseñar ejes de transmisión de potencia, para fatiga mediante análisis estocástico, en el cual la eficacia del diseño no estará dada por el factor de seguridad, sino por un porcentaje de probabilidad de falla o confiabilidad.

3.2 Objetivos específicos

- Desarrollar un programa que permita el diseño de un eje, dimensionando cada una de sus secciones, el cual sea flexible² al establecer los requisitos de tipo de cargas y potencia transmitida. El programa debe guiar al usuario para aclarar y depurar la entrada de datos al programa.
- Calcular los diferentes factores de Marín en su versión estocástica, sección por sección, teniendo en cuenta las condiciones del eje (factores de concentración, potencial fluctuante, número de engranes, etc.) siguiendo la metodología plasmada por el autor Joseph Shigley ([2]. Shigley).
- Calcular el tiempo de vida dependiendo de las condiciones de trabajo a las cuales será sometido el eje así mismo la sobrecarga máxima que este puede soportar para los tipos de condiciones de carga considerados.
- Verificar en un caso simple los resultados del diseño estocástico obtenido mediante ensayos de carga de un eje en condiciones de laboratorio.
- Escribir un artículo científico para divulgación de los resultados.
- Redactar una guía del usuario para el uso correcto de la aplicación.

² Se establece un rango de variaciones y posibles condiciones de carga amplio pero flexible.

4 PROPUESTA INNOVADORA

La propuesta innovadora de nuestro proyecto de grado es ofrecer el análisis y diseño estocástico, para ejes, en una sola aplicación computacional, el cual se realiza en términos de probabilidad partiendo de variables aleatorias. Además de resumir tablas de concentradores de esfuerzo, y muy diversos criterios reunidos en la misma, se ofrece un diseño más confiable, ya que se tienen cuenta la naturaleza aleatoria de la resistencia de los materiales mediante variables como:

El límite de resistencia a la fatiga: se tiene en cuenta la anisotropía de la materia del cual este compuesto el eje a diseñar, ya que muchos de los materiales para el uso del eje, no mantiene sus características a lo largo de su contenido, dando como resultado un valor variable en la resistencia última del material.

Cargas aplicadas: se tendrá en cuenta que la cargas aplicadas también variar, esto puede ser generado por discontinuidades en los procesos que podrían ser generados por malos ajustes, o sobre cargas.

Condiciones de operación: algunos de los factores de los cuales depende el diseño, son variables, como por ejemplo la temperatura de trabajo, la confiabilidad en el maquinado y otras.

5 MARCO CONCEPTUAL

5.1 Resistencia del material como variable aleatoria

La resistencia última de los materiales se puede expresar mediante variables aleatorias donde se describan la media y la desviación estándar de un conjunto de datos del material estudiado. Para esto se utilizan comúnmente las siguientes distribuciones:

- Distribución Gaussiana
- Distribución Log-Normal
- Distribución Weibull

Pero, se hace más énfasis en la distribución normal debido a la simplicidad de sus ecuaciones y cálculos para establecer la resistencia última como variable aleatoria.

Distribución Gaussiana

La función de densidad de probabilidad de Gauss está dada mediante la siguiente ecuación, ([3]. Shigley)

$$f(x) = \frac{1}{\widehat{G}_x \sqrt{\pi}} \exp \left[-\frac{1}{2} \left(\frac{x - \mu_x}{\widehat{G}_x} \right)^2 \right] \quad (1)$$

Donde la variable x normalmente distribuida se denota como, ([3]. Shigley)

$$x = N(\mu_x, \widehat{G}_x) \quad (2)$$

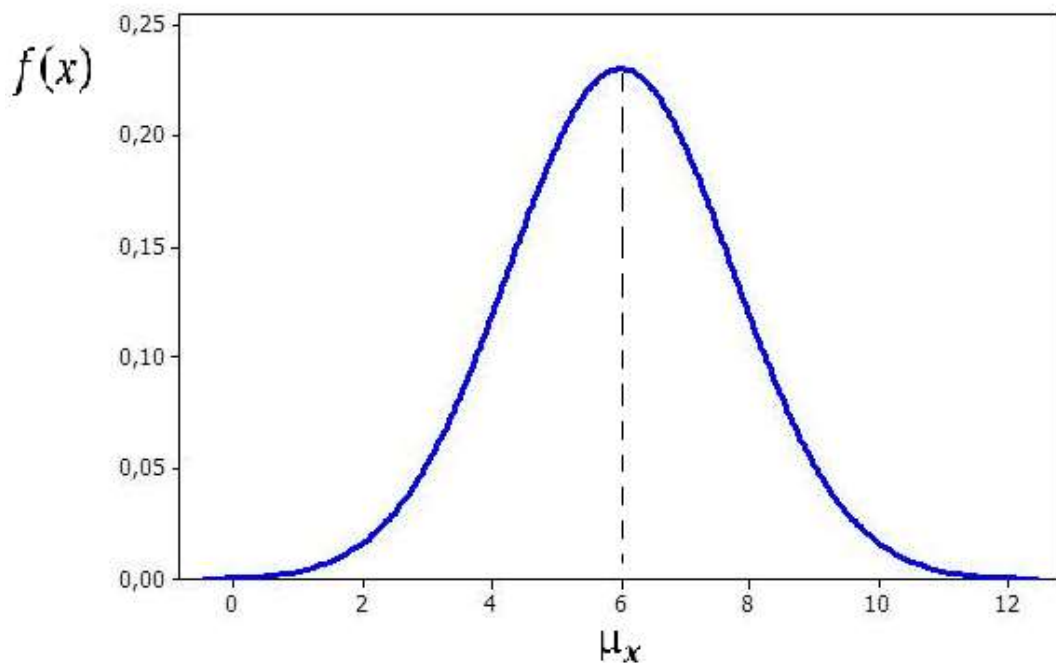
Donde μ_x es el valor medio de la resistencia última y \widehat{G}_x es su desviación estándar.

La desviación de la media se expresa en unidades de desviación estándar mediante la siguiente transformada, ([3]. Shigley)

$$z = \frac{x - \mu_x}{\widehat{G}_x} \quad (3)$$

Donde z hace referencia a la función de distribución acumulada de la distribución normal cuyos valores se encuentran tabulados y normalmente distribuida con una media de 1 y desviación estándar 0.

La gráfica de distribución normal es simétrica respecto a la media como se muestra en la siguiente gráfica, ([3]. Shigley)

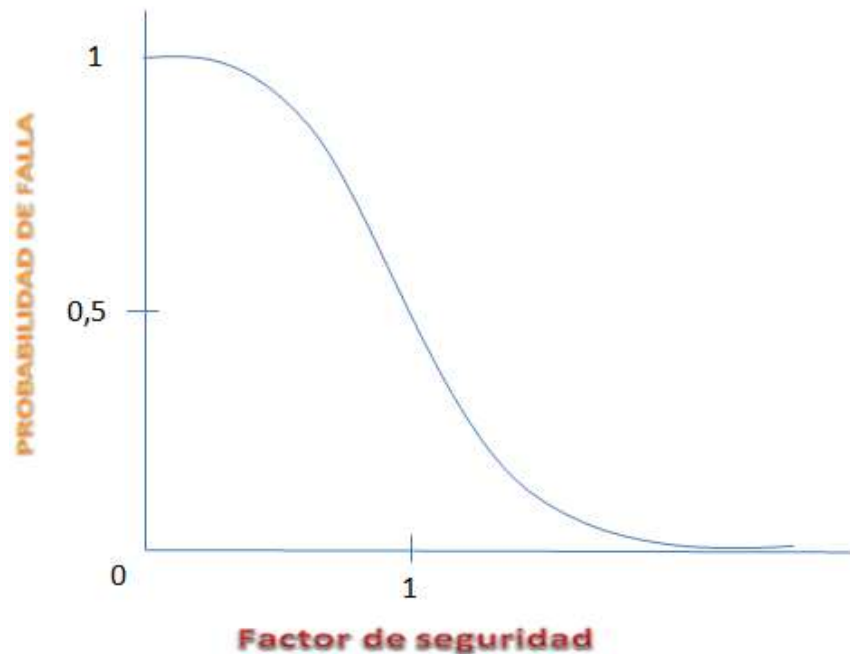


Gráfica 1. Distribución normal

Para valores mayores de desviaciones estándar, la campana tiende a ser más pequeña y ancha, mientras que para valores menores, la campana tiende a ser más grande y menos ancha.

5.2 Relación entre la probabilidad de falla y el factor de seguridad para diseño.

Al momento de relacionar estos dos parámetros, en un proceso de diseño, se evidencia que existe una relación inversa entre ambos, es decir, si escogemos un factor de seguridad alto sabremos que la probabilidad de falla será menor, pero si escogemos uno bajo se espera que la probabilidad de falla sea más alta. Gráficamente se ve de la siguiente forma:



Grafica 2. Relación factor de seguridad – probabilidad de falla³

Por medio de la gráfica de la relación de factor de seguridad y probabilidad de falla se puede observar que un factor de seguridad de 1.0 equivale a una probabilidad de falla de 0.5, es decir, del 50%. Además se evidencian otros dos casos en los cuales el factor de seguridad es 0.0 cuando la probabilidad de falla sea 1.0, y si se tiene un factor de seguridad alto, la probabilidad de falla tendera a 0, pero esto dependerá de la desviación estándar que se de en la variable aleatoria.

Si se quisiera saber a qué probabilidad de falla equivale un factor de seguridad entre 0 y 1, no podríamos manejar la gráfica como si fuese un caso de inversa proporcionalidad entre los dos factores, ya que la ecuación de la línea graficada no es de esta forma, puesto que la curva cambia dependiendo de la desviación de estándar que se tenga.

Una de las ventajas que nos brinda el saber la probabilidad de falla es que nos muestra una representación más útil de la confiabilidad que puede tener un diseño. El factor de seguridad mide aproximadamente cuánta incertidumbre existe en la predicción de la rotura del material. Sin embargo, la probabilidad de falla (y su complemento, la confiabilidad) nos dan medidas precisas usando variables aleatorias del proceso de ruptura de materiales que claramente es un proceso

³ <http://www.scg.org.co/wp-content/uploads/FACTORES-DE-SEGURIDAD-BASICOS-E-INDIRECTOS-AJGG2.pdf>

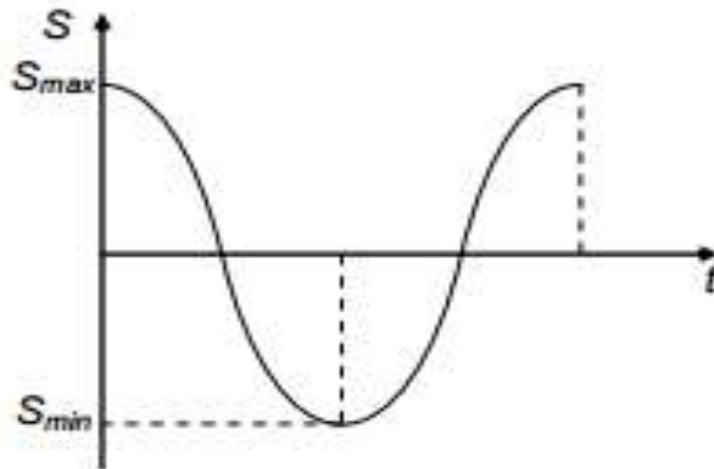
estocástico. Dando así una certeza mayor de lo que se habla y por tanto mejorando la confiabilidad del diseño.

5.3 Diseño de ejes sometidos a fatiga

Las metodologías para el diseño a la fatiga y su naturaleza experimental se pueden encontrar en libros de Diseño Mecánico, tales como el escrito por Joseph E. Shigley y sus seguidores ([2]. Shigley), en el cual se muestran los procedimientos para finalmente determinar las dimensiones del eje, dependiendo de las condiciones de trabajo a las cuales este sujeto

Los ejes de transmisión de potencia experimentan cargas variables, ya que estos ejes además de estar sometidos a torsión, están sujetos a cargas radiales, debido a la transmisión a la que se encuentren acoplados, ya sea por engranes, correas, entre otros. Estas cargas radiales ejercen esfuerzos de flexión sobre el eje.

Debido a que el eje se encuentra girando a cierto número de revoluciones y con un torque, el esfuerzo de flexión generado por las cargas radiales, varía de compresión a tensión, en determinados puntos del eje, provocando la rotura de este a causa de la fatiga.



Gráfica 3. Esfuerzo de flexión en eje

En resumen, una ecuación (criterio de falla) evalúa una combinación algebraica de los esfuerzos presentes en el eje y el esfuerzo límite de fatiga y la resistencia última del material para determinar si la condición del eje es segura o no en cada punto del mismo. Esta ecuación se puede evaluar determinísticamente contra un factor de seguridad o con variables aleatorias para evaluar la probabilidad de falla.

5.4 Diseño estocástico de ejes sometidos a fatiga

5.4.1 Vida para fatiga estocástica

El procedimiento del diseño estocástico empieza con el calculo de la resistencia a la fatiga, el cual depende para los aceros de la resistencia ultima del material. La resistencia a la fatiga muestra cuánto puede soportar determinado material esta condición de cargas variables. Este valor de resistencia a la fatiga se calcula como una variable aleatoria, debido a que esta propiedad, que depende del material puede no ser igual a lo largo de todo el eje y por ello presenta una desviacion en su valor. En general, se simplifica la distribución de probabilidad para mayor facilidad utilizando la distribución Log-Normal. Para los valores usuales de desviación estándar esto produce resultados suficientemente precisos. Típicamente se expresa dicho valor como:

$$Se' = Se * LN(1.0, d) \quad (4)$$

5.4.2 Factores de Marin estocásticos K_a , K_b , K_c , K_d

Luego sigue el calculo de los factores de Marin, los cuales son factores que modifican la resistencia a la fatiga del material, estos son:

- K_c . Factor de carga, este factor esta relacionado con el tipo de carga a la que se someta el eje.
- K_d . Factor de temperatura, este factor esta relacionado con la temperatura de trabajo.
- K_a . Factor de superficie, este factor depende del acabado superficial que tenga el eje.
- K_b . Factor de tamaño, este factor esta relacionado con las dimesiones del eje.

Estos factores tambien presentan desviaciones en su valor, ya que algunas condiciones de trabajo, como lo son la temperaura y el tipo de carga, pueden cambiar en las condiciones de uso. De igual modo se utilizan como variables Log-Normales.

5.4.3 Factor de Concentracion K_f , K_t y sensibilidad a la muesca.

Hay otro factor K_f el cual esta relacionado con la concentración de los esfuerzos. Se requiere obtenerlo a partir del concentrador de esfuerzos estático K_t y la sensibilidad a la muesca q . Para el calculo de K_f se usó el libro de Diseño de

Petterson ([4]. Petterson), en el que se encuentran distintos tipos de concentradores de esfuerzos graficados, tabulados y calculados analíticamente usando polinomios de grado 6 en general, entre ellos hay ranuras, redondeos, agujeros y otros más. Luego se multiplican estos factores con la resistencia a la fatiga del material, para modificar su valor. Dependiendo del material y su resistencia última, el valor de q se modifica según tablas y asimismo se obtiene su desviación estándar.

En este trabajo se incluye en el código dichos polinomios, gráficas y tablas codificados como parte de la aplicación en Scilab ([1]. Scilab 5.5.1).

5.4.4 Tipos de cargas y condiciones diseño para los ejes a diseñar.

Consideraremos en este trabajo, el diseño de ejes escalonados de transmisión de potencia con superficies maquinadas, chaflanes y redondeos. Sobre los mismos consideraremos diversas secciones de engranes helicoidales y apoyos simples de cojinetes. El valor de las cargas también puede ser considerado como una variable aleatoria, ya que la magnitud de las cargas puede variar. Con las cargas y las dimensiones del eje se calculan los esfuerzos debidos a torsión y a flexión.

A continuación se muestra un bosquejo simple de un eje de transmisión de potencia con estas condiciones.

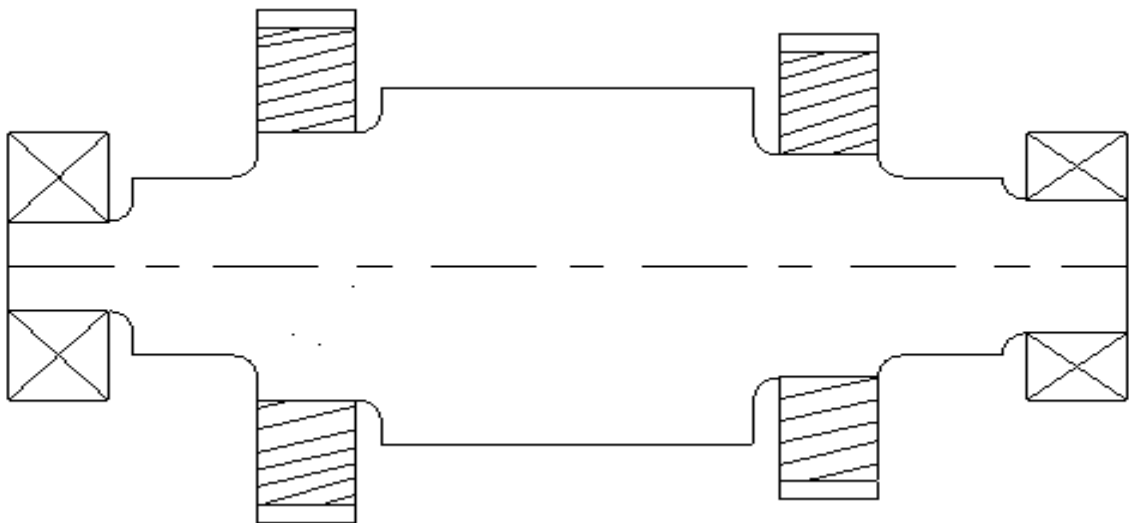
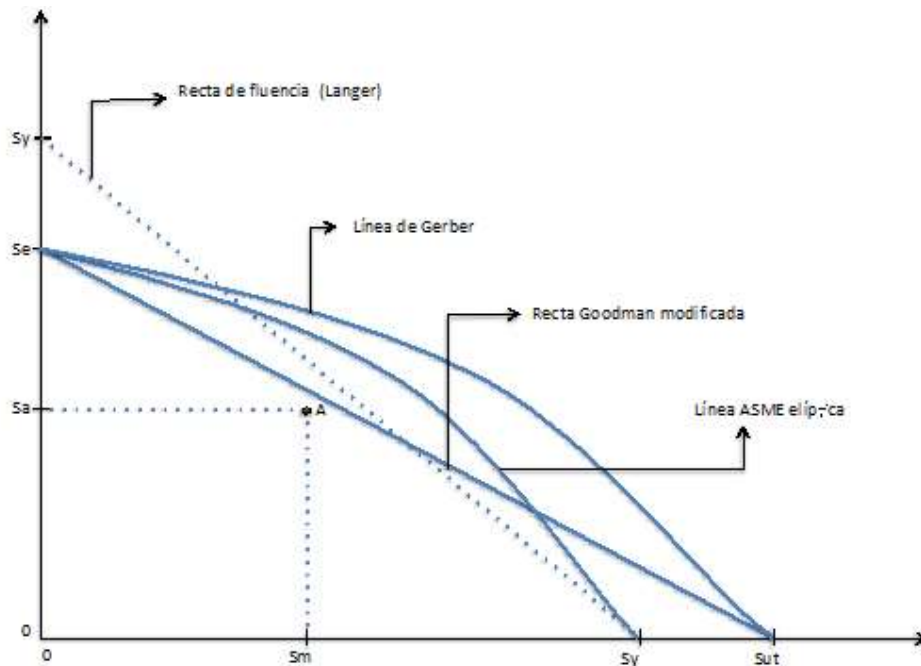


Figura 1. Eje de transmisión de potencia

Luego, usando el tensor de esfuerzos se calculan los esfuerzos principales y con ello el esfuerzo de Von Mises, en el que se tiene en cuenta la teoría de distorsión

para materiales dúctiles la cual predice que un material dúctil sufrirá un ruptura cuando la energía elástica de distorsión sobrepase un valor determinado.

Finalmente se calcula probabilidad de falla de acuerdo al criterio, ya sea Gerber, Goodman o ASME, los cuales son criterios que evalúan el diseño del eje dependiente del esfuerzo medio, el esfuerzo alternante, la resistencia a la fatiga y la resistencia última del material. La siguiente grafica muestra como se relacionan los esfuerzos y las resistencias mencionadas ([5]. Shigley):



Gráfica 4. Criterios de falla para fatiga

Como se observa en la grafica, el criterio de Gerber da una zona de seguridad mayor por lo tanto permite usar dimensiones de diseño menores, mientras que el criterio de Goodman es mas conservador ya que cubre un menor rango de esfuerzos resultando en dimensiones de diseño mayores y ejes de mayor peso y volumen.

6 METODOLOGÍA

6.1 Elaboración de la aplicación

Se escribió el código, para la aplicación, utilizando Scilab 5.5.1 ([1]. Scilab 5.5.1), aquí se aplicaron teorías y procedimientos del libro de diseño mecánico de Joseph Shigley ([2]. Shigley), para el diseño estocástico de ejes de transmisión de potencia, con el fin de que la aplicación sea confiable para el diseño del eje. A continuación se muestra como fue el desarrollo de la aplicación.

6.1.1 Introducción a la aplicación

Antes de la explicación detallada de la aplicación, se presentarán algunas herramientas, funciones y estructuras brindadas por Scilab que se utilizaron en el programa:

Clc-Clear all: Borra todo lo que hay en la pantalla de ejecución. Libera las variables para una nueva ejecución.

Cdfnor: Permite evaluar numéricamente las integrales bajo la distribución gaussiana. Con la transformación adecuada, también permite evaluar los valores de probabilidad acumulada para distribuciones log-normales.

La aplicación se encuentra fraccionada y organizada de tal forma que se puede encontrar cada proceso llevado a cabo de forma individual. Existe un archivo principal el cual llama al resto de divisiones por medio de la función “exec” y añadiendo la dirección correspondiente a la ubicación del archivo pretendido.

Los archivos divididos se encuentran organizados por nombre y número dentro de una carpeta.

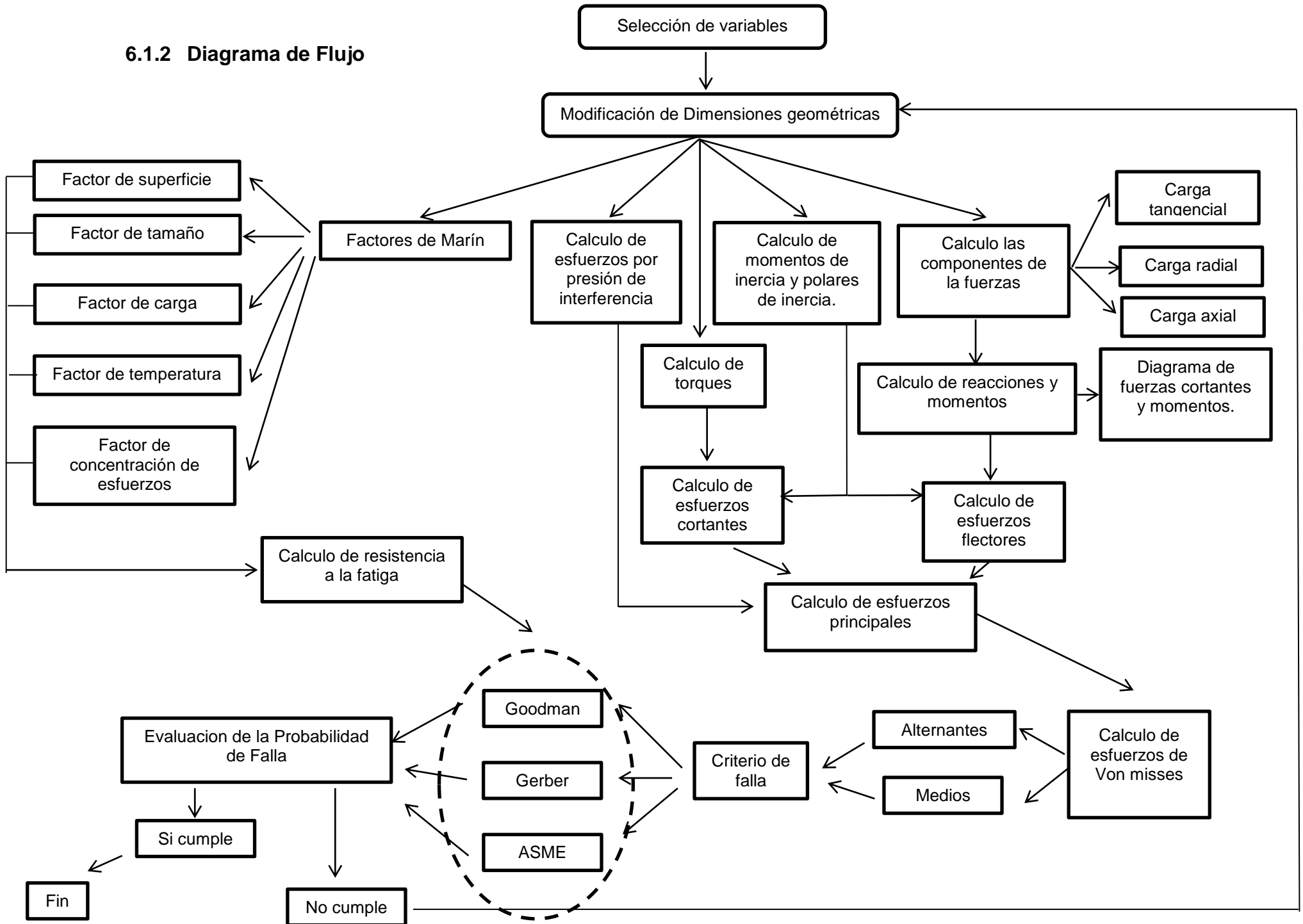
La lista de los archivos tiene el siguiente orden:

1. ScilabPrincipal.sce
2. 00_FuncionesEstocásticas.sci
3. 01_TablasPresionInterferencia.sci
4. 02_TablaMateriales.sci
5. 03_SePrima.sci
6. 04_FactorSuperficie.sci
7. 05_FactorDeCarga.sci
8. 06_FactorTemperatura.sci
9. 07_FactorConcentracionEsfuerzo.sci

10.08_EscogerCriterio.sci
11.09_CantidadEngranosFrecuenciaLongitudEje.sci
12.10_UnEngrane.sci
13.10.1_DistanciaSesionesMuecasYUbicacionEngrane.sci
14.10.2_OpcionesInterferencia.sci
15.10.3_InterferenciaUno.sci
16.10.4_InterferenciaDos.sci
17.10.5_InterferenciaTres.sci
18.10.7_TorquesMomentosYGraficas.sci
19.10.8_MultiplicacionKaKd.sci
20.10.9_IteracionesDeLosDiametros.sci
21.10.10_VidaRemanente.sci
22.11_DosEngranos.sci
23.11.1_DistanciaSesionesMuecasYUbicacionEngrane.sci
24.11.1.1_DistanciasFuerzasAnguloUbicacionEngraneUno.sci
25.11.1.2_DistanciasFuerzasAnguloUbicacionEngraneDos.sci
26.11.1.3_DistanciasFuerzasAnguloUbicacionEngraneTres.sci
27.11.1.4_DistanciasFuerzasAnguloUbicacionEngraneCuatro.sci
28.11.1.5_DistanciasFuerzasAnguloUbicacionEngraneCinco.sci
29.11.2_OpcionesInterferenciaUno.sci
30.11.3_OpcionesInterferenciaDos.sci
31.11.4_InterferenciaUno.sci
32.11.5_InterferenciaDos.sci
33.11.6_InterferenciaTres.sci
34.11.7_TorquesMomentosYGraficas.sci
35.11.9_IteracionesDeLosDiametros.sci
36.11.10_VidaRemanente.sci
37.12_TresEngranos.sci
38.12.1_DistanciaSesionesMuecasYUbicacionEngrane.sci
39.12.2_OpcionesInterferenciaTres.sci
40.12.3_InterferenciaUno.sci
41.12.4_InterferenciaDos.sci
42.12.5_InterferenciaTres.sci
43.12.6_TorquesMomentosYGraficas.sci
44.12.7_IteracionesDeLosDiametros.sci
45.12.8_VidaRemanente.sci
46.13_CuatroEngranos.sci
47.13.1_DistanciaSesionesMuecasYUbicacionEngrane.sci
48.13.2_OpcionesInterferenciaCuatro.sci
49.13.3_InterferenciaUno.sci
50.13.4_InterferenciaDos.sci
51.13.5_InterferenciaTres.sci
52.13.6_IteracionesDeLosDiametros.sci
53.13.7_TorquesMomentosYGraficas.sci
54.13.8_VidaRemanente.sci

55. 14_CincoEngranes.sci
56. 14.1_DistanciaSesionesMuestrasYUbicacionEngrane.sci
57. 14.2_OpcionesInterferenciaCinco.sci
58. 14.3_InterferenciaUno.sci
59. 14.4_InterferenciaDos.sci
60. 14.5_InterferenciaTres.sci
61. 14.6_TorquesMomentosYGraficas.sci
62. 14.7_IteracionesDeLosDiametros.sci
63. 14.8_VidaRemanente.sci
64. 16_VidaRemanente_UnEngrane.sci
65. 17_VidaRemanente_DosEngranes.sci
66. 18_VidaRemanente_TresEngranes.sci
67. 19_VidaRemanente_CuatroEngranes.sci

6.1.2 Diagrama de Flujo



6.1.3 Selección del material

Este es el primer paso en la interacción del usuario con la aplicación. Se encuentra un menú el cual muestra los materiales disponibles para el diseño. Los materiales posibles mostrados en el menú son los siguientes:

1. Acero 1018 CD
2. Acero 1035 HR
3. Acero 1045 CD
4. Acero 1117 CD
5. Acero 1137 CD
6. Acero 12L14 CD
7. Acero 12L14 CD
8. Acero 1038 Pernos HT
9. ASTM40
10. Acero 35018 Maleable
11. Acero 32510 Maleable
12. Acero Maleable Perlítico
13. Acero 604515 Nodular
14. Acero 100-70-04 Nodular
15. Acero 201ss CD
16. Acero 301ss CD
17. Acero 301ss A
18. Acero 304ss A
19. Acero 310ss A
20. Acero 403ss
21. Acero 17-7Pss
22. Acero AM350ss A
23. Ti 6AL 4V
24. Acero 2024 0
25. Acero 2024 T4
26. Acero 2024 T6
27. Acero 7075 T6.0.25

Las resistencias últimas con sus respectivas desviaciones estándar de los materiales fueron tomadas de libro de Diseño mecánico de Joseph Shigley ([6]. Shigley) y están guardadas en una tabla contenida en la aplicación.

El algoritmo que se utiliza para que el usuario escoja el material es el siguiente:

1-Se le pide al usuario que escoja una de las 26 opciones de tipo de material que se muestran en el menú.

2-Se valida por medio de un condicional "If" si la opción no está entre 1 y 26, entonces un ciclo "while" regresara al usuario al menú para que nuevamente escoja el material mostrado entre las opciones.

3-Luego que la opción sea correcta, por medio de condicionales “if” se encuentran cuáles son los datos del material correspondiente en la tabla incluida.

4-Se muestra al usuario los datos del material escogido.

5-Se guardan los datos del material en variables nuevas.

6.1.4 Cálculo del límite de resistencia a la fatiga

El cálculo de límite de resistencia a la fatiga depende de la resistencia última del material y se calcula mediante la ecuación, ([7]. Shigley)

$$S_e' = \begin{cases} 0.506S_{ut}LN(1,0.138)Kpsi \text{ o } Mpa & S_{ut} \leq 212 Kpsi \\ 107LN(1,0.139)Kpsi & S_{ut} > 212Kpsi \end{cases} \quad (5)$$

El algoritmo que se utiliza para el cálculo de este factor es el siguiente:

- 1- Se calcula el límite de resistencia a la fatiga de las dos formas, y se guarda en una tabla.
- 2- Se generan dos variables numéricas nuevas con valor cero: “Se_prima_media” y “Se_prima_desv”.
- 3- Por medio de condicionales “if” se observa si la resistencia última del material es mayor que 212 o es menor o igual que 212. Según sea el caso, se toman los valores correspondientes calculados en el primer paso y se guardan en las variables declaradas; en una se guarda el valor medio y en la otra la desviación estándar.
- 4- Se guardan los valores finales, en una variable de tipo vector llamada “Se_Prima”.

6.1.5 Factor de acabado superficial

El factor de acabado superficial se calcula con la ecuación, ([7]. Shigley)

$$K_a = aS_{ut}^bLN(1, C) \quad (6)$$

Los valores de las constantes a y b y el coeficiente de variación C dependen del tipo de acabado superficial y están contenidas en la aplicación.

El algoritmo que se utilizó para el cálculo del factor de superficie es el siguiente:

1- Se proyecta un menú al usuario donde visualice la opción de acabado superficial deseada las cuales son descritas en la anterior tabla.

2- Se valida la selección del usuario por medio de un condicional "If". Si la opción no está entre las opciones 1 y 4, entonces un ciclo "while" regresará al usuario al menú mostrado primero para que escoja nuevamente la opción deseada.

3- Por medio del condicional "If" se comprueba la opción tomada por el usuario, y se toman los valores correspondientes al acabado seleccionado.

4- Se calcula el valor del factor de superficie con su respectiva desviación estándar y se guardan en una variable de tipo vector nombrada "Acabado_superficial". Finalmente se muestran al usuario los valores anteriores.

6.1.6 Factor de temperatura

El factor de temperatura, el cual tiene en cuenta la temperatura del ambiente en el cual va a trabajar el eje se calcula mediante ecuación, ([7]. Shigley)

$$K_d = KdLN(1,0.11) \quad (7)$$

El valor de Kd depende de la temperatura de operación y sus valores se encuentran guardados en la aplicación.

El algoritmo realizado para el cálculo del factor de temperatura es el siguiente:

1- Se proyecta un menú al usuario donde puede escoger la temperatura a la que va a trabajar el eje, las opciones de temperatura mostradas son: 20°C, 50°C, 100°C, 150°C, 200°C, 250°C, 300°C, 350°C, 400°C, 450°C y 500°C.

2- Se valida por medio de un condicional "If". Si la opción no está entre 1 y 11, entonces un ciclo "while" regresará al usuario al menú para que nuevamente escoja entre las opciones.

3- Por medio del condicional "If", se comprueba la opción tomada por el usuario, y se toman los valores correspondientes a la temperatura escogida.

4- Se calcula el valor del factor de temperatura con su respectiva desviación estándar y se guardan en una variable de tipo vector llamada "Factor_temperatura". Finalmente se muestran al usuario los valores anteriores.

6.1.7 Factor de concentración de esfuerzo

El factor de concentración de esfuerzo depende del diámetro, es por esto que se encuentra dentro de la iteración de los diámetros. Además este solo se calcula en el

caso de que exista un cambio de tamaño en la sección. Existen tres tipos: axial, torsión y flexión. Todos los anteriores se calculan por medio de las ecuaciones de Peterson.

El cálculo de este lo podemos encontrar en el siguiente archivo "10.9_IteracionesDeLosDiametros.sci".

Para el cálculo del factor de concentración de esfuerzo axial, flexión y torsión se utiliza la siguiente ecuación descrita en el libro de Petterson, ([4]. Petterson)

$$K_t = C_1 + C_2 \frac{2h}{D} + C_3 \left(\frac{2h}{D}\right)^2 + C_4 \left(\frac{2h}{D}\right)^3 \quad (8)$$

El valor de las constantes difieren del tipo de carga y concentrador que se presente, los valores de las constantes se encuentran guardadas en la aplicación para todas los tipos.

El proceso que la aplicación sigue para calcular al factor de concentración de esfuerzo axial es el siguiente:

1-Se calcula la relación que hay entre las secciones donde se está efectuando el muesca, la cual es la diferencia de los radios entre el radio de la muesca.

2-Se calcula el valor de las constantes (C1, C2, C3 y C4) para todos los casos.

3-Por medio de condicionales "if", se mira el rango en el que se encuentra la relación calculada en el paso 1. Dependiendo de cuál sea el caso se guardan las constantes en nuevas variables (C3_A1, C3_A2, C3_A3 Y C3_A4). Los números que aparecen en el nombre de las variables pueden cambiar según sea el número de la sección del eje que se esté trabajando dentro de la aplicación.

4-Si la relación no se encuentra en ninguno de los dos rangos encontrados en la tabla, entonces $K_t = 1$.

5-Se hace un cambio de variable, se la asigna el valor de este factor a una variable llamada "Kt3_A".

6-Luego, se completa el cálculo de este factor aplicando la ecuación, ([7]. Shigley)

$$kf = \frac{K_t}{1 + \frac{2(K_t - 1)\sqrt{a}}{K_t * \sqrt{r}}} \quad (9)$$

6.1.8 Factor de tamaño

El factor de tamaño depende del diámetro, es por esto que el proceso para que pueda ser calculado se encuentra dentro de la iteración que se realiza para encontrar los diámetros adecuados, en el siguiente archivo "10.9_IteracionesDeLosDiametros.sci", a partir de la línea 15.

Para ser calculado se debe tener en cuenta en que rango está el diámetro seleccionado, por medio de la ecuación, ([7]. Shigley)

$$k_b = \begin{cases} 1 & 0.11 > d \\ 0.879d^{-0.107} & 0.11 \leq d \leq 2\text{pulg.} \\ 0.91d^{-0.157} & 2 < d \leq 10\text{pulg.} \end{cases} \quad (10)$$

Dentro del proceso algorítmico, se realizó por medio de condicionales "if", para remplazar las diferentes opciones que se presentar al calcular el factor de tamaño. De la siguiente forma:

1- Si el diámetro es menor que 0.11 pulgadas entonces el factor de tamaño es igual a 1.

2- Si el diámetro es menor o igual que 0.11 y también es menor o igual que 2 pulgadas, entonces el factor de tamaño es igual a $0.879d^{-0.107}$.

3- Si el diámetro es menor que 2 y también es menor o igual que 10 pulgadas, entonces el factor de tamaño es igual a $0.91d^{-0.157}$.

6.1.9 Factor de modificación por la carga

Existen tres factores de modificación por la carga:

- Factor de carga por flexión
- Factor de carga axial
- Factor de carga por torsión

Para el cálculo del factor de carga K_c para cargas de flexión, torsión y axial se usa la ecuación, ([7]. Shigley)

$$K_c = a0.328Sut^{-\beta}LN(1, C) \quad (11)$$

Las constantes a y β dependen del tipo de carga y se encuentran guardadas en la aplicación.

El algoritmo utilizado para realizar el cálculo del factor de carga es el siguiente:

1- Se le da el valor por omisión al valor medio del factor de carga por flexión igual a uno y su desviación estándar se le asigna cero.

2- Se calcula el valor del factor de carga axial y se guarda en la variable “f_carga_axial”. De igual forma se calcula su desviación y se guarda en la variable “f_carga_axial_desv”.

3- Se calcula el valor del factor de carga por torsión y se guarda en la variable “f_carga_torsion”. De igual forma se calcula su desviación y se guarda en la variable “f_carga_torsion_desv”.

6.1.10 Cálculo de la resistencia a la fatiga

Una vez calculados los factores de Marin y el límite de resistencia a la fatiga, se procedió a hallar la resistencia a la fatiga. Como respecta un cálculo estocástico multiplicativo, se recurrió al uso de las ecuaciones estocásticas para la multiplicación de medias y desviaciones estándar.

Valor medio de la multiplicación de medias ([3]. Shigley):

$$\mu = \mu_x \mu_y \quad (12)$$

Desviación estándar de la multiplicación de medias ([3]. Shigley):

$$\mu_x \mu_y (C_x^2 + C_y^2 + C_x^2 C_y^2)^{1/2} \quad (13)$$

Los factores que intervienen en esta multiplicación de medias son el factor de superficie, el factor de temperatura y el límite de resistencia a la fatiga. Por lo que el cálculo de la resistencia a la fatiga se reduce a, ([3]. Shigley)

$$S_e = k_a k_d S'_e \quad (14)$$

Por medio de la función “function” y “mult” de scilab se calcula el valor medio resultante y la desviación estándar resultante a través del archivo denotado como “00_FuncionesEstocasticas.sci”

6.1.11 Análisis de fuerzas en engranes helicoidales

Para el cálculo de las componentes de las fuerzas se realizó el siguiente procedimiento:

1- Se calcula el ángulo tangencial con la siguiente formula, ([8]. Shigley)

$$\phi_t = \tan^{-1} \frac{\tan \phi_n}{\cos \Psi} \quad (15)$$

ϕ_n = ángulo de presión

Ψ = ángulo de hélice

2- Se calcula la velocidad lineal de paso en *pies/min*, el cálculo se realiza de la siguiente fórmula, ([8]. Shigley)

$$V = (\text{radiodel engrane})/(\text{velocidad angular} * 12) \quad (16)$$

3- La carga transmitida o la componente tangencia, fue calculada de la siguiente manera, ([8]. Shigley)

$$W_t = \frac{33000H}{V} \quad (17)$$

H = potencia transmitida en horse power

4- Luego se calcularon las componentes de las fuerzas restantes con las siguientes formulas, ([8]. Shigley)

$$W_r = W_t \tan \phi_t \quad (18)$$

$$W_a = W_t \tan \Psi \quad (19)$$

$$W = \frac{W_t}{\cos \phi_n \cos \Psi} \quad (20)$$

La siguiente figura muestra las cargas a las cuales se encuentra sujeto un eje el cual transfiere potencia por medio de engranes helicoidales ([8]. Shigley):

W =fuerza total

W_r =componente radial

W_a =componente axial

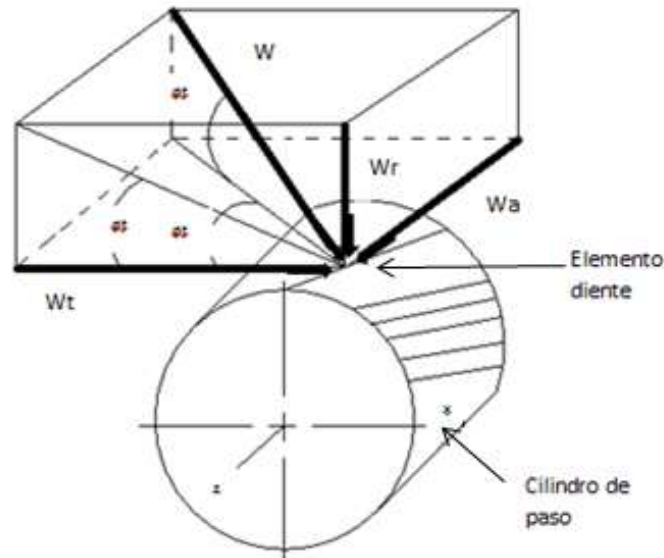


Figura 2. Fuerzas en engranes helicoidales

Como se puede observar en el anterior procedimiento, para poder calcular las componentes de las fuerzas, se requiere de la potencia transmitida, del radio del engrane, del ángulo de presión y del ángulo de hélice del engrane. La componente radial, es una carga perpendicular al eje, este generará los momentos flectores, la componente axial genera compresión del eje y la componente tangencial, da el torque al eje.

Este procedimiento se realiza para cada engrane que transmita potencia hasta, teniendo en cuenta que el eje puede tener acoplados hasta un máximo de cinco engranes.

El procedimiento anterior es ejecutado en los archivos Scilab,

“11.1_DistanciaSesionesMuecasYUbicacionEngrane.sci

11.1.1_DistanciasFuerzasAnguloUbicacionEngraneUno.sci

11.1.2_DistanciasFuerzasAnguloUbicacionEngraneDos.sci

11.1.3_DistanciasFuerzasAnguloUbicacionEngraneTres.sci

11.1.4_DistanciasFuerzasAnguloUbicacionEngraneCuatro.sci

11.1.5_DistanciasFuerzasAnguloUbicacionEngraneCinco.sci”

6.1.12 Análisis y ubicación de los momentos resultantes

Los engranes pueden estar ubicados en dos planos posibles, así que los momentos se analizan, teniendo en cuenta que los engranes pueden ejercer fuerza en dos planos diferentes.

El número de secciones o diámetros diferentes dependerá del número de engranes que este posea, si el eje posee un engrane el eje tendrá cinco secciones, si el eje posee dos engranes este estará dividido en 7 secciones, si tiene tres estará dividido 9 secciones, si tiene 4 estará dividido en 11 secciones y si tiene 5 estará dividido en 13 secciones. Esto se puede ver gráficamente en el manual del usuario.

Luego de calcular la componente de fuerza radial, se calcularon los momentos flectores a lo largo de todo el eje, teniendo en cuenta que los engranes pueden estar ubicados dos planos diferentes, es decir, que las componentes de las fuerzas pueden ejercer en dos direcciones diferentes. Para el cálculo de momento se realizó el siguiente procedimiento:

1- Primero se calcularon las fuerzas cortantes en el eje, para ello se usó la siguiente formula, ([9]. Shigley)

$$\int_{V_A}^{V_B} dV = \int_{x_A}^{x_B} qdx = V_B - V_A \quad (21)$$

En donde el cambio de la fuerza cortante entre los engranes que transmiten potencia, es igual al área del diagrama de la carga distribuida entre x_A y x_B , es decir entre las cargas puntuales que generan los engranes.

2- Luego se calcularon los momentos flectores en el eje ([9]. Shigley)

$$\int_{M_A}^{M_B} dM = \int_{x_A}^{x_B} Vdx = M_B - M_A \quad (22)$$

Donde se establece que el cambio de momento desde A hasta B es igual al área del diagrama de la fuerza cortante entre x_A y x_B .

3- En caso de que los engranes tengan ubicaciones diferentes, se realizó el cálculo de momentos flectores en cada plano u ubicación. Luego se el momento resultante para cada sección con la siguiente formula,

$$M = \sqrt{M_{xy}^2 + M_{xz}^2} \quad (23)$$

Donde M es el momento resultante, M_{xy} y M_{xz} son los momentos flectores calculados en los pasos anteriores para cada ubicación.

4- Luego se analizó cada sección para tomar el momento máximo correspondiente. Esto se realizó por medio de un ciclo for, el cual calcula los valores de momentos correspondientes a cada punto del eje, que luego por medio de un condicional if, se compara el punto de la sección que tiene el momento máximo dentro de la sección y próximamente se guarda en otra variable.

5- Además de calcular los momentos, estos se grafican a lo largo del eje, por medio de la función plot, usando el ciclo mencionado en el paso anterior.

El cálculo de momentos se realizó para todas las configuraciones posibles, el programa calculara para la configuración seleccionada en el menú.

El torque en el eje es calculado con la componente tangencial multiplicado por el radio del engrane. El procedimiento anterior es ejecutado en los archivos Scilab,

“10.7_TorquesMomentosYGraficas.sci

11.7_TorquesMomentosYGraficas.sci

12.6_TorquesMomentosYGraficas.sci

13.7_TorquesMomentosYGraficas.sci

14.6_TorquesMomentosYGraficas.sci”

6.1.13 Cálculo de esfuerzos

Se calculó el esfuerzo flector y el esfuerzo cortante con el siguiente procedimiento:

1- Se halló el momento polar de inercia para secciones transversales circulares, con la siguiente formula, ([10]. Shigley)

$$J = \frac{\pi d^2}{32} \quad (24)$$

Donde d es el diámetro del eje, que es la variable a calcular para el diseño del eje. Que finalmente es corroborada con la teoría seleccionada para el diseño.

2- Se calculó el momento de inercia con la siguiente formula, ([10]. Shigley)

$$I = \frac{J}{2} \quad (25)$$

3- Se calculó el esfuerzo flector con la siguiente formula, ([10]. Shigley)

$$\sigma = \frac{Mc}{I} \quad (26)$$

Donde c, es igual a la mitad del diámetro para secciones circulares y M es el momento flector

4- Se calculó el esfuerzo cortante con la siguiente formula, ([10]. Shigley)

$$\tau = \frac{Tc}{J} \quad (27)$$

Donde T es el torque.

El procedimiento anterior se realiza para cada sección del eje, que depende del número de engranes que transmiten potencia. El esfuerzo cortante solo aplica a las secciones en las que se encuentra el engrane.

6.1.14 Esfuerzos medios

El esfuerzo cortante fue considerado como esfuerzo medio, ya que el esfuerzo actúa de forma constante. Para cálculo de este esfuerzo se usó la siguiente formula, ([5]. Shigley)

$$\tau_m = \frac{\tau_{min} + \tau_{max}}{2} \quad (28)$$

Donde el esfuerzo cortante mínimo es igual a 0, y el esfuerzo máximo es el calculado anteriormente, ([5]. Shigley)

$$\tau_m = \frac{\tau}{2} \quad (29)$$

6.1.15 Esfuerzos alternantes

El esfuerzo flector fue considerado como alternante, ya que la flexión del eje en cualquier punto, varía de compresión a tensión, a causa del giro del eje. Se calculó mediante la siguiente formula, ([5]. Shigley)

$$\sigma_a = \left| \frac{\sigma_{min} - \sigma_{max}}{2} \right| \quad (30)$$

$$\sigma_a = \left| \frac{-\sigma - \sigma}{2} \right| \quad (31)$$

$$\sigma_a = \sigma \quad (32)$$

6.1.16 Ajustes por interferencia

El tipo de ajuste utilizado para el ensamble entre los engranes y el eje es el ajuste por interferencia que puede ser de tres tipos:

- Ajuste de interferencia localizada (H7/p6)
- Ajuste de impulso medio (H7/s6)
- Ajuste forzado (H7/u6)

Las ecuaciones que gobiernan estas dimensiones son:

Tamaño máximo del agujero ([11]. Shigley):

$$D_{max} = D + \Delta D \quad (33)$$

Tamaño mínimo del agujero ([11]. Shigley):

$$D_{min} = D \quad (34)$$

Tamaño máximo del eje ([11]. Shigley):

$$d_{max} = d + \delta F \quad (35)$$

Tamaño mínimo del eje ([11]. Shigley):

$$d_{min} = d + \delta F - \Delta d \quad (36)$$

Ahora se expresa la definición de la configuración de los tipos de ajuste:

Ajuste de interferencia localizada: Una vez definido el tamaño básico, se selecciona de la tabla de la serie IT7 el grado de tolerancia del agujero correspondiente al tamaño básico y de la misma manera se obtiene la desviación superior del eje de la tabla a partir de la sigla p.

Ajuste de impulso medio: Una vez definido el tamaño básico, se selecciona de la tabla de la serie IT7 el grado de tolerancia del agujero correspondiente al tamaño básico y de la misma manera se obtiene la desviación superior del eje de la tabla a partir de la sigla s.

Ajuste forzado: Una vez definido el tamaño básico, se selecciona de la tabla de la serie IT7 el grado de tolerancia del agujero correspondiente al tamaño básico y de la misma manera se obtiene la desviación superior del eje de la tabla a partir de la sigla u.

Para los tipos de ajuste mencionados, la desviación fundamental del eje se define mediante la serie IT correspondiente al tamaño básico.

El algoritmo utilizado para calcular el tamaño máximo del eje, tamaño mínimo del eje, tamaño máximo del agujero y tamaño mínimo del agujero es el siguiente:

- 1- Se proyecta un menú al usuario donde visualice la opción del tipo de ajuste por interferencia que desea seleccionar
- 2- Se valida la selección del usuario por medio de un condicional "If". Si la opción no está entre las opciones 1 y 3, entonces un ciclo "while" regresará al usuario al menú mostrado primero para que escoja nuevamente el tipo de ajuste adecuado.
- 3- Se le pide al usuario que ingrese el tamaño básico
- 4- Se valida la selección del usuario por medio de un condicional "If". Si la opción de tamaño básico no está entre los nombrados en la tabla, entonces un ciclo "while" regresará al usuario y le mostrará un mensaje para que vuelva a ingresar la opción de tamaño básico correcta.
- 5- A partir del tamaño básico escogido, por medio de un condicional "if" se escoge de "tabla_13" el grado de tolerancia del agujero denominado como "it7" y el grado de tolerancia del eje declarado como "it6"; se guarda la desviación fundamental del eje dependiendo del tipo de ajuste.
- 6- Los resultados se guardan en las variables "tolerancia_agujero" y "tolerancia_eje".
- 7- Se calcula el diámetro máximo del agujero y se guarda en la variable "Diametro_max_agujero".
- 8- Se calcula el diámetro mínimo del agujero y se guarda en la variable "Diametro_min_agujero".
- 9- Se calcula el diámetro máximo del eje y se guarda en la variable "Diametro_max_eje".
- 10- Se calcula el diámetro mínimo del eje y se guarda en la variable "Diametro_min_eje".

6.1.17 Cálculo del esfuerzo debido a la presión por interferencia

Para cada interferencia entre el engrane y el eje cuyo valor corresponde a la primera casilla de los tensores de esfuerzos medios en estos acoples. El esfuerzo debido a la presión por interferencia es el esfuerzo máximo, lo cual se calculó mediante la ecuación, ([11]. Shigley)

$$p = \frac{E \delta_{\text{máx}}}{2d^3} \left[\frac{(d_o^2 - d^2)(d^2 - d_i^2)}{d_o^2 - d_i^2} \right] \quad (37)$$

El procedimiento de cálculo se realiza automáticamente en los siguientes archivos nombrados en Scilab como:

“10.9_IteracionesDeLosDiametros.sci

11.9_IteracionesDeLosDiametros.sci

12.7_IteracionesDeLosDiametros.sci

13.6_IteracionesDeLosDiametros.sci

14.7_IteracionesDeLosDiametros.sci”

6.1.18 Cálculo del tensor de esfuerzos

Se realizó un tensor de esfuerzos medio y alternante, y con ello se obtendrán esfuerzos de Von misses medio y esfuerzos de Von misses alternantes. Las condiciones a las que se someten las secciones son diferentes, así que el tensor de esfuerzos es agrupado de forma diferente en cada sección. El procedimiento fue el siguiente:

1-Se armó el tensor de esfuerzos medios

Para las dos secciones de los extremos del eje se usó el siguiente tensor medio:

$$\mathbf{tensor\ medio} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (38)$$

Para las secciones en donde se encuentren los engranes se usó el siguiente tensor medio:

$$\mathbf{tensor\ medio} = \begin{bmatrix} p & kf\sigma m & 0 \\ kf\sigma m & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (39)$$

Y para las secciones intermedias que no tienen engranes se usó el siguiente tensor:

$$\mathbf{tensor\ medio} = \begin{bmatrix} 0 & kf\sigma m & 0 \\ kf\sigma m & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (40)$$

Donde $kf\sigma m$ Es el esfuerzo cortante medio multiplicado por el factor de concentración de esfuerzo por torsión. P La presión de interferencia.

Los procedimientos que se dieron en este capítulo son ejecutados en los archivos Scilab,

“10.9_IteracionesDeLosDiametros.sci

11.9_IteracionesDeLosDiametros.sci

12.7_IteracionesDeLosDiametros.sci

13.6_IteracionesDeLosDiametros.sci

14.7_IteracionesDeLosDiametros.sci”

2-Se armó el tensor de esfuerzos alternantes

Para las dos secciones de los extremos del eje se usó el siguiente tensor alternante:

$$\mathbf{tensor\ alternante} = \begin{bmatrix} Kf\sigma & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (41)$$

Para las secciones en donde se encuentren los engranes se usó el siguiente tensor alternante:

$$\mathbf{tensor\ alternante} = \begin{bmatrix} Kf\sigma & kfst\alpha & 0 \\ kfst\alpha & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (42)$$

Y para las secciones intermedias que no tienen engranes se usó el siguiente tensor:

$$\mathbf{tensor\ alternante} = \begin{bmatrix} Kf\sigma & kfst\alpha & 0 \\ kfst\alpha & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (43)$$

Donde $kfst\alpha$ Es el esfuerzo cortante medio multiplicado por el factor de concentración de esfuerzo por torsión. P La presión de interferencia.

Donde $Kf\sigma$, es el esfuerzo flector multiplicado por el factor de concentración de esfuerzo por flexión. $kfst\alpha$ Es el esfuerzo cortante alternante multiplicado por el factor de concentración de esfuerzo por torsión.

3- Se calcularon los auto valores o esfuerzos principales por medio de la función spec() en Scilab

4-Se calculó el esfuerzo de Von misses medio y alternante por medio de la ecuación, ([12]. Shigley)

$$\sigma' = \left[\frac{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2}{2} \right]^{1/2} \quad (44)$$

Los procedimientos que se dieron en este capítulo son ejecutados en los archivos Scilab:

“10.9_IteracionesDeLosDiametros.sci

11.9_IteracionesDeLosDiametros.sci

12.7_IteracionesDeLosDiametros.sci

13.6_IteracionesDeLosDiametros.sci

14.7_IteracionesDeLosDiametros.sci”

6.1.19 Selección de los criterios de falla por fatiga

El usuario tiene la opción de elegir cuál de los tres criterios de falla desea que sea usado para el diseño del eje. A este se le muestra un menú con las tres opciones y digita el número de la opción que desee. La opción es guardada en una variable llamada “criterio falla”.

Criterio de la ASME – Elíptica

La ecuación que dictamina este criterio es la siguiente, ([5]. Shigley)

$$\left(\frac{\sigma_a}{S_e}\right)^2 + \left(\frac{\sigma_m}{S_{ut}}\right)^2 = 1 \quad (45)$$

Si el usuario digita esta opción, seguirá el siguiente proceso para calcular la primera parte de la igualdad de la ecuación anterior (podemos encontrar este proceso en la línea 780 del archivo iteración de los diámetros):

1- Se realiza la división entre el esfuerzo de Von Misses alternante y la resistencia a la fatiga del material. Esto se realiza llamando la función llamada “div”, la cual realiza división de variables con coeficiente de variación.

2- El valor obtenido por la división del paso anterior se eleva al cuadrado por medio de la función llamada “pot”.

3- Se realiza el mismo procedimientos que en los pasos uno y dos, pero esta vez se realiza con las variables esfuerzo de Von Misses alternante y resistencia ultima del material.

4- Se realiza la suma de los dos valores obtenidos por medio de la función “suma”.

5- Se guarda en un vector llamado ASME, el valor obtenido con su respectiva desviación estándar.

6- Se calcula el coeficiente de variación, dividiendo el valor que se encuentra en el segundo cajón del vector ASME, entre el primer cajón. Este valor se guarda en una variable llamada "CF_CRITERIO_F".

7- Se reemplaza la variable ASME por una llamada "CRITERIO_F".

Criterio de Gerber

Si el usuario digita la segunda opción, por medio de un condicional "if" la aplicación tendrá en cuenta este criterio, el cual se rige por la siguiente ecuación, ([5]. Shigley)

$$\frac{\sigma_a}{S_e} + \left(\frac{\sigma_m}{S_{ut}}\right)^2 = 1 \quad (46)$$

El proceso que se lleva a cabo para el cálculo de la primera parte de la igualdad en la ecuación anterior en el siguiente:

1-Se realiza la división entre el esfuerzo de Von Misses alternante y la resistencia a la fatiga del material. Esto se realiza llamando la función llamada "div", la cual realiza división de variables con coeficiente de variación.

2-Se realiza el mismo procedimiento que en el paso uno , pero esta vez se realiza con las variables esfuerzo de Von Misses alternante y resistencia ultima del material.

3-El resultado de la división anterior se eleva al cuadrado por medio de la función "pot".

4-Se realiza la suma de los dos valores obtenidos por medio de la función "suma".

5-Se guarda en un vector llamado GERBER, el valor obtenido con su respectiva desviación estándar.

6-Se calcula el coeficiente de variación, dividiendo el valor que se encuentra en el segundo cajón del vector GERBER, entre el primer cajón. Este valor se guarda en una variable llamada "CF_CRITERIO_F".

7-Se reemplaza la variable GERBER por una llamada "CRITERIO_F".

Criterio de Goodman

De igual forma si el usuario digita la tercera opción, por medio de un condicional "if" la aplicación tendrá en cuenta este criterio, el cual se rige por la siguiente ecuación, ([5]. Shigley)

$$\frac{\sigma_a}{Se} + \frac{\sigma_m}{Sut} = 1 \quad (47)$$

El proceso que se lleva a cabo para el cálculo de la primera parte de la igualdad en la ecuación anterior en el siguiente:

1-Se realiza la división entre el esfuerzo de Von Misses alternante y la resistencia a la fatiga del material. Esto se realiza llamando la función llamada “div”, la cual realiza división de variables con coeficiente de variación.

2-Se realiza el mismo procedimiento que en el paso uno , pero esta vez se realiza con las variables esfuerzo de Von Misses alternante y resistencia ultima del material.

3-Se realiza la suma de los dos valores obtenidos por medio de la función “suma”.

4-Se guarda en un vector llamado GOODMAN, el valor obtenido con su respectiva desviación estándar.

5-Se calcula el coeficiente de variación, dividiendo el valor que se encuentra en el segundo cajón del vector GOODMAN, entre el primer cajón. Este valor se guarda en una variable llamada “CF_CRITERIO_F”.

6-Se reemplaza la variable GOODMAN, por una llamada “CRITERIO_F”.

6.1.20 Cálculo de la distribución de probabilidad

La distribución de probabilidad utilizada fue la log normal cuyos logaritmos de sus parámetros tienen un comportamiento normal.

La función de densidad de probabilidad de la distribución normal es igual a la unidad ([3]. Shigley):

$$F(x, \hat{\sigma}_x, \mu_x) = \frac{1}{\hat{\sigma}_x \sqrt{2\pi}} \int_{-\infty}^{\infty} \exp \left[-\frac{1}{2} \left(\frac{x - \mu_x}{\hat{\sigma}_x} \right)^2 \right] dx = 1 \quad (48)$$

Existe una conversión de esta distribución de probabilidad para facilidades de cálculo y se expresa mediante la siguiente transformada, ([3]. Shigley)

$$Z_\alpha = \frac{x - \mu_x}{\hat{\sigma}_x} \quad (49)$$

La integral de la transformada Z_α se encuentra normalmente distribuida con una media de cero y desviación estándar igual a uno.

Una vez que se hayan calculado la desviación estándar y el valor medio del criterio de falla escogido, se aplican las siguientes para el valor medio y la desviación estándar de la distribución log normal ([3]. Shigley).

$$\mu_y = \ln \mu_x - \ln \sqrt{1 + C_x^2} \quad (50)$$

$$\hat{\sigma}_y = \sqrt{\ln(1 + C_x^2)} \quad (51)$$

Y luego de este paso se aplica la transformada Z_α reemplazando a μ_x por μ_y y $\hat{\sigma}_x$ por $\hat{\sigma}_y$ cuya variable aleatoria x corresponde al valor del criterio de falla; en este caso x es igual a $\log(1)$ debido a que se requiere hallar la probabilidad que el valor esperado sea menor o igual a 1.

Lo anterior se ejecuta en Scilab mediante el comando “cdfnor” en el cual la primera casilla corresponde a la posición de la variable aleatoria, la segunda casilla le da lugar al valor de la media y el tercer lugar a la desviación estándar como se ilustra a continuación:

$$[\text{Probabilifad}, \text{Confiabilidad}] = \text{cdfnor}(\text{"PQ"}, x, \text{MEDIA}, \text{DESVIACIÓN})$$

Por medio de un condicional “if” se comprueba si la probabilidad de falla está entre un rango establecido por la aplicación.

Si está en el rango establecido, el valor del diámetro de la sección es guardado automáticamente en una variable que tiene el mismo nombre del diámetro y finaliza el proceso iterativo.

6.1.21 Manual para la aplicación

Se realizó manual de la aplicación, en la cual se le muestra al usuario como seleccionar el material, el coeficiente de variación de las cargas, el acabado superficial, la temperatura de trabajo, el tipo de muesca y el criterio de falla.

Además se muestra el correcto ingreso de las dimensiones y condiciones de trabajo, dependiendo del número de engranes del eje a diseñar.

7 VERIFICACIÓN DE LA APLICACIÓN

En este trabajo se realizó una verificación de los resultados arrojados por la aplicación de diseño estocástico de ejes, por medio de un ensayo simple usando un eje sometido a cargas de transmisión de potencia. Para ello se usó un eje transmitiendo 500 Wt desde un motor 11000rpm acoplado directamente al eje a un freno de zapato conformado por una polea y una banda de caucho. El montaje se ilustra en el diagrama.

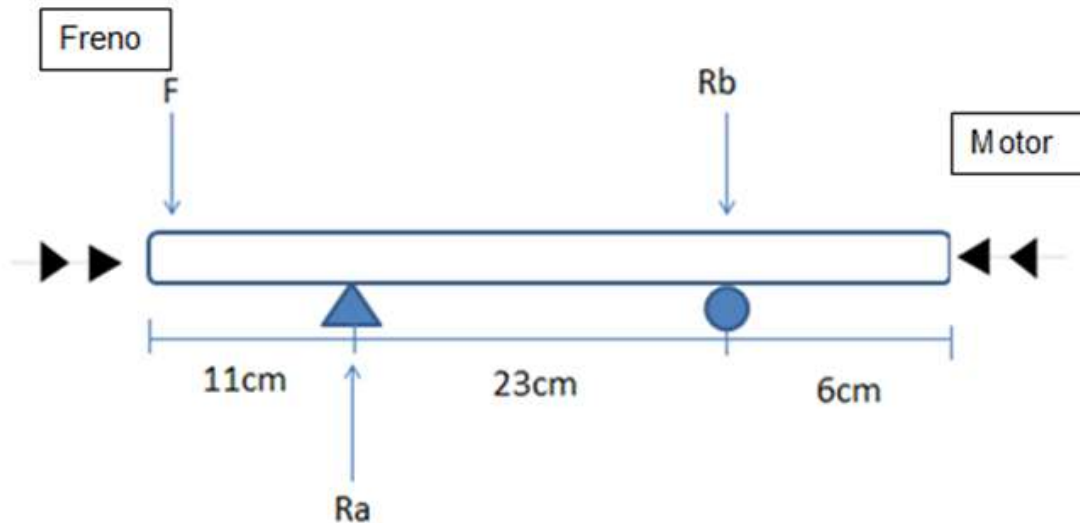


Figura 3. Diagrama de cuerpo libre

Para ello se estableció un diámetro del eje de 4,2 mm, y un número de 5000 ciclos. Con estos valores se calculó la resistencia a la fatiga para vida finita y finalmente la probabilidad de falla con las cargas aplicadas.

Luego de realizar el montaje del eje, se realizaron cinco ensayos y se evaluó experimentalmente la probabilidad de falla obtenida.

7.1 Probabilidad de falla para fatiga en vida finita del eje.

7.1.1 Potencia absorbida por el freno

La potencia del motor es de 500 watts a una revolución de 11000 rpm, por medio de la formula,

$$T = \frac{P}{\omega} \quad (52)$$

Se calculó el torque del motor, dando 434.06 N-mm

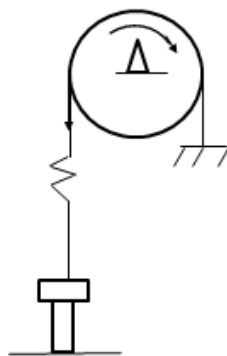


Figura 4. Diagrama de sistema de frenado

El torque se transmite a través del eje hacia el freno, y se calcula la fuerza que ejerce el motor hacia el resorte.

$$F = \frac{T}{R.polea} \quad (53)$$

La fuerza es de 6.2N.

A partir de la fórmula del coeficiente de elasticidad, se calculó la longitud que se debe estirar el resorte. Dando 2.5 mm

7.1.2 Calculo resorte

Se midió la longitud del resorte sin carga, dando 7.8 cm, y se midió la longitud con una carga de 99.612N, dando 11.8cm.

Con la formula,

$$F = -kx \quad (54)$$

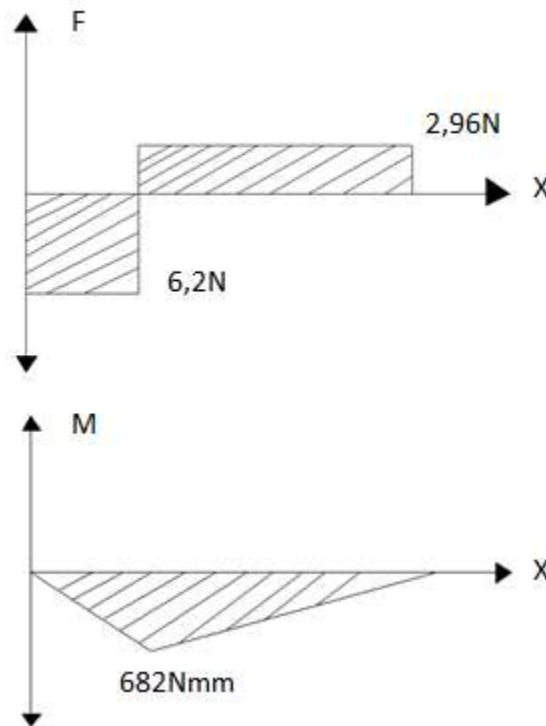
Se calculó el coeficiente de elasticidad del resorte, dando 2.49N/mm



7.1.3 Momento máximo en el eje

Se realizó análisis de cargas, y de momento para determinar el momento flector máximo del eje y en qué punto del eje se daba.

El momento máximo del eje es de 682.1 N-mm y se da a 11cm del lado de la polea. En ese punto diámetro del eje debe de ser el de diseño.



Grafica 5. Diagrama de momento y fuerza

7.1.4 Medición de resistencia ultima del material

Para obtener la resistencia última del material, se realizaron ensayos de micro dureza y para con ello la resistencia última del material.



Los resultados obtenidos de los ensayos de dureza Rockwell C, fueron los siguientes,

Rockwell C	Resistencia ultima (Mpa)
20	770
20	770
20	770
20	770
20	770
15	675
23	820
25	865
25	865

Tabla 1. Dureza del material

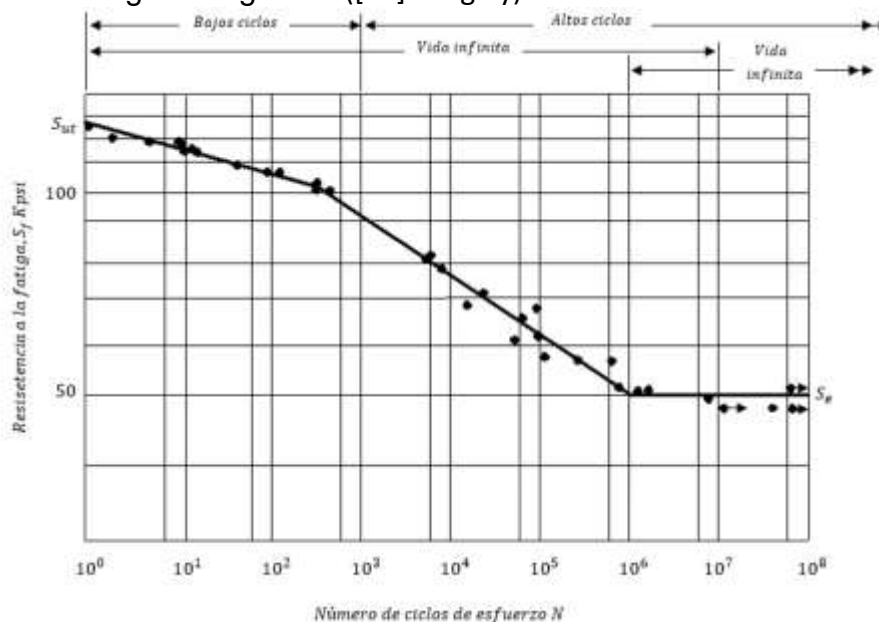
La resistencia última del material fue hallada a partir de la dureza del material⁴. Obteniendo un promedio de resistencia ultima de 786,11Mpa con una desviación estándar de 58,35 Mpa.

7.1.5 Calculo de esfuerzos

A partir del diámetro establecido, momentos flectores y torques se calcularon los esfuerzos de Von Misses alternante y medio. Dando un Von Misses medio de 768,2 Mpa y un alternante de 156,3 Mpa.

7.1.6 Calculo de la resistencia a la fatiga

Por medio de la siguiente gráfica ([13]. Shigley):



Gráfica 6. Numero de ciclos

Se calculó la resistencia a la fatiga, a partir de la resistencia última del material, el número de ciclos establecido anteriormente y el esfuerzo de Von misses, dando 1230Mpa, luego se calculó la resistencia a la fatiga multiplicándola con los factores de Marín dando 294.6Mpa.

⁴ http://ingemecanica.com/tutoriales/tabla_dureza.html

7.1.7 Probabilidad de falla

Finalmente, usando la aplicación⁵, se calculó la probabilidad de falla, aplicando el criterio ASME- Elíptica, dando 0.273.

7.2 Construcción del prototipo

Se realizó montaje de ejes sometidos a cargas de torsión y flexión, para corroborar la teoría estocástica, y con ello, la aplicación elaborada para el diseño de ejes de transmisión de potencia.



Los ejes usados, tienen una longitud de 40 cm, este se encuentra acoplado a un motor de 500 watts a 11000 rpm y a un freno, que absorbe la energía del motor, transmitida el eje. El freno consiste en una polea y una correa la cual tiene un resorte, con el fin de medir la fuerza radial a la cual se somete el eje.



⁵ El código de la aplicación usada para el diseño se encuentra en el "Anexo A. Código en Scilab ensayo" de este documento.

8 RESULTADOS DE LOS ENSAYOS

Se realizaron 5 ensayos, con ejes y montajes de iguales dimensiones, para verificar la probabilidad de falla calculada anteriormente (0.273). El motor que se usó para los ensayos tiene una velocidad de 11000 rpm, entonces se espera que 2 de los cinco ejes presente ruptura antes de los 5000 ciclos.

Ensayo #1

Montaje



Resultado





Se presentó la rotura en un tiempo 10 segundos, correspondiendo a 1833 ciclos.

Ensayo #2

Montaje



Resultado



Luego de un tiempo de 40 para un total de 7333 ciclos el eje no presento rotura pero, se muestra con deflexión.

Ensayo #3

Montaje



Resultado



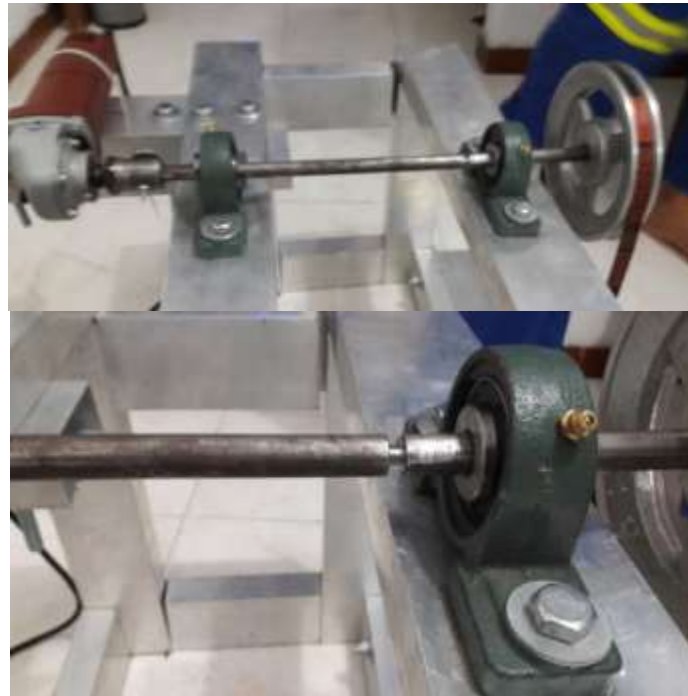
Se presentó la rotura en un tiempo 15 segundos, correspondiendo a 2750 ciclos.

Ensayo #4

Montaje



Resultado



Luego de un tiempo de 30 segundos para un total de 5500 ciclos el eje no presento rotura pero, se muestra con deflexión.

Ensayo #5

Montaje



Resultado



Luego de un tiempo de 50 segundos para un total de 9166 el eje no presento rotura pero, se muestra con deflexión.

Análisis de resultados

Luego de poner a prueba a los ejes, se obtuvieron los resultados esperados, ya que 2 de los 5 ejes ensayados presentaron rotura antes de 5000 ciclos, lo que corresponde a una probabilidad de falla de 0.4. y la probabilidad calculada es de 0.273.

Numero de eje	Tiempo de falla (s)	Numero de ciclos	Tipo de falla
Eje 1	10	1833	Rotura
Eje 2	40	7333	Deflexión
Eje 3	15	2750	Rotura
Eje 4	30	5500	Deflexión
Eje 5	50	9166	Deflexión

Tabla 2. Resultados de ensayos de ejes

El error en la evaluación de la probabilidad de falla es del 33%, lo cual se considera muy satisfactorio, entendiendo la naturaleza estocástica de todo el proceso y del numero de ensayos realizados, quedando como trabajo posterior el desarrollo de prototipos de ensayo que puedan efectuar evaluaciones experimentales en serie para reducir los márgenes de las variables aleatorias.

Con estos resultados de la verificación, tenemos una certeza de la consistencia lógica del proceso de diseño implementado en la aplicación.

9 CONCLUSIONES

Desarrollar este proyecto de grado fue muy importante para nosotros porque nos ayudó a desarrollar la capacidad de investigación y solución de problemas que se nos presentaron para llevar a cabo este proyecto de grado.

Durante el desarrollo de la aplicación fue difícil generar un código que tuviera en cuenta todas las condiciones de carga posibles, ya que para calcular los momentos flectores en cada plano, era necesario analizar los diagramas de carga estática dependiendo de la configuración de los engranes en el eje a diseñar, para con ello calcular el momento flector en cada sección del eje. A pesar de esto se logró una estructura de código que permite variar específicamente las condiciones de ubicación y carga transmitida en ejes escalonados sometidos a las condiciones de carga consideradas.

En el montaje del prototipo, tuvimos inconvenientes para encontrar un sistema que absorba el torque generado por el motor y así generar esfuerzos en el eje. Para ello se usó un sistema de frenado con polea, correa y resorte, con el fin de poder medir dicho torque y forzar el eje. El montaje obtenido fue suficiente para lograr evaluar la probabilidad de falla en un eje transmisor de potencia y de este modo verificar (en condiciones de vida finita por motivos económicos) las predicciones hechas por nuestro código.

La aplicación permite el diseño de ejes de transmisión de potencia con hasta cinco engranajes sobre los mismos, diseñando para confiabilidad cada una de las secciones de dichos ejes usando los criterios de fatiga estocástica de acuerdo con las metodologías indicadas en este documento. Incluir la información de tablas, gráficas, correlaciones y múltiples fuentes de datos para dicho proceso constituye una gran ayuda para los diseñadores, y el uso de la herramienta computacional simplifica el proceso. El uso de análisis estocástico es así mismo una fortaleza innovadora frente a las aplicaciones actuales que utilizan únicamente factor de seguridad.

En el futuro se puede continuar este trabajo diseñando y construyendo montajes experimentales para evaluar fatiga estocástica para vida infinita en ejes. Esto implica la evaluación sistemática de muestras en el rango de 10^6 – 10^8 ciclos con el consecuente costo de tiempo y dinero en dichos ensayos.

De igual modo hacia el futuro la aplicación puede ser extendida incluyendo modelos de función densidad de probabilidad de Weibull para la resistencia del material y el análisis estocástico del cálculo de la confiabilidad según los criterios de falla de fatiga usando variables de Weibull. El presente trabajo se limita a variables Log-Normales.

En resumen, se logró realizar una aplicación para el diseño estocástico de ejes de transmisión de potencia y se corroboró el diseño en la práctica, por medio del montaje de un prototipo. Por lo tanto podemos afirmar con total éxito la realización del proyecto de grado, ya que consideramos haber cumplido de manera adecuada todos los objetivos propuestos.

10 BIBLIOGRAFÍA

- [1]. Scilab 5.5.1, E. (s.f.). Scilab 5.5.1 es un paquete de software libre y de código abierto distribuido con la licencia CeCILL – GPL compatible. Desarrollado por Scilab Enterprises. Disponible desde: <http://www.scilab.org>.
- [10]. Shigley, J. (s.f.). Esfuerzos por carga axial, flexión y torsión. En *Diseño en ingeniería mecánica de Shigley. 9na edición*. (págs. 85-107). Mc Graw Hill.
- [11]. Shigley, J. (s.f.). Límites y ajustes. En *Diseño en ingeniería mecánica de Shigley. 9na edición* (págs. 377-382). Mc Graw Hill.
- [12]. Shigley, J. (s.f.). Teoría de la energía de la distorsión para materiales dúctiles. En *Diseño en ingeniería mecánica de Shigley. 9na edición*. (págs. 212-216). Mc Graw Hill.
- [13]. Shigley, J. (s.f.). Método esfuerzo-vida. En *Diseño en ingeniería mecánica de Shigley. 9na edición*. (págs. 261-263). Mc Graw Hill.
- [14]. Código Scilab 5.5.1, E. (s.f.). Este código ha sido escrito con Scilab 5.5.1 que es un paquete de software libre y de código abierto distribuido con la licencia CeCILL – GPL compatible. Desarrollado por Scilab Enterprises. Disponible desde: <http://www.scilab.org>.
- [2]. Shigley, J. (s.f.). *Diseño en ingeniería mecánica de Shigley*. Mc Graw Hill.
- [3]. Shigley, J. (s.f.). Distribución de probabilidad. En *Diseño en ingeniería mecánica de Shigley. 9na edición*. (págs. 948-957). Mc Graw Hill.
- [4]. Peterson, A. (s.f.). Factor de concentración de esfuerzos.
- [5]. Shigley, J. (s.f.). Esfuerzos fluctuantes y criterios de falla. En *Diseño en ingeniería mecánica de Shigley. 9na edición*. (págs. 286-294). Mc Graw Hill.
- [6]. Shigley, J. (s.f.). Tablas del libro. En *En Diseño en ingeniería mecánica de Shigley. 9na edición* (págs. 985-1022). Mc Graw Hill.
- [7]. Shigley, J. (s.f.). Análisis estocástico. En *Diseño en ingeniería mecánica de Shigley. 9na* (págs. 315-323). Mc Graw Hill.
- [8]. Shigley, J. (s.f.). Análisis de fuerzas en engranes helicoidales. En *Diseño en ingeniería mecánica de Shigley. 9na edición*. (págs. 684-687). Mc Graw Hill.
- [9]. Shigley, J. (s.f.). Fuerza cortante y momentos flectores. En *Diseño en ingeniería mecánica de Shigley. 9na edición*. (págs. 73-77). Mc Graw Hill.

11 ANEXOS

Anexo A. Código en Scilab aplicación y montaje

A continuación se muestra el código en Scilab que se realizó para la aplicación

([14]. Código Scilab 5.5.1):

ScilabPrincipal.sce

```
clear
clear all
//----- No mensajes de alerta por redefinir funciones
funcprot(0);
//----- Memoria para variables y funciones
// (medido en unidades de numeros de doble precision)
stacksize(250000000);
cd "C:\Users\Usuario\Desktop\Jorge\TESIS OK\codigo";

exec('00_FuncionesEstocásticas.sci',-1); // Funciones para
transformaciones estocásticas.
exec('01_TablasPresionInterferencia.sci',-1); //tablas de presion
de interferencia
exec('02_TablaMateriales.sci',-1); // Menu y tabla de los
materiales a escoger

// FACTORES DE MARIN QUE NO DEPENDEN DEL
DIAMETRO
exec('03_SePrima.sci',-1); // Factor de Se prima
exec('04_FactorSuperficie.sci',-1); // Factor de superficie
exec('05_FactorDeCarga.sci',-1); // factores de carga
exec('06_FactorTemperatura.sci',-1); //Factor de temperatura,
grados centigrados
exec('07_FactorConcentracionEsfuerzo.sci',-1); //Factor de
concentracion de esfuerzos

//-----
-----

exec('08_EscogerCriterio.sci',-1); //Menú para escoger criterio
de falla

//-----
-----

c=0
while c==0
c=1
exec('09_CantidadEngranajesFrecuenciaLongitudEje.sci',-1);
//preguntas de engrane, longitud de eje y frecuencia.

if (cant_engrane<1 | cant_engrane>5)
c=0
disp('numero de engranes incorrecto, debe estar entre 1 y 5.
Escriba resume para volver a ver el menu anterior')
pause
end
end
if (cant_engrane == 1) then
exec('10_UnEngrane.sci',-1);

end
```

```
if (cant_engrane == 2) then
exec('11_DosEngranajes.sci',-1);

end

if (cant_engrane == 3) then
exec('12_TresEngranajes.sci',-1);

end

if (cant_engrane == 4) then
exec('13_CuatroEngranajes.sci',-1);

end

if (cant_engrane == 5) then
exec('14_CincoEngranajes.sci',-1);

end
```

00_FuncionesEstocásticas.sci

```
clc
clear
function [media, desv]=suma(a, b)
media=a(1)+b(1)
desv=sqrt((a(2)/a(1))^2+(b(2)/b(1))^2)
endfunction

function [media1, desv1]=resta(a, b)
media1=a(1)-b(1)
desv1=sqrt((a(2)/a(1))^2+(b(2)/b(1))^2)
endfunction

function [media2, desv2]=mult(a, b)
media2=a(1)*b(1)
desv2=a(1)*b(1)*sqrt((a(2)/a(1))^2+(b(2)/b(1))^2+((a(2)/a(1)
)^2)*(b(2)/b(1))^2)
endfunction

function [media3, desv3]=div(a, b)
media3=a(1)/b(1)
desv3=a(1)/b(1)*sqrt(((a(2)/a(1))^2+(b(2)/b(1))^2)/(1+(b(2)/b
(1))^2))
endfunction

function [media4, desv4]=pot(a, n)
media4=(a(1)^n)*(1+((n*(n-1)/2)*((a(2)/a(1))^2)))
desv4=n*a(1)*a(2)/a(1)*1+(((n-1)^2)/4)*((a(2)/a(1))^2))
endfunction

function [media5, desv5]=pot1(a)
media5=sqrt(a(1))*(1-((1/8)*a(2)/a(1))^2)
desv5=((sqrt(a(1))*a(2)/a(1))/2)*(1+((1/16)*a(2)/a(1))^2)
```

```
endfunction
```

```
function [media6, desv6]=caso1(a)//1/x  
media6=(1/a(1))*(1+((a(2)/a(1))^2))  
desv6=((a(2)/a(1))/a(1))*(1+((a(2)/a(1))^2))  
endfunction
```

01_TablasPresionInterferencia.sci

```
clc  
clear all
```

```
//tabla presion por interferencia
```

```
Tabla_13=[0.0002 0.0004 0.0006 0.0010 0.0016 0.0024;  
0.0003 0.0005 0.0007 0.0012 0.0019 0.0030; 0.0004 0.0006  
0.0009 0.0014 0.0023 0.0035; 0.0004 0.0007 0.0011 0.0017  
0.0028 0.0043; 0.0005 0.0008 0.0013 0.0020 0.0033 0.0051;  
0.0006 0.0010 0.0015 0.0024 0.0039 0.0063; 0.0007 0.0012  
0.0018 0.0029 0.0047 0.0075; 0.0009 0.0014 0.0021 0.0034  
0.0055 0.0087; 0.0010 0.0016 0.0025 0.0039 0.0063 0.0098;  
0.0011 0.0018 0.0028 0.0045 0.0073 0.0114; 0.0013 0.0020  
0.0032 0.0051 0.0083 0.0126; 0.0014 0.0022 0.0035 0.0055  
0.0091 0.0142]
```

```
//spu
```

```
tabla14=[0.0002 0.0006 0.0007; 0.0005 0.0007 0.0009; 0.0006  
0.0009 0.0011; 0.0007 0.0011 0.0013; 0.0009 0.0014 0.0016;  
0.0009 0.0014 0.0019; 0.0010 0.0017 0.0024; 0.0010 0.0017  
0.0028; 0.0013 0.0021 0.0034; 0.0013 0.0023 0.0040; 0.0015  
0.0028 0.0049; 0.0015 0.0031 0.0057; 0.0017 0.0036 0.0067;  
0.0017 0.0039 0.0075; 0.0017 0.0043 0.0083; 0.0020 0.0048  
0.0093; 0.0020 0.0051 0.0102; 0.0020 0.0055 0.0112; 0.0022  
0.0062 0.0124; 0.0022 0.0067 0.0130; 0.0024 0.0075 0.0154;  
0.0024 0.0082 0.0171]
```

02_TablaMateriales.sci

```
clc  
clear all
```

```
Tabla_materiales=[87.6 5.74; 86.2 3.92; 117.7 7.13; 83.1 5.25;  
106.5 6.15; 79.6 6.92; 133.4 3.38; 44.5 4.34; 53.3 1.59; 53.4  
2.68; 93.9 3.83; 64.8 3.77; 122.2 7.65; 195.9 7.76; 191.2 5.82;  
105.0 5.68; 85.0 4.14; 84.8 4.23; 105.3 3.09; 198.8 9.51; 149.1  
8.29; 175.4 7.91; 28.1 1.73; 64.9 1.64; 67.5 1.50; 75.5 2.10 ]
```

```
c=0
```

```
while c==0
```

```
c=1
```

```
disp("*****OPCIONES PARA ESCOGER EL  
MATERIAL*****")
```

```
disp("1-Acero 1018 CD")  
disp("2-Acero 1035 HR")  
disp("3-Acero 1045 CD")  
disp("4-Acero 1117 CD")  
disp("5-Acero 1137 CD")  
disp("6-Acero 12L14 CD")  
disp("7-Acero 1038 Pernos HT")  
disp("8-ASTM40")  
disp("9-Acero 35018 Maleable")  
disp("10-Acero 32510 Maleable")  
disp("11-Acero Maleable Perlítico")  
disp("12-Acero 604515 Nodular")
```

```
disp("13-Acero 100-70-04 Nodular")  
disp("14-Acero 201SS CD")  
disp("15-Acero 301SS CD")  
disp("16-Acero 301SS A")  
disp("17-Acero 304SS A")  
disp("18-Acero 310SS A")  
disp("19-Acero 403SS")  
disp("20-Acero 17-7PSS")  
disp("21-Acero AM350SS A")  
disp("22-Ti 6AL 4V")  
disp("23-Acero 2024 0")  
disp("24-Acero 2024 T4")  
disp("25-Acero 2024 T6")  
disp("26-Acero 7075 T6.0.25")
```

```
Material=input('Ingrese la opción del material deseado para el  
eje: ')
```

```
if (Material<1 | Material>26)
```

```
c=0
```

```
disp('Opción incorrecta. Escriba resume para volver a ver el  
menu anterior')
```

```
pause
```

```
end
```

```
end
```

```
coef_variacion_carga=input('Ingrese el coeficiente de  
variación de las cargas (%): ')
```

```
Coef_variacion_carga=coef_variacion_carga/100
```

```
Sut_material_media=0
```

```
Sut_material_desv=0
```

```
if (Material==1)
```

```
Sut_material_media=Tabla_materiales(1,1)
```

```
Sut_material_desv=Tabla_materiales(1,2)
```

```
end
```

```
if (Material==2)
```

```
Sut_material_media=Tabla_materiales(2,1)
```

```
Sut_material_desv=Tabla_materiales(2,2)
```

```
end
```

```
if (Material==3)
```

```
Sut_material_media=Tabla_materiales(3,1)
```

```
Sut_material_desv=Tabla_materiales(3,2)
```

```
end
```

```
if (Material==4)
```

```
Sut_material_media=Tabla_materiales(4,1)
```

```
Sut_material_desv=Tabla_materiales(4,2)
```

```
end
```

```
if (Material==5)
```

```
Sut_material_media=Tabla_materiales(5,1)
```

```
Sut_material_desv=Tabla_materiales(5,2)
```

```
end
```

```
if (Material==6)
```

```
Sut_material_media=Tabla_materiales(6,1)
```

```
Sut_material_desv=Tabla_materiales(6,2)
```

```
end
```

```
if (Material==7)
```

```
Sut_material_media=Tabla_materiales(7,1)
```

```
Sut_material_desv=Tabla_materiales(7,2)
```

```
end
```

```
if (Material==8)
```

```
Sut_material_media=Tabla_materiales(8,1)
```

```
Sut_material_desv=Tabla_materiales(8,2)
```

```
end
```

```
if (Material==9)
```

```
Sut_material_media=Tabla_materiales(9,1)
```

```
Sut_material_desv=Tabla_materiales(9,2)
```

```
end
```

```
if (Material==10)
```

```

    Sut_material_media=Tabla_materiales(10,1)
    Sut_material_desv=Tabla_materiales(10,2)
end
if (Material==11)
    Sut_material_media=Tabla_materiales(11,1)
    Sut_material_desv=Tabla_materiales(11,2)
end
if (Material==12)
    Sut_material_media=Tabla_materiales(12,1)
    Sut_material_desv=Tabla_materiales(12,2)
end
if (Material==13)
    Sut_material_media=Tabla_materiales(13,1)
    Sut_material_desv=Tabla_materiales(13,2)
end
if (Material==14)
    Sut_material_media=Tabla_materiales(14,1)
    Sut_material_desv=Tabla_materiales(14,2)
end
if (Material==15)
    Sut_material_media=Tabla_materiales(15,1)
    Sut_material_desv=Tabla_materiales(15,2)
end
if (Material==16)
    Sut_material_media=Tabla_materiales(16,1)
    Sut_material_desv=Tabla_materiales(16,2)
end

```

03_SePrima.sci

```

clc
clear all

//FACTOR DE SE_PRIMA
Se_prima_tabl=[0.506*Valor_medio_sut*1
0.506*Valor_medio_sut*0.138; 107*Valor_medio_sut*1
0.139*Valor_medio_sut*107]
Se_prima_media=0
Se_prima_desv=0
if (Valor_medio_sut<=212)
    Se_prima_media=Se_prima_tabl(1,1)
    Se_prima_desv=Se_prima_tabl(1,2)
end
if (Valor_medio_sut>=212)
    Se_prima_media=Se_prima_tabl(2,1)
    Se_prima_desv=Se_prima_tabl(2,2)
end
Se_Prima=[Se_prima_media,Se_prima_desv]

```

04_FactorSuperficie.sci

```

clc
clear all

//FACTOR DE SUPERFICIE
acabado_superficial_tabl=[1.34 -0.086 0.120; 2.67 -0.265
0.058; 14.5 -0.719 0.110;39.8 -0.995 0.145 ]

c=0
while c==0
c=1
disp("*****OPCIONES PARA ESCOGER EL ACABADO
SUPERFICIAL*****")

```

```

disp("1-Esmerilado")
disp("2- Maquinado o laminado en frío")
disp("3-Laminado en caliente")
disp("4-Como sale de la forja")
a_superficial=input("Ingrese la opción de acabado superficial
del material deseado:")
if (a_superficial<1 | a_superficial>4)
c=0
disp('Opcion incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end
end
A_superficial_media=0
A_superficial_coeficient=0
if (a_superficial==1)

```

```

A_superficial_media=(acabado_superficial_tabl(1,1))*Valor_
medio_sut^(acabado_superficial_tabl(1,2))
A_superficial_coeficient=acabado_superficial_tabl(1,3)
end
if (a_superficial==2)

```

```

A_superficial_media=(acabado_superficial_tabl(2,1))*Valor_
medio_sut^(acabado_superficial_tabl(2,2))
A_superficial_coeficient=acabado_superficial_tabl(2,3)
end
if (a_superficial==3)

```

```

A_superficial_media=(acabado_superficial_tabl(3,1))*Valor_
medio_sut^(acabado_superficial_tabl(3,2))
A_superficial_coeficient=acabado_superficial_tabl(3,3)
end
if (a_superficial==4)

```

```

A_superficial_media=(acabado_superficial_tabl(4,1))*Valor_
medio_sut^(acabado_superficial_tabl(4,2))
A_superficial_coeficient=acabado_superficial_tabl(4,3)
end
Acabado_superficial=[A_superficial_media,A_superficial_coe
ficient*A_superficial_media]
disp("Valor medio del factor de superficie (Kpsi):")
disp(Acabado_superficial(1))
disp("Desviación estándar del factor de superficie (Kpsi):")
disp(Acabado_superficial(2))

```

05_FactorDeCarga.sci

```

clc
clear all

//FACTOR DE CARGA
f_carga_flexion=1
f_carga_flexion_desv=0
f_carga_axial=1.23*(Valor_medio_sut^(-0.0778))
f_carga_axial_desv=0.125*1.23*(Valor_medio_sut^(-0.0778))
f_carga_torsion=0.328*(Valor_medio_sut^0.125)
f_carga_torsion_desv=0.125*0.328*(Valor_medio_sut^0.125)

```

06_FactorTemperatura.sci

```

clc
clear all

```

```

//FACTOR DE TEMPERATURA-GRADOS CENTÍGRADOS
temperatura_tabla=[1 1.010 1.020 1.025 1.020 1 0.975 0.943
0.9 0.843 0.768 0.672 0.549]
c=0
while c==0
c=1
disp("****OPCIONES PARA LA TEMPERATURA DE
TRABAJO****")
disp("1-20°")
disp("2-50°C")
disp("3-100°C")
disp("4-150°C")
disp("5-200°C")
disp("6-250°C")
disp("7-300°")
disp("8-350°C")
disp("9-400°C")
disp("10-450°C")
disp("11-500°C")

temperatura=input('Ingrese la opción correspondiente a la
temperatura de trabajo (Centígrados): ')
if (temperatura<1 | temperatura>11)
c=0
disp('Opcion incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end
end
f_temperatura=0
if (temperatura==1)
f_temperatura=temperatura_tabla(1)
end
if (temperatura==2)
f_temperatura=temperatura_tabla(2)
end
if (temperatura==3)
f_temperatura=temperatura_tabla(3)
end
if (temperatura==4)
f_temperatura=temperatura_tabla(4)
end
if (temperatura==5)
f_temperatura=temperatura_tabla(5)
end
if (temperatura==6)
f_temperatura=temperatura_tabla(6)
end
if (temperatura==7)
f_temperatura=temperatura_tabla(7)
end
if (temperatura==8)
f_temperatura=temperatura_tabla(8)
end
if (temperatura==9)
f_temperatura=temperatura_tabla(9)
end
if (temperatura==10)
f_temperatura=temperatura_tabla(10)
end
if (temperatura==11)
f_temperatura=temperatura_tabla(11)
end
Factor_temperatura=[f_temperatura,f_temperatura*0.11]
disp('Valor medio del factor de temperatura(°C): ')
disp(Factor_temperatura(1))
disp('Desviación estándar del factor de temperatura(°C): ')

```

```
disp(Factor_temperatura(2))
```

07_FactorConcentracionEsfuerzo.sci

```

clc
clear all

//FACTOR DE CONCENTRACIÓN DE ESFUERZO
tipo_de_muesca_tabl=[5/Sut(1) 4/Sut(1) 3/Sut(1)]
coef_tipo_muesca_tabl=[0.10 0.11 0.15]
c=0
while c==0
c=1
disp("****OPCIONES PARA EL TIPO DE MUESCA
DESEADO****")
disp("1-Agujero transversal")
disp("2-Hombro")
disp("3-Ranura")
raiz_de_a=input('Seleccione el tipo de muesca: ')
if (raiz_de_a<1 | raiz_de_a>3)
c=0
disp('Opción incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end
end
Raiz_de_a=0
Coef_muesca=0
if (raiz_de_a==1)
Raiz_de_a=tipo_de_muesca_tabl(1)
Coef_muesca=coef_tipo_muesca_tabl(1)
end
if (raiz_de_a==2)
Raiz_de_a=tipo_de_muesca_tabl(2)
Coef_muesca=coef_tipo_muesca_tabl(2)
end
if (raiz_de_a==3)
Raiz_de_a=tipo_de_muesca_tabl(3)
Coef_muesca=coef_tipo_muesca_tabl(3)
end
RAIZ_DE_A=Raiz_de_a
COEF_MUESCA=Coef_muesca

```

08_EscogerCriterio.sci

```

clc
clear all

//CRITERIO DE FALLA DESEADO
c=0
while c==0
c=1
disp("****OPCIONES PARA CRITERIOS DE
FALLA****")
disp("1- CRITERIO DE FALLA DE LA ASME-
ELÍPTICA")
disp("2-CRITERIO DE FALLA DE GERBER")
disp("3-CRITERIO DE FALLA DE GOODMAN")
criterio_falla=input('Ingrese la opción correspondiente al
critetio de falla deseado: ')
if (criterio_falla<1 | criterio_falla>3)
c=0
disp('Opción incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end

```

```
end
```

09_CantidadEngranesFrecuenciaLongitudEje.sci

```
clc
clear all

//NÚMERO DE ENGRANES
cant_engrane=input('Ingrese la cantidad de engranes:')
Frecuencia=input('Ingrese la velocidad angular en rpm:')
w_angular=(Frecuencia*2*pi) //Velocidad angular en rad/min
longitud_eje=input('Ingrese la longitud del eje:')
```

10.1_DistanciaSesionesMuecasYUbicacionEngrane.sci

```
clc
clear all

//Datos de entrada

sesion_1=input('Ingrese la distancia de la sesion 1:')
Radio_muesca=input('Ingrese el Radio de la muesca de la
sesión 1:')

c=0
while c==0
c=1
sesion_2=input('Ingrese la distancia de la sesion 2:')
if (sesion_2<=sesion_1 | sesion_2 >= longitud_eje)
c=0
disp('Distancia de la sesión 2 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca3=input('Ingrese el Radio de la muesca de la
sesión 2:')

c=0
while c==0
c=1
sesion_3=input('Ingrese la distancia de la sesion 3:')

if (sesion_3<=sesion_1 | sesion_3 <= sesion_2 | sesion_3
>=longitud_eje)
c=0
disp('Distancia de la sesión 3 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca4=input('Ingrese el Radio de la muesca de la
sesión 3:')

c=0
while c==0
c=1
sesion_4=input('Ingrese la distancia de la sesion 4:')

if (sesion_4<=sesion_1 | sesion_4 <= sesion_2 | sesion_4 <=
sesion_3 | sesion_4>= longitud_eje)
c=0
disp('Distancia de la sesion 4 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca5=input('Ingrese el Radio de la muesca de la
sesión 4:')
```

```
E=30000; //Módulo de elasticidad del acero
```

```
c=0
while c==0
c=1
distancia_1=input('Ingrese la distancia a la que se encuentra
el engrane:')

if (distancia_1<=sesion_2 | distancia_1 >= sesion_3 |
distancia_1 >= longitud_eje)
c=0
disp('La distancia a la que se ubica el engrane no es correcta.
Escriba resume para volver a ingresar el valor')
pause
end
end

potencia_1=input('Ingrese la potencia transmitida al eje:')
Radio_engrane1=input('Ingrese el radio del engrane:')
c=0
while c==0
c=1
disp('***OPCIONES DE LOS PLANOS PARALELOS A
LA POTENCIA TRANSMITIDA POR EL ENGRANE 1***')
disp("1- Plano XY")
disp("2- Plano XZ")
ubicacion_1=input('Ingrese la opción del plano paralelo a la
potencia transmitida por el primer engrane:')
if (ubicacion_1<1 | ubicacion_1>2)
c=0
disp('Opcion incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end
end
angulo_presion1=input('ingrese el angulo de presion del
engrane 1:')
angulo_helice1=input('ingrese el angulo de helice del
engrane 1:')

angulo_tangencial1=atand(tand(angulo_presion1))/cosd(angul
o_helice1)
Vel_lineal1=(Radio_engrane1)/12*w_angular //Velocidad
lineal
W_tangencial1=33000*potencia_1/Vel_lineal1//Fuerza
tangencial

W_radial1=W_tangencial1*(tand(angulo_tangencial1))/Fuerza
a Radial
W_axial1=W_tangencial1*tand(angulo_helice1)//Fuerza
axial

10.10_VidaRemanente.sci
clc
clear all
if sigvma1 > sigvma5 & sigvma1 > sigvma4 & sigvma1 >
sigvma3 & sigvma1 > sigvma2 then
sigvma=sigvma1
end
if sigvma2 > sigvma5 & sigvma2 > sigvma4 & sigvma2 >
sigvma3 & sigvma2 > sigvma1 then
sigvma=sigvma2
end
if sigvma3 > sigvma5 & sigvma3 > sigvma4 & sigvma3 >
sigvma2 & sigvma3 > sigvma1 then
sigvma=sigvma3
end
```

```

if sigvma4 > sigvma5 & sigvma4 > sigvma3 & sigvma4 >
sigvma2 & sigvma4 > sigvma1 then
sigvma=sigvma4
end
if sigvma5 > sigvma4 & sigvma5 > sigvma3 & sigvma5 >
sigvma2 & sigvma5 > sigvma1 then
sigvma=sigvma3
end
SUT=0.9*Sut (1)
m1=(SUT-SE(1))
m2=(3-6)
pend=m1/m2 // calculo de la pendiente
x=((sigvma)-SUT+(pend*3))/pend //valor de x para el esfuerzo
alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)

```

10.2_OperacionesInterferencia.sci

```

clc
clear all
c=0
while c==0
c=1
disp("OPCIONES PARA EL TIPO DE AJUSTE POR
INTERFERENCIA")
disp("1- Ajuste de interferencia localizada H7/P6")
disp("2- Ajuste de impulso medio H7/S6")
disp("3- Ajuste forzado H7/U6")

interferencia = input('Seleccione la opción de ajuste
interferencia deseada para el primer engrane: ')
if (interferencia<1 | interferencia>3)
c=0
disp('Opcion incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end
end

```

10.3_InterferenciaUno.sci

```

clc
clear all
c=0
while c==0
c=1
tamano_basico = input('Ingrese el tamaño básico:')
p=0;
if (tamano_basico<0 | tamano_basico>16)
c=0
disp('Tamano básico incorrecto. Escriba resume para volver
a ingresar el valor.')
pause
end
end
if tamano_basico >=0 & tamano_basico < 0.12 then
p=tabla14(1,1)
end

if tamano_basico >= 0.12 & tamano_basico < 0.24 then
p=tabla14(2,1)
end

if tamano_basico >= 0.24 & tamano_basico < 0.40 then
p=tabla14(3,1)
end

```

```

if tamano_basico >= 0.40 & tamano_basico < 0.72 then
p=tabla14(4,1)
end

if tamano_basico >= 0.72 & tamano_basico < 0.96 then
p=tabla14(5,1)
end

if tamano_basico >= 0.96 & tamano_basico < 1.20 then
p=tabla14(6,1)
end

if tamano_basico >= 1.20 & tamano_basico < 1.60 then
p=tabla14(7,1)
end

if tamano_basico >= 1.60 & tamano_basico < 2.00 then
p=tabla14(8,1)
end

if tamano_basico >= 2.00 & tamano_basico < 2.60 then
p=tabla14(9,1)
end

if tamano_basico >= 2.60 & tamano_basico < 3.20 then
p=tabla14(10,1)
end

if tamano_basico >= 3.20 & tamano_basico < 4.00 then
p=tabla14(11,1)
end

if tamano_basico >= 4.00 & tamano_basico < 4.80 then
p=tabla14(12,1)
end

if tamano_basico >= 4.80 & tamano_basico < 5.60 then
p=tabla14(13,1)
end

if tamano_basico >= 5.60 & tamano_basico < 6.40 then
p=tabla14(14,1)
end

if tamano_basico >= 6.40 & tamano_basico < 7.20 then
p=tabla14(15,1)
end

if tamano_basico >= 7.20 & tamano_basico < 8.00 then
p=tabla14(16,1)
end

if tamano_basico >= 8.00 & tamano_basico < 9.00 then
p=tabla14(17,1)
end

if tamano_basico >= 9.00 & tamano_basico < 10.00 then
p=tabla14(18,1)
end

if tamano_basico >= 10.00 & tamano_basico < 11.20 then
p=tabla14(19,1)
end

if tamano_basico >= 11.20 & tamano_basico < 12.60 then
p=tabla14(20,1)
end

```

```

end

if tamaño_basico >= 12.60 & tamaño_basico < 14.20 then
    p=tabla14(21,1)
end

if tamaño_basico >= 14.20 & tamaño_basico < 16.00 then
    p=tabla14(22,1)
end

it6=0
it7=0

if tamaño_basico>=0 & tamaño_basico<0.12 then
    it6=Tabla_13(1,1)
    it7=Tabla_13(1,2)
end
if tamaño_basico>=0.12 & tamaño_basico<0.24 then
    it6=Tabla_13(2,1)
    it7=Tabla_13(2,2)
end
if tamaño_basico>=0.24 & tamaño_basico<0.40 then
    it6=Tabla_13(3,1)
    it7=Tabla_13(3,2)
end
if tamaño_basico>=0.40 & tamaño_basico<0.72 then
    it6=Tabla_13(4,1)
    it7=Tabla_13(4,2)
end
if tamaño_basico>=0.72 & tamaño_basico<1.20 then
    it6=Tabla_13(5,1)
    it7=Tabla_13(5,2)
end
if tamaño_basico>=1.20 & tamaño_basico<2.00 then
    it6=Tabla_13(6,1)
    it7=Tabla_13(6,2)
end
if tamaño_basico>=2.00 & tamaño_basico<3.20 then
    it6=Tabla_13(7,1)
    it7=Tabla_13(7,2)
end
if tamaño_basico>=3.20 & tamaño_basico<4.80 then
    it6=Tabla_13(8,1)
    it7=Tabla_13(8,2)
end
if tamaño_basico >= 4.80 & tamaño_basico < 7.20 then
    it6=Tabla_13(9,1)
    it7=Tabla_13(9,2)
end
if tamaño_basico >= 7.20 & tamaño_basico < 10.00 then
    it6=Tabla_13(10,1)
    it7=Tabla_13(10,2)
end
if tamaño_basico >= 10.00 & tamaño_basico < 12.60 then
    it6=Tabla_13(11,1)
    it7=Tabla_13(11,2)
end
if tamaño_basico >= 12.60 & tamaño_basico < 16.00 then
    it6=Tabla_13(12,1)
    it7=Tabla_13(12,2)
end

tolerancia_agujero=it7
tolerancia_eje=it6
Diametro_max_agujero=tamaño_basico + tolerancia_agujero
Diametro_min_agujero=tamaño_basico
Diametro_max_eje= tamaño_basico + p + tolerancia_eje

```

```

Diametro_min_eje= tamaño_basico + p

Interf_maza_eje_min=Diametro_min_eje-
Diametro_max_agujero
Interf_maza_eje_max=Diametro_max_eje-
Diametro_min_agujero

```

10.4_InterferenciaDos.sci

```

clc
clear all
c=0
while c==0
    c=1
    tamaño_basico = input('Ingrese el tamaño básico:')
    s=0;
    if (tamaño_basico<0 | tamaño_basico>16)
        c=0
        disp('Tamaño básico incorrecto. Escriba resume para volver
a ingresar el valor.')
        pause
    end
end
if tamaño_basico >=0 & tamaño_basico < 0.12 then
    s=tabla14(1,2)
end

if tamaño_basico >= 0.12 & tamaño_basico < 0.24 then
    s=tabla14(2,2)
end

if tamaño_basico >= 0.24 & tamaño_basico < 0.40 then
    s=tabla14(3,2)
end

if tamaño_basico >= 0.40 & tamaño_basico < 0.72 then
    s=tabla14(4,2)
end

if tamaño_basico >= 0.72 & tamaño_basico < 0.96 then
    s=tabla14(5,2)
end

if tamaño_basico >= 0.96 & tamaño_basico < 1.20 then
    s=tabla14(6,2)
end

if tamaño_basico >= 1.20 & tamaño_basico < 1.60 then
    s=tabla14(7,2)
end

if tamaño_basico >= 1.60 & tamaño_basico < 2.00 then
    s=tabla14(8,2)
end

if tamaño_basico >= 2.00 & tamaño_basico < 2.60 then
    s=tabla14(9,2)
end

if tamaño_basico >= 2.60 & tamaño_basico < 3.20 then
    s=tabla14(10,2)
end

if tamaño_basico >= 3.20 & tamaño_basico < 4.00 then
    s=tabla14(11,2)
end

if tamaño_basico >= 4.00 & tamaño_basico < 4.80 then

```



```

s=tabla14(12,2)
end

if tamaño_basico >= 4.80 & tamaño_basico < 5.60 then
s=tabla14(13,2)
end

if tamaño_basico >= 5.60 & tamaño_basico < 6.40 then
s=tabla14(14,2)
end

if tamaño_basico >= 6.40 & tamaño_basico < 7.20 then
s=tabla14(15,2)
end

if tamaño_basico >= 7.20 & tamaño_basico < 8.00 then
s=tabla14(16,2)
end

if tamaño_basico >= 8.00 & tamaño_basico < 9.00 then
s=tabla14(17,2)
end

if tamaño_basico >= 9.00 & tamaño_basico < 10.00 then
s=tabla14(18,2)
end

if tamaño_basico >= 10.00 & tamaño_basico < 11.20 then
s=tabla14(19,2)
end

if tamaño_basico >= 11.20 & tamaño_basico < 12.60 then
s=tabla14(20,2)
end

if tamaño_basico >= 12.60 & tamaño_basico < 14.20 then
s=tabla14(21,2)
end

if tamaño_basico >= 14.20 & tamaño_basico < 16.00 then
s=tabla14(22,2)
end

it6=0
it7=0

if tamaño_basico>=0 & tamaño_basico<0.12 then
it6=Tabla_13(1,1)
it7=Tabla_13(1,2)
end
if tamaño_basico>=0.12 & tamaño_basico<0.24 then
it6=Tabla_13(2,1)
it7=Tabla_13(2,2)
end
if tamaño_basico>=0.24 & tamaño_basico<0.40 then
it6=Tabla_13(3,1)
it7=Tabla_13(3,2)
end
if tamaño_basico>=0.40 & tamaño_basico<0.72 then
it6=Tabla_13(4,1)
it7=Tabla_13(4,2)
end
if tamaño_basico>=0.72 & tamaño_basico<1.20 then
it6=Tabla_13(5,1)
it7=Tabla_13(5,2)
end
if tamaño_basico>=1.20 & tamaño_basico<2.00 then

```

```

it6=Tabla_13(6,1)
it7=Tabla_13(6,2)
end
if tamaño_basico>=2.00 & tamaño_basico<3.20 then
it6=Tabla_13(7,1)
it7=Tabla_13(7,2)
end
if tamaño_basico>=3.20 & tamaño_basico<4.80 then
it6=Tabla_13(8,1)
it7=Tabla_13(8,2)
end
if tamaño_basico >= 4.80 & tamaño_basico < 7.20 then
it6=Tabla_13(9,1)
it7=Tabla_13(9,2)
end
if tamaño_basico >= 7.20 & tamaño_basico < 10.00 then
it6=Tabla_13(10,1)
it7=Tabla_13(10,2)
end
if tamaño_basico >= 10.00 & tamaño_basico < 12.60 then
it6=Tabla_13(11,1)
it7=Tabla_13(11,2)
end
if tamaño_basico >= 12.60 & tamaño_basico < 16.00 then
it6=Tabla_13(12,1)
it7=Tabla_13(12,2)
end

```

```

tolerancia_agujero=it7
tolerancia_eje=it6
Diametro_max_agujero=tamaño_basico + tolerancia_agujero
Diametro_min_agujero=tamaño_basico
Diametro_max_eje= tamaño_basico + s + tolerancia_eje
Diametro_min_eje= tamaño_basico + s

Interf_maza_eje_min=Diametro_min_eje-
Diametro_max_agujero
Interf_maza_eje_max=Diametro_max_eje-
Diametro_min_agujero

```

10.5 InterferenciaTres.sci

```

clc
clear all
c=0
while c==0
c=1
tamaño_basico = input('Ingrese el tamaño básico:')
u=0;
if (tamaño_basico<0 | tamaño_basico>16)
c=0
disp('Tamaño básico incorrecto. Escriba resume para volver
a ingresar el valor.')
pause
end
end
if tamaño_basico >=0 & tamaño_basico < 0.12 then
u=tabla14(1,3)
end
if tamaño_basico >= 0.12 & tamaño_basico < 0.24 then
u=tabla14(2,3)
end
if tamaño_basico >= 0.24 & tamaño_basico < 0.40 then
u=tabla14(3,3)
end

```

```

if tamaño_basico >= 0.40 & tamaño_basico < 0.72 then
u=tabla14(4,3)
end

if tamaño_basico >= 0.72 & tamaño_basico < 0.96 then
u=tabla14(5,3)
end

if tamaño_basico >= 0.96 & tamaño_basico < 1.20 then
u=tabla14(6,3)
end

if tamaño_basico >= 1.20 & tamaño_basico < 1.60 then
u=tabla14(7,3)
end

if tamaño_basico >= 1.60 & tamaño_basico < 2.00 then
u=tabla14(8,3)
end

if tamaño_basico >= 2.00 & tamaño_basico < 2.60 then
u=tabla14(9,3)
end

if tamaño_basico >= 2.60 & tamaño_basico < 3.20 then
u=tabla14(10,3)
end

if tamaño_basico >= 3.20 & tamaño_basico < 4.00 then
u=tabla14(11,3)
end

if tamaño_basico >= 4.00 & tamaño_basico < 4.80 then
u=tabla14(12,3)
end

if tamaño_basico >= 4.80 & tamaño_basico < 5.60 then
u=tabla14(13,3)
end

if tamaño_basico >= 5.60 & tamaño_basico < 6.40 then
u=tabla14(14,3)
end

if tamaño_basico >= 6.40 & tamaño_basico < 7.20 then
u=tabla14(15,3)
end

if tamaño_basico >= 7.20 & tamaño_basico < 8.00 then
u=tabla14(16,3)
end

if tamaño_basico >= 8.00 & tamaño_basico < 9.00 then
u=tabla14(17,3)
end

if tamaño_basico >= 9.00 & tamaño_basico < 10.00 then
u=tabla14(18,3)
end

if tamaño_basico >= 10.00 & tamaño_basico < 11.20 then
u=tabla14(19,3)
end

if tamaño_basico >= 11.20 & tamaño_basico < 12.60 then
u=tabla14(20,3)
end

```

```

if tamaño_basico >= 12.60 & tamaño_basico < 14.20 then
u=tabla14(21,3)
end

if tamaño_basico >= 14.20 & tamaño_basico < 16.00 then
u=tabla14(22,3)
end

it6=0
it7=0

if tamaño_basico >= 0 & tamaño_basico < 0.12 then
it6=Tabla_13(1,1)
it7=Tabla_13(1,2)
end

if tamaño_basico >= 0.12 & tamaño_basico < 0.24 then
it6=Tabla_13(2,1)
it7=Tabla_13(2,2)
end

if tamaño_basico >= 0.24 & tamaño_basico < 0.40 then
it6=Tabla_13(3,1)
it7=Tabla_13(3,2)
end

if tamaño_basico >= 0.40 & tamaño_basico < 0.72 then
it6=Tabla_13(4,1)
it7=Tabla_13(4,2)
end

if tamaño_basico >= 0.72 & tamaño_basico < 1.20 then
it6=Tabla_13(5,1)
it7=Tabla_13(5,2)
end

if tamaño_basico >= 1.20 & tamaño_basico < 2.00 then
it6=Tabla_13(6,1)
it7=Tabla_13(6,2)
end

if tamaño_basico >= 2.00 & tamaño_basico < 3.20 then
it6=Tabla_13(7,1)
it7=Tabla_13(7,2)
end

if tamaño_basico >= 3.20 & tamaño_basico < 4.80 then
it6=Tabla_13(8,1)
it7=Tabla_13(8,2)
end

if tamaño_basico >= 4.80 & tamaño_basico < 7.20 then
it6=Tabla_13(9,1)
it7=Tabla_13(9,2)
end

if tamaño_basico >= 7.20 & tamaño_basico < 10.00 then
it6=Tabla_13(10,1)
it7=Tabla_13(10,2)
end

if tamaño_basico >= 10.00 & tamaño_basico < 12.60 then
it6=Tabla_13(11,1)
it7=Tabla_13(11,2)
end

if tamaño_basico >= 12.60 & tamaño_basico < 16.00 then
it6=Tabla_13(12,1)
it7=Tabla_13(12,2)
end

tolerancia_agujero=it7
tolerancia_eje=it6
Diametro_max_agujero=tamaño_basico + tolerancia_agujero
Diametro_min_agujero=tamaño_basico
Diametro_max_eje= tamaño_basico + u + tolerancia_eje
Diametro_min_eje= tamaño_basico + u

```

```

Interf_maza_eje_min=Diametro_min_eje-
Diametro_max_agujero
Interf_maza_eje_max=Diametro_max_eje-
Diametro_min_agujero

```

10.7_TorquesMomentosYGraficas.sci

```

//torques
W_tangencial11=W_tangencial1/1000
W_radial11=W_radial1/1000
Torque_1=W_tangencial11*Radio_engrane1

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_max5=0

// momentos
Reaccion_Axy=(W_radial11)-
((W_radial11*distancia_1)/longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(0-Momento_1xy)/(longitud_eje-distancia_1)

for x1=0.0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
if Momento_1XY>Momento_max1 then
Momento_max1=Momento_1XY
Xmax_1=x1
end
end

for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x1-0)+0)
if Momento_2XY>Momento_max2 then
Momento_max2=Momento_2XY
Xmax_2=x2
end
end

for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(x1-0)+0)
if Momento_parte1_3XY>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3XY
Xmax_parte1_3=parte1_x3
end
end

for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
longitud_eje)+0)
if Momento_parte2_3XY>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3XY
Xmax_parte2_3=parte2_x3
end
end

if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end

if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end

if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end

```

```

for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_2xy*(x4-longitud_eje)+0)
if Momento_4XY>Momento_max4 then
Momento_max4=Momento_4XY
Xmax_4=x4
end
end

for x5=sesion_4:0.1:longitud_eje
Momento_5XY=(pendiente_2xy*(x5-longitud_eje)+0)
if Momento_5XY>Momento_max5 then
Momento_max5=Momento_5XY
Xmax_5=x5
end
end

x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then
s=(pendiente_1xy*(x-0)+0)
elseif x>=sesion_1 & x< sesion_2 then
s=(pendiente_1xy*(x-0)+0)
elseif x>=sesion_2 & x< distancia_1 then
s=(pendiente_1xy*(x-0)+0)
elseif x>=distancia_1 & x< sesion_3 then
s=(pendiente_2xy*(parte2_x3-longitud_eje)+0)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_2xy*(x-longitud_eje)+0)
elseif x>=sesion_4 & x < longitud_eje then
s=(pendiente_2xy*(x-longitud_eje)+0)
else
s=0
end
endfunction

```

```
endfunction
```

```

plot(x, MomentoXY1, "r")
title("Diagrama de momento")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]

```

10.8_MultiplicacionKakd.sci

```

clc
clear all

function [media2, desv2]=mult(a, b)
media2=a(1)*b(1)
desv2=a(1)*b(1)*sqrt(((a(2)/a(1))^2+(b(2)/b(1))^2+((a(2)/a(1))^2)*((b(2)/b(1))^2))
endfunction

```

```

[Media_KaKd,Desv_KaKd]=mult(Acabado_superficial,Factor_temperatura)
Ka_Kd=[Media_KaKd,Desv_KaKd]

```

10.9_IteracionesDeLosDiametros.sci

```

clc
clear all

tol=0.001
//Iteraciones de los diámetros
for Diametro_eje1=tol:tol:6

Radio_eje1=Diametro_eje1/2

// inercias y momentos polares
I1=(%pi*((Radio_eje1*2)^4))/4)

```

```

J1=I1*2
sigal=(Momento_max1*Radio_eje1/I1)

//FACTOR DE TAMAÑO

if (Diametro_eje1 < 0.11)
f_tama1=1
end
if (Diametro_eje1 >= 0.11 & Diametro_eje1 <= 2)
f_tama1=0.879*Diametro_eje1^-0.107
end
if (Diametro_eje1 >= 3 & Diametro_eje1 <= 10)
f_tama1=0.91*(Diametro_eje1^(-0.157))
end

Am1=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio//

[Em]=spec(Am1)//Funcion para los autovalores
sigvmm1=(((Em(1)-Em(2))^2)+((Em(2)-Em(3))^2)+((Em(3)-Em(1))^2))/2)^0.5 //Vmisses medio

Aa1=[(sigal*f_carga_flexion/f_tama1) 0 0;0 0 0;0 0 0]
//tensor de esfuerzo alternante

[Ea]=spec(Aa1)//Funcion para los autovalores
sigvma1=(((Ea(1)-Ea(2))^2)+((Ea(2)-Ea(3))^2)+((Ea(3)-Ea(1))^2))/2)^0.5 //Vmisses alternante

V_misses_med_LN1=[sigvmm1,
Coef_variacion_carga*sigvmm1]
V_misses_alt_LN1=[sigvma1,
Coef_variacion_carga*sigvma1]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
D_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

[Div_Va_SE_1_Cuad, Desv_Div_Va_SE_1_Cuad]=
pot(D_Va_SE1,2)
Division_VaSE1 = [Div_Va_SE_1_Cuad,
Desv_Div_Va_SE_1_Cuad]

Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)

ASME_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_ASME_1=ASME_1(2)/ASME_1(1)
CRITERIO_F=ASME_1
CF_CRITERIO_F=Cf_ASME_1
end

if criterio_falla == 2 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_VaSE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

```

```

Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)

GERBER_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GERBER_1=GERBER_1(2)/GERBER_1(1)
CRITERIO_F=GERBER_1
CF_CRITERIO_F=Cf_GERBER_1
end

if criterio_falla == 3 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_Va_SE1(1)+Division_VmSut1(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_Va_SE1(2)/Division_Va_SE1(1))^2+(0)^2)
GOODMAN_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GOODMAN_1=GOODMAN_1(2)/GOODMAN_1(1)
CRITERIO_F=GOODMAN_1
CF_CRITERIO_F=Cf_GOODMAN_1
end

MEDCnormal_1=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_1=CF_CRITERIO_F

[Prob_1,Conf_1]=cdfnor("PQ",0,abs(MEDCnormal_1),DESVnormal_1)

DIAMETRO_EJE1=0

if Prob_1<=1.5*10^-6 & Prob_1>=0.5*10^-6 then
DIAMETRO_EJE1=Diametro_eje1
break;
end
end
disp("El diametro del tramo 1 es")
disp(DIAMETRO_EJE1)

for Diametro_eje2=tol:tol:6
Radio_eje2=Diametro_eje2/2

I2=(%pi*(((Radio_eje2*2)^4))/4)/inercia
J2=I2*2

// calculo de esfuerzos
siga2=(Momento_max2*Radio_eje2/I2) // esfuerzo flector maximo
//FACTOR DE TAMAÑO
if (Diametro_eje2 < 0.11) then
f_tama2=1
end
if (Diametro_eje2 >= 0.11 & Diametro_eje2 <= 2)
f_tama2=0.879*Diametro_eje2^-0.107
end
if (Diametro_eje2 > 2 & Diametro_eje2 <= 10)
f_tama2=0.91*(Diametro_eje2^(-0.157))

```

end

```
//concentracion de esfuerzo  
h2=Radio_eje2-Radio_eje1  
relacion2=h2/Radio_muesca
```

```
//Kt para esfuerzos de flexión (Kf)
```

```
//0.1 <= h/r <= 2.0  
C2_f1=0.947+1.206*sqrt(h2/Radio_muesca)-  
0.131*(h2/Radio_muesca)  
C2_f2=0.022-  
3.405*sqrt(h2/Radio_muesca)+0.915*(h2/Radio_muesca)  
C2_f3=0.869+1.777*sqrt(h2/Radio_muesca)-  
0.555*(h2/Radio_muesca)  
C2_f4=-0.810+0.422*sqrt(h2/Radio_muesca)-  
0.260*(h2/Radio_muesca)
```

```
//2.0 <= h/r <= 20  
C2_f11=1.232+0.832*sqrt(h2/Radio_muesca)-  
0.008*(h2/Radio_muesca)  
C2_f22=-3.813+0.968*sqrt(h2/Radio_muesca)-  
0.260*(h2/Radio_muesca)  
C2_f33=7.423-  
4.868*sqrt(h2/Radio_muesca)+0.869*(h2/Radio_muesca)  
C2_f44=-3.839+3.070*sqrt(h2/Radio_muesca)-  
0.600*(h2/Radio_muesca)
```

```
if (relacion2 >=0.1 & relacion2 <= 2) then
```

```
    C2_F1=C2_f1  
    C2_F2=C2_f2  
    C2_F3=C2_f3  
    C2_F4=C2_f4
```

```
Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/  
Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
```

end

```
if (relacion2 > 2 & relacion2 <= 20) then
```

```
    C2_F1=C2_f11  
    C2_F2=C2_f22  
    C2_F3=C2_f33  
    C2_F4=C2_f44
```

```
Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/  
Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
```

end

```
if (relacion2 < 0.1 ) then
```

```
    Kt2_f=1
```

end

```
if (relacion2 > 0.1 ) then
```

```
    Kt2_f=1
```

end

```
KT2_F=Kt2_f
```

```
//Factor de concentración de esfuerzo de fatiga (Kf_a - axial),  
(Kf_f- flexión), (Kf_t-torsión)
```

```
constante_reemplazo_f2=(2*(KT2_F-  
1)/KT2_F)*(RAIZ_DE_A/sqrt(Radio_muesca))  
Kf2_f=KT2_F/((1+constante_reemplazo_f2))
```

```
Am2=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio  
[Em2]=spec(Am2)//Funcion para los autovalores  
sigvmm2=(((Em2(1)-Em2(2))^2)+((Em2(2)-  
Em2(3))^2)+((Em2(3)-Em2(1))^2))/2^0.5 //Vmises medio
```

```
Aa2=[(sigma2*f_carga_flexion*Kf2_f/f_tama2) 0 0;0 0 0;0 0 0]
```

```
//tensor de esfuerzo alternante
```

```
[Ea2]=spec(Aa2)//Funcion para los autovalores
```

```
sigvma2=(((Ea2(1)-Ea2(2))^2)+((Ea2(2)-  
Ea2(3))^2)+((Ea2(3)-Ea2(1))^2))/2^0.5 //Vmises alternante
```

```
V_misses_med_LN2=[sigvmm2,  
Coef_variacion_carga*sigvmm2]  
V_misses_alt_LN2=[sigvma2,  
Coef_variacion_carga*sigvma2]
```

```
[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)
```

```
SE=[Media_Se, Desv_Se]
```

```
Sut=[Valor_medio_sut,Desv_sut]
```

```
if criterio_falla == 1 then
```

```
    [Div_Va_SE_2,  
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)  
    D_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]
```

```
    [Div_Va_SE_2_Cuad, Desv_Div_Va_SE_2_Cuad]=  
pot(D_Va_SE2,2)  
    Division_VaSE2 = [Div_Va_SE_2_Cuad,  
Desv_Div_Va_SE_2_Cuad]
```

```
Division_VmSut2 = [0,0]
```

```
Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2  
(1)
```

```
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio  
n_VaSE2(1))^2+(0)^2)
```

```
ASME_2=[Suma2_VmSut_VaSE,  
Desv_Suma1_VmSut_VaSE]  
Cf_ASME_2=ASME_2(2)/ASME_2(1)  
CRITERIO_F=ASME_2  
CF_CRITERIO_F=Cf_ASME_2  
end
```

```
if criterio_falla == 2 then
```

```
    [Div_Va_SE_2,  
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)  
    Division_VaSE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]
```

```
Division_VmSut2 = [0,0]
```

```
Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2  
(1)
```

```
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio  
n_VaSE2(1))^2+(0)^2)
```

```
GERBER_2=[Suma2_VmSut_VaSE,  
Desv_Suma2_VmSut_VaSE]  
Cf_GERBER_2=GERBER_2(2)/GERBER_2(1)  
CRITERIO_F=GERBER_2  
CF_CRITERIO_F=Cf_GERBER_2  
end
```

```
if criterio_falla == 3 then
```

```
    [Div_Va_SE_2,  
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)  
    Division_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]
```

```
Division_VmSut2 = [0,0]
```

```

Suma2_VmSut_VaSE=Division_Va_SE2(1)+Division_VmSut
2(1)
Desv_Suma2_VmSut_VaSE=sqrt((Division_Va_SE2(2)/Divisi
on_Va_SE2(1))^2+(0)^2)
GOODMAN_2=[Suma2_VmSut_VaSE,
Desv_Suma2_VmSut_VaSE]
Cf_GOODMAN_2=GOODMAN_2(2)/GOODMAN_2(1)
CRITERIO_F=GOODMAN_2
CF_CRITERIO_F=Cf_GOODMAN_2
end

```

```

MEDCnormal_2=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_2=CF_CRITERIO_F

```

```

[Prob_2,Conf_2]=cdfnorr("PQ",0,abs(MEDCnormal_2),DESV
normal_2)

```

```
DIAMETRO_EJE2=0
```

```

if Prob_2<=1.5*10^-6 & Prob_2>=0.5*10^-6 then
    DIAMETRO_EJE2=Diametro_eje2
    break;
end
end

```

```

disp("El diametro del tramo 2 es")
disp(DIAMETRO_EJE2)

```

```
for Diametro_eje3=tol:tol:6
```

```
Radio_eje3=Diametro_eje3/2 // Radio del eje en el tramo del
engrane 1
```

```

Torque3=(W_tangencial1/1000)*((Radio_engrane1)-
Radio_eje3) // torque generado por el primer engrane
I3=(%pi)*(((Radio_eje3^2)^4)/4)//momento de inercia
J3=I3*2 // momento polar de inercia
// calculo de esfuerzos para tramo 3
sigma3=(Momento_max3*Radio_eje3/I3) // esfuerzo flexor
maximo
taotormed3=(Torque3*Radio_eje3/J3) // esfuerzo cortante
torsor alternante
taotoralt3=(Torque3*Radio_eje3/J3) // esfuerzo cortante torsor
medio

```

```
//FACTOR DE TAMAÑO
```

```

if (Diametro_eje3 < 0.11)
    f_tama3=1
end
if (Diametro_eje3 >= 0.11 & Diametro_eje3 <= 2)
    f_tama3=0.879*Diametro_eje3^-0.107
end
if (Diametro_eje3 >= 3 & Diametro_eje3 <= 10)
    f_tama3=0.91*(Diametro_eje3^(-0.157))
end

```

```
//factor de concentracion de esfuerzo
```

```
h3=Radio_eje3-Radio_eje2
relacion3=h3/Radio_muesca3
```

```
//0.1<=h/r<=2.0
```

```

C3_a1=0.926+(1.157*sqrt(h3/Radio_muesca3))-
(0.099*(h3/Radio_muesca3))
C3_a2=0.012-
3.036*sqrt(h3/Radio_muesca3)+0.961*(h3/Radio_muesca3)

```

```

C3_a3=-0.302+3.977*sqrt(h3/Radio_muesca3)-
1.744*(h3/Radio_muesca3)
C3_a4=0.365-
2.098*sqrt(h3/Radio_muesca3)+0.878*(h3/Radio_muesca3)

```

```
//2.0<=h/r<=20
```

```

C3_a11=1.2+0.860*sqrt(h3/Radio_muesca3)-
0.022*(h3/Radio_muesca3)
C3_a22=-1.805-0.346*sqrt(h3/Radio_muesca3)-
0.038*(h3/Radio_muesca3)
C3_a33=2.198-
0.486*sqrt(h3/Radio_muesca3)+0.165*(h3/Radio_muesca3)
C3_a44=-0.593-0.028*sqrt(h3/Radio_muesca3)-
0.106*(h3/Radio_muesca3)

```

```

if (relacion3 >= 0.1 & relacion3 <= 2) then
    C3_A1=C3_a1
    C3_A2=C3_a2
    C3_A3=C3_a3
    C3_A4=C3_a4

```

```

Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end

```

```
if (relacion3 > 2 & relacion3 <= 20) then
```

```

    C3_A1=C3_a11
    C3_A2=C3_a22
    C3_A3=C3_a33
    C3_A4=C3_a44

```

```

Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end

```

```
if relacion3 < 0.1 then
```

```
    Kt3_a=1
```

```
end
```

```
if relacion3 > 20 then
```

```
    Kt3_a=1
```

```
end
```

```
KT3_A=Kt3_a
```

```
//Kt para esfuerzos de flexión (Ktf)
```

```
//0.1<=h/r<=2.0
```

```

C3_f1=0.947+1.206*sqrt(h3/Radio_muesca3)-
0.131*(h3/Radio_muesca3)
C3_f2=0.022-
3.405*sqrt(h3/Radio_muesca3)+0.915*(h3/Radio_muesca3)
C3_f3=0.869+1.777*sqrt(h3/Radio_muesca3)-
0.555*(h3/Radio_muesca3)
C3_f4=-0.810+0.422*sqrt(h3/Radio_muesca3)-
0.260*(h3/Radio_muesca3)

```

```
//2.0<=h/r<=20
```

```

C3_f11=1.232+0.832*sqrt(h3/Radio_muesca3)-
0.008*(h3/Radio_muesca3)
C3_f22=-3.813+0.968*sqrt(h3/Radio_muesca3)-
0.260*(h3/Radio_muesca3)
C3_f33=7.423-
4.868*sqrt(h3/Radio_muesca3)+0.869*(h3/Radio_muesca3)
C3_f44=-3.839+3.070*sqrt(h3/Radio_muesca3)-
0.600*(h3/Radio_muesca3)

```

```
if (relacion3 >= 0.1 & relacion3 <= 2) then
```

```

C3_F1=C3_f1
C3_F2=C3_f2
C3_F3=C3_f3
C3_F4=C3_f4
Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end

if (relacion3 > 2 & relacion3 <= 20) then
  C3_F1=C3_f11
  C3_F2=C3_f22
  C3_F3=C3_f33
  C3_F4=C3_f44
  Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
  Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.1 then
  Kt3_f=1
end

if relacion3 > 20 then
  Kt3_f=1
end

KT3_F=Kt3_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C3_t1=0.905+0.783*sqrt(h3/Radio_muesca3)-
0.075*(h3/Radio_muesca3)
C3_t2=-0.437-
1.969*sqrt(h3/Radio_muesca3)+0.553*(h3/Radio_muesca3)
C3_t3=1.557+1.073*sqrt(h3/Radio_muesca3)-
0.578*(h3/Radio_muesca3)
C3_t4=-1.061+0.171*sqrt(h3/Radio_muesca3)-
0.086*(h3/Radio_muesca3)

if (relacion3 >=0.25 & relacion3 <= 4.0)
  C3_T1=C3_t1
  C3_T2=C3_t2
  C3_T3=C3_t3
  C3_T4=C3_t4
  Kt3_t=C3_T1+C3_T2*(2*h3/Diametro_eje3)+C3_T3*(2*h3/
  Diametro_eje3)^2+C3_T4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.25 then
  Kt3_t=1
end

KT3_T=Kt3_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f- flexión), (Kf_t-torsión)
constante_reemplazo_a3=(2*(KT3_A-
1)/KT3_A)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf3_a=KT3_A/((1+constante_reemplazo_a3))

constante_reemplazo_f3=(2*(KT3_F-
1)/KT3_F)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf3_f=KT3_F/((1+constante_reemplazo_f3))

constante_reemplazo_t3=(2*(KT3_T-
1)/KT3_T)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf3_t=KT3_T/((1+constante_reemplazo_t3))

```

//presión por interferencia

```

m=(((2*Radio_engrane1)^2-
Diametro_eje3^2))*Diametro_eje3^2))
n=(2*Radio_engrane1)
PP=((E*Interferen_maza_eje_max)/(2*Diametro_eje3^3))*(m/
n)

```

```

Am3=[(PP*f_carga_axial*Kf3_a/f_tama3)
(taotormed3*f_carga_torsion*Kf3_t/f_tama3)
0;(taotormed3*f_carga_torsion*Kf3_t/f_tama3) 0 0;0 0 0]

```

//tensor de esfuerzo medio

//función de autovalores

```

[Em3]=spec(Am3)
sigvmm3=(((Em3(1)-Em3(2))^2)+((Em3(2)-
Em3(3))^2)+((Em3(3)-Em3(1))^2)/2)^0.5//calculo esfuerzo
Von Mises medio

```

```

Aa3=[(sigma3*f_carga_flexion*Kf3_f/f_tama3)
(taotoralt3*f_carga_torsion*Kf3_t/f_tama3) 0;0
(taotoralt3*f_carga_torsion*Kf3_t/f_tama3) 0;0 0 0] //tensor
de esfuerzo alterante

```

[Ea3]=spec(Aa3)

```

sigvma3=(((Ea3(1)-Ea3(2))^2)+((Ea3(2)-
Ea3(3))^2)+((Ea3(3)-Ea3(1))^2)/2)^0.5//calculo esfuerzo Von
Mises medio

```

```

V_misses_med_LN3=[sigvmm3,
Coef_variacion_carga*sigvmm3]
V_misses_alt_LN3=[sigvma3,
Coef_variacion_carga*sigvma3]

```

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

```

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

```

if criterio_falla == 1 then

```

[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
D_Va_SE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

```

```

[Div_Va_SE_3_Cuad, Desv_Div_Va_SE_3_Cuad]=
pot(D_Va_SE3,2)
Division_VaSE3 = [Div_Va_SE_3_Cuad,
Desv_Div_Va_SE_3_Cuad]

```

```

[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

```

```

[Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]

```

```

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)

```

ASME_3=[Suma3_VmSut,Desv_Suma3_VmSut]

Cf_ASME_3=ASME_3(2)/ASME_3(1)

CRITERIO_F=ASME_3

CF_CRITERIO_F=Cf_ASME_3

end

if criterio_falla == 2 then


```

[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

[Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)

GERBER_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GERBER_3=GERBER_3(2)/GERBER_3(1)
CRITERIO_F=GERBER_3
CF_CRITERIO_F=Cf_GERBER_3
end

if criterio_falla == 3 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]
[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
Division_VmSut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)
GOODMAN_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GOODMAN_3=GOODMAN_3(2)/GOODMAN_3(1)
CRITERIO_F=GOODMAN_3
CF_CRITERIO_F=Cf_GOODMAN_3
end

MEDCnormal_3=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_3=CF_CRITERIO_F

[Prob_3,Conf_3]=cdfnor("PQ",0,abs(MEDCnormal_3),DESV
normal_3)

DIAMETRO_EJE3=0

if Prob_3<=1.5*10^-6 & Prob_3>=0.5*10^-6 then
DIAMETRO_EJE3=Diametro_eje3
break;
end
end
disp("El diametro del tramo 3 es ")
disp(DIAMETRO_EJE3)

for Diametro_eje4=tol:tol:6

Radio_eje4=Diametro_eje4/2
Torque4=(W_tangencial1/1000)*((Radio engrane1)-
Radio_eje3)
I4=(%pi*(((Radio_eje4*2)^4))/4)//inercia
J4=I4*2
// calculo de esfuerzos
siga4=(Momento_max4*Radio_eje4/I4) // esfuerzo flector
maximo
taotormed4=(Torque4*Radio_eje4/J4) // esfuerzo cortante
torsor maximo

```

```

taotoralt4=(Torque4*Radio_eje4/J4)

//FACTOR DE TAMAÑO
if (Diametro_eje4 < 0.11)
f_tama4=1
end
if (Diametro_eje4 >= 0.11 & Diametro_eje4 <= 2)
f_tama4=0.879*Diametro_eje4^-0.107
end
if (Diametro_eje4 >= 3 & Diametro_eje4 <= 10)
f_tama4=0.91*(Diametro_eje4^(-0.157))
end
//factor de concentracion de esfuerzo
h4=Radio_eje4-Radio_eje3
relacion4=h4/Radio_muesca4

//Kt para esfuerzos de flexión (Ktf)
//0.1<=h/r<=2.0
C4_f1=0.947+1.206*sqrt(h4/Radio_muesca4)-
0.131*(h4/Radio_muesca4)
C4_f2=0.022-
3.405*sqrt(h4/Radio_muesca4)+0.915*(h4/Radio_muesca4)
C4_f3=0.869+1.777*sqrt(h4/Radio_muesca4)-
0.555*(h4/Radio_muesca4)
C4_f4=-0.810+0.422*sqrt(h4/Radio_muesca4)-
0.260*(h4/Radio_muesca4)

//2.0<=h/r<=20
C4_f11=1.232+0.832*sqrt(h4/Radio_muesca4)-
0.008*(h4/Radio_muesca4)
C4_f22=-3.813+0.968*sqrt(h4/Radio_muesca4)-
0.260*(h4/Radio_muesca4)
C4_f33=7.423-
4.868*sqrt(h4/Radio_muesca4)+0.869*(h4/Radio_muesca4)
C4_f44=-3.839+3.070*sqrt(h4/Radio_muesca4)-
0.600*(h4/Radio_muesca4)

if (relacion4 >=0.1 & relacion4 <= 2)
C4_F1=C4_f1
C4_F2=C4_f2
C4_F3=C4_f3
C4_F4=C4_f4

Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if (relacion4 > 2 & relacion4 <= 20)
C4_F1=C4_f11
C4_F2=C4_f22
C4_F3=C4_f33
C4_F4=C4_f44

Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end
if relacion4 < 0.1 then
Kt4_f=1
end
if relacion4 > 20 then
Kt4_f=1
end

KT4_F=Kt4_f

//Kt para esfuerzos de torsión (Ktf)

```



```

//0.25<=h/r<=4.0
C4_t1=0.905+0.783*sqrt(h4/Radio_muesca4)-
0.075*(h4/Radio_muesca4)
C4_t2=-0.437-
1.969*sqrt(h4/Radio_muesca4)+0.553*(h4/Radio_muesca4)
C4_t3=1.557+1.073*sqrt(h4/Radio_muesca4)-
0.578*(h4/Radio_muesca4)
C4_t4=-1.061+0.171*sqrt(h4/Radio_muesca4)-
0.086*(h4/Radio_muesca4)

if (relacion4 >=0.25 & relacion4 <= 4.0)
  C4_T1=C4_t1
  C4_T2=C4_t2
  C4_T3=C4_t3
  C4_T4=C4_t4
Kt4_t=C4_T1+C4_T2*(2*h4/Diametro_eje4)+C4_T3*(2*h4/
Diametro_eje4)^2+C4_T4*(2*h4/Diametro_eje4)^3
end

if relacion4 < 0.25 then
Kt4_t=1
end
if relacion4 > 4.0 then
Kt4_t=1
end
KT4_T=Kt4_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f- flexión), (Kf_t-torsión)

constante_reemplazo_f4=(2*(KT4_F-
1)/KT4_F)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf4_f=KT4_F/((1+constante_reemplazo_f4))

constante_reemplazo_t4=(2*(KT4_T-
1)/KT4_T)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf4_t=KT4_T/((1+constante_reemplazo_t4))

Am4=[0 (taotormed4*f_carga_torsion*Kf4_t/f_tama4)
0;(taotormed4*f_carga_torsion*Kf4_t/f_tama4) 0 0;0 0 0]
//tensor de esfuerzo medio
[Em4]=spec(Am4)
sigvmm4=(((Em4(1)-Em4(2))^2)+((Em4(2)-
Em4(3))^2)+((Em4(3)-Em4(1))^2))/2)^0.5//Vmises medio

Aa4=[(siga4*f_carga_flexion*Kf4_f/f_tama4)
(taotoralt4*f_carga_torsion*Kf4_t/f_tama4) 0;0
(taotoralt4*f_carga_torsion*Kf4_t/f_tama4) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea4]=spec(Aa4)
sigvma4=(((Ea4(1)-Ea4(2))^2)+((Ea4(2)-
Ea4(3))^2)+((Ea4(3)-Ea4(1))^2))/2)^0.5//Vmises alternante

V_misses_med_LN4=[sigvmm4,
Coef_variacion_carga*sigvmm4]
V_misses_alt_LN4=[sigvma4,
Coef_variacion_carga*sigvma4]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)

```

```

D_Va_SE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

[Div_Va_SE_4_Cuad, Desv_Div_Va_SE_4_Cuad]=
pot(D_Va_SE4,2)
Division_VaSE4 = [Div_Va_SE_4_Cuad,
Desv_Div_Va_SE_4_Cuad]

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Div_Vm_Sut_4_Cuad, Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4 = [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

ASME_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_ASME_4=ASME_4(2)/ASME_4(1)
CRITERIO_F=ASME_4
CF_CRITERIO_F=Cf_ASME_4
end

if criterio_falla == 2 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Div_Vm_Sut_4_Cuad, Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4 = [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

GERBER_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GERBER_4=GERBER_4(2)/GERBER_4(1)
CRITERIO_F=GERBER_4
CF_CRITERIO_F=Cf_GERBER_4
end

if criterio_falla == 3 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]
[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
Division_VmSut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)
GOODMAN_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GOODMAN_4=GOODMAN_4(2)/GOODMAN_4(1)
CRITERIO_F=GOODMAN_4
CF_CRITERIO_F=Cf_GOODMAN_4
end

MEDCnormal_4=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_4=CF_CRITERIO_F

```

```
[Prob_4,Conf_4]=cdfnorr("PQ",0,abs(MEDCnormal_4),DESV  
normal_4)
```

```
DIAMETRO_EJE4=0
```

```
if Prob_4<=1.5*10^-6 & Prob_4>=0.5*10^-6 then  
    DIAMETRO_EJE4=Diametro_eje4  
    break;
```

```
end  
disp("El diametro del tramo 4 es")  
disp(DIAMETRO_EJE4)
```

```
for Diametro_eje5=tol:tol:6
```

```
Radio_eje5=Diametro_eje5/2  
Torque5=(W_tangencial1/1000)*((Radio_engrane1)-  
Radio_eje3)  
I5=(%pi*(((Radio_eje5^2)^5)/4))/inercia  
J5=I5*2  
// calculo de esfuerzos  
sigma5=(Momento_max5*Radio_eje5/I5) // esfuerzo flexor  
maximo  
taotormed5=(Torque5*Radio_eje5/J5) // esfuerzo cortante  
torsor maximo  
taotoralt5=(Torque5*Radio_eje5/J5)
```

```
//FACTOR DE TAMAÑO
```

```
if (Diametro_eje5 < 0.11)  
    f_tama5=1  
end  
if (Diametro_eje5 >= 0.11 & Diametro_eje5 <= 2)  
    f_tama5=0.879*Diametro_eje5^-0.107  
end  
if (Diametro_eje5 >= 3 & Diametro_eje5 <= 10)  
    f_tama5=0.91*(Diametro_eje5^(-0.157))  
end
```

```
//factor de concentracion de esfuerzo
```

```
h5=Radio_eje5-Radio_eje4  
relacion5=h5/Radio_muesca5
```

```
//Kt para esfuerzos de flexión (Ktf)
```

```
//0.1<=h/r<=2.0  
C5_f1=0.947+1.206*sqrt(h5/Radio_muesca5)-  
0.131*(h5/Radio_muesca5)  
C5_f2=0.022-  
3.405*sqrt(h5/Radio_muesca5)+0.915*(h5/Radio_muesca5)  
C5_f3=0.869+1.777*sqrt(h5/Radio_muesca5)-  
0.555*(h5/Radio_muesca5)  
C5_f4=-0.810+0.422*sqrt(h5/Radio_muesca5)-  
0.260*(h5/Radio_muesca5)
```

```
//2.0<=h/r<=20
```

```
C5_f11=1.232+0.832*sqrt(h5/Radio_muesca5)-  
0.008*(h5/Radio_muesca5)  
C5_f22=-3.813+0.968*sqrt(h5/Radio_muesca5)-  
0.260*(h5/Radio_muesca5)  
C5_f33=7.423-  
4.868*sqrt(h5/Radio_muesca5)+0.869*(h5/Radio_muesca5)  
C5_f44=-3.839+3.070*sqrt(h5/Radio_muesca5)-  
0.600*(h5/Radio_muesca5)
```

```
if (relacion5 >=0.1 & relacion5 <= 2)  
    C5_F1=C5_f1
```

```
C5_F2=C5_f2  
C5_F3=C5_f3  
C5_F4=C5_f4
```

```
Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/  
Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3  
end
```

```
if (relacion5 > 2 & relacion5 <= 20)
```

```
C5_F1=C5_f11  
C5_F2=C5_f22  
C5_F3=C5_f33  
C5_F4=C5_f44
```

```
Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/  
Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3  
end
```

```
if relacion5 < 0.1 then
```

```
    Kt5_f=1
```

```
end
```

```
if relacion5 > 20 then
```

```
    Kt5_f=1
```

```
end
```

```
KT5_F = Kt5_f
```

```
//Kt para esfuerzos de torsión (Ktf)
```

```
//0.25<=h/r<=4.0
```

```
C5_t1=0.905+0.783*sqrt(h5/Radio_muesca5)-  
0.075*(h5/Radio_muesca5)  
C5_t2=-0.437-  
1.969*sqrt(h5/Radio_muesca5)+0.553*(h5/Radio_muesca5)  
C5_t3=1.557+1.073*sqrt(h5/Radio_muesca5)-  
0.578*(h5/Radio_muesca5)  
C5_t4=-1.061+0.171*sqrt(h5/Radio_muesca5)-  
0.086*(h5/Radio_muesca5)
```

```
if (relacion5 >=0.25 & relacion5 <= 4.0)
```

```
C5_T1=C5_t1  
C5_T2=C5_t2  
C5_T3=C5_t3  
C5_T4=C5_t4
```

```
Kt5_t=C5_T1+C5_T2*(2*h5/Diametro_eje5)+C5_T3*(2*h5/  
Diametro_eje5)^2+C5_T4*(2*h5/Diametro_eje5)^3  
end
```

```
if relacion5 < 0.25 then
```

```
    Kt5_t=1
```

```
end
```

```
if relacion5 > 4.0 then
```

```
    Kt5_t=1
```

```
end
```

```
KT5_T=Kt5_t
```

```
constante_reemplazo_f5=(2*(KT5_F-  
1)/KT5_F)*(RAIZ_DE_A/sqrt(Radio_muesca5))  
Kf5_f=KT5_F/((1+constante_reemplazo_f5))
```

```
constante_reemplazo_t5=(2*(KT5_T-  
1)/KT5_T)*(RAIZ_DE_A/sqrt(Radio_muesca5))  
Kf5_t=KT5_T/((1+constante_reemplazo_t5))
```

```

Am5=[0 (taotormed5*f_carga_torsion*Kf5_t/f_tama5)
0;(taotormed5*f_carga_torsion*Kf5_t/f_tama5) 0 0;0 0 0]
//tensor de esfuerzo medio
[Em5]=spec(Am5)
sigvmm5=(((Em5(1)-Em5(2))^2)+((Em5(2)-
Em5(3))^2)+((Em5(3)-Em5(1))^2)/2)^0.5

Aa5=[(siga5*f_carga_flexion*Kf5_f/f_tama5)
(taotoralt5*f_carga_torsion*Kf5_t/f_tama5) 0;0
(taotoralt5*f_carga_torsion*Kf5_t/f_tama5) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea5]=spec(Aa5)
sigvma5=(((Ea5(1)-Ea5(2))^2)+((Ea5(2)-
Ea5(3))^2)+((Ea5(3)-Ea5(1))^2)/2)^0.5

V_misses_med_LN5=[sigvmm5,
Coef_variacion_carga*sigvmm5]
V_misses_alt_LN5=[sigvma5,
Coef_variacion_carga*sigvma5]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
D_Va_SE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

[Div_Va_SE_5_Cuad, Desv_Div_Va_SE_5_Cuad]=
pot(D_Va_SE5,2)
Division_VaSE5 = [Div_Va_SE_5_Cuad,
Desv_Div_Va_SE_5_Cuad]

[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]

[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)

ASME_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_ASME_5=ASME_5(2)/ASME_5(1)
CRITERIO_F=ASME_5
CF_CRITERIO_F=Cf_ASME_5
end

if criterio_falla == 2 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]

```

```

[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)

GERBER_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_GERBER_5=GERBER_5(2)/GERBER_5(1)
CRITERIO_F=GERBER_5
CF_CRITERIO_F=Cf_GERBER_5
end

if criterio_falla == 3 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
Division_VmSut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
GOODMAN_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_GOODMAN_5=GOODMAN_5(2)/GOODMAN_5(1)
CRITERIO_F=GOODMAN_5
CF_CRITERIO_F=Cf_GOODMAN_5
end

MEDCnormal_5=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_5=CF_CRITERIO_F

[Prob_5,Conf_5]=cdfnor("PQ",0,abs(MEDCnormal_5),DESV
normal_5)

DIAMETRO_EJE5=0

if Prob_5<=1.5*10^-6 & Prob_5>=0.5*10^-6 then
DIAMETRO_EJE5=Diametro_eje5
break;
end
end
disp("El diametro del tramo 5 es")
disp(DIAMETRO_EJE5)

Sut9=0.9*Sut (1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=(sigvma5-Sut9+(pend*3))/pend //valor de x para el esfuerzo
alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida del eje es :")
disp(ViRe)

```

10_UnEngrane.sci

```

clc
clear all

exec('10.1_DistanciaSesionesMuestrasYUbicacionEngrane.sci',
-1) //entrada de datos de las sesiones y muescas

exec('10.2_OpcionesInterferencia.sci',-1) // entrada del ajuste
por interferencia del usuario

if interferencia==1 then

exec('10.3_InterferenciaUno.sci',-1) //Para interferencia 1
end

```

```

if interferencia==2 then
exec('10.4_InterferenciaDos.sci',-1) //Para interferencia 2
end

if interferencia==3 then
exec('10.5_InterferenciaTres.sci',-1) //Para interferencia 3
end

Interferen_maza_eje_min=Interf_maza_eje_min
Interferen_maza_eje_max=Interf_maza_eje_max
exec('10.7_TorquesMomentosYGraficas.sci',-1)
exec('10.8_MultiplicacionKaKd.sci',-1)
exec('10.9 IteracionesDeLosDiametros.sci',-1)

11. 1. 1_DistanciaFuerzasAnguloUbicacionEngraneUno.sci
clc
clear all

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 1:

c=0
while c==0
c=1
distancia_1=input('Ingrese la distancia a la que se encuentra
el engrane 1:')
if (distancia_1 <=sesion_2 | distancia_1 >= sesion_3)
c=0
disp('La distancia a la que se encuentra el engrane 1 es
incorrecta. Escriba resume para volver a ingresar el valor')
pause
end
end
potencia_1=input('Ingrese la potencia transmitida al engrane
1:')
Radio_engrane1=input('Ingrese el radio del engrane 1:')

c=0
while c==0
c=1
disp('***OPCIONES DE LOS PLANOS PARALELOS A
LA POTENCIA TRANSMITIDA POR EL ENGRANE 1***')
disp("1- Plano XY")
disp("2- Plano XZ")
ubicacion_1=input('Ingrese el plano paralelo a la potencia
transmitida por el primer engrane:')
if (ubicacion_1<1 | ubicacion_1>2)
c=0
disp('Opción incorrecta. Escriba resume para volver a ver el
menú anterior')
pause
end
end

angulo_presion1=input('Ingrese el ángulo de presión del
engrane 1: ')
angulo_helice1=input('Ingrese el ángulo de hélice del
engrane 1: ')

angulo_tangencial1=atand(tand(angulo_presion1))/cosd(angul
o_helice1)
Vel_lineal1=(Radio_engrane1/12)*w_angular //Velocidad
lineal
W_tangencial1=33000*potencia_1/Vel_lineal1//Fuerza
tangencial (Klbf)

```

```

W_radial1=W_tangencial1*(tand(angulo_tangencial1))/Fuerz
a Radial
W_axial1=W_tangencial1*tand(angulo_helice1)//Fuerza
axial

```

```

11. 1. 2_DistanciasFuerzasAnguloUbicacionEngraneDos.sci
clc
clear all
//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 2:
c=0
while c==0
c=1
distancia_2=input('Ingrese la distancia a la que se encuentra
el engrane 2: ')
if (distancia_2 <=sesion_4 | distancia_2 >= sesion_5)
c=0
disp('La distancia a la que se encuentra el engrane 2 es
incorrecta. Escriba resume para volver a ingresar el valor')
pause
end
end
potencia_2=input('Ingrese la potencia transmitida al engrane
2: ')
Radio_engrane2=input('Ingrese el radio del engrane 2: ')
c=0
while c==0
c=1
disp('***OPCIONES DE LOS PLANOS PARALELOS A
LA POTENCIA TRANSMITIDA POR EL ENGRANE 2***')
disp("1- Plano XY")
disp("2- Plano XZ")
ubicacion_2=input('Ingrese el plano paralelo a la potencia
transmitida por el segundo engrane: ')
if (ubicacion_2 < 1 | ubicacion_2 > 2)
c=0
disp('Opción incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end
end
angulo_presion2=input('Ingrese el ángulo de presión del
engrane 2: ')
angulo_helice2=input('Ingrese el ángulo de hélice del
engrane 2: ')

angulo_tangencial2=atand(tand(angulo_presion2))/cosd(angul
o_helice2)
Vel_lineal2=(Radio_engrane2/12)*w_angular //Velocidad
lineal
W_tangencial2=33000*potencia_2/Vel_lineal2//Fuerza
tangencial

W_radial2=W_tangencial2*(tand(angulo_tangencial2))/Fuerz
a Radial
W_axial2=W_tangencial2*tand(angulo_helice2)//Fuerza
axial

11. 1. 3_DistanciasFuerzasAnguloUbicacionEngraneTres.s
ci
clc
clear all
//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 2:

```

```

c=0
while c==0
c=1
distancia_2=input('Ingrese la distancia a la que se encuentra
el engrane 2: ')
if (distancia_2 <= sesion_4 | distancia_2 >= sesion_5)
c=0
disp('La distancia a la que se encuentra el engrane 2 es
incorrecta. Escriba resume para volver a ingresar el valor')
pause
end
end
potencia_2=input('Ingrese la potencia transmitida al engrane
2: ')
Radio_engrane2=input('Ingrese el radio del engrane 2: ')
c=0
while c==0
c=1
disp('***OPCIONES DE LOS PLANOS PARALELOS A
LA POTENCIA TRANSMITIDA POR EL ENGRANE 2***')
disp("1- Plano XY")
disp("2- Plano XZ")
ubicacion_2=input('Ingrese el plano paralelo a la potencia
transmitida por el segundo engrane: ')
if (ubicacion_2 < 1 | ubicacion_2 > 2)
c=0
disp('Opción incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end
end
angulo_presion2=input('Ingrese el ángulo de presión del
engrane 2: ')
angulo_helice2=input('Ingrese el ángulo de hélice del
engrane 2: ')

angulo_tangencial2=atand(tand(angulo_presion2))/cosd(angul
o_helice2)
Vel_lineal2=(Radio_engrane2/12)*w_angular //Velocidad
lineal
W_tangencial2=33000*potencia_2/Vel_lineal2//Fuerza
tangencial

W_radial2=W_tangencial2*(tand(angulo_tangencial2))/Fuerz
a Radial
W_axial2=W_tangencial2*tand(angulo_helice2)//Fuerza
axial

```

11. 4_DistanciasFuerzasAnguloUbicacionEngraneCuatro .sci

```

clc
clear all

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 1:

```

```

c=0
while c==0
c=1
distancia_4=input('Ingrese la distancia a la que se encuentra
el engrane 4: ')
if (distancia_4 <= sesion_8 | distancia_4 >= sesion_9)
c=0

```

```

disp('La distancia a la que se ubica el engrane 4 incorrecta.
Escriba resume para volver a ingresar el valor')

```

```

pause
end
end
potencia_4=input('Ingrese la potencia transmitida al engrane
4: ')

```

```

Radio_engrane4=input('Ingrese el radio del engrane 4: ')
c=0
while c==0
c=1
disp('***OPCIONES DE LOS PLANOS PARALELOS A
LA POTENCIA TRANSMITIDA POR EL ENGRANE 2***')

```

```

disp("1- Plano XY")
disp("2- Plano XZ")
ubicacion_4=input('Ingrese el plano paralelo a la potencia
transmitida por el cuarto engrane: ')

```

```

if (ubicacion_4 < 1 | ubicacion_4 > 2)
c=0
disp('Opción incorrecta. Escriba resume para volver a ver el
menu anterior')

```

```

pause
end
end

```

```

angulo_presion4=input('Ingrese el ángulo de presión del
engrane 4: ')

```

```

angulo_helice4=input('Ingrese el ángulo de hélice del
engrane 4: ')

```

```

angulo_tangencial4=atand(tand(angulo_presion4))/cosd(angul
o_helice4)

```

```

Vel_lineal4=(Radio_engrane4/12)*w_angular //Velocidad
lineal

```

```

W_tangencial4=33000*potencia_4/Vel_lineal4//Fuerza
tangencial

```

```

W_radial4=W_tangencial4*(tand(angulo_tangencial4))/Fuerz
a Radial

```

```

W_axial4=W_tangencial4*tand(angulo_helice4)//Fuerza
axial

```

11. 5_DistanciasFuerzasAnguloUbicacionEngraneCinco .sci

```

clc
clear all

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 1:

```

```

c=0
while c==0
c=1
distancia_5=input('Ingrese la distancia a la que se encuentra
el engrane 5: ')
if (distancia_5 <= sesion_10 | distancia_5 >= sesion_11)
c=0

```

```

disp('La distancia a la que se ubica el engrane 5 es
incorrecta. Escriba resume para volver a ingresar el valor')

```

```

pause
end
end

```

```

potencia_5=input('Ingrese la potencia transmitida al engrane
5: ')

```

```

Radio_engrane5=input('Ingrese el radio del engrane 5: ')

```

```

c=0
while c==0
c=1
disp('***OPCIONES DE LOS PLANOS PARALELOS A
LA POTENCIA TRANSMITIDA POR EL ENGRANE 2***')
disp("1- Plano XY")
disp("2- Plano XZ")
ubicacion_5=input('Ingrese el plano paralelo a la potencia
transmitida por el segundo engrane: ')
if (ubicacion_5 < 1 | ubicacion_5 > 2)
c=0
disp('Opción incorrecta. Escriba resume para volver a ver el
menu anterior')
pause
end
end

```

```

angulo_presion5=input('Ingrese el ángulo de presión del
engrane 5: ')
angulo_helice5=input('Ingrese el ángulo de hélice del
engrane 5: ')

```

```

angulo_tangencial5=atand(tand(angulo_presion5))/cosd(angul
o_helice5)
Vel_lineal5=(Radio_engrane5/12)*w_angular //Velocidad
lineal
W_tangencial5=33000*potencia_5/Vel_lineal5//Fuerza
tangencial

```

```

W_radial5=W_tangencial5*(tand(angulo_tangencial5))//Fuerz
a Radial

```

```

W_axial5=W_tangencial5*tand(angulo_helice5)//Fuerza
axial

```

```

11.
1_1_DistanciasSesionesMuecasYUbicacionEngrane.s
ci

```

```

clc
clear all

```

```

//SESIONES Y MUECAS:

```

```

sesion_1=input('Ingrese la distancia de la sesion 1:')
Radio_muesca=input('Ingrese el radio de la muesca de la
sesión 1: ')

```

```

c=0
while c==0
c=1
sesion_2=input('Ingrese la distancia de la sesion 2:')

```

```

if (sesion_2<=sesion_1 | sesion_2 >= longitud_eje)
c=0
disp('Distancia de la sesion 2 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca2=input('Ingrese el radio de la muesca de la
sesión 2: ')

```

```

c=0
while c==0
c=1
sesion_3=input('Ingrese la distancia de la sesion 3:')
if (sesion_3 <= sesion_1 | sesion_3 <= sesion_2 | sesion_3
>= longitud_eje)

```

```

disp('Distancia de la sesion 3 incorrecta. Escriba resume
para volver a ingresar el valor')

```

```

pause
end
end
Radio_muesca3=input('Ingrese el radio de la muesca de la
sesión 3: ')

```

```

c=0
while c==0
c=1
sesion_4=input('Ingrese la distancia de la sesion 4:')
if (sesion_4 <= sesion_1 | sesion_4 <= sesion_2 | sesion_4
<= sesion_3 | sesion_4 >= longitud_eje)
disp('Distancia de la sesion 4 incorrecta. Escriba resume
para volver a ingresar el valor')

```

```

pause
end
end
Radio_muesca4=input('Ingrese el radio de la muesca de la
sesión 4: ')

```

```

c=0
while c==0
c=1
sesion_5=input('Ingrese la distancia de la sesion 5:')
if (sesion_5 <=sesion_1 | sesion_5 <= sesion_2 | sesion_5
<= sesion_3 | sesion_5 <= sesion_4 | sesion_5>=longitud_eje)
disp('Distancia de la sesion 5 incorrecta. Escriba resume
para volver a ingresar el valor')

```

```

pause
end
end
Radio_muesca5=input('Ingrese el radio de la muesca de la
sesión 5: ')

```

```

c=0
while c==0
c=1
sesion_6=input('Ingrese la distancia de la sesion 6:')
if (sesion_6 <= sesion_1 | sesion_6 <= sesion_2 | sesion_6
<= sesion_3 | sesion_6 <= sesion_4 | sesion_6 <= sesion_5 |
sesion_6>=longitud_eje)
disp('Distancia de la sesion 6 incorrecta. Escriba resume
para volver a ingresar el valor')

```

```

pause
end
end
Radio_muesca6=input('Ingrese el radio de la muesca de la
sesión 6: ')

```

```

//MÓDULO DE ELASTICIDAD DEL ACERO:

```

```

E= 30000

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 1:

```

```

exec('11.1.1_DistanciasFuerzasAnguloUbicacionEngraneUno.
sci',-1)

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 2:

```

```

exec('11.1.2_DistanciasFuerzasAnguloUbicacionEngraneDos.s
ci',-1)

```

```

11.10_VidaRemanate.sci

```

```

clc
clear all

//SESIONES Y MUESCAS:
sesion_1=input('Ingrese la distancia de la sesion 1:')
Radio_muesca=input('Ingrese el radio de la muesca de la
sesión 1:')

c=0
while c==0
c=1
sesion_2=input('Ingrese la distancia de la sesion 2:')

if (sesion_2<=sesion_1 | sesion_2 >= longitud_eje)
c=0
disp('Distancia de la sesion 2 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca2=input('Ingrese el radio de la muesca de la
sesión 2:')

c=0
while c==0
c=1
sesion_3=input('Ingrese la distancia de la sesion 3:')
if (sesion_3 <= sesion_1 | sesion_3 <= sesion_2 | sesion_3
>= longitud_eje)
disp('Distancia de la sesion 3 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca3=input('Ingrese el radio de la muesca de la
sesión 3:')

c=0
while c==0
c=1
sesion_4=input('Ingrese la distancia de la sesion 4:')
if (sesion_4 <= sesion_1 | sesion_4 <= sesion_2 | sesion_4
<= sesion_3 | sesion_4 >= longitud_eje)
disp('Distancia de la sesion 4 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca4=input('Ingrese el radio de la muesca de la
sesión 4:')

c=0
while c==0
c=1
sesion_5=input('Ingrese la distancia de la sesion 5:')
if (sesion_5 <=sesion_1 | sesion_5 <= sesion_2 | sesion_5
<= sesion_3 | sesion_5 <= sesion_4 | sesion_5>=longitud_eje)
disp('Distancia de la sesion 5 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca5=input('Ingrese el radio de la muesca de la
sesión 5:')

c=0

```

```

while c==0
c=1
sesion_6=input('Ingrese la distancia de la sesion 6:')
if (sesion_6 <= sesion_1 | sesion_6 <= sesion_2 | sesion_6
<= sesion_3 | sesion_6 <= sesion_4 | sesion_6 <= sesion_5 |
sesion_6>=longitud_eje)
disp('Distancia de la sesion 6 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca6=input('Ingrese el radio de la muesca de la
sesión 6:')

```

//MÓDULO DE ELASTICIDAD DEL ACERO:

E= 30000

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL ENGRANE 1:

exec('11.1.1_DistanciasFuerzasAnguloUbicacionEngraneUno.
sci',-1)

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL ENGRANE 2:

exec('11.1.2_DistanciasFuerzasAnguloUbicacionEngraneDos.s
ci',-1)

11. 2_OpcionesInterfrecuenciaUno.sci

```

clc
clear all
c=0
while c==0
c=1
disp("OPCIONES PARA EL TIPO DE AJUSTE POR
INTERFERENCIA")
disp("1- Ajuste de interferencia localizada H7/P6")
disp("2- Ajuste de impulso medio H7/S6")
disp("3- Ajuste forzado H7/U6")

```

interferencia = input('Seleccione la opción de ajuste por
interferencia deseada para el primer engrane: ')

```

if (interferencia<1 | interferencia>3)
c=0
disp('Opción incorrecta. Escriba resume para volver a
ingresar el valor')
pause
end
end

```

11. 3_OpcionesInferenciaDos.sci

```

clc
clear all
c=0
while c==0
c=1
disp("OPCIONES PARA EL TIPO DE AJUSTE POR
INTERFERENCIA")
disp("1- Ajuste de interferencia localizada H7/P6")
disp("2- Ajuste de impulso medio H7/S6")
disp("3- Ajuste forzado H7/U6")

```



```
interferencia2 = input('Seleccione la opción de ajuste por interferencia deseada para el segundo engrane: ')
```

```
if (interferencia2<1 | interferencia2>3)
    c=0
    disp('Opción incorrecta. Escriba resume para volver a ingresar el valor')
    pause
end
end
```

11. 4_InterferenciaaUno.sci

```
clc
clear all
c=0
while c==0
c=1
tamano_basico2 = input('Ingrese el tamaño básico:')
p2=0;
if (tamano_basico2<0 | tamano_basico2>16)
    c=0
    disp('Opción de tamaño básico incorrecto. Escriba resume para volver a ver el menu anterior')
    pause
end
end
if tamano_basico2 >=0 & tamano_basico2 < 0.12 then
    p2=tabla14(1,1)
end

if tamano_basico2 >= 0.12 & tamano_basico2 < 0.24 then
    p2=tabla14(2,1)
end

if tamano_basico2 >= 0.24 & tamano_basico2 < 0.40 then
    p2=tabla14(3,1)
end

if tamano_basico2 >= 0.40 & tamano_basico2 < 0.72 then
    p2=tabla14(4,1)
end

if tamano_basico2 >= 0.72 & tamano_basico2 < 0.96 then
    p2=tabla14(5,1)
end

if tamano_basico2 >= 0.96 & tamano_basico2 < 1.20 then
    p2=tabla14(6,1)
end

if tamano_basico2 >= 1.20 & tamano_basico2 < 1.60 then
    p2=tabla14(7,1)
end

if tamano_basico2 >= 1.60 & tamano_basico2 < 2.00 then
    p2=tabla14(8,1)
end

if tamano_basico2 >= 2.00 & tamano_basico2 < 2.60 then
    p2=tabla14(9,1)
end

if tamano_basico2 >= 2.60 & tamano_basico2 < 3.20 then
    p2=tabla14(10,1)
```

```
end
```

```
if tamano_basico2 >= 3.20 & tamano_basico2 < 4.00 then
    p2=tabla14(11,1)
end
```

```
if tamano_basico2 >= 4.00 & tamano_basico2 < 4.80 then
    p2=tabla14(12,1)
end
```

```
if tamano_basico2 >= 4.80 & tamano_basico2 < 5.60 then
    p2=tabla14(13,1)
end
```

```
if tamano_basico2 >= 5.60 & tamano_basico2 < 6.40 then
    p2=tabla14(14,1)
end
```

```
if tamano_basico2 >= 6.40 & tamano_basico2 < 7.20 then
    p2=tabla14(15,1)
end
```

```
if tamano_basico2 >= 7.20 & tamano_basico2 < 8.00 then
    p2=tabla14(16,1)
end
```

```
if tamano_basico2 >= 8.00 & tamano_basico2 < 9.00 then
    p2=tabla14(17,1)
end
```

```
if tamano_basico2 >= 9.00 & tamano_basico2 < 10.00 then
    p2=tabla14(18,1)
end
```

```
if tamano_basico2 >= 10.00 & tamano_basico2 < 11.20 then
    p2=tabla14(19,1)
end
```

```
if tamano_basico2 >= 11.20 & tamano_basico2 < 12.60 then
    p2=tabla14(20,1)
end
```

```
if tamano_basico2 >= 12.60 & tamano_basico2 < 14.20 then
    p2=tabla14(21,1)
end
```

```
if tamano_basico2 >= 14.20 & tamano_basico2 < 16.00 then
    p2=tabla14(22,1)
end
```

```
it6_2=0
it7_2=0
```

```
if tamano_basico2>=0 & tamano_basico2<0.12 then
    it6_2=Tabla_13(1,1)
    it7_2=Tabla_13(1,2)
end
```

```
if tamano_basico2>=0.12 & tamano_basico2<0.24 then
    it6_2=Tabla_13(2,1)
    it7_2=Tabla_13(2,2)
end
```

```
if tamano_basico2>=0.24 & tamano_basico2<0.40 then
    it6_2=Tabla_13(3,1)
    it7_2=Tabla_13(3,2)
end
```

```
if tamano_basico2>=0.40 & tamano_basico2<0.72 then
    it6_2=Tabla_13(4,1)
```



```

    it7_2=Tabla_13(4,2)
end
if tamano_basico2>=0.72 & tamano_basico2<1.20 then
    it6_2=Tabla_13(5,1)
    it7_2=Tabla_13(5,2)
end
if tamano_basico2>=1.20 & tamano_basico2<2.00 then
    it6_2=Tabla_13(6,1)
    it7_2=Tabla_13(6,2)
end
if tamano_basico2>=2.00 & tamano_basico2<3.20 then
    it6_2=Tabla_13(7,1)
    it7_2=Tabla_13(7,2)
end
if tamano_basico2>=3.20 & tamano_basico2<4.80 then
    it6_2=Tabla_13(8,1)
    it7_2=Tabla_13(8,2)
end
if tamano_basico2 >= 4.80 & tamano_basico2 < 7.20 then
    it6_2=Tabla_13(9,1)
    it7_2=Tabla_13(9,2)
end
if tamano_basico2 >= 7.20 & tamano_basico2 < 10.00 then
    it6_2=Tabla_13(10,1)
    it7_2=Tabla_13(10,2)
end
if tamano_basico2 >= 10.00 & tamano_basico2 < 12.60 then
    it6_2=Tabla_13(11,1)
    it7_2=Tabla_13(11,2)
end
if tamano_basico2 >= 12.60 & tamano_basico2 < 16.00 then
    it6_2=Tabla_13(12,1)
    it7_2=Tabla_13(12,2)
end

tolerancia_agujero2 = it7_2
tolerancia_eje2=it6_2
Diametro_max_agujero2=tamano_basico2 +
tolerancia_agujero2
Diametro_min_agujero2=tamano_basico2
Diametro_max_eje2= tamano_basico2 + p2 + tolerancia_eje2
Diametro_min_eje2= tamano_basico2 + p2

Interf_maza_eje_min2=Diametro_min_eje2-
Diametro_max_agujero2
Interf_maza_eje_max2=Diametro_max_eje2-
Diametro_min_agujero2

```

11. 5_InterferenciaDos.sci

```

clc
clear all
c=0
while c==0
c=1
tamano_basico2 = input('Ingrese el tamaño básico:')
s2=0;
if (tamano_basico2<0 | tamano_basico2>16)
c=0
disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
pause
end

```

```

end
if tamano_basico2 >=0 & tamano_basico2 < 0.12 then
s2=tabla14(1,2)
end
if tamano_basico2 >= 0.12 & tamano_basico2 < 0.24 then
s2=tabla14(2,2)
end
if tamano_basico2 >= 0.24 & tamano_basico2 < 0.40 then
s2=tabla14(3,2)
end
if tamano_basico2 >= 0.40 & tamano_basico2 < 0.72 then
s2=tabla14(4,2)
end
if tamano_basico2 >= 0.72 & tamano_basico2 < 0.96 then
s2=tabla14(5,2)
end
if tamano_basico2 >= 0.96 & tamano_basico2 < 1.20 then
s2=tabla14(6,2)
end
if tamano_basico2 >= 1.20 & tamano_basico2 < 1.60 then
s2=tabla14(7,2)
end
if tamano_basico2 >= 1.60 & tamano_basico2 < 2.00 then
s2=tabla14(8,2)
end
if tamano_basico2 >= 2.00 & tamano_basico2 < 2.60 then
s2=tabla14(9,2)
end
if tamano_basico2 >= 2.60 & tamano_basico2 < 3.20 then
s2=tabla14(10,2)
end
if tamano_basico2 >= 3.20 & tamano_basico2 < 4.00 then
s2=tabla14(11,2)
end
if tamano_basico2 >= 4.00 & tamano_basico2 < 4.80 then
s2=tabla14(12,2)
end
if tamano_basico2 >= 4.80 & tamano_basico2 < 5.60 then
s2=tabla14(13,2)
end
if tamano_basico2 >= 5.60 & tamano_basico2 < 6.40 then
s2=tabla14(14,2)
end
if tamano_basico2 >= 6.40 & tamano_basico2 < 7.20 then
s2=tabla14(15,2)
end
if tamano_basico2 >= 7.20 & tamano_basico2 < 8.00 then
s2=tabla14(16,2)
end
if tamano_basico2 >= 8.00 & tamano_basico2 < 9.00 then
s2=tabla14(17,2)

```

```

end

if tamano_basico2 >= 9.00 & tamano_basico2 < 10.00 then
    s2=tabla14(18,2)
end

if tamano_basico2 >= 10.00 & tamano_basico2 < 11.20 then
    s2=tabla14(19,2)
end

if tamano_basico2 >= 11.20 & tamano_basico2 < 12.60 then
    s2=tabla14(20,2)
end

if tamano_basico2 >= 12.60 & tamano_basico2 < 14.20 then
    s2=tabla14(21,2)
end

if tamano_basico2 >= 14.20 & tamano_basico2 < 16.00 then
    s2=tabla14(22,2)
end

it6_2=0
it7_2=0

if tamano_basico2>=0 & tamano_basico2<0.12 then
    it6_2=Tabla_13(1,1)
    it7_2=Tabla_13(1,2)
end
if tamano_basico2>=0.12 & tamano_basico2<0.24 then
    it6_2=Tabla_13(2,1)
    it7_2=Tabla_13(2,2)
end
if tamano_basico2>=0.24 & tamano_basico2<0.40 then
    it6_2=Tabla_13(3,1)
    it7_2=Tabla_13(3,2)
end
if tamano_basico2>=0.40 & tamano_basico2<0.72 then
    it6_2=Tabla_13(4,1)
    it7_2=Tabla_13(4,2)
end
if tamano_basico2>=0.72 & tamano_basico2<1.20 then
    it6_2=Tabla_13(5,1)
    it7_2=Tabla_13(5,2)
end
if tamano_basico2>=1.20 & tamano_basico2<2.00 then
    it6_2=Tabla_13(6,1)
    it7_2=Tabla_13(6,2)
end
if tamano_basico2>=2.00 & tamano_basico2<3.20 then
    it6_2=Tabla_13(7,1)
    it7_2=Tabla_13(7,2)
end
if tamano_basico2>=3.20 & tamano_basico2<4.80 then
    it6_2=Tabla_13(8,1)
    it7_2=Tabla_13(8,2)
end
if tamano_basico2 >= 4.80 & tamano_basico2 < 7.20 then
    it6_2=Tabla_13(9,1)
    it7_2=Tabla_13(9,2)
end
if tamano_basico2 >= 7.20 & tamano_basico2 < 10.00 then
    it6_2=Tabla_13(10,1)
    it7_2=Tabla_13(10,2)
end

```

```

if tamano_basico2 >= 10.00 & tamano_basico2 < 12.60 then
    it6_2=Tabla_13(11,1)
    it7_2=Tabla_13(11,2)
end
if tamano_basico2 >= 12.60 & tamano_basico2 < 16.00 then
    it6_2=Tabla_13(12,1)
    it7_2=Tabla_13(12,2)
end

tolerancia_agujero2 = it7_2
tolerancia_eje2=it6_2
Diametro_max_agujero2=tamano_basico2 +
tolerancia_agujero2
Diametro_min_agujero2=tamano_basico2
Diametro_max_eje2= tamano_basico2 + s2 + tolerancia_eje2
Diametro_min_eje2= tamano_basico2 + s2

Interf_maza_eje_min2=Diametro_min_eje2-
Diametro_max_agujero2
Interf_maza_eje_max2=Diametro_max_eje2-
Diametro_min_agujero2

```

11. 6 InterferenciaTres.sci

```

clc
clear all
c=0
while c==0
    c=1
    tamano_basico2 = input('Ingrese el tamaño básico:')
    u2=0;
    if (tamano_basico2<0 | tamano_basico2>16)
        c=0
        disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
        pause
    end
    if tamano_basico2 >=0 & tamano_basico2 < 0.12 then
        u2=tabla14(1,3)
    end
    if tamano_basico2 >= 0.12 & tamano_basico2 < 0.24 then
        u2=tabla14(2,3)
    end
    if tamano_basico2 >= 0.24 & tamano_basico2 < 0.40 then
        u2=tabla14(3,3)
    end
    if tamano_basico2 >= 0.40 & tamano_basico2 < 0.72 then
        u2=tabla14(4,3)
    end
    if tamano_basico2 >= 0.72 & tamano_basico2 < 0.96 then
        u2=tabla14(5,3)
    end
    if tamano_basico2 >= 0.96 & tamano_basico2 < 1.20 then
        u2=tabla14(6,3)
    end
    if tamano_basico2 >= 1.20 & tamano_basico2 < 1.60 then
        u2=tabla14(7,3)
    end
end

```

```

if tamano_basico2 >= 1.60 & tamano_basico2 < 2.00 then
    u2=tabla14(8,3)
end

if tamano_basico2 >= 2.00 & tamano_basico2 < 2.60 then
    u2=tabla14(9,3)
end

if tamano_basico2 >= 2.60 & tamano_basico2 < 3.20 then
    u2=tabla14(10,3)
end

if tamano_basico2 >= 3.20 & tamano_basico2 < 4.00 then
    u2=tabla14(11,3)
end

if tamano_basico2 >= 4.00 & tamano_basico2 < 4.80 then
    u2=tabla14(12,3)
end

if tamano_basico2 >= 4.80 & tamano_basico2 < 5.60 then
    u2=tabla14(13,3)
end

if tamano_basico2 >= 5.60 & tamano_basico2 < 6.40 then
    u2=tabla14(14,3)
end

if tamano_basico2 >= 6.40 & tamano_basico2 < 7.20 then
    u2=tabla14(15,3)
end

if tamano_basico2 >= 7.20 & tamano_basico2 < 8.00 then
    u2=tabla14(16,3)
end

if tamano_basico2 >= 8.00 & tamano_basico2 < 9.00 then
    u2=tabla14(17,3)
end

if tamano_basico2 >= 9.00 & tamano_basico2 < 10.00 then
    u2=tabla14(18,3)
end

if tamano_basico2 >= 10.00 & tamano_basico2 < 11.20 then
    u2=tabla14(19,3)
end

if tamano_basico2 >= 11.20 & tamano_basico2 < 12.60 then
    u2=tabla14(20,3)
end

if tamano_basico2 >= 12.60 & tamano_basico2 < 14.20 then
    u2=tabla14(21,3)
end

if tamano_basico2 >= 14.20 & tamano_basico2 < 16.00 then
    u2=tabla14(22,3)
end

it6_2=0
it7_2=0

if tamano_basico2>=0 & tamano_basico2<0.12 then
    it6_2=Tabla_13(1,1)
    it7_2=Tabla_13(1,2)

```

```

end
if tamano_basico2>=0.12 & tamano_basico2<0.24 then
    it6_2=Tabla_13(2,1)
    it7_2=Tabla_13(2,2)
end
if tamano_basico2>=0.24 & tamano_basico2<0.40 then
    it6_2=Tabla_13(3,1)
    it7_2=Tabla_13(3,2)
end
if tamano_basico2>=0.40 & tamano_basico2<0.72 then
    it6_2=Tabla_13(4,1)
    it7_2=Tabla_13(4,2)
end
if tamano_basico2>=0.72 & tamano_basico2<1.20 then
    it6_2=Tabla_13(5,1)
    it7_2=Tabla_13(5,2)
end
if tamano_basico2>=1.20 & tamano_basico2<2.00 then
    it6_2=Tabla_13(6,1)
    it7_2=Tabla_13(6,2)
end
if tamano_basico2>=2.00 & tamano_basico2<3.20 then
    it6_2=Tabla_13(7,1)
    it7_2=Tabla_13(7,2)
end
if tamano_basico2>=3.20 & tamano_basico2<4.80 then
    it6_2=Tabla_13(8,1)
    it7_2=Tabla_13(8,2)
end
if tamano_basico2 >= 4.80 & tamano_basico2 < 7.20 then
    it6_2=Tabla_13(9,1)
    it7_2=Tabla_13(9,2)
end
if tamano_basico2 >= 7.20 & tamano_basico2 < 10.00 then
    it6_2=Tabla_13(10,1)
    it7_2=Tabla_13(10,2)
end
if tamano_basico2 >= 10.00 & tamano_basico2 < 12.60 then
    it6_2=Tabla_13(11,1)
    it7_2=Tabla_13(11,2)
end
if tamano_basico2 >= 12.60 & tamano_basico2 < 16.00 then
    it6_2=Tabla_13(12,1)
    it7_2=Tabla_13(12,2)
end

tolerancia_agujero2 = it7_2
tolerancia_eje2=it6_2
Diametro_max_agujero2=tamano_basico2 +
tolerancia_agujero2
Diametro_min_agujero2=tamano_basico2
Diametro_max_eje2= tamano_basico2 + u2 + tolerancia_eje2
Diametro_min_eje2= tamano_basico2 + u2

Interf_maza_eje_min2=Diametro_min_eje2-
Diametro_max_agujero2
Interf_maza_eje_max2=Diametro_max_eje2-
Diametro_min_agujero2

```

11.7_TorquesMomentosYGraficas.sci

```

clc
clear all

W_tangencial11=W_tangencial1/1000

```

```

W_radial11=W_radial1/1000
W_tangencial22=W_tangencial2/1000
W_radial22=W_radial2/1000
//Reacciones y momentos
if (ubicacion_1 == ubicacion_2) then

Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial22*lon
gitud_eje)-(W_radial11*distancia_1)-
(W_radial22*distancia_2))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
pendiente_3xy=(0-Momento_2xy)/(longitud_eje-
distancia_2)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_max7=0

for x1=0:0.1:sesion_1
Momento_1XY=pendiente_1xy*(x1-0)+0
if Momento_1XY>Momento_max1 then
Momento_max1=Momento_1XY
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=pendiente_1xy*(x2-0)+0
if Momento_2XY>Momento_max2 then
Momento_max2=Momento_2XY
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=pendiente_1xy*(parte1_x3-0)+0
if Momento_parte1_3XY>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3XY
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=pendiente_2xy*(parte1_x3-
distancia_1)+Momento_1xy
if Momento_parte2_3XY > Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3XY
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3

```

```

Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=pendiente_2xy*(x4-
distancia_1)+Momento_1xy
if Momento_4XY>Momento_max4 then
Momento_max4=Momento_4XY
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy
if Momento_parte1_5XY>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5XY
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=pendiente_3xy*(parte2_x5-
longitud_eje)+0
if Momento_parte2_5XY>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5XY
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=pendiente_3xy*(x6-longitud_eje)+0
if Momento_6XY>Momento_max6 then
Momento_max6=Momento_6XY
Xmax_6=x6
end
end
for x7=sesion_6:0.1:longitud_eje
Momento_7XY=pendiente_3xy*(x7-longitud_eje)+0
if Momento_7XY>Momento_max7 then
Momento_max7=Momento_7XY
Xmax_7=x7
end
end
if (ubicacion_1 == ubicacion_2)
x=0:0.1:longitud_eje
function s=Momento_XY(x)
if x>=0 & x < sesion_1 then
s=(pendiente_1xy*(x-0)+0)
elseif x>=sesion_1 & x < sesion_2 then
s=(pendiente_1xy*(x-0)+0)
elseif x>=sesion_2 & x < distancia_1 then
s=(pendiente_1xy*(x-0)+0)
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_2xy*(x-distancia_1)+Momento_1xy
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_2xy*(x-distancia_1)+Momento_1xy
elseif x>=sesion_4 & x < distancia_2 then

```

```

    s= pendiente_2xy*(x-distancia_1)+Momento_1xy
elseif x>=distancia_2 & x < sesion_5 then
    s=pendiente_3xy*(x-longitud_eje)+0
elseif x>=sesion_5 & x < sesion_6 then
    s=pendiente_3xy*(x-longitud_eje)+0
elseif x>=sesion_6 & x < longitud_eje then
    s=pendiente_3xy*(x-longitud_eje)+0
else
    s=0
end
endfunction
plot(x,Momento_XY,"r")
title("Diagrama de momento")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
end
if (ubicacion_1 <> ubicacion_2) then
    Reaccion_Axy=W_radial11-
    ((W_radial11*distancia_1)/longitud_eje)
    Momento_1xy=Reaccion_Axy*distancia_1
    Reaccion_Axz=W_radial22-
    ((W_radial22*distancia_2)/longitud_eje)
    Momento_1xz=Reaccion_Axz*distancia_2
    pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
    pendiente_2xy=(0-Momento_1xy)/(longitud_eje-
    distancia_1)
    pendiente_1xz=(Momento_1xz-0)/(distancia_2-0)
    pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
    distancia_2)

    Momento_max1=0
    Momento_max2=0
    Momento_parte1_max3=0
    Momento_parte2_max3=0
    Momento_max4=0
    Momento_parte1_max5=0
    Momento_parte2_max5=0
    Momento_max6=0
    Momento_max7=0

    for x1=0.0:0.1:sesion_1
        Momento_1XY=(pendiente_1xy*(x1-0)+0)
        Momento_1XZ=(pendiente_1xz*(x1-0)+0)
    end
    Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    if Momento_1>Momento_max1 then
        Momento_max1=Momento_1
        Xmax_1=x1
    end
end
for x2=sesion_1:0.1:sesion_2
    Momento_2XY=(pendiente_1xy*(x2-0)+0)
    Momento_2XZ=(pendiente_1xz*(x2-0)+0)
end
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
    Momento_max2=Momento_2
    Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
    Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
    0)+0)
    Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
    Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
    ento_parte1_3XZ)^2))
    if Momento_parte1_3>Momento_parte1_max3 then
        Momento_parte1_max3=Momento_parte1_3
        Xmax_parte1_3=parte1_x3
    end
end
for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
    longitud_eje)+0)
    Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
    Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
    ento_parte2_3XZ)^2))
    if Momento_parte2_3>Momento_parte2_max3 then
        Momento_parte2_max3=Momento_parte2_3
        Xmax_parte2_3=parte2_x3
    end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_2xy*(x4-longitud_eje)+0)
    Momento_4XZ=(pendiente_1xz*(x4-0)+0)
end
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
    Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
    longitud_eje)+0)
    Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
    Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
    ento_parte1_5XZ)^2))
    if Momento_parte1_5>Momento_parte1_max5 then
        Momento_parte1_max5=Momento_parte1_5
        Xmax_parte1_5=parte1_x5
    end
end
end
for parte2_x5=distancia_2:0.1:sesion_5
    Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
    longitud_eje)+0)
    Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
    longitud_eje)+0)
    Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
    ento_parte2_5XZ)^2))
    if Momento_parte2_5>Momento_parte2_max5 then
        Momento_parte2_max5=Momento_parte2_5
        Xmax_parte2_5=parte2_x5
    end
end
end

```

```

        if Momento_parte2_max5 > Momento_parte1_max5 then
            Momento_max5=Momento_parte2_max5
            Xmax_5= Xmax_parte2_5
        end
        if Momento_parte1_max5 > Momento_parte2_max5 then
            Momento_max5=Momento_parte1_max5
            Xmax_5=Xmax_parte1_5
        end
        if Momento_parte1_max5 == Momento_parte2_max5 then
            Momento_max5=Momento_parte1_max5
            Xmax_5=Xmax_parte1_5
        end
        for x6=sesion_5:0.1:sesion_6
            Momento_6XY=(pendiente_2xy*(x6-longitud_eje)+0)
            Momento_6XZ=(pendiente_2xz*(x6-longitud_eje)+0)
        end
        Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
        if Momento_6>Momento_max6 then
            Momento_max6=Momento_6
            Xmax_6=x6
        end
        for x7=sesion_6:0.1:longitud_eje
            Momento_7XY=(pendiente_2xy*(x7-longitud_eje)+0)
            Momento_7XZ=(pendiente_2xz*(x7-longitud_eje)+0)
        end
        Momento_7=sqrt(((Momento_7XY)^2)+((Momento_7XZ)^2))
        if Momento_7>Momento_max7 then
            Momento_max7=Momento_7
            Xmax_7=x7
        end
    end
end
if (ubicacion_1 <> ubicacion_2)
    x=0:0.0:longitud_eje
    function s=MomentoXY(x)
        if x>=0 & x < sesion_1 then
            s=(pendiente_1xy*(x-0)+0)
        elseif x>=sesion_1 & x < sesion_2 then
            s=(pendiente_1xy*(x-0)+0)
        elseif x>=sesion_2 & x < distancia_1 then
            s=(pendiente_1xy*(x-0)+0)
        elseif x>=distancia_1 & x < sesion_3 then
            s=(pendiente_2xy*(x-longitud_eje)+0)
        elseif x>=sesion_3 & x < sesion_4 then
            s=(pendiente_2xy*(x-longitud_eje)+0)
        elseif x>=sesion_4 & x < distancia_2 then
            s=(pendiente_2xy*(x-longitud_eje)+0)
        elseif x>=distancia_2 & x < sesion_5 then
            s=(pendiente_2xy*(x-longitud_eje)+0)
        elseif x>=sesion_5 & x < sesion_6 then
            s=(pendiente_2xy*(x-longitud_eje)+0)
        elseif x>=sesion_6 & x < longitud_eje then
            s=(pendiente_2xy*(x-longitud_eje)+0)
        else
            s=0
        end
    end
endfunction
function s=MomentoXZ(x)
    if x>=0 & x < sesion_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_1 & x < sesion_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_2 & x < distancia_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_1xz*(x-0)+0)
    end
end

```

```

        elseif x>=sesion_3 & x < sesion_4 then
            s=(pendiente_1xz*(x-0)+0)
        elseif x>=sesion_4 & x < distancia_2 then
            s=(pendiente_1xz*(x-0)+0)
        elseif x>=distancia_2 & x < sesion_5 then
            s=(pendiente_2xz*(x-longitud_eje)+0)
        elseif x>=sesion_5 & x < sesion_6 then
            s=(pendiente_2xz*(x-longitud_eje)+0)
        elseif x>=sesion_6 & x < longitud_eje then
            s=(pendiente_2xz*(x-longitud_eje)+0)
        else
            s=0
        end
    end
endfunction

function s=MomentoResultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then
        s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
    elseif x>=distancia_1 & x < sesion_3 then
        s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
    elseif x>=sesion_3 & x < sesion_4 then
        s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x < distancia_2 then
        s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
    elseif x>=distancia_2 & x < sesion_5 then
        s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < longitud_eje then
        s=sqrt(((Momento_7XY)^2)+((Momento_7XZ)^2))
    else
        s=0
    end
endfunction

plot(x,MomentoXY,"r")
plot(x,MomentoXZ,"g")
plot(x,MomentoResultante,"b")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
end

```

11. 9 IteracionesDeLosDiametros.sci

```

clc
clear all

//MULTIPLICACIÓN DE SE_PRIMA POR Ka POR Kd:
[Media_Se,Desv_Se]=mult(Se_Prima,Ka,Kd)

```

```

SE=[Media_Se, Desv_Se]

//VALOR MEDIO Y DESVIACIÓN ESTÁNDAR DE LA
RESISTENCIA ÚLTIMA:
Sut=[Valor_medio_sut,Desv_sut]

//TOLERANCIA DE ITERACIÓN:
tol=0.001
//ITERACIÓN DEL PRIMER DIÁMETRO
for Diametro_eje1=tol:tol:6

Radio_eje1=Diametro_eje1/2

//MOMENTO DE INERCIA:
I1=(%pi*((Radio_eje1*2)^4))/4
//MOMENTO POLAR DE INERCIA:
J1=I1*2

sigal=(Momento_max1*Radio_eje1/I1)

//FACTOR DE TAMAÑO:
if (Diametro_eje1 < 0.11)
f_tama1=1
end
if (Diametro_eje1 >= 0.11 & Diametro_eje1 <= 2)
f_tama1=0.879*Diametro_eje1^-0.107
end
if (Diametro_eje1 >= 3 & Diametro_eje1 <= 10)
f_tama1=0.91*(Diametro_eje1^(-0.157))
end
//TENSOR DE ESFUERZOS MEDIOS:
Am1=[0 0 0;0 0 0;0 0 0]
//FUNCIÓN PARA CALCULAR LOS AUTOVALORES:
[Em]=spec(Am1)
//CRITERIO DE VON-MISSES:
sigvmm1=(((Em(1)-Em(2))^2)+((Em(2)-Em(3))^2)+((Em(3)-Em(1))^2))/2^0.5
V_misses_med_LN1=[sigvmm1,
Coef_variacion_carga*sigvmm1]

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa1=(sigal*f_carga_flexion/f_tama1) 0 0 0;0 0 0;0 0 0]
//FUNCIÓN PARA CALCULAR LOS AUTOVALORES:
[Ea]=spec(Aa1)
//CRITERIO DE VON-MISSES:
sigvma1=(((Ea(1)-Ea(2))^2)+((Ea(2)-Ea(3))^2)+((Ea(3)-Ea(1))^2))/2^0.5
V_misses_alt_LN1=[sigvma1,
Coef_variacion_carga*sigvma1]

//EVALUACIÓN DE LOS CRITERIOS DE FALLA
ESCOGIDOS:
if criterio_falla == 1 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
D_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

[Div_Va_SE_1_Cuad,          Desv_Div_Va_SE_1_Cuad]=
pot(D_Va_SE1,2)
Division_VaSE1          =          [Div_Va_SE_1_Cuad,
Desv_Div_Va_SE_1_Cuad]

Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1
(1)

```

```

Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)

ASME_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_ASME_1=ASME_1(2)/ASME_1(1)
CRITERIO_F=ASME_1
CF_CRITERIO_F=Cf_ASME_1
end

if criterio_falla == 2 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_VaSE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1
(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)

GERBER_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GERBER_1=GERBER_1(2)/GERBER_1(1)
CRITERIO_F=GERBER_1
CF_CRITERIO_F=Cf_GERBER_1
end

if criterio_falla == 3 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_Va_SE1(1)+Division_VmSut1
(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_Va_SE1(2)/Division_Va_SE1(1))^2+(0)^2)
GOODMAN_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GOODMAN_1=GOODMAN_1(2)/GOODMAN_1(1)
CRITERIO_F=GOODMAN_1
CF_CRITERIO_F=Cf_GOODMAN_1
end

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL:
MEDCnormal_1=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_1=CF_CRITERIO_F

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_1,Conf_1]=cdfnor("PQ",0,abs(MEDCnormal_1),DESVnormal_1)

DIAMETRO_EJE1=0

if Prob_1<=1.5*10^-6 & Prob_1>=0.5*10^-6 then
DIAMETRO_EJE1=Diametro_eje1
break;
end
end
disp("El diametro del tramo 1 es")
disp(DIAMETRO_EJE1)

```



```

//ITERACIÓN DEL SEGUNDO DIÁMETRO
for Diametro_eje2=tol.tol:6

Radio_eje2=Diametro_eje2/2

//MOMENTO DE INERCIA:
I2=(%pi*((Radio_eje2^2)^4))/4
//MOMENTO POLAR DE INERCIA:
J2=I2*2

// CÁLCULO DE ESFUERZOS:
//ESFUERZO FLECTOR MÁXIMO:
sigma2=(Momento_max2*Radio_eje2/I2)

//FACTOR DE TAMAÑO
if (Diametro_eje2 < 0.11)
  f_tama2=1
end
if (Diametro_eje2 >= 0.11 & Diametro_eje2 <= 2)
  f_tama2=0.879*Diametro_eje2^-0.107
end
if (Diametro_eje2 >= 3 & Diametro_eje2 <= 10)
  f_tama2=0.91*(Diametro_eje2^(-0.157))
end

//CONCENTRACIÓN DE ESFUERZO:
h2=Radio_eje2-Radio_eje1
relacion2=h2/Radio_muesca

//FACTOR DE CONCENTRACIÓN DE ESFUERZO DE
PETERSON:

//CONSTANTES PARA CONCENTRADOR DE ESFUERZO
PARA FLEXIÓN:

//0.1 <= h/r <= 2.0
C2_f1=0.947+1.206*sqrt(h2/Radio_muesca)-
0.131*(h2/Radio_muesca)
C2_f2=0.022-
3.405*sqrt(h2/Radio_muesca)+0.915*(h2/Radio_muesca)
C2_f3=0.869+1.777*sqrt(h2/Radio_muesca)-
0.555*(h2/Radio_muesca)
C2_f4=-0.810+0.422*sqrt(h2/Radio_muesca)-
0.260*(h2/Radio_muesca)

//2.0 <= h/r <= 20
C2_f11=1.232+0.832*sqrt(h2/Radio_muesca)-
0.008*(h2/Radio_muesca)
C2_f22=-3.813+0.968*sqrt(h2/Radio_muesca)-
0.260*(h2/Radio_muesca)
C2_f33=7.423-
4.868*sqrt(h2/Radio_muesca)+0.869*(h2/Radio_muesca)
C2_f44=-3.839+3.070*sqrt(h2/Radio_muesca)-
0.600*(h2/Radio_muesca)

if (relacion2 >=0.1 & relacion2 <= 2)
  C2_F1=C2_f1
  C2_F2=C2_f2
  C2_F3=C2_f3
  C2_F4=C2_f4
  Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/
  Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
end

if (relacion2 > 2 & relacion2 <= 20)
  C2_F1=C2_f11
  C2_F2=C2_f22
  C2_F3=C2_f33
  C2_F4=C2_f44
  Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/
  Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
end

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f2=(2*(KT2_F-
1)/KT2_F)*(RAIZ_DE_A/sqrt(Radio_muesca))
Kf2_f=KT2_F/((1+constante_reemplazo_f2))

//TENSO DE ESFUERZOS MEDIOS:
Am2=[0 0 0;0 0 0;0 0 0]
//CÁLCULO DE LOS AUTOVALORES:
[Em2]=spec(Am2)
//CRITERIO DE VON-MISSES:
sigvmm2=(((Em2(1)-Em2(2))^2)+((Em2(2)-
Em2(3))^2)+((Em2(3)-Em2(1))^2))/2^0.5

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa2=[(sigma2*f_carga_flexion*Kf2_f/f_tama2) 0 0;0 0 0;0 0 0]

//CÁLCULO DE AUTOVALORES:
[Ea2]=spec(Aa2)
//CRITERIO DE VON-MISSES:
sigvma2=(((Ea2(1)-Ea2(2))^2)+((Ea2(2)-
Ea2(3))^2)+((Ea2(3)-Ea2(1))^2))/2^0.5

V_misses_med_LN2=[sigvmm2,
Coef_variacion_carga*sigvmm2]
V_misses_alt_LN2=[sigvma2,
Coef_variacion_carga*sigvma2]

//EVALUACIÓN DEL CRITERIO DE FALLA ESCOGIDO:

if criterio_falla == 1 then
  [Div_Va_SE_2,
  Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
  D_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]

  [Div_Va_SE_2_Cuad, Desv_Div_Va_SE_2_Cuad]=
  pot(D_Va_SE2,2)
  Division_VaSE2 = [Div_Va_SE_2_Cuad,
  Desv_Div_Va_SE_2_Cuad]

  Division_VmSut2 = [0,0]

  Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2
  (1)
  Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio
  n_VaSE2(1))^2+(0)^2)

  ASME_2=[Suma2_VmSut_VaSE,
  Desv_Suma1_VmSut_VaSE]
  Cf_ASME_2=ASME_2(2)/ASME_2(1)

```



```

CRITERIO_F=ASME_2
CF_CRITERIO_F=Cf_ASME_2
end

if criterio_falla == 2 then
[Div_Va_SE_2,
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
Division_VaSE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]

Division_VmSut2 = [0,0]

Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2
(1)
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio
n_VaSE2(1))^2+(0)^2)

GERBER_2=[Suma2_VmSut_VaSE,
Desv_Suma2_VmSut_VaSE]
Cf_GERBER_2=GERBER_2(2)/GERBER_2(1)
CRITERIO_F=GERBER_2
CF_CRITERIO_F=Cf_GERBER_2
end

if criterio_falla == 3 then
[Div_Va_SE_2,
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
Division_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]

Division_VmSut2 = [0,0]

Suma2_VmSut_VaSE=Division_Va_SE2(1)+Division_VmSut
2(1)
Desv_Suma2_VmSut_VaSE=sqrt((Division_Va_SE2(2)/Divisio
n_Va_SE2(1))^2+(0)^2)
GOODMAN_2=[Suma2_VmSut_VaSE,
Desv_Suma2_VmSut_VaSE]
Cf_GOODMAN_2=GOODMAN_2(2)/GOODMAN_2(1)
CRITERIO_F=GOODMAN_2
CF_CRITERIO_F=Cf_GOODMAN_2
end
//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL:
MEDCnormal_2=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F^2)
DESVnormal_2=CF_CRITERIO_F

//EVALUCIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_2,Conf_2]=cdfnor("PQ",0,abs(MEDCnormal_2),DESV
normal_2)

DIAMETRO_EJE2=0

if Prob_2<=1.5*10^-6 & Prob_2>=0.5*10^-6 then
DIAMETRO_EJE2=Diametro_eje2
break;
end
end

disp("El diametro del tramo 2 es")
disp(DIAMETRO_EJE2)

//ITERACIÓN DEL TERCER DIÁMETRO

for Diametro_eje3=tol.tol:6

// Radio del eje en el tramo del engrane 1:
Radio_eje3=Diametro_eje3/2

```

```

Torque3=(W_tangencial1/1000)*((Radio engrane1)-
Radio_eje3) // torque generado por el primer engrane
//MOMENTO DE INERCIA:
I3=(%pi*(((Radio_eje3*2)^4))/4)
//MOMENTO POLAR DE INERCIA:
J3=I3*2
// Cálculo de esfuerzos para tramo 3:
// Esfuerzo flector máximo:
sigma3=(Momento_max3*Radio_eje3/I3)
//Esfuerzo cortante torsor alternante:
taotormed3=(Torque3*Radio_eje3/J3)
//Esfuerzo cortante torsor medio:
taotoralt3=(Torque3*Radio_eje3/J3)

//FACTOR DE TAMAÑO
if (Diametro_eje3 < 0.11)
f_tama3=1
end
if (Diametro_eje3 >= 0.11 & Diametro_eje3 <= 2)
f_tama3=0.879*Diametro_eje3^-0.107
end
if (Diametro_eje3 >= 3 & Diametro_eje3 <= 10)
f_tama3=0.91*(Diametro_eje3^(-0.157))
end
//Factor de concentración de esfuerzo:
h3=Radio_eje3-Radio_eje2
relacion3=h3/Radio_muesca2

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON:

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON PARA CARGA AXIAL:

//0.1<=h/r<=2.0
C3_a1= 0.926+(1.157*sqrt(h3/Radio_muesca2))-
(0.099*(h3/Radio_muesca2))
C3_a2=0.012-
3.036*sqrt(h3/Radio_muesca2)+0.961*(h3/Radio_muesca2)
C3_a3=-0.302+3.977*sqrt(h3/Radio_muesca2)-
1.744*(h3/Radio_muesca2)
C3_a4=0.365-
2.098*sqrt(h3/Radio_muesca2)+0.878*(h3/Radio_muesca2)

//2.0<=h/r<=20
C3_a11=1.2+0.860*sqrt(h3/Radio_muesca2)-
0.022*(h3/Radio_muesca2)
C3_a22=-1.805-0.346*sqrt(h3/Radio_muesca2)-
0.038*(h3/Radio_muesca2)
C3_a33=2.198-
0.486*sqrt(h3/Radio_muesca2)+0.165*(h3/Radio_muesca2)
C3_a44=-0.593-0.028*sqrt(h3/Radio_muesca2)-
0.106*(h3/Radio_muesca2)

if (relacion3>=0.1 & relacion3 <= 2)
C3_A1=C3_a1
C3_A2=C3_a2
C3_A3=C3_a3
C3_A4=C3_a4
Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end

if (relacion3 > 2 & relacion3 <= 20)
C3_A1=C3_a11
C3_A2=C3_a22

```

```

C3_A3=C3_a33
C3_A4=C3_a44
Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.1 then
    Kt3_a=1
end
if relacion3 > 20 then
    Kt3_a=1
end
KT3_A=Kt3_a

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON PARA FLEXIÓN:

//0.1<=h/r<=2.0
C3_f1=0.947+1.206*sqrt(h3/Radio_muesca2)-
0.131*(h3/Radio_muesca2)
C3_f2=0.022-
3.405*sqrt(h3/Radio_muesca2)+0.915*(h3/Radio_muesca2)
C3_f3=0.869+1.777*sqrt(h3/Radio_muesca2)-
0.555*(h3/Radio_muesca2)
C3_f4=-0.810+0.422*sqrt(h3/Radio_muesca2)-
0.260*(h3/Radio_muesca2)

//2.0<=h/r<=20
C3_f11=1.232+0.832*sqrt(h3/Radio_muesca2)-
0.008*(h3/Radio_muesca2)
C3_f22=-3.813+0.968*sqrt(h3/Radio_muesca2)-
0.260*(h3/Radio_muesca2)
C3_f33=7.423-
4.868*sqrt(h3/Radio_muesca2)+0.869*(h3/Radio_muesca2)
C3_f44=-3.839+3.070*sqrt(h3/Radio_muesca2)-
0.600*(h3/Radio_muesca2)

if (relacion3 >=0.1 & relacion3 <= 2)
    C3_F1=C3_f1
    C3_F2=C3_f2
    C3_F3=C3_f3
    C3_F4=C3_f4
    Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
    Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end

if (relacion3 > 2 & relacion3 <= 20)
    C3_F1=C3_f11
    C3_F2=C3_f22
    C3_F3=C3_f33
    C3_F4=C3_f44
    Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
    Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.1 then
    Kt3_f=1
end
if relacion3 > 20 then
    Kt3_f=1
end

KT3_F=Kt3_f

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON PARA TORSIÓN:

```

```

//0.25<=h/r<=4.0
C3_t1=0.905+0.783*sqrt(h3/Radio_muesca2)-
0.075*(h3/Radio_muesca2)
C3_t2=-0.437-
1.969*sqrt(h3/Radio_muesca2)+0.553*(h3/Radio_muesca2)
C3_t3=1.557+1.073*sqrt(h3/Radio_muesca2)-
0.578*(h3/Radio_muesca2)
C3_t4=-1.061+0.171*sqrt(h3/Radio_muesca2)-
0.086*(h3/Radio_muesca2)

if (relacion3 >=0.25 & relacion3 <= 4.0)
    C3_T1=C3_t1
    C3_T2=C3_t2
    C3_T3=C3_t3
    C3_T4=C3_t4
    Kt3_t=C3_T1+C3_T2*(2*h3/Diametro_eje3)+C3_T3*(2*h3/
    Diametro_eje3)^2+C3_T4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.25 then
    Kt3_t=1
end

if relacion3 > 4.0 then
    Kt3_t=1
end

KT3_T=Kt3_t
//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f- flexión), (Kf_t-torsión)
constante_reemplazo_a3=(2*(KT3_A-
1)/KT3_A)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_a=KT3_A/((1+constante_reemplazo_a3))

constante_reemplazo_f3=(2*(KT3_F-
1)/KT3_F)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_f=KT3_F/((1+constante_reemplazo_f3))

constante_reemplazo_t3=(2*(KT3_T-
1)/KT3_T)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_t=KT3_T/((1+constante_reemplazo_t3))

//CÁLCULO DE LA PRESIÓN POR INTERFERENCIA
m=(((2*Radio_engrane1)^2-
Diametro_eje3^2))*(Diametro_eje3^2))
n=(2*Radio_engrane1)
PP1=((E*Interferen_maza_eje_max)/(2*Diametro_eje3^3))*(
m/n)

//TENSOR DE ESFUERZOS MEDIOS:
Am3=[(PP1*f_carga_axial*Kf3_a/f_tama3)
(taotormed3*f_carga_torsion*Kf3_t/f_tama3)
0;(taotormed3*f_carga_torsion*Kf3_t/f_tama3) 0 0 0 0]
//CÁLCULO DE AUTOVALORES:
[Em3]=spec(Am3)
//CRITERIO DE VON-MISSES:
sigvmm3=(((Em3(1)-Em3(2))^2)+((Em3(2)-
Em3(3))^2)+((Em3(3)-Em3(1))^2)/2)^0.5

//TENSOR DE ESFUERZOS ALTERNANTES
Aa3=[(siga3*f_carga_flexion*Kf3_f/f_tama3)
(taotoralt3*f_carga_torsion*Kf3_t/f_tama3)
(taotoralt3*f_carga_torsion*Kf3_t/f_tama3) 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Ea3]=spec(Aa3)
//CRITERIO DE VON-MISSES:

```

0;0

```
sigvma3=(((Ea3(1)-Ea3(2))^2)+((Ea3(2)-Ea3(3))^2)+((Ea3(3)-Ea3(1))^2)/2)^0.5
```

```
V_misses_med_LN3=[sigvmm3,
Coef_variacion_carga*sigvmm3]
V_misses_alt_LN3=[sigvma3,
Coef_variacion_carga*sigvma3]
```

```
//EVALUACIÓN DEL CRITERIO DE FALLA ESCOGIDO:
```

```
if criterio_falla == 1 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
D_Va_SE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]
```

```
[Div_Va_SE_3_Cuad, Desv_Div_Va_SE_3_Cuad]=
pot(D_Va_SE3,2)
Division_VaSE3 = [Div_Va_SE_3_Cuad,
Desv_Div_Va_SE_3_Cuad]
```

```
[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]
```

```
[Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]
```

```
[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)
```

```
ASME_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_ASME_3=ASME_3(2)/ASME_3(1)
CRITERIO_F=ASME_3
CF_CRITERIO_F=Cf_ASME_3
end
```

```
if criterio_falla == 2 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]
```

```
[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]
```

```
[Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]
```

```
[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)
```

```
GERBER_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GERBER_3=GERBER_3(2)/GERBER_3(1)
CRITERIO_F=GERBER_3
CF_CRITERIO_F=Cf_GERBER_3
end
```

```
if criterio_falla == 3 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]
[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
Division_VmSut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]
```

```
[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)
GOODMAN_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GOODMAN_3=GOODMAN_3(2)/GOODMAN_3(1)
CRITERIO_F=GOODMAN_3
CF_CRITERIO_F=Cf_GOODMAN_3
end
```

```
//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A LA DISTRIBUCIÓN NORMAL:
```

```
MEDCnormal_3=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_3=CF_CRITERIO_F
```

```
//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
```

```
[Prob_3,Conf_3]=cdfnor("PQ",0,abs(MEDCnormal_3),DESVnormal_3)
```

```
DIAMETRO_EJE3=0
```

```
if Prob_3<=1.5*10^-6 & Prob_3>=0.5*10^-6 then
DIAMETRO_EJE3=Diametro_eje3
break;
```

```
end
end
disp("El diametro del tramo 3 es ")
disp(DIAMETRO_EJE3)
```

```
//ITERACIÓN DEL CUARTO DIÁMETRO:
```

```
for Diametro_eje4=tol:tol:6
```

```
Radio_eje4=Diametro_eje4/2
Torque4=(W_tangencial1/1000)*((Radio_engrane1)-Radio_eje3)
```

```
//MOMENTO DE INERCIA:
```

```
I4=(%pi*(((Radio_eje4*2)^4))/4)
```

```
//MOMENTO POLAR DE INERCIA:
```

```
J4=I4*2
```

```
//CÁLCULO DE ESFUERZOS:
```

```
//Esfuerzo flector máximo:
```

```
sigma4=(Momento_max4*Radio_eje4/I4)
```

```
//Esfuerzo cortante torsor máximo medio:
```

```
taotormed4=(Torque4*Radio_eje4/J4)
```

```
//Esfuerzo cortante torsor máximo alternante:
```

```
taotoralt4=(Torque4*Radio_eje4/J4)
```

```
//FACTOR DE TAMAÑO
```

```
if (Diametro_eje4 < 0.11)
```

```
f_tama4=1
```

```
end
```

```
if (Diametro_eje4 >= 0.11 & Diametro_eje4 <= 2)
```

```
f_tama4=0.879*Diametro_eje4^-0.107
```

```
end
```

```
if (Diametro_eje4 >= 3 & Diametro_eje4 <= 10)
```

```
f_tama4=0.91*(Diametro_eje4^(-0.157))
```

```
end
```

```
//Factor de concentración de esfuerzo
```

```
h4=Radio_eje4-Radio_eje3
```

```
relacion4=h4/Radio_muesca3
```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE PETERSON:
```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE PETERSON PARA FLEXIÓN:
```

```
//0.1<=h/r<=2.0
```

```

C4_f1=0.947+1.206*sqrt(h4/Radio_muesca3)-
0.131*(h4/Radio_muesca3)
C4_f2=0.022-
3.405*sqrt(h4/Radio_muesca3)+0.915*(h4/Radio_muesca3)
C4_f3=0.869+1.777*sqrt(h4/Radio_muesca3)-
0.555*(h4/Radio_muesca3)
C4_f4=-0.810+0.422*sqrt(h4/Radio_muesca3)-
0.260*(h4/Radio_muesca3)

//2.0<=h/r<=20
C4_f11=1.232+0.832*sqrt(h4/Radio_muesca3)-
0.008*(h4/Radio_muesca3)
C4_f22=-3.813+0.968*sqrt(h4/Radio_muesca3)-
0.260*(h4/Radio_muesca3)
C4_f33=7.423-
4.868*sqrt(h4/Radio_muesca3)+0.869*(h4/Radio_muesca3)
C4_f44=-3.839+3.070*sqrt(h4/Radio_muesca3)-
0.600*(h4/Radio_muesca3)

if (relacion4 >=0.1 & relacion4 <= 2)
  C4_F1=C4_f1
  C4_F2=C4_f2
  C4_F3=C4_f3
  C4_F4=C4_f4
  Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
  Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if (relacion4 > 2 & relacion4 <= 20)
  C4_F1=C4_f11
  C4_F2=C4_f22
  C4_F3=C4_f33
  C4_F4=C4_f44
  Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
  Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if relacion4 < 0.1 then
  Kt4_f=1
end

if relacion4 > 0.1 then
  Kt4_f=1
end

KT4_F= Kt4_f

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON PARA TORSIÓN:

//0.25<=h/r<=4.0
C4_t1=0.905+0.783*sqrt(h4/Radio_muesca3)-
0.075*(h4/Radio_muesca3)
C4_t2=-0.437-
1.969*sqrt(h4/Radio_muesca3)+0.553*(h4/Radio_muesca3)
C4_t3=1.557+1.073*sqrt(h4/Radio_muesca3)-
0.578*(h4/Radio_muesca3)
C4_t4=-1.061+0.171*sqrt(h4/Radio_muesca3)-
0.086*(h4/Radio_muesca3)

if (relacion4 >=0.25 & relacion4 <= 4.0)
  C4_T1=C4_t1
  C4_T2=C4_t2
  C4_T3=C4_t3
  C4_T4=C4_t4
  Kt4_t=C4_T1+C4_T2*(2*h4/Diametro_eje4)+C4_T3*(2*h4/
  Diametro_eje4)^2+C4_T4*(2*h4/Diametro_eje4)^3

```

```

end

if relacion4 < 0.25 then
  Kt4_t=1
end

if relacion4 > 4.0 then
  Kt4_t=1
end

KT4_T=Kt4_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f4=(2*(KT4_F-
1)/KT4_F)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf4_f=KT4_F/((1+constante_reemplazo_f4))

constante_reemplazo_t4=(2*(KT4_T-
1)/KT4_T)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf4_t=KT4_T/((1+constante_reemplazo_t4))

//TENSOR DE ESFUERZOS MEDIOS:
Am4=[0 (taotormed4*f_carga_torsion*Kf4_t/f_tama4)
0;(taotormed4*f_carga_torsion*Kf4_t/f_tama4) 0 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Em4]=spec(Am4)
//CRITERIO DE VON-MISSES:
sigvmm4=(((Em4(1)-Em4(2))^2)+((Em4(2)-
Em4(3))^2)+((Em4(3)-Em4(1))^2))/2^0.5

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa4=[(siga4*f_carga_flexion*Kf4_f/f_tama4)
(taotoralt4*f_carga_torsion*Kf4_t/f_tama4) 0;0 0]
(taotoralt4*f_carga_torsion*Kf4_t/f_tama4) 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Ea4]=spec(Aa4)
//CRITERIO DE VON-MISSES:
sigvma4=(((Ea4(1)-Ea4(2))^2)+((Ea4(2)-
Ea4(3))^2)+((Ea4(3)-Ea4(1))^2))/2^0.5

V_misses_med_LN4=[sigvmm4,
Coef_variacion_carga*sigvmm4]
V_misses_alt_LN4=[sigvma4,
Coef_variacion_carga*sigvma4]

//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:
if criterio_falla == 1 then
  [Div_Va_SE_4, Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
  D_Va_SE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

  [Div_Va_SE_4_Cuad, Desv_Div_Va_SE_4_Cuad]=
  pot(D_Va_SE4,2)
  Division_VaSE4 = [Div_Va_SE_4_Cuad,
  Desv_Div_Va_SE_4_Cuad]

  [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
  D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

  [Div_Vm_Sut_4_Cuad, Desv_Div_Vm_Sut_4_Cuad]=
  pot(D_Vm_Sut4,2)
  Division_VmSut4 = [Div_Vm_Sut_4_Cuad,
  Desv_Div_Vm_Sut_4_Cuad]

```

```

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

ASME_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_ASME_4=ASME_4(2)/ASME_4(1)
CRITERIO_F=ASME_4
CF_CRITERIO_F=Cf_ASME_4
end

if criterio_falla == 2 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Div_Vm_Sut_4_Cuad, Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4 = [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

GERBER_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GERBER_4=GERBER_4(2)/GERBER_4(1)
CRITERIO_F=GERBER_4
CF_CRITERIO_F=Cf_GERBER_4
end

if criterio_falla == 3 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]
[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
Division_VmSut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)
GOODMAN_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GOODMAN_4=GOODMAN_4(2)/GOODMAN_4(1)
CRITERIO_F=GOODMAN_4
CF_CRITERIO_F=Cf_GOODMAN_4
end

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL:
MEDCnormal_4=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESNormal_4=CF_CRITERIO_F

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_4,Conf_4]=cdfnorf("PQ",0,abs(MEDCnormal_4),DESNormal_4)

DIAMETRO_EJE4=0

if Prob_4<=1.5*10^-6 & Prob_4>=0.5*10^-6 then
DIAMETRO_EJE4=Diametro_eje4
break;
end
end
disp("El diametro del tramo 4 es")
disp(DIAMETRO_EJE4)

```

//ITERACIÓN DEL QUINTO DIÁMETRO

```
for Diametro_eje5=tol:tol:6
```

```
Radio_eje5=Diametro_eje5/2
```

```
Torque5=(W_tangencial2/1000)*((Radio engrane2)-
Radio_eje5) // el torque aplicado es la fuerza tangencial por
el Radio del engranaje menos el Radio del eje.
```

```
//MOMENTO DE INERCIA:
```

```
I5=(%pi*(((Radio_eje5*2)^4))/4)
```

```
//MOMENTO DE ÁREA:
```

```
Q5=(2/3)*(Radio_eje5^3)
```

```
//MOMENTO POLAR DE INERCIA:
```

```
J5=I5*2
```

```
//CÁLCULO DE ESFUERZOS:
```

```
//Esfuerzo flector máximo:
```

```
sigma5=(Momento_max5*Radio_eje5/I5)
```

```
//Esfuerzo cortante torsor máximo medio:
```

```
taotormed5=(Torque5*Radio_eje5/J5)
```

```
//Esfuerzo cortante torsor máximo alternante:
```

```
taotoralt5=(Torque5*Radio_eje5/J5)
```

```
//FACTOR DE TAMAÑO
```

```
if (Diametro_eje5 < 0.11)
```

```
f_tama5=1
```

```
end
```

```
if (Diametro_eje5 >= 0.11 & Diametro_eje5 <= 2)
```

```
f_tama5=0.879*Diametro_eje5^-0.107
```

```
end
```

```
if (Diametro_eje5 >= 3 & Diametro_eje5 <= 10)
```

```
f_tama5=0.91*(Diametro_eje5^(-0.157))
```

```
end
```

```
//factor de concentracion de esfuerzo
```

```
h5=Radio_eje5-Radio_eje4
```

```
relacion5=h5/Radio_muesca4
```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON:
```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON PARA CARGA AXIAL:
```

```
//0.1 <= h/r <= 2.0
```

```
C5_a1= 0.926+(1.157*sqrt(h5/Radio_muesca4))-
(0.099*(h5/Radio_muesca4))
```

```
C5_a2=0.012-
```

```
3.036*sqrt(h5/Radio_muesca4)+0.961*(h5/Radio_muesca4)
```

```
C5_a3=-0.302+3.977*sqrt(h5/Radio_muesca4)-
```

```
1.744*(h5/Radio_muesca4)
```

```
C5_a4=0.365-
```

```
2.098*sqrt(h5/Radio_muesca4)+0.878*(h5/Radio_muesca4)
```

```
//2.0 <= h/r <= 20
```

```
C5_a11=1.2+0.860*sqrt(h5/Radio_muesca4)-
```

```
0.022*(h5/Radio_muesca4)
```

```
C5_a22=-1.805-0.346*sqrt(h5/Radio_muesca4)-
```

```
0.038*(h5/Radio_muesca4)
```

```
C5_a33=2.198-
```

```
0.486*sqrt(h5/Radio_muesca4)+0.165*(h5/Radio_muesca4)
```

```
C5_a44=-0.593-0.028*sqrt(h5/Radio_muesca4)-
```

```
0.106*(h5/Radio_muesca4)
```

```
if (relacion5 >= 0.1 & relacion5 <= 2)
```

```
C5_A1=C5_a1
```

```

C5_A2=C5_a2
C5_A3=C5_a3
C5_A4=C5_a4
Kt5_a=C5_A1+C5_A2*(2*h5/Diametro_eje5)+C5_A3*(2*h5/
Diametro_eje5)^2+C5_A4*(2*h5/Diametro_eje5)^3
end

```

```

if (relacion5 > 2 & relacion5 <= 20)

```

```

  C5_A1=C5_a11
  C5_A2=C5_a22
  C5_A3=C5_a33
  C5_A4=C5_a44
  Kt5_a=C5_A1+C5_A2*(2*h5/Diametro_eje5)+C5_A3*(2*h5/
  Diametro_eje5)^2+C5_A4*(2*h5/Diametro_eje5)^3
end

```

```

if relacion5 < 0.1 then

```

```

  Kt5_a=1
end

```

```

if relacion5 > 20 then

```

```

  Kt5_a=1
end

```

```

KT5_A=Kt5_a

```

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE PETERSON PARA FLEXIÓN:

```

//0.1 <= h/r <= 2.0

```

```

C5_f1=0.947+1.206*sqrt(h5/Radio_muesca4)-
0.131*(h5/Radio_muesca4)
C5_f2=0.022-
3.405*sqrt(h5/Radio_muesca4)+0.915*(h5/Radio_muesca4)
C5_f3=0.869+1.777*sqrt(h5/Radio_muesca4)-
0.555*(h5/Radio_muesca4)
C5_f4=-0.810+0.422*sqrt(h5/Radio_muesca4)-
0.260*(h5/Radio_muesca4)

```

```

//2.0 <= h/r <= 20

```

```

C5_f11=1.232+0.832*sqrt(h5/Radio_muesca4)-
0.008*(h5/Radio_muesca4)
C5_f22=-3.813+0.968*sqrt(h5/Radio_muesca4)-
0.260*(h5/Radio_muesca4)
C5_f33=7.423-
4.868*sqrt(h5/Radio_muesca4)+0.869*(h5/Radio_muesca4)
C5_f44=-3.839+3.070*sqrt(h5/Radio_muesca4)-
0.600*(h5/Radio_muesca4)

```

```

if (relacion5 >= 0.1 & relacion5 <= 2)

```

```

  C5_F1=C5_f1
  C5_F2=C5_f2
  C5_F3=C5_f3
  C5_F4=C5_f4
  Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
  Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

```

```

if (relacion5 > 2 & relacion5 <= 20)

```

```

  C5_F1=C5_f11
  C5_F2=C5_f22
  C5_F3=C5_f33
  C5_F4=C5_f44
  Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
  Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

```

```

if relacion5 < 0.1 then

```

```

  Kt5_f=1
end

```

```

if relacion5 < 20 then

```

```

  Kt5_f=1
end

```

```

KT5_F=Kt5_f

```

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE PETERSON PARA TORSIÓN:

```

//0.25 <= h/r <= 4.0

```

```

C5_t1=0.905+0.783*sqrt(h5/Radio_muesca4)-
0.075*(h5/Radio_muesca4)
C5_t2=-0.437-
1.969*sqrt(h5/Radio_muesca4)+0.553*(h5/Radio_muesca4)
C5_t3=1.557+1.073*sqrt(h5/Radio_muesca4)-
0.578*(h5/Radio_muesca4)
C5_t4=-1.061+0.171*sqrt(h5/Radio_muesca4)-
0.086*(h5/Radio_muesca4)

```

```

if (relacion5 >= 0.25 & relacion5 <= 4.0)

```

```

  C5_T1=C5_t1
  C5_T2=C5_t2
  C5_T3=C5_t3
  C5_T4=C5_t4
  Kt5_t=C5_T1+C5_T2*(2*h5/Diametro_eje5)+C5_T3*(2*h5/
  Diametro_eje5)^2+C5_T4*(2*h5/Diametro_eje5)^3
end

```

```

if relacion5 < 0.25 then

```

```

  Kt5_t=1
end

```

```

if relacion5 > 4.0 then

```

```

  Kt5_t=1
end

```

```

KT5_T=Kt5_t

```

//Factor de concentración de esfuerzo de fatiga (Kf_a - axial), (Kf_f - flexión), (Kf_t - torsión)

```

constante_reemplazo_a5=(2*(KT5_A-
1)/KT5_A)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_a=Kt5_a/((1+constante_reemplazo_a5))

```

```

constante_reemplazo_f5=(2*(KT5_F-
1)/KT5_F)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_f=Kt5_f/((1+constante_reemplazo_f5))

```

```

constante_reemplazo_t5=(2*(KT5_T-
1)/KT5_T)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_t=Kt5_t/((1+constante_reemplazo_t5))

```

//CÁLCULO DE LA PRESIÓN POR INTERFERENCIA:

```

m2=(((2*Radio_engrane2)^2-
Diametro_eje5^2))*(Diametro_eje5^2))
n2=(2*Radio_engrane2)
PP2=((E*Interferen_maza_eje_max2)/(2*Diametro_eje5^3))*(
m2/n2)

```

//TENSOR DE ESFUERZOS MEDIOS:

```

Am5=[(PP2*f_carga_axial*Kf5_a/f_tama5)
(taotormed5*f_carga_torsion*Kf5_t/f_tama5)
0;(taotormed5*f_carga_torsion*Kf5_t/f_tama5) 0 0;0 0 0]

```

//CÁLCULO DE AUTOVALORES:


```

[Em5]=spec(Am5)
//CRITERIO DE VON-MISSES
sigvmm5=(((Em5(1)-Em5(2))^2)+((Em5(2)-
Em5(3))^2)+((Em5(3)-Em5(1))^2))/2^0.5

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa5=[(siga5*f_carga_flexion*Kf5_f/f_tama5)
(taotoralt5*f_carga_torsion*Kf5_t/f_tama5)
(taotoralt5*f_carga_torsion*Kf5_t/f_tama5) 0;0 0 0]
0;0
//CÁLCULO DE AUTOVALORES:
[Ea5]=spec(Aa5)
//CRITERIO DE VON-MISSES:
sigvma5=(((Ea5(1)-Ea5(2))^2)+((Ea5(2)-
Ea5(3))^2)+((Ea5(3)-Ea5(1))^2))/2^0.5

V_misses_med_LN5=[sigvmm5,
Coef_variacion_carga*sigvmm5]
V_misses_alt_LN5=[sigvma5,
Coef_variacion_carga*sigvma5]

//EVALUACIÓN DEL CRITERIO DE FALLA ESCOGIDO:
if criterio_falla == 1 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
D_Va_SE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

[Div_Va_SE_5_Cuad, Desv_Div_Va_SE_5_Cuad]=
pot(D_Va_SE5,2)
Division_VaSE5 = [Div_Va_SE_5_Cuad,
Desv_Div_Va_SE_5_Cuad]

[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]

[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)

ASME_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_ASME_5=ASME_5(2)/ASME_5(1)
CRITERIO_F=ASME_5
CF_CRITERIO_F=Cf_ASME_5
end

if criterio_falla == 2 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]

[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)

GERBER_5=[Suma5_VmSut,Desv_Suma5_VmSut]

```

```

Cf_GERBER_5=GERBER_5(2)/GERBER_5(1)
CRITERIO_F=GERBER_5
CF_CRITERIO_F=Cf_GERBER_5
end

if criterio_falla == 3 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
Division_VmSut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
GOODMAN_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_GOODMAN_5=GOODMAN_5(2)/GOODMAN_5(1)
CRITERIO_F=GOODMAN_5
CF_CRITERIO_F=Cf_GOODMAN_5
end

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL:
MEDCnormal_5=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_5=CF_CRITERIO_F

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_5,Conf_5]=cdfnor("PQ",0,abs(MEDCnormal_5),DESV
normal_5)

DIAMETRO_EJE5=0

if Prob_5<=1.5*10^-6 & Prob_5>=0.5*10^-6 then
DIAMETRO_EJE5=Diametro_eje5
break;
end
end
disp("El diametro del tramo 5 es: ")
disp(DIAMETRO_EJE5)

//ITERACIÓN DEL SEXTO DIÁMETRO:
for Diametro_eje6=tol:tol:6

Radio_eje6=Diametro_eje6/2
//MOMENTO DE INERCIA:
I6=(%pi)*(((Radio_eje6*2)^4))/4)
//MOMENTO POLAR DE INERCIA:
J6=I6*2

//CÁLCULO DE ESFUERZOS:
//Esfuerzo flector máximo:
siga6=(Momento_max6*Radio_eje6/16)

//FACTOR DE TAMAÑO
if (Diametro_eje6 < 0.11)
f_tama6=1
end
if (Diametro_eje6 >= 0.11 & Diametro_eje6 <= 2)
f_tama6=0.879*Diametro_eje6^-0.107
end
if (Diametro_eje6 >= 3 & Diametro_eje6 <= 10)
f_tama6=0.91*(Diametro_eje6^(-0.157))
end
//Factor de concentración de esfuerzo:
h6=Radio_eje6-Radio_eje5
relacion6=h6/Radio_muesca5

```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON:
```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON PARA FLEXIÓN:
```

```
//Kt para esfuerzos de flexión (Kt)
```

```
//0.1 <= h/r <= 2.0
```

```
C6_f1=0.947+1.206*sqrt(h6/Radio_muesca5)-
0.131*(h6/Radio_muesca5)
C6_f2=0.022-
3.405*sqrt(h6/Radio_muesca5)+0.915*(h6/Radio_muesca5)
C6_f3=0.869+1.777*sqrt(h6/Radio_muesca5)-
0.555*(h6/Radio_muesca5)
C6_f4=-0.810+0.422*sqrt(h6/Radio_muesca5)-
0.260*(h6/Radio_muesca5)
```

```
//2.0 <= h/r <= 20
```

```
C6_f11=1.232+0.832*sqrt(h6/Radio_muesca5)-
0.008*(h6/Radio_muesca5)
C6_f22=-3.813+0.968*sqrt(h6/Radio_muesca5)-
0.260*(h6/Radio_muesca5)
C6_f33=7.423-
4.868*sqrt(h6/Radio_muesca5)+0.869*(h6/Radio_muesca5)
C6_f44=-3.839+3.070*sqrt(h6/Radio_muesca5)-
0.600*(h6/Radio_muesca5)
```

```
if (relacion6 >= 0.1 & relacion6 <= 2)
```

```
    C6_F1=C6_f1
    C6_F2=C6_f2
    C6_F3=C6_f3
    C6_F4=C6_f4
```

```
Kt6_f=C6_F1+C6_F2*(2*h6/Diametro_eje6)+C6_F3*(2*h6/
Diametro_eje6)^2+C6_F4*(2*h6/Diametro_eje6)^3
end
```

```
if (relacion6 > 2 & relacion6 <= 20)
```

```
    C6_F1=C6_f11
    C6_F2=C6_f22
    C6_F3=C6_f33
    C6_F4=C6_f44
```

```
Kt6_f=C6_F1+C6_F2*(2*h6/Diametro_eje6)+C6_F3*(2*h6/
Diametro_eje6)^2+C6_F4*(2*h6/Diametro_eje6)^3
end
```

```
if relacion6 < 0.1 then
```

```
    Kt6_f=1
```

```
end
```

```
if relacion6 > 20 then
```

```
    Kt6_f=1
```

```
end
```

```
KT6_F=Kt6_f
```

```
constante_reemplazo_f6=(2*(KT6_F-
1)/KT6_F)*(RAIZ_DE_A/sqrt(Radio_muesca5))
Kf6_f=KT6_F/((1+constante_reemplazo_f6))
```

```
//TENSOR DE ESFUERZOS MEDIOS:
```

```
Am6=[0 0 0;0 0 0;0 0 0]
```

```
//CÁLCULO DE AUTOVALORES:
```

```
[Em6]=spec(Am6)
```

```
//CRITERIO DE VON-MISSES:
```

```
sigvmm6=(((Em6(1)-Em6(2))^2)+((Em6(2)-
Em6(3))^2)+((Em6(3)-Em6(1))^2))/2)^0.5
```

```
//TENSOR DE ESFUERZOS ALTERNANTES:
```

```
Aa6=[(sigma6*f_carga_flexion*Kf6_f/f_tama6) 0 0;0 0 0;0 0 0]
```

```
//CÁLCULO DE AUTOVALORES
```

```
[Ea6]=spec(Aa6)
```

```
//CRITERIO DE VON-MISSES:
```

```
sigvma6=(((Ea6(1)-Ea6(2))^2)+((Ea6(2)-
Ea6(3))^2)+((Ea6(3)-Ea6(1))^2))/2)^0.5
```

```
V_misses_med_LN6=[sigvmm6,
```

```
Coef_variacion_carga*sigvmm6]
```

```
V_misses_alt_LN6=[sigvma6,
```

```
Coef_variacion_carga*sigvma6]
```

```
//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:
```

```
if criterio_falla == 1 then
```

```
    [Div_Va_SE_6,
```

```
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
```

```
D_Va_SE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]
```

```
[Div_Va_SE_6_Cuad, Desv_Div_Va_SE_6_Cuad]=
```

```
pot(D_Va_SE6,2)
```

```
Division_VaSE6 = [Div_Va_SE_6_Cuad,
```

```
Desv_Div_Va_SE_6_Cuad]
```

```
Division_VmSut6 = [0,0]
```

```
Suma6_VmSut_VaSE=Division_VaSE6(1)+Division_VmSut6
```

```
(1)
```

```
Desv_Suma6_VmSut_VaSE=sqrt((Division_VaSE6(2)/Divisio
n_VaSE6(1))^2+(0)^2)
```

```
ASME_6=[Suma6_VmSut_VaSE,
```

```
Desv_Suma6_VmSut_VaSE]
```

```
Cf_ASME_6=ASME_6(2)/ASME_6(1)
```

```
CRITERIO_F=ASME_6
```

```
CF_CRITERIO_F=Cf_ASME_6
```

```
end
```

```
if criterio_falla == 2 then
```

```
    [Div_Va_SE_6,
```

```
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
```

```
Division_VaSE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]
```

```
Division_VmSut6 = [0,0]
```

```
Suma6_VmSut_VaSE=Division_VaSE6(1)+Division_VmSut6
```

```
(1)
```

```
Desv_Suma6_VmSut_VaSE=sqrt((Division_VaSE6(2)/Divisio
n_VaSE6(1))^2+(0)^2)
```

```
GERBER_6=[Suma6_VmSut_VaSE,
```

```
Desv_Suma6_VmSut_VaSE]
```

```
Cf_GERBER_6=GERBER_6(2)/GERBER_6(1)
```

```
CRITERIO_F=GERBER_6
```

```
CF_CRITERIO_F=Cf_GERBER_6
```

```
end
```

```
if criterio_falla == 3 then
```

```
    [Div_Va_SE_6,
```

```
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
```

```
Division_Va_SE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]
```

```
Division_VmSut6 = [0,0]
```

```
Suma6_VmSut_VaSE=Division_Va_SE6(1)+Division_VmSut
```

```
6(1)
```



```

Desv_Suma6_VmSut_VaSE=sqrt((Division_Va_SE6(2)/Divisi
on_Va_SE6(1))^2+(0)^2)
GOODMAN_6=[Suma6_VmSut_VaSE,
Desv_Suma6_VmSut_VaSE]
Cf_GOODMAN_6=GOODMAN_6(2)/GOODMAN_6(1)
CRITERIO_F=GOODMAN_6
CF_CRITERIO_F=Cf_GOODMAN_6
end

```

```

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL:
MEDCnormal_6=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_6=CF_CRITERIO_F

```

```

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_6,Conf_6]=cdfnorr("PQ",0,abs(MEDCnormal_6),DESV
normal_6)

```

```
DIAMETRO_EJE6=0
```

```

if Prob_6<=1.5*10^-6 & Prob_6>=0.5*10^-6 then
    DIAMETRO_EJE6=Diametro_eje6
    break;
end
end
disp("El diametro del tramo 6 es")
disp(DIAMETRO_EJE6)

```

```

//ITERACIÓN DEL SÉPTIMO DIÁMETRO:
for Diametro_eje7=tol:tol:6

```

```

Radio_eje7=Diametro_eje7/2
//MOMENTO DE INERCIA:
I7=(%pi*(((Radio_eje7^2)^4))/4)
//MOMENTO DE ÁREA:
Q7=(2/3)*(Radio_eje7^3)
//MOMENTO POLAR DE INERCIA:
J7=I7*2
//CÁLCULO DE ESFUERZOS:
//Esfuerzo flector máximo:
siga7=(Momento_max7*Radio_eje7/I7)

```

```

//FACTOR DE TAMAÑO
if (Diametro_eje7 < 0.11)
    f_tama7=1
end
if (Diametro_eje7 >= 0.11 & Diametro_eje7 <= 2)
    f_tama7=0.879*Diametro_eje7^-0.107
end
if (Diametro_eje7 >= 3 & Diametro_eje7 <= 10)
    f_tama7=0.91*(Diametro_eje7^(-0.157))
end

```

```

//Factor de concentración de esfuerzo
h7=Radio_eje7-Radio_eje6
relacion7=h7/Radio_muesca6

```

```

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON:
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON PARA FLEXIÓN:
//0.1<=h/r<=2.0
C7_f1=0.947+1.206*sqrt(h7/Radio_muesca6)-
0.131*(h7/Radio_muesca6)
C7_f2=0.022-
3.405*sqrt(h7/Radio_muesca6)+0.915*(h7/Radio_muesca6)

```

```

C7_f3=0.869+1.777*sqrt(h7/Radio_muesca6)-
0.555*(h7/Radio_muesca6)
C7_f4=-0.810+0.422*sqrt(h7/Radio_muesca6)-
0.260*(h7/Radio_muesca6)

```

```

//2.0<=h/r<=20
C7_f11=1.232+0.832*sqrt(h7/Radio_muesca6)-
0.008*(h7/Radio_muesca6)
C7_f22=-3.813+0.968*sqrt(h7/Radio_muesca6)-
0.260*(h7/Radio_muesca6)
C7_f33=7.423-
4.868*sqrt(h7/Radio_muesca6)+0.869*(h7/Radio_muesca6)
C7_f44=-3.839+3.070*sqrt(h7/Radio_muesca6)-
0.600*(h7/Radio_muesca6)

```

```

if (relacion7 >=0.1 & relacion7 <= 2)
    C7_F1=C7_f1
    C7_F2=C7_f2
    C7_F3=C7_f3
    C7_F4=C7_f4
    Kt7_f=C7_F1+C7_F2*(2*h7/Diametro_eje7)+C7_F3*(2*h7/
Diametro_eje7)^2+C7_F4*(2*h7/Diametro_eje7)^3
end

```

```

if (relacion7 > 2 & relacion7 <= 20)
    C7_F1=C7_f11
    C7_F2=C7_f22
    C7_F3=C7_f33
    C7_F4=C7_f44
    Kt7_f=C7_F1+C7_F2*(2*h7/Diametro_eje7)+C7_F3*(2*h7/
Diametro_eje7)^2+C7_F4*(2*h7/Diametro_eje7)^3
end

```

```

if relacion7 < 0.1 then
    Kt7_f=1
end

```

```

if relacion7 > 20 then
    Kt7_f=1
end

```

```
KT7_F=Kt7_f
```

```

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f- flexión), (Kf_t-torsión)

```

```

constante_reemplazo_f7=(2*(KT7_F-
1)/KT7_F)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_f=KT7_F/((1+constante_reemplazo_f7))

```

```

//TENSOR DE ESFUERZOS MEDIOS:
Am7=[0 0 0;0 0 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Em7]=spec(Am7)
//CRITERIO DE VON-MISSES:
sigvmm7=(((Em7(1)-Em7(2))^2)+((Em7(2)-
Em7(3))^2)+((Em7(3)-Em7(1))^2))/2^0.5

```

```

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa7=[(siga7*f_carga_flexion*Kf7_f/f_tama7) 0 0;0 0 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Ea7]=spec(Aa7)
//CRITERIO DE VON-MISSES:
sigvma7=(((Ea7(1)-Ea7(2))^2)+((Ea7(2)-
Ea7(3))^2)+((Ea7(3)-Ea7(1))^2))/2^0.5

```

```

V_misses_med_LN7=[sigvmm7,
Coef_variacion_carga*sigvmm7]
V_misses_alt_LN7=[sigvma7,
Coef_variacion_carga*sigvma7]

//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:
if criterio_falla == 1 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
D_Va_SE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

[Div_Va_SE_7_Cuad,          Desv_Div_Va_SE_7_Cuad]=
pot(D_Va_SE7,2)
Division_VaSE7          =          [Div_Va_SE_7_Cuad,
Desv_Div_Va_SE_7_Cuad]

Division_VmSut7 = [0,0]

Suma7_VmSut_VaSE=Division_VaSE7(1)+Division_VmSut7
(1)
Desv_Suma7_VmSut_VaSE=sqrt((Division_VaSE7(2)/Divisio
n_VaSE7(1))^2+(0)^2)

ASME_7=[Suma7_VmSut_VaSE,
Desv_Suma7_VmSut_VaSE]
Cf_ASME_7=ASME_7(2)/ASME_7(1)
CRITERIO_F=ASME_7
CF_CRITERIO_F=Cf_ASME_7
end

if criterio_falla == 2 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
Division_VaSE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

Division_VmSut7 = [0,0]

Suma7_VmSut_VaSE=Division_VaSE7(1)+Division_VmSut7
(1)
Desv_Suma7_VmSut_VaSE=sqrt((Division_VaSE7(2)/Divisio
n_VaSE7(1))^2+(0)^2)

GERBER_7=[Suma7_VmSut_VaSE,
Desv_Suma7_VmSut_VaSE]
Cf_GERBER_7=GERBER_7(2)/GERBER_7(1)
CRITERIO_F=GERBER_7
CF_CRITERIO_F=Cf_GERBER_7
end

if criterio_falla == 3 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
Division_Va_SE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

Division_VmSut7 = [0,0]

Suma7_VmSut_VaSE=Division_Va_SE7(1)+Division_VmSut
7(1)
Desv_Suma7_VmSut_VaSE=sqrt((Division_Va_SE7(2)/Divisi
on_Va_SE7(1))^2+(0)^2)
GOODMAN_7=[Suma7_VmSut_VaSE,
Desv_Suma7_VmSut_VaSE]
Cf_GOODMAN_7=GOODMAN_7(2)/GOODMAN_7(1)
CRITERIO_F=GOODMAN_7
CF_CRITERIO_F=Cf_GOODMAN_7
end

```

```

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL
MEDCnormal_7=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_7=CF_CRITERIO_F

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_7,Conf_7]=cdfnor("PQ",0,abs(MEDCnormal_7),DESV
normal_7)

DIAMETRO_EJE7=0

if Prob_7<=1.5*10^-6 & Prob_7>=0.5*10^-6 then
DIAMETRO_EJE7=Diametro_eje7
break;
end
end
disp("El diametro del tramo 7 es")
disp(DIAMETRO_EJE7)

Sut9=0.9*Sut(1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=((sigvma7)-Sut9+(pend*3))/pend //valor de x para el
esfuerzo alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida del eje es :")
disp(ViRe)

```

11_DosEngranes.sci

```

clc
clear all
//Entrada de datos de las sesiones, muescas, ángulos, ubicación
y fuerzas:
exec('11.1_DistanciaSesionesMuecasYUbicacionEngrane.sci',
-1)

//Entrada del ajuste por interferencia para el primer engrane:
exec('11.2_OpcionesInterferenciaUno.sci',-1)

if interferencia == 1 then
exec('10.3_InterferenciaUno.sci',-1) //Para la opción 1
end

if interferencia == 2 then
exec('10.4_InterferenciaDos.sci',-1) // Para la opción 2
end

if interferencia == 3 then
exec('10.5_InterferenciaTres.sci',-1) //Para la opción 3
end

Interferen_maza_eje_min=Interf_maza_eje_min
Interferen_maza_eje_max=Interf_maza_eje_max

//Entrada del ajuste por interferencia para el segundo
engrane:
exec('11.3_OpcionesInterferenciaDos.sci',-1)

```

```

if interferencia2 == 1 then
exec('11.4_InterferenciaUno.sci',-1) //Para la opción 1
end

```

```

if interferencia2 == 2 then
exec('11.5_InterferenciaDos.sci',-1) //Para la opción 2
end

```

```

if interferencia2 == 3 then
exec('11.6_InterferenciaTres.sci',-1) //Para la opción 3
end

```

```

Interferen_maza_eje_min2=Interf_maza_eje_min2
Interferen_maza_eje_max2=Interf_maza_eje_max2

```

```

//Cálculo de diagramas de momentos con sus diagramas:
exec('11.7_TorquesMomentosYGraficas.sci',-1)
//Multiplicación de los factores de Marin Ka por Kd:
exec('10.8_MultiplicacionKaKd.sci',-1)
//Iteraciones de los diámetros:
exec('11.9_IteracionesDeLosDiametros.sci',-1)

```

12. 1_DistanciaSesionesMuestrasYUbicacionEngrane.sci

```

clc
clear all

```

```

sesion_1=input('Ingrese la distancia de la sesión 1:')
Radio_muesca=input('Ingrese el radio de la muesca de la
sesión 1:')

```

```

c=0
while c==0
c=1
sesion_2=input('Ingrese la distancia de la sesión 2:')
if (sesion_2 <= sesion_1 | sesion_2 >= longitud_eje)
c=0
disp('Distancia de la sesion 2 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca2=input('Ingrese el radio de la muesca de la
sesión 2:')

```

```

c=0
while c==0
c=1
sesion_3=input('Ingrese la distancia de la sesión 3:')
if (sesion_3 <= sesion_1 | sesion_3 <= sesion_2 | sesion_3
>= longitud_eje)
c=0
disp('Distancia de la sesion 3 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca3=input('Ingrese el radio de la muesca de la
sesión 3:')

```

```

c=0
while c==0

```

```

c=1
sesion_4=input('Ingrese la distancia de la sesión 4:')
if (sesion_4 <= sesion_1 | sesion_4 <= sesion_2 | sesion_4
<= sesion_3 | sesion_4 >= longitud_eje)
c=0
disp('Distancia de la sesion 4 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca4=input('Ingrese el radio de la muesca de la
sesión 4:')

```

```

c=0
while c==0
c=1
sesion_5=input('Ingrese la distancia de la sesion 5:')
if (sesion_5 <= sesion_4 | sesion_5 <= sesion_3 | sesion_5
<= sesion_2 | sesion_5 <= sesion_1 | sesion_5 >= longitud_eje)
c=0
disp('Distancia de la sesion 5 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca5=input('Ingrese el radio de la muesca de la
sesión 5:')

```

```

c=0
while c==0
c=1
sesion_6=input('Ingrese la distancia de la sesion 6:')
if (sesion_6 <= sesion_5 | sesion_6 <= sesion_4 | sesion_6 <=
sesion_3 | sesion_6 <= sesion_2 | sesion_6 <= sesion_1 |
sesion_6 >= longitud_eje)
c=0
disp('Distancia de la sesion 6 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca6=input('Ingrese el radio de la muesca de la
sesión 6:')

```

```

c=0
while c==0
c=1
sesion_7=input('Ingrese la distancia de la sesión 7:')
if (sesion_7 <= sesion_6 | sesion_7 <= sesion_5 | sesion_7
<= sesion_4 | sesion_7 <= sesion_3 | sesion_7 <= sesion_2 |
sesion_7 <= sesion_1 | sesion_7 >= longitud_eje)
c=0
disp('Distancia de la sesion 7 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca7=input('Ingrese el radio de la muesca de la
sesión 7:')

```

```

c=0
while c==0
c=1
sesion_8=input('Ingrese la distancia de la sesión 8:')
if (sesion_8 <= sesion_7 | sesion_8 <= sesion_6 | sesion_8
<= sesion_5 | sesion_8 <= sesion_4 | sesion_8 <= sesion_3 |

```

```

sesion_8 <= sesion_2 | sesion_8 <= sesion_1 | sesion_8 >=
longitud_eje)
    c=0
    disp('Distancia de la sesion 8 incorrecta. Escriba resume
para volver a ingresar el valor')
    pause
end
end
Radio_muesca8=input('Ingrese el radio de la muesca de la
sesión 8: ')

```

```

//MÓDULO DE ELASTICIDAD DEL ACERO:
E= 30000

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 1:
exec('11.1.1_DistanciasFuerzasAnguloUbicacionEngraneUno.
sci',-1)

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 2:
exec('11.1.2_DistanciasFuerzasAnguloUbicacionEngraneDos.
ci',-1)

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 3:
exec('11.1.3_DistanciasFuerzasAnguloUbicacionEngraneTres.
sci',-1)

```

12. 2_OpcionesInterferenciaTres.sci

```

clc
clear all

disp("OPCIONES PARA EL TIPO DE AJUSTE POR
INTERFERENCIA")
disp("1- Ajuste de interferencia localizada H7/P6")
disp("2- Ajuste de impulso medio H7/S6")
disp("3- Ajuste forzado H7/U6")

interferencia3 = input('Seleccione la opción de ajuste por
interferencia deseada para el tercer engrane: ')
while c==0
c=1
if (interferencia3<1 | interferencia3>3)
    c=0
    disp('Opción incorrecta. Escriba resume para volver a ver el
menu anterior')
    pause
end
end
end

```

12. 3_InterferenciaUno.sci

```

clc
clear all
c=0
while c==0
c=1
tamano_basico3 = input('Ingrese el tamaño básico:')
p2=0;
if (tamano_basico3<0 | tamano_basico3>16)

```

```

    c=0
    disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
    pause
end
end
p3=0;

if tamano_basico3 >=0 & tamano_basico3 < 0.12 then
    p3=tabla14(1,1)
end

if tamano_basico3 >= 0.12 & tamano_basico3 < 0.24 then
    p3=tabla14(2,1)
end

if tamano_basico3 >= 0.24 & tamano_basico3 < 0.40 then
    p3=tabla14(3,1)
end

if tamano_basico3 >= 0.40 & tamano_basico3 < 0.72 then
    p3=tabla14(4,1)
end

if tamano_basico3 >= 0.72 & tamano_basico3 < 0.96 then
    p3=tabla14(5,1)
end

if tamano_basico3 >= 0.96 & tamano_basico3 < 1.20 then
    p3=tabla14(6,1)
end

if tamano_basico3 >= 1.20 & tamano_basico3 < 1.60 then
    p3=tabla14(7,1)
end

if tamano_basico3 >= 1.60 & tamano_basico3 < 2.00 then
    p3=tabla14(8,1)
end

if tamano_basico3 >= 2.00 & tamano_basico3 < 2.60 then
    p3=tabla14(9,1)
end

if tamano_basico3 >= 2.60 & tamano_basico3 < 3.20 then
    p3=tabla14(10,1)
end

if tamano_basico3 >= 3.20 & tamano_basico3 < 4.00 then
    p3=tabla14(11,1)
end

if tamano_basico3 >= 4.00 & tamano_basico3 < 4.80 then
    p3=tabla14(12,1)
end

if tamano_basico3 >= 4.80 & tamano_basico3 < 5.60 then
    p3=tabla14(13,1)
end

if tamano_basico3 >= 5.60 & tamano_basico3 < 6.40 then
    p3=tabla14(14,1)
end

if tamano_basico3 >= 6.40 & tamano_basico3 < 7.20 then
    p3=tabla14(15,1)
end
end

```

```

if tamaño_basico3 >= 7.20 & tamaño_basico3 < 8.00 then
    p3=tabla14(16,1)
end

if tamaño_basico3 >= 8.00 & tamaño_basico3 < 9.00 then
    p3=tabla14(17,1)
end

if tamaño_basico3 >= 9.00 & tamaño_basico3 < 10.00 then
    p3=tabla14(18,1)
end

if tamaño_basico3 >= 10.00 & tamaño_basico3 < 11.20 then
    p3=tabla14(19,1)
end

if tamaño_basico3 >= 11.20 & tamaño_basico3 < 12.60 then
    p3=tabla14(20,1)
end

if tamaño_basico3 >= 12.60 & tamaño_basico3 < 14.20 then
    p3=tabla14(21,1)
end

if tamaño_basico3 >= 14.20 & tamaño_basico3 < 16.00 then
    p3=tabla14(22,1)
end

it6_3=0
it7_3=0

if tamaño_basico3>=0 & tamaño_basico3<0.12 then
    it6_3=Tabla_13(1,1)
    it7_3=Tabla_13(1,2)
end
if tamaño_basico3>=0.12 & tamaño_basico3<0.24 then
    it6_3=Tabla_13(2,1)
    it7_3=Tabla_13(2,2)
end
if tamaño_basico3>=0.24 & tamaño_basico3<0.40 then
    it6_3=Tabla_13(3,1)
    it7_3=Tabla_13(3,2)
end
if tamaño_basico3>=0.40 & tamaño_basico3<0.72 then
    it6_3=Tabla_13(4,1)
    it7_3=Tabla_13(4,2)
end
if tamaño_basico3>=0.72 & tamaño_basico3<1.20 then
    it6_3=Tabla_13(5,1)
    it7_3=Tabla_13(5,2)
end
if tamaño_basico3>=1.20 & tamaño_basico3<2.00 then
    it6_3=Tabla_13(6,1)
    it7_3=Tabla_13(6,2)
end
if tamaño_basico3>=2.00 & tamaño_basico3<3.20 then
    it6_3=Tabla_13(7,1)
    it7_3=Tabla_13(7,2)
end
if tamaño_basico3>=3.20 & tamaño_basico3<4.80 then
    it6_3=Tabla_13(8,1)
    it7_3=Tabla_13(8,2)
end
if tamaño_basico3 >= 4.80 & tamaño_basico3 < 7.20 then
    it6_3=Tabla_13(9,1)
    it7_3=Tabla_13(9,2)

```

```

end
if tamaño_basico3 >= 7.20 & tamaño_basico3 < 10.00 then
    it6_3=Tabla_13(10,1)
    it7_3=Tabla_13(10,2)
end
if tamaño_basico3 >= 10.00 & tamaño_basico3 < 12.60 then
    it6_3=Tabla_13(11,1)
    it7_3=Tabla_13(11,2)
end
if tamaño_basico3 >= 12.60 & tamaño_basico3 < 16.00 then
    it6_3=Tabla_13(12,1)
    it7_3=Tabla_13(12,2)
end

tolerancia_agujero3 = it7_3
tolerancia_eje3=it6_3
Diametro_max_agujero3=tamaño_basico3 +
tolerancia_agujero3
Diametro_min_agujero3=tamaño_basico3
Diametro_max_eje3= tamaño_basico3 + p3 + tolerancia_eje3
Diametro_min_eje3= tamaño_basico3 + p3

Interf_maza_eje_min3=Diametro_min_eje3-
Diametro_max_agujero3
Interf_maza_eje_max3=Diametro_max_eje3-
Diametro_min_agujero3

```

12. 4_InterferenciaDos.sci

```

clc
clear all

c=0
while c==0
    c=1
    tamaño_basico3 = input('Ingrese el tamaño básico:')
    s3=0;
    if (tamaño_basico3<0 | tamaño_basico3>16)
        c=0
        disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
        pause
    end
end
s3=0;

if tamaño_basico3 >=0 & tamaño_basico3 < 0.12 then
    s3=tabla14(1,2)
end

if tamaño_basico3 >= 0.12 & tamaño_basico3 < 0.24 then
    s3=tabla14(2,2)
end

if tamaño_basico3 >= 0.24 & tamaño_basico3 < 0.40 then
    s3=tabla14(3,2)
end

if tamaño_basico3 >= 0.40 & tamaño_basico3 < 0.72 then
    s3=tabla14(4,2)
end

if tamaño_basico3 >= 0.72 & tamaño_basico3 < 0.96 then
    s3=tabla14(5,2)
end

```

```

if tamano_basico3 >= 0.96 & tamano_basico3 < 1.20 then
s3=tabla14(6,2)
end

if tamano_basico3 >= 1.20 & tamano_basico3 < 1.60 then
s3=tabla14(7,2)
end

if tamano_basico3 >= 1.60 & tamano_basico3 < 2.00 then
s3=tabla14(8,2)
end

if tamano_basico3 >= 2.00 & tamano_basico3 < 2.60 then
s3=tabla14(9,2)
end

if tamano_basico3 >= 2.60 & tamano_basico3 < 3.20 then
s3=tabla14(10,2)
end

if tamano_basico3 >= 3.20 & tamano_basico3 < 4.00 then
s3=tabla14(11,2)
end

if tamano_basico3 >= 4.00 & tamano_basico3 < 4.80 then
s3=tabla14(12,2)
end

if tamano_basico3 >= 4.80 & tamano_basico3 < 5.60 then
s3=tabla14(13,2)
end

if tamano_basico3 >= 5.60 & tamano_basico3 < 6.40 then
s3=tabla14(14,2)
end

if tamano_basico3 >= 6.40 & tamano_basico3 < 7.20 then
s3=tabla14(15,2)
end

if tamano_basico3 >= 7.20 & tamano_basico3 < 8.00 then
s3=tabla14(16,2)
end

if tamano_basico3 >= 8.00 & tamano_basico3 < 9.00 then
s3=tabla14(17,2)
end

if tamano_basico3 >= 9.00 & tamano_basico3 < 10.00 then
s3=tabla14(18,2)
end

if tamano_basico3 >= 10.00 & tamano_basico3 < 11.20 then
s3=tabla14(19,2)
end

if tamano_basico3 >= 11.20 & tamano_basico3 < 12.60 then
s3=tabla14(20,2)
end

if tamano_basico3 >= 12.60 & tamano_basico3 < 14.20 then
s3=tabla14(21,2)
end

if tamano_basico3 >= 14.20 & tamano_basico3 < 16.00 then
s3=tabla14(22,2)

```

```

end

it6_3=0
it7_3=0

if tamano_basico3>=0 & tamano_basico3<0.12 then
it6_3=Tabla_13(1,1)
it7_3=Tabla_13(1,2)
end

if tamano_basico3>=0.12 & tamano_basico3<0.24 then
it6_3=Tabla_13(2,1)
it7_3=Tabla_13(2,2)
end

if tamano_basico3>=0.24 & tamano_basico3<0.40 then
it6_3=Tabla_13(3,1)
it7_3=Tabla_13(3,2)
end

if tamano_basico3>=0.40 & tamano_basico3<0.72 then
it6_3=Tabla_13(4,1)
it7_3=Tabla_13(4,2)
end

if tamano_basico3>=0.72 & tamano_basico3<1.20 then
it6_3=Tabla_13(5,1)
it7_3=Tabla_13(5,2)
end

if tamano_basico3>=1.20 & tamano_basico3<2.00 then
it6_3=Tabla_13(6,1)
it7_3=Tabla_13(6,2)
end

if tamano_basico3>=2.00 & tamano_basico3<3.20 then
it6_3=Tabla_13(7,1)
it7_3=Tabla_13(7,2)
end

if tamano_basico3>=3.20 & tamano_basico3<4.80 then
it6_3=Tabla_13(8,1)
it7_3=Tabla_13(8,2)
end

if tamano_basico3 >= 4.80 & tamano_basico3 < 7.20 then
it6_3=Tabla_13(9,1)
it7_3=Tabla_13(9,2)
end

if tamano_basico3 >= 7.20 & tamano_basico3 < 10.00 then
it6_3=Tabla_13(10,1)
it7_3=Tabla_13(10,2)
end

if tamano_basico3 >= 10.00 & tamano_basico3 < 12.60 then
it6_3=Tabla_13(11,1)
it7_3=Tabla_13(11,2)
end

if tamano_basico3 >= 12.60 & tamano_basico3 < 16.00 then
it6_3=Tabla_13(12,1)
it7_3=Tabla_13(12,2)
end

tolerancia_agujero3 = it7_3
tolerancia_eje3=it6_3
Diametro_max_agujero3=tamano_basico3 +
tolerancia_agujero3
Diametro_min_agujero3=tamano_basico3
Diametro_max_eje3= tamano_basico3 + s3 + tolerancia_eje3
Diametro_min_eje3= tamano_basico3 + s3

Interf_maza_eje_min3=Diametro_min_eje3-
Diametro_max_agujero3
Interf_maza_eje_max3=Diametro_max_eje3-
Diametro_min_agujero3

```

12. 5_InterferenciTres.sci

```
clc
clear all

c=0
while c==0
c=1
tamano_basico3 = input('Ingrese el tamaño básico:');
s3=0;
if (tamano_basico3<0 | tamano_basico3>16)
c=0
disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
pause
end
end
s3=0;

if tamano_basico3 >=0 & tamano_basico3 < 0.12 then
s3=tabla14(1,2)
end

if tamano_basico3 >= 0.12 & tamano_basico3 < 0.24 then
s3=tabla14(2,2)
end

if tamano_basico3 >= 0.24 & tamano_basico3 < 0.40 then
s3=tabla14(3,2)
end

if tamano_basico3 >= 0.40 & tamano_basico3 < 0.72 then
s3=tabla14(4,2)
end

if tamano_basico3 >= 0.72 & tamano_basico3 < 0.96 then
s3=tabla14(5,2)
end

if tamano_basico3 >= 0.96 & tamano_basico3 < 1.20 then
s3=tabla14(6,2)
end

if tamano_basico3 >= 1.20 & tamano_basico3 < 1.60 then
s3=tabla14(7,2)
end

if tamano_basico3 >= 1.60 & tamano_basico3 < 2.00 then
s3=tabla14(8,2)
end

if tamano_basico3 >= 2.00 & tamano_basico3 < 2.60 then
s3=tabla14(9,2)
end

if tamano_basico3 >= 2.60 & tamano_basico3 < 3.20 then
s3=tabla14(10,2)
end

if tamano_basico3 >= 3.20 & tamano_basico3 < 4.00 then
s3=tabla14(11,2)
end

if tamano_basico3 >= 4.00 & tamano_basico3 < 4.80 then
s3=tabla14(12,2)
end

if tamano_basico3 >= 4.80 & tamano_basico3 < 5.60 then
s3=tabla14(13,2)
end

if tamano_basico3 >= 5.60 & tamano_basico3 < 6.40 then
s3=tabla14(14,2)
end

if tamano_basico3 >= 6.40 & tamano_basico3 < 7.20 then
s3=tabla14(15,2)
end

if tamano_basico3 >= 7.20 & tamano_basico3 < 8.00 then
s3=tabla14(16,2)
end

if tamano_basico3 >= 8.00 & tamano_basico3 < 9.00 then
s3=tabla14(17,2)
end

if tamano_basico3 >= 9.00 & tamano_basico3 < 10.00 then
s3=tabla14(18,2)
end

if tamano_basico3 >= 10.00 & tamano_basico3 < 11.20 then
s3=tabla14(19,2)
end

if tamano_basico3 >= 11.20 & tamano_basico3 < 12.60 then
s3=tabla14(20,2)
end

if tamano_basico3 >= 12.60 & tamano_basico3 < 14.20 then
s3=tabla14(21,2)
end

if tamano_basico3 >= 14.20 & tamano_basico3 < 16.00 then
s3=tabla14(22,2)
end

it6_3=0
it7_3=0

if tamano_basico3>=0 & tamano_basico3<0.12 then
it6_3=Tabla_13(1,1)
it7_3=Tabla_13(1,2)
end

if tamano_basico3>=0.12 & tamano_basico3<0.24 then
it6_3=Tabla_13(2,1)
it7_3=Tabla_13(2,2)
end

if tamano_basico3>=0.24 & tamano_basico3<0.40 then
it6_3=Tabla_13(3,1)
it7_3=Tabla_13(3,2)
end

if tamano_basico3>=0.40 & tamano_basico3<0.72 then
it6_3=Tabla_13(4,1)
it7_3=Tabla_13(4,2)
end

if tamano_basico3>=0.72 & tamano_basico3<1.20 then
it6_3=Tabla_13(5,1)
it7_3=Tabla_13(5,2)
end

if tamano_basico3>=1.20 & tamano_basico3<2.00 then
it6_3=Tabla_13(6,1)
it7_3=Tabla_13(6,2)
end

end
```

```

if tamano_basico3 >= 2.00 & tamano_basico3 < 3.20 then
    it6_3=Tabla_13(7,1)
    it7_3=Tabla_13(7,2)
end
if tamano_basico3 >= 3.20 & tamano_basico3 < 4.80 then
    it6_3=Tabla_13(8,1)
    it7_3=Tabla_13(8,2)
end
if tamano_basico3 >= 4.80 & tamano_basico3 < 7.20 then
    it6_3=Tabla_13(9,1)
    it7_3=Tabla_13(9,2)
end
if tamano_basico3 >= 7.20 & tamano_basico3 < 10.00 then
    it6_3=Tabla_13(10,1)
    it7_3=Tabla_13(10,2)
end
if tamano_basico3 >= 10.00 & tamano_basico3 < 12.60 then
    it6_3=Tabla_13(11,1)
    it7_3=Tabla_13(11,2)
end
if tamano_basico3 >= 12.60 & tamano_basico3 < 16.00 then
    it6_3=Tabla_13(12,1)
    it7_3=Tabla_13(12,2)
end

tolerancia_agujero3 = it7_3
tolerancia_eje3=it6_3
Diametro_max_agujero3=tamano_basico3 +
tolerancia_agujero3
Diametro_min_agujero3=tamano_basico3
Diametro_max_eje3= tamano_basico3 + s3 + tolerancia_eje3
Diametro_min_eje3= tamano_basico3 + s3

Interf_maza_eje_min3=Diametro_min_eje3-
Diametro_max_agujero3
Interf_maza_eje_max3=Diametro_max_eje3-
Diametro_min_agujero3

```

12. 6_TorquesMomentosYGraficas.sci

```

clc
clear all
c=0
while c==0
    c=1
    tamano_basico3 = input('Ingrese el tamaño básico:')
    u3=0;
    if (tamano_basico3<0 | tamano_basico3>16)
        c=0
        disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
        pause
    end
end
u3=0;

if tamano_basico3 >= 0 & tamano_basico3 < 0.12 then
    u3=tabla14(1,3)
end

if tamano_basico3 >= 0.12 & tamano_basico3 < 0.24 then
    u3=tabla14(2,3)
end

if tamano_basico3 >= 0.24 & tamano_basico3 < 0.40 then

```

```

    u3=tabla14(3,3)
end

if tamano_basico3 >= 0.40 & tamano_basico3 < 0.72 then
    u3=tabla14(4,3)
end

if tamano_basico3 >= 0.72 & tamano_basico3 < 0.96 then
    u3=tabla14(5,3)
end

if tamano_basico3 >= 0.96 & tamano_basico3 < 1.20 then
    u3=tabla14(6,3)
end

if tamano_basico3 >= 1.20 & tamano_basico3 < 1.60 then
    u3=tabla14(7,3)
end

if tamano_basico3 >= 1.60 & tamano_basico3 < 2.00 then
    u3=tabla14(8,3)
end

if tamano_basico3 >= 2.00 & tamano_basico3 < 2.60 then
    u3=tabla14(9,3)
end

if tamano_basico3 >= 2.60 & tamano_basico3 < 3.20 then
    u3=tabla14(10,3)
end

if tamano_basico3 >= 3.20 & tamano_basico3 < 4.00 then
    u3=tabla14(11,3)
end

if tamano_basico3 >= 4.00 & tamano_basico3 < 4.80 then
    u3=tabla14(12,3)
end

if tamano_basico3 >= 4.80 & tamano_basico3 < 5.60 then
    u3=tabla14(13,3)
end

if tamano_basico3 >= 5.60 & tamano_basico3 < 6.40 then
    u3=tabla14(14,3)
end

if tamano_basico3 >= 6.40 & tamano_basico3 < 7.20 then
    u3=tabla14(15,3)
end

if tamano_basico3 >= 7.20 & tamano_basico3 < 8.00 then
    u3=tabla14(16,3)
end

if tamano_basico3 >= 8.00 & tamano_basico3 < 9.00 then
    u3=tabla14(17,3)
end

if tamano_basico3 >= 9.00 & tamano_basico3 < 10.00 then
    u3=tabla14(18,3)
end

if tamano_basico3 >= 10.00 & tamano_basico3 < 11.20 then
    u3=tabla14(19,3)
end
end

```



```

if tamano_basico3 >= 11.20 & tamano_basico3 < 12.60 then
    u3=tabla14(20,3)
end

if tamano_basico3 >= 12.60 & tamano_basico3 < 14.20 then
    u3=tabla14(21,3)
end

if tamano_basico3 >= 14.20 & tamano_basico3 < 16.00 then
    u3=tabla14(22,3)
end

it6_3=0
it7_3=0

if tamano_basico3>=0 & tamano_basico3<0.12 then
    it6_3=Tabla_13(1,1)
    it7_3=Tabla_13(1,2)
end
if tamano_basico3>=0.12 & tamano_basico3<0.24 then
    it6_3=Tabla_13(2,1)
    it7_3=Tabla_13(2,2)
end
if tamano_basico3>=0.24 & tamano_basico3<0.40 then
    it6_3=Tabla_13(3,1)
    it7_3=Tabla_13(3,2)
end
if tamano_basico3>=0.40 & tamano_basico3<0.72 then
    it6_3=Tabla_13(4,1)
    it7_3=Tabla_13(4,2)
end
if tamano_basico3>=0.72 & tamano_basico3<1.20 then
    it6_3=Tabla_13(5,1)
    it7_3=Tabla_13(5,2)
end
if tamano_basico3>=1.20 & tamano_basico3<2.00 then
    it6_3=Tabla_13(6,1)
    it7_3=Tabla_13(6,2)
end
if tamano_basico3>=2.00 & tamano_basico3<3.20 then
    it6_3=Tabla_13(7,1)
    it7_3=Tabla_13(7,2)
end
if tamano_basico3>=3.20 & tamano_basico3<4.80 then
    it6_3=Tabla_13(8,1)
    it7_3=Tabla_13(8,2)
end
if tamano_basico3 >= 4.80 & tamano_basico3 < 7.20 then
    it6_3=Tabla_13(9,1)
    it7_3=Tabla_13(9,2)
end
if tamano_basico3 >= 7.20 & tamano_basico3 < 10.00 then
    it6_3=Tabla_13(10,1)
    it7_3=Tabla_13(10,2)
end
if tamano_basico3 >= 10.00 & tamano_basico3 < 12.60 then
    it6_3=Tabla_13(11,1)
    it7_3=Tabla_13(11,2)
end
if tamano_basico3 >= 12.60 & tamano_basico3 < 16.00 then
    it6_3=Tabla_13(12,1)
    it7_3=Tabla_13(12,2)
end

tolerancia_agujero3 = it7_3
tolerancia_eje3=it6_3

```

```

Diametro_max_agujero3=tamano_basico3 +
tolerancia_agujero3
Diametro_min_agujero3=tamano_basico3
Diametro_max_eje3= tamano_basico3 + u3 + tolerancia_eje3
Diametro_min_eje3= tamano_basico3 + u3

Interf_maza_eje_min3=Diametro_min_eje3-
Diametro_max_agujero3
Interf_maza_eje_max3=Diametro_max_eje3-
Diametro_min_agujero3

```

12. 7 InteraccionesDeLosDiametros.sci

```

clc
clear all
tol=0.001

for Diametro_eje1=tol:tol:6

Radio_eje1=Diametro_eje1/2

// inercias y momentos polares
I1=(%pi*((Radio_eje1*2)^4))/4)
J1=I1*2
sigal=(Momento_max1*Radio_eje1/I1)

//FACTOR DE TAMAÑO
if (Diametro_eje1 < 0.11)
    f_tama1=1
end
if (Diametro_eje1 >= 0.11 & Diametro_eje1 <= 2)
    f_tama1=0.879*Diametro_eje1^-0.107
end
if (Diametro_eje1 >= 3 & Diametro_eje1 <= 10)
    f_tama1=0.91*(Diametro_eje1^(-0.157))
end

Am1=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio//
[Em]=spec(Am1)//Funcion para los autovalores
sigvmm1=(((Em(1)-Em(2))^2)+((Em(2)-Em(3))^2)+((Em(3)-
Em(1))^2))/2)^0.5 //Vmises medio
Aa1=[(sigal*f_carga_flexion/f_tama1) 0 0;0 0 0;0 0 0]
//tensor de esfuerzo alternante
[Ea]=spec(Aa1)//Funcion para los autovalores
sigvma1=(((Ea(1)-Ea(2))^2)+((Ea(2)-Ea(3))^2)+((Ea(3)-
Ea(1))^2))/2)^0.5 //Vmises alternante

V_misses_med_LN1=[sigvmm1,
Coef_variacion_carga*sigvmm1]
V_misses_alt_LN1=[sigvma1,
Coef_variacion_carga*sigvma1]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
D_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

[Div_Va_SE_1_Cuad, Desv_Div_Va_SE_1_Cuad]=
pot(D_Va_SE1,2)

```

```
Division_VaSE1 = [Div_Va_SE_1_Cuad,
Desv_Div_Va_SE_1_Cuad]
```

```
Division_VmSut1 = [0,0]
```

```
Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1(1)
```

```
Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)
```

```
ASME_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_ASME_1=ASME_1(2)/ASME_1(1)
CRITERIO_F=ASME_1
CF_CRITERIO_F=Cf_ASME_1
end
```

```
if criterio_falla == 2 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_VaSE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]
```

```
Division_VmSut1 = [0,0]
```

```
Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1(1)
```

```
Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)
```

```
GERBER_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GERBER_1=GERBER_1(2)/GERBER_1(1)
CRITERIO_F=GERBER_1
CF_CRITERIO_F=Cf_GERBER_1
end
```

```
if criterio_falla == 3 then
[Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]
```

```
Division_VmSut1 = [0,0]
```

```
Suma1_VmSut_VaSE=Division_Va_SE1(1)+Division_VmSut1(1)
```

```
Desv_Suma1_VmSut_VaSE=sqrt((Division_Va_SE1(2)/Division_Va_SE1(1))^2+(0)^2)
```

```
GOODMAN_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GOODMAN_1=GOODMAN_1(2)/GOODMAN_1(1)
CRITERIO_F=GOODMAN_1
CF_CRITERIO_F=Cf_GOODMAN_1
end
```

```
MEDCnormal_1=log(CRITERIO_F(1)-
((0.5)*(CF_CRITERIO_F)^2))
DESVnormal_1=CF_CRITERIO_F
```

```
[Prob_1,Conf_1]=cdfnorr("PQ",0,abs(MEDCnormal_1),DESVnormal_1)
```

```
DIAMETRO_EJE1=0
```

```
if Prob_1<=1.5*10^-6 & Prob_1>=0.5*10^-6 then
DIAMETRO_EJE1=Diametro_eje1
break;
end
```

```
end
disp("El diametro del tramo 1 es")
disp(DIAMETRO_EJE1)
```

```
for Diametro_eje2=tol:tol:6
```

```
Radio_eje2=Diametro_eje2/2
```

```
I2=(%pi*((Radio_eje2*2)^4))/4//inercia
J2=I2*2
```

```
// calculo de esfuerzos
sigma2=(Momento_max2*Radio_eje2/I2) // esfuerzo flexor maximo
```

```
//FACTOR DE TAMAÑO
```

```
if (Diametro_eje2 < 0.11)
f_tama2=1
end
if (Diametro_eje2 >= 0.11 & Diametro_eje2 <= 2)
f_tama2=0.879*Diametro_eje2^-0.107
end
if (Diametro_eje2 >= 3 & Diametro_eje2 <= 10)
f_tama2=0.91*(Diametro_eje2^(-0.157))
end
```

```
//concentracion de esfuerzo
```

```
h2=Radio_eje2-Radio_eje1
relacion2=h2/Radio_muesca
```

```
//Kt para esfuerzos de flexión (Ktf)
```

```
//0.1<=h/r<=2.0
```

```
C2_f1=0.947+1.206*sqrt(h2/Radio_muesca)-
0.131*(h2/Radio_muesca)
C2_f2=0.022-
3.405*sqrt(h2/Radio_muesca)+0.915*(h2/Radio_muesca)
C2_f3=0.869+1.777*sqrt(h2/Radio_muesca)-
0.555*(h2/Radio_muesca)
C2_f4=-0.810+0.422*sqrt(h2/Radio_muesca)-
0.260*(h2/Radio_muesca)
```

```
//2.0<=h/r<=20
```

```
C2_f11=1.232+0.832*sqrt(h2/Radio_muesca)-
0.008*(h2/Radio_muesca)
C2_f22=-3.813+0.968*sqrt(h2/Radio_muesca)-
0.260*(h2/Radio_muesca)
C2_f33=7.423-
4.868*sqrt(h2/Radio_muesca)+0.869*(h2/Radio_muesca)
C2_f44=-3.839+3.070*sqrt(h2/Radio_muesca)-
0.600*(h2/Radio_muesca)
```

```
if (relacion2 >=0.1 & relacion2 <= 2)
```

```
C2_F1=C2_f1
C2_F2=C2_f2
C2_F3=C2_f3
C2_F4=C2_f4
Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
end
```

```
if (relacion2 > 2 & relacion2 <= 20)
```

```
C2_F1=C2_f11
C2_F2=C2_f22
C2_F3=C2_f33
C2_F4=C2_f44
Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
```

```

end

if relacion2 < 0.1 then
    Kt2_f=1
end

if relacion2 > 20 then
    Kt2_f=1
end

KT2_F=Kt2_f

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f2=(2*(KT2_F-
1)/KT2_F)*(RAIZ_DE_A/sqrt(Radio_muesca))
Kf2_f=KT2_F/((1+constante_reemplazo_f2))

Am2=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio
[Em2]=spec(Am2)//Funcion para los autovalores
sigvmm2=(((Em2(1)-Em2(2))^2)+((Em2(2)-
Em2(3))^2)+((Em2(3)-Em2(1))^2))/2)^0.5 //Vmises medio
Aa2=[(sigma2*f_carga_flexion*Kf2_f/f_tama2) 0 0;0 0 0;0 0 0]
//tensor de esfuerzo alternante
[Ea2]=spec(Aa2)//Funcion para los autovalores
sigvma2=(((Ea2(1)-Ea2(2))^2)+((Ea2(2)-
Ea2(3))^2)+((Ea2(3)-Ea2(1))^2))/2)^0.5 //Vmises alternante

V_misses_med_LN2=[sigvmm2,
Coef_variacion_carga*sigvmm2]
V_misses_alt_LN2=[sigvma2,
Coef_variacion_carga*sigvma2]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_2,
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
D_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]

[Div_Va_SE_2_Cuad,          Desv_Div_Va_SE_2_Cuad]=
pot(D_Va_SE2,2)
Division_VaSE2          =          [Div_Va_SE_2_Cuad,
Desv_Div_Va_SE_2_Cuad]

Division_VmSut2 = [0,0]

Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2
(1)
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio
n_VaSE2(1))^2+(0)^2)

ASME_2=[Suma2_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_ASME_2=ASME_2(2)/ASME_2(1)
CRITERIO_F=ASME_2
CF_CRITERIO_F=Cf_ASME_2
end

if criterio_falla == 2 then
[Div_Va_SE_2,
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
Division_VaSE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]

```

```
Division_VmSut2 = [0,0]
```

```
Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2
(1)
```

```
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio
n_VaSE2(1))^2+(0)^2)
```

```
GERBER_2=[Suma2_VmSut_VaSE,
Desv_Suma2_VmSut_VaSE]
Cf_GERBER_2=GERBER_2(2)/GERBER_2(1)
CRITERIO_F=GERBER_2
CF_CRITERIO_F=Cf_GERBER_2
end
```

```
if criterio_falla == 3 then
```

```
[Div_Va_SE_2,
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
Division_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]
```

```
Division_VmSut2 = [0,0]
```

```
Suma2_VmSut_VaSE=Division_Va_SE2(1)+Division_VmSut2
(1)
```

```
Desv_Suma2_VmSut_VaSE=sqrt((Division_Va_SE2(2)/Divisi
on_Va_SE2(1))^2+(0)^2)
```

```
GOODMAN_2=[Suma2_VmSut_VaSE,
Desv_Suma2_VmSut_VaSE]
Cf_GOODMAN_2=GOODMAN_2(2)/GOODMAN_2(1)
CRITERIO_F=GOODMAN_2
CF_CRITERIO_F=Cf_GOODMAN_2
end
```

```
MEDCnormal_2=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_2=CF_CRITERIO_F
```

```
[Prob_2,Conf_2]=cdfnor("PQ",0,abs(MEDCnormal_2),DESV
normal_2)
```

```
DIAMETRO_EJE2=0
```

```
if Prob_2<=1.5*10^-6 & Prob_2>=0.5*10^-6 then
```

```
    DIAMETRO_EJE2=Diametro_eje2
    break;
end
end
```

```
disp("El diametro del tramo 2 es")
disp(DIAMETRO_EJE2)
```

```
for Diametro_eje3=tol:tol:6
```

```
Radio_eje3=Diametro_eje3/2 // Radio del eje en el tramo del
engrane 1
```

```
Torque3=(W_tangencial1/1000)*((Radio_engrane1)-
Radio_eje3) // torque generado por el primer engrane
I3=(%pi*(((Radio_eje3*2)^4)/4))/momento de inercia
J3=I3*3 // momento polar de inercia
// calculo de esfuerzos para tramo 3
sigma3=(Momento_max3*Radio_eje3/I3) // esfuerzo flector
maximo
taotormed3=(Torque3*Radio_eje3/J3) // esfuerzo cortante
torsor alternante
taotoralt3=(Torque3*Radio_eje3/J3) // esfuerzo cortante torsor
medio
```

//FACTOR DE TAMAÑO

```
if (Diametro_eje3 >= 0.11)
  f_tama3=1
end
```

```
if (Diametro_eje3 >= 0.11 & Diametro_eje3 <= 2)
  f_tama3=0.879*Diametro_eje3^(-0.107)
end
```

```
if (Diametro_eje3 >= 3 & Diametro_eje3 <= 10)
  f_tama3=0.91*(Diametro_eje3^(-0.157))
end
```

//factor de concentracion de esfuerzo

```
h3=Radio_eje3-Radio_eje2
relacion3=h3/Radio_muesca2
```

//0.1 <= h/r <= 2.0

```
C3_a1= 0.926+(1.157*sqrt(h3/Radio_muesca2))-
(0.099*(h3/Radio_muesca2))
C3_a2=0.012-
3.036*sqrt(h3/Radio_muesca2)+0.961*(h3/Radio_muesca2)
C3_a3=-0.302+3.977*sqrt(h3/Radio_muesca2)-
1.744*(h3/Radio_muesca2)
C3_a4=0.365-
2.098*sqrt(h3/Radio_muesca2)+0.878*(h3/Radio_muesca2)
```

//2.0 <= h/r <= 20

```
C3_a11=1.2+0.860*sqrt(h3/Radio_muesca2)-
0.022*(h3/Radio_muesca2)
C3_a22=-1.805-0.346*sqrt(h3/Radio_muesca2)-
0.038*(h3/Radio_muesca2)
C3_a33=2.198-
0.486*sqrt(h3/Radio_muesca2)+0.165*(h3/Radio_muesca2)
C3_a44=-0.593-0.028*sqrt(h3/Radio_muesca2)-
0.106*(h3/Radio_muesca2)
```

```
if (relacion3 >= 0.1 & relacion3 <= 2)
```

```
  C3_A1=C3_a1
  C3_A2=C3_a2
  C3_A3=C3_a3
  C3_A4=C3_a4
  Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
  Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end
```

```
if (relacion3 > 2 & relacion3 <= 20)
```

```
  C3_A1=C3_a11
  C3_A2=C3_a22
  C3_A3=C3_a33
  C3_A4=C3_a44
  Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
  Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end
```

```
if relacion3 < 0.1 then
```

```
  Kt3_a=1
end
```

```
if relacion3 > 20 then
```

```
  Kt3_a=0
end
```

```
KT3_A=Kt3_a
```

//Kt para esfuerzos de flexión (Ktf)

//0.1 <= h/r <= 2.0

```
C3_f1=0.947+1.206*sqrt(h3/Radio_muesca2)-
0.131*(h3/Radio_muesca2)
C3_f2=0.022-
3.405*sqrt(h3/Radio_muesca2)+0.915*(h3/Radio_muesca2)
C3_f3=0.869+1.777*sqrt(h3/Radio_muesca2)-
0.555*(h3/Radio_muesca2)
C3_f4=-0.810+0.422*sqrt(h3/Radio_muesca2)-
0.260*(h3/Radio_muesca2)
```

//2.0 <= h/r <= 20

```
C3_f11=1.232+0.832*sqrt(h3/Radio_muesca2)-
0.008*(h3/Radio_muesca2)
C3_f22=-3.813+0.968*sqrt(h3/Radio_muesca2)-
0.260*(h3/Radio_muesca2)
C3_f33=7.423-
4.868*sqrt(h3/Radio_muesca2)+0.869*(h3/Radio_muesca2)
C3_f44=-3.839+3.070*sqrt(h3/Radio_muesca2)-
0.600*(h3/Radio_muesca2)
```

```
if (relacion3 >= 0.1 & relacion3 <= 2)
```

```
  C3_F1=C3_f1
  C3_F2=C3_f2
  C3_F3=C3_f3
  C3_F4=C3_f4
  Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
  Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end
```

```
if (relacion3 > 2 & relacion3 <= 20)
```

```
  C3_F1=C3_f11
  C3_F2=C3_f22
  C3_F3=C3_f33
  C3_F4=C3_f44
  Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
  Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end
```

```
if relacion3 < 0.1 then
```

```
  Kt3_f=1
end
```

```
if relacion3 > 20 then
```

```
  Kt3_f=1
end
KT3_F=Kt3_f
```

//Kt para esfuerzos de torsión (Ktf)

//0.25 <= h/r <= 4.0

```
C3_t1=0.905+0.783*sqrt(h3/Radio_muesca2)-
0.075*(h3/Radio_muesca2)
C3_t2=-0.437-
1.969*sqrt(h3/Radio_muesca2)+0.553*(h3/Radio_muesca2)
C3_t3=1.557+1.073*sqrt(h3/Radio_muesca2)-
0.578*(h3/Radio_muesca2)
C3_t4=-1.061+0.171*sqrt(h3/Radio_muesca2)-
0.086*(h3/Radio_muesca2)
```

```
if (relacion3 >= 0.25 & relacion3 <= 4.0)
```

```
  C3_T1=C3_t1
  C3_T2=C3_t2
  C3_T3=C3_t3
  C3_T4=C3_t4
  Kt3_t=C3_T1+C3_T2*(2*h3/Diametro_eje3)+C3_T3*(2*h3/
  Diametro_eje3)^2+C3_T4*(2*h3/Diametro_eje3)^3
```

```

end
if relacion3 < 0.25 then
  Kt3_t=1
end
if relacion3 > 4.0 then
  Kt3_t=1
end
KT3_T=Kt3_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)
constante_reemplazo_a3=(2*(KT3_A-
1)/KT3_A)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_a=KT3_A/((1+constante_reemplazo_a3))

constante_reemplazo_f3=(2*(KT3_F-
1)/KT3_F)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_f=KT3_F/((1+constante_reemplazo_f3))

constante_reemplazo_t3=(2*(KT3_T-
1)/KT3_T)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_t=KT3_T/((1+constante_reemplazo_t3))

//presion por interferencia
m3=(((2*Radio_engrane1)^2-
Diametro_eje3^2)*(Diametro_eje3^2))
n3=(2*Radio_engrane1)
PP3=((E*Interferen_maza_eje_max)/(2*Diametro_eje3^3))*(
m3/n3)

Am3=[(PP3*f_carga_axial*Kf3_a/f_tama3)
(taotormed3*f_carga_torsion*Kf3_t/f_tama3)
0;(taotormed3*f_carga_torsion*Kf3_t/f_tama3) 0 0;0 0 0]
//tensor de esfuerzo medio
[Em3]=spec(Am3)
sigvmm3=(((Em3(1)-Em3(2))^2)+(Em3(2)-
Em3(3))^2)+((Em3(3)-Em3(1))^2)/2)^0.5//calculo esfuerzo
Von Mises medio
Aa3=[(siga3*f_carga_flexion*Kf3_f/f_tama3)
(taotort3*f_carga_torsion*Kf3_t/f_tama3) 0;0
(taotort3*f_carga_torsion*Kf3_t/f_tama3) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea3]=spec(Aa3)
sigma3=(((Ea3(1)-Ea3(2))^2)+(Ea3(2)-
Ea3(3))^2)+((Ea3(3)-Ea3(1))^2)/2)^0.5//calculo esfuerzo Von
Mises medio

V_misses_med_LN3=[sigvmm3,
Coef_variacion_carga*sigvmm3]
V_misses_alt_LN3=[sigvma3,
Coef_variacion_carga*sigvma3]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
D_Va_SE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

[Div_Va_SE_3_Cuad, Desv_Div_Va_SE_3_Cuad]=
pot(D_Va_SE3,2)
Division_VaSE3 = [Div_Va_SE_3_Cuad,
Desv_Div_Va_SE_3_Cuad]

[Div_Va_SE_3_Cuad, Desv_Div_Va_SE_3_Cuad]=
pot(D_Va_SE3,2)
Division_VaSE3 = [Div_Va_SE_3_Cuad,
Desv_Div_Va_SE_3_Cuad]

if criterio_falla == 1 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
D_Va_SE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

[Div_Va_SE_3_Cuad, Desv_Div_Va_SE_3_Cuad]=
pot(D_Va_SE3,2)
Division_VaSE3 = [Div_Va_SE_3_Cuad,
Desv_Div_Va_SE_3_Cuad]

```

```

[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

[Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)

ASME_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_ASME_3=ASME_3(2)/ASME_3(1)
CRITERIO_F=ASME_3
CF_CRITERIO_F=Cf_ASME_3
end

if criterio_falla == 2 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

[Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)

GERBER_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GERBER_3=GERBER_3(2)/GERBER_3(1)
CRITERIO_F=GERBER_3
CF_CRITERIO_F=Cf_GERBER_3
end

if criterio_falla == 3 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]
[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
Division_VmSut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)
GOODMAN_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GOODMAN_3=GOODMAN_3(2)/GOODMAN_3(1)
CRITERIO_F=GOODMAN_3
CF_CRITERIO_F=Cf_GOODMAN_3
end

MEDCnormal_3=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESNnormal_3=CF_CRITERIO_F

[Prob_3,Conf_3]=cdfnor("PQ",0,abs(MEDCnormal_3),DESN
normal_3)

DIAMETRO_EJE3=0

if Prob_3<=1.5*10^-6 & Prob_3>=0.5*10^-6 then

```

```

DIAMETRO_EJE3=Diametro_eje3
break;
end
end
disp("El diametro del tramo 3 es ")
disp(DIAMETRO_EJE3)
for Diametro_eje4=tol:tol:6

Radio_eje4=Diametro_eje4/2
Torque4=(W_tangencial1/1000)*((Radio_engrane1)-
Radio_eje3)
I4=(%pi*(((Radio_eje4*2)^4))/4)/inercia
J4=I4*2
// calculo de esfuerzos
sigma4=(Momento_max4*Radio_eje4/I4) // esfuerzo flector
maximo
taotormed4=(Torque4*Radio_eje4/J4) // esfuerzo cortante
torsor maximo
taotoralt4=(Torque4*Radio_eje4/J4)

//FACTOR DE TAMAÑO
if (Diametro_eje4 < 0.11)
f_tama4=1
end
if (Diametro_eje4 >= 0.11 & Diametro_eje4 <= 2)
f_tama4=0.879*Diametro_eje4^(-0.107)
end
if (Diametro_eje4 >= 3 & Diametro_eje4 <= 10)
f_tama4=0.91*(Diametro_eje4^(-0.157))
end

//factor de concentracion de esfuerzo
h4=Radio_eje4-Radio_eje3
relacion4=h4/Radio_muesca3

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C4_f1=0.947+1.206*sqrt(h4/Radio_muesca3)-
0.131*(h4/Radio_muesca3)
C4_f2=0.022-
3.405*sqrt(h4/Radio_muesca3)+0.915*(h4/Radio_muesca3)
C4_f3=0.869+1.777*sqrt(h4/Radio_muesca3)-
0.555*(h4/Radio_muesca3)
C4_f4=-0.810+0.422*sqrt(h4/Radio_muesca3)-
0.260*(h4/Radio_muesca3)

//2.0<=h/r<=20
C4_f11=1.232+0.832*sqrt(h4/Radio_muesca3)-
0.008*(h4/Radio_muesca3)
C4_f22=-3.813+0.968*sqrt(h4/Radio_muesca3)-
0.260*(h4/Radio_muesca3)
C4_f33=7.423-
4.868*sqrt(h4/Radio_muesca3)+0.869*(h4/Radio_muesca3)
C4_f44=-3.839+3.070*sqrt(h4/Radio_muesca3)-
0.600*(h4/Radio_muesca3)

if (relacion4 >=0.1 & relacion4 <= 2)
C4_F1=C4_f1
C4_F2=C4_f2
C4_F3=C4_f3
C4_F4=C4_f4
Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if (relacion4 > 2 & relacion4 <= 20)
C4_F1=C4_f11
C4_F2=C4_f22
C4_F3=C4_f33
C4_F4=C4_f44
Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if relacion4 < 0.1 then
Kt4_f=1
end
if relacion4 > 20 then
Kt4_f=1
end

KT4_F=Kt4_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C4_t1=0.905+0.783*sqrt(h4/Radio_muesca3)-
0.075*(h4/Radio_muesca3)
C4_t2=-0.437-
1.969*sqrt(h4/Radio_muesca3)+0.553*(h4/Radio_muesca3)
C4_t3=1.557+1.073*sqrt(h4/Radio_muesca3)-
0.578*(h4/Radio_muesca3)
C4_t4=-1.061+0.171*sqrt(h4/Radio_muesca3)-
0.086*(h4/Radio_muesca3)

if (relacion4 >=0.25 & relacion4 <= 4.0)
C4_T1=C4_t1
C4_T2=C4_t2
C4_T3=C4_t3
C4_T4=C4_t4
Kt4_t=C4_T1+C4_T2*(2*h4/Diametro_eje4)+C4_T3*(2*h4/
Diametro_eje4)^2+C4_T4*(2*h4/Diametro_eje4)^3
end

if relacion4 < 0.25 then
Kt4_t=1
end
if relacion4 > 4.0 then
Kt4_t=1
end
KT4_T=Kt4_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f4=(2*(KT4_F-
1)/KT4_F)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf4_f=KT4_F/((1+constante_reemplazo_f4))

constante_reemplazo_t4=(2*(KT4_T-
1)/KT4_T)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf4_t=KT4_T/((1+constante_reemplazo_t4))

Am4=[0 (taotormed4*f_carga_torsion*Kf4_t/f_tama4)
0;(taotormed4*f_carga_torsion*Kf4_t/f_tama4) 0 0; 0 0]
//tensor de esfuerzo medio
[Em4]=spec(Am4)
sigvmm4=(((Em4(1)-Em4(2))^2)+(Em4(2)-
Em4(3))^2)+(Em4(3)-Em4(1))^2)/2^0.5//Mises medio
Aa4=((sigma4*f_carga_flexion*Kf4_f/f_tama4)
(taotoralt4*f_carga_torsion*Kf4_t/f_tama4) 0;0
(taotoralt4*f_carga_torsion*Kf4_t/f_tama4) 0;0 0] //tensor
de esfuerzo alternante

```

```

[Ea4]=spec(Aa4)
sigvma4=(((Ea4(1)-Ea4(2))^2)+(Ea4(2)-Ea4(3))^2)+(Ea4(3)-Ea4(1))^2)/2^0.5//Vmises alternante

V_misses_med_LN4=[sigvmm4,
Coef_variacion_carga*sigvmm4]
V_misses_alt_LN4=[sigvma4,
Coef_variacion_carga*sigvma4]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
D_Va_SE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

[Div_Va_SE_4_Cuad,          Desv_Div_Va_SE_4_Cuad]=
pot(D_Va_SE4,2)
Division_VaSE4          =          [Div_Va_SE_4_Cuad,
Desv_Div_Va_SE_4_Cuad]

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Div_Vm_Sut_4_Cuad,          Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4          =          [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

ASME_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_ASME_4=ASME_4(2)/ASME_4(1)
CRITERIO_F=ASME_4
CF_CRITERIO_F=Cf_ASME_4
end

if criterio_falla == 2 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Div_Vm_Sut_4_Cuad,          Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4          =          [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

GERBER_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GERBER_4=GERBER_4(2)/GERBER_4(1)
CRITERIO_F=GERBER_4
CF_CRITERIO_F=Cf_GERBER_4
end

if criterio_falla == 3 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Div_Vm_Sut_4_Cuad,          Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4          =          [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

GOODMAN_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GOODMAN_4=GOODMAN_4(2)/GOODMAN_4(1)
CRITERIO_F=GOODMAN_4
CF_CRITERIO_F=Cf_GOODMAN_4
end

MEDCnormal_4=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_4=CF_CRITERIO_F

[Prob_4,Conf_4]=cdfnorr("PQ",0,abs(MEDCnormal_4),DESV
normal_4)

DIAMETRO_EJE4=0

if Prob_4<=1.5*10^-6 & Prob_4>=0.5*10^-6 then
DIAMETRO_EJE4=Diametro_eje4
break;
end
end
disp("El diametro del tramo 4 es")
disp(DIAMETRO_EJE4)

for Diametro_eje5=tol:tol:6

Radio_eje5=Diametro_eje5/2

Torque5=(W_tangencial2/1000)*((Radio engrane2)-
Radio_eje5) // el torque aplicado es la fuerza tangencial por
el Radio del engranaje menos el Radio del eje
I5=(%pi*((Radio_eje5^2)^4))/4//ineracia
Q5=(2/3)*(Radio_eje5^3)
J5=I5*2

//calculo de esfuerzos
siga5=(Momento_max5*Radio_eje5/I5) // esfuerzo flexor
maximo
taotormed5=(Torque5*Radio_eje5/J5) // esfuerzo cortante
torsor maximo
taotoralt5=(Torque5*Radio_eje5/J5)

//FACTOR DE TAMAÑO
if (Diametro_eje5 < 0.11)
f_tama5=1
end
if (Diametro_eje5 >= 0.11 & Diametro_eje5 <= 2)
f_tama5=0.879*Diametro_eje5^-0.107
end
if (Diametro_eje5 >= 3 & Diametro_eje5 <= 10)
f_tama5=0.91*(Diametro_eje5^(-0.157))
end
//factor de concentracion de esfuerzo
h5=Radio_eje5-Radio_eje4
relacion5=h5/Radio_muesca4

//0.1<=h/r<=2.0
C5_a1=          0.926+(1.157*sqrt(h5/Radio_muesca4))-
(0.099*(h5/Radio_muesca4))

```



```

C5_a2=0.012-
3.036*sqrt(h5/Radio_muesca4)+0.961*(h5/Radio_muesca4)
C5_a3=-0.302+3.977*sqrt(h5/Radio_muesca4)-
1.744*(h5/Radio_muesca4)
C5_a4=0.365-
2.098*sqrt(h5/Radio_muesca4)+0.878*(h5/Radio_muesca4)

```

```

//2.0<=h/r<=20
C5_a11=1.2+0.860*sqrt(h5/Radio_muesca4)-
0.022*(h5/Radio_muesca4)
C5_a22=-1.805-0.346*sqrt(h5/Radio_muesca4)-
0.038*(h5/Radio_muesca4)
C5_a33=2.198-
0.486*sqrt(h5/Radio_muesca4)+0.165*(h5/Radio_muesca4)
C5_a44=-0.593-0.028*sqrt(h5/Radio_muesca4)-
0.106*(h5/Radio_muesca4)

```

```

if (relacion5 >=0.1 & relacion5 <= 2)
    C5_A1=C5_a1
    C5_A2=C5_a2
    C5_A3=C5_a3
    C5_A4=C5_a4
    Kt5_a=C5_A1+C5_A2*(2*h5/Diametro_eje5)+C5_A3*(2*h5/
    Diametro_eje5)^2+C5_A4*(2*h5/Diametro_eje5)^3
end

```

```

if (relacion5 > 2 & relacion5 <= 20)
    C5_A1=C5_a11
    C5_A2=C5_a22
    C5_A3=C5_a33
    C5_A4=C5_a44
    Kt5_a=C5_A1+C5_A2*(2*h5/Diametro_eje5)+C5_A3*(2*h5/
    Diametro_eje5)^2+C5_A4*(2*h5/Diametro_eje5)^3
end
if relacion5 < 0.1 then
    Kt5_a=1
end

```

```

if relacion5 > 20 then
    Kt5_a=1
end
KT5_A=Kt5_a
//Kt para esfuerzos de flexión (Kf)

```

```

//0.1<=h/r<=2.0
C5_f1=0.947+1.206*sqrt(h5/Radio_muesca4)-
0.131*(h5/Radio_muesca4)
C5_f2=0.022-
3.405*sqrt(h5/Radio_muesca4)+0.915*(h5/Radio_muesca4)
C5_f3=0.869+1.777*sqrt(h5/Radio_muesca4)-
0.555*(h5/Radio_muesca4)
C5_f4=-0.810+0.422*sqrt(h5/Radio_muesca4)-
0.260*(h5/Radio_muesca4)

```

```

//2.0<=h/r<=20
C5_f11=1.232+0.832*sqrt(h5/Radio_muesca4)-
0.008*(h5/Radio_muesca4)
C5_f22=-3.813+0.968*sqrt(h5/Radio_muesca4)-
0.260*(h5/Radio_muesca4)
C5_f33=7.423-
4.868*sqrt(h5/Radio_muesca4)+0.869*(h5/Radio_muesca4)
C5_f44=-3.839+3.070*sqrt(h5/Radio_muesca4)-
0.600*(h5/Radio_muesca4)

```

```

if (relacion5 >=0.1 & relacion5 <= 2)
    C5_F1=C5_f1
    C5_F2=C5_f2

```

```

C5_F3=C5_f3
C5_F4=C5_f4
Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

```

```

if (relacion5 > 2 & relacion5 <= 20)
    C5_F1=C5_f11
    C5_F2=C5_f22
    C5_F3=C5_f33
    C5_F4=C5_f44
    Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
    Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

```

```

if relacion5 < 0.1 then
    Kt5_f=1
end
if relacion5 > 20 then
    Kt5_f=1
end
KT5_F=Kt5_f

```

//Kt para esfuerzos de torsión (Ktf)

```

//0.25<=h/r<=4.0
C5_t1=0.905+0.783*sqrt(h5/Radio_muesca4)-
0.075*(h5/Radio_muesca4)
C5_t2=-0.437-
1.969*sqrt(h5/Radio_muesca4)+0.553*(h5/Radio_muesca4)
C5_t3=1.557+1.073*sqrt(h5/Radio_muesca4)-
0.578*(h5/Radio_muesca4)
C5_t4=-1.061+0.171*sqrt(h5/Radio_muesca4)-
0.086*(h5/Radio_muesca4)

```

```

if (relacion5 >=0.25 & relacion5 <= 4.0)
    C5_T1=C5_t1
    C5_T2=C5_t2
    C5_T3=C5_t3
    C5_T4=C5_t4
    Kt5_t=C5_T1+C5_T2*(2*h5/Diametro_eje5)+C5_T3*(2*h5/
    Diametro_eje5)^2+C5_T4*(2*h5/Diametro_eje5)^3
end
if relacion5 < 0.25 then
    Kt5_t=1
end
if relacion5 > 4.0 then
    Kt5_t=1
end
KT5_T=Kt5_t

```

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial), (Kf_f-flexión), (Kf_t-torsión)
constante_reemplazo_a5=(2*(KT5_A-1)/KT5_A)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_a=KT5_A/((1+constante_reemplazo_a5))

```

constante_reemplazo_f5=(2*(KT5_F-1)/KT5_F)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_f=KT5_F/((1+constante_reemplazo_f5))

```

```

constante_reemplazo_t5=(2*(KT5_T-1)/KT5_T)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_t=KT5_T/((1+constante_reemplazo_t5))

```

//presion por interferencia


```

m2=(((2*Radio_engrane2)^2-
Diametro_eje5^2))*(Diametro_eje5^2))
n2=(2*Radio_engrane2)
PP5=(E*Interferen_maza_eje_max2)/(2*Diametro_eje5^3))*(
m2/n2)

```

```

Am5=[(PP5*f_carga_axial*Kf5_a/f_tama5)
(taotormed5*f_carga_torsion*Kf5_t/f_tama5)
0;(taotormed5*f_carga_torsion*Kf5_t/f_tama5) 0 0;0 0 0]
//tensor de esfuerzo medio//
[Ea5]=spec(Aa5)
sigvmm5=(((Ea5(1)-Ea5(2))^2)+((Ea5(2)-
Ea5(3))^2)+((Ea5(3)-Ea5(1))^2))/2)^0.5

```

```

Aa5=[(siga5*f_carga_flexion*Kf5_f/f_tama5)
(taotoralt5*f_carga_torsion*Kf5_t/f_tama5) 0;0
(taotoralt5*f_carga_torsion*Kf5_t/f_tama5) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea5]=spec(Aa5)
sigvma5=(((Ea5(1)-Ea5(2))^2)+((Ea5(2)-
Ea5(3))^2)+((Ea5(3)-Ea5(1))^2))/2)^0.5

```

```

V_misses_med_LN5=[sigvmm5,
Coef_variacion_carga*sigvmm5]
V_misses_alt_LN5=[sigvma5,
Coef_variacion_carga*sigvma5]

```

```
[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)
```

```
SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]
```

```

if criterio_falla == 1 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
D_Va_SE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

```

```

[Div_Va_SE_5_Cuad, Desv_Div_Va_SE_5_Cuad]=
pot(D_Va_SE5,2)
Division_VaSE5 = [Div_Va_SE_5_Cuad,
Desv_Div_Va_SE_5_Cuad]

```

```

[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

```

```

[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]

```

```
[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
```

```

ASME_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_ASME_5=ASME_5(2)/ASME_5(1)
CRITERIO_F=ASME_5
CF_CRITERIO_F=Cf_ASME_5
end

```

```

if criterio_falla == 2 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

```

```

[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)

```

```
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]
```

```

[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]

```

```
[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
```

```

GERBER_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_GERBER_5=GERBER_5(2)/GERBER_5(1)
CRITERIO_F=GERBER_5
CF_CRITERIO_F=Cf_GERBER_5
end

```

```

if criterio_falla == 3 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
Division_VmSut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

```

```

[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
GOODMAN_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_GOODMAN_5=GOODMAN_5(2)/GOODMAN_5(1)
CRITERIO_F=GOODMAN_5
CF_CRITERIO_F=Cf_GOODMAN_5
end

```

```

MEDCnormal_5=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_5=CF_CRITERIO_F

```

```
[Prob_5,Conf_5]=cdfnor("PQ",0,abs(MEDCnormal_5),DESV
normal_5)
```

```
DIAMETRO_EJE5=0
```

```

if Prob_5<=1.5*10^-6 & Prob_5>=0.5*10^-6 then
DIAMETRO_EJE5=Diametro_eje5
break;
end
end

```

```

disp("El diametro del tramo 5 es")
disp(DIAMETRO_EJE5)

```

```
for Diametro_eje6=0+tol:tol:10
```

```

Radio_eje6=Diametro_eje6/2
Torque6=(W_tangencial2/1000)*((Radio_engrane2)-
Radio_eje5)
I6=(%pi*(((Radio_eje6*2)^4))/4)/inercia
J6=I6*2

```

```

// calculo de esfuerzos
siga6=(Momento_max6*Radio_eje6/I6) // esfuerzo flector
maximo
taotormed6=(Torque6*Radio_eje6/J6) // esfuerzo cortante
torsor maximo
taotoralt6=(Torque6*Radio_eje6/J6)

```

```

//FACTOR DE TAMAÑO
if (Diametro_eje6 < 0.11)
f_tama6=1
end

```

```

if (Diametro_eje6 >= 0.11 & Diametro_eje6 <= 2)
  f_tama6=0.879*Diametro_eje6^-0.107
end
if (Diametro_eje6 >= 3 & Diametro_eje6 <= 10)
  f_tama6=0.91*(Diametro_eje6^(-0.157))
end

```

//factor de concentracion de esfuerzo

```

h6=Radio_eje6-Radio_eje5
relacion6=h6/Radio_muesca5

```

//Kt para esfuerzos de flexión (Ktf)

```

//0.1<=h/r<=2.0
C6_f1=0.947+1.206*sqrt(h6/Radio_muesca5)-
0.131*(h6/Radio_muesca5)
C6_f2=0.022-
3.405*sqrt(h6/Radio_muesca5)+0.915*(h6/Radio_muesca5)
C6_f3=0.869+1.777*sqrt(h6/Radio_muesca5)-
0.555*(h6/Radio_muesca5)
C6_f4=-0.810+0.422*sqrt(h6/Radio_muesca5)-
0.260*(h6/Radio_muesca5)

```

//2.0<=h/r<=20

```

C6_f11=1.232+0.832*sqrt(h6/Radio_muesca5)-
0.008*(h6/Radio_muesca5)
C6_f22=-3.813+0.968*sqrt(h6/Radio_muesca5)-
0.260*(h6/Radio_muesca5)
C6_f33=7.423-
4.868*sqrt(h6/Radio_muesca5)+0.869*(h6/Radio_muesca5)
C6_f44=-3.839+3.070*sqrt(h6/Radio_muesca5)-
0.600*(h6/Radio_muesca5)

```

```

if (relacion6 >= 0.1 & relacion6 <= 2)

```

```

  C6_F1=C6_f1
  C6_F2=C6_f2
  C6_F3=C6_f3
  C6_F4=C6_f4
  Kt6_f=C6_F1+C6_F2*(2*h6/Diametro_eje6)+C6_F3*(2*h6/
Diametro_eje6)^2+C6_F4*(2*h6/Diametro_eje6)^3
end

```

```

if (relacion6 > 2 & relacion6 <= 20)

```

```

  C6_F1=C6_f11
  C6_F2=C6_f22
  C6_F3=C6_f33
  C6_F4=C6_f44
  Kt6_f=C6_F1+C6_F2*(2*h6/Diametro_eje6)+C6_F3*(2*h6/
Diametro_eje6)^2+C6_F4*(2*h6/Diametro_eje6)^3
end

```

```

if relacion6 < 0.1 then

```

```

  Kt6_f=1
end
if relacion6 > 20 then
  Kt6_f=1
end
Kt6_F=Kt6_f

```

//Kt para esfuerzos de torsión (Ktf)

```

//0.25<=h/r<=4.0
C6_t1=0.905+0.783*sqrt(h6/Radio_muesca5)-
0.075*(h6/Radio_muesca5)
C6_t2=-0.437-
1.969*sqrt(h6/Radio_muesca5)+0.553*(h6/Radio_muesca5)
C6_t3=1.557+1.073*sqrt(h6/Radio_muesca5)-
0.578*(h6/Radio_muesca5)

```

```

C6_t4=-1.061+0.171*sqrt(h6/Radio_muesca5)-
0.086*(h6/Radio_muesca5)

```

```

if (relacion6 >= 0.25 & relacion6 <= 4.0)

```

```

  C6_T1=C6_t1
  C6_T2=C6_t2
  C6_T3=C6_t3
  C6_T4=C6_t4
  Kt6_t=C6_T1+C6_T2*(2*h6/Diametro_eje6)+C6_T3*(2*h6/
Diametro_eje6)^2+C6_T4*(2*h6/Diametro_eje6)^3
end

```

```

if relacion6 < 0.25 then

```

```

  Kt6_t=1
end

```

```

if relacion6 > 20 then

```

```

  Kt6_t=1
end

```

```

Kt6_T=Kt6_t

```

```

constante_reemplazo_f6=(2*(Kt6_F-
1)/Kt6_F)*(RAIZ_DE_A/sqrt(Radio_muesca5))
Kf6_f=Kt6_F/((1+constante_reemplazo_f6))

```

```

constante_reemplazo_t6=(2*(Kt6_T-
1)/Kt6_T)*(RAIZ_DE_A/sqrt(Radio_muesca5))
Kf6_t=Kt6_T/((1+constante_reemplazo_t6))

```

```

Am6=[0 (taotormed6*f_carga_torsion*Kf6_t/f_tama6)
0;(taotormed6*f_carga_torsion*Kf6_t/f_tama6) 0 0; 0 0]

```

//tensor de esfuerzo medio

```

[Em6]=spec(Am6)
sigvmm6=(((Em6(1)-Em6(2))^2)+(Em6(2)-
Em6(3))^2)+(Em6(3)-Em6(1))^2)/2^0.5

```

```

Aa6=[(sigma6*f_carga_flexion*Kf6_f/f_tama6)
(taotoralt6*f_carga_torsion*Kf6_t/f_tama6) 0;0 0;0 0]
(taotoralt6*f_carga_torsion*Kf6_t/f_tama6) 0;0 0 0] //tensor
de esfuerzo alternante

```

//tensor de esfuerzo alternante

```

[Ea6]=spec(Aa6)
sigvma6=(((Ea6(1)-Ea6(2))^2)+(Ea6(2)-
Ea6(3))^2)+(Ea6(3)-Ea6(1))^2)/2^0.5

```

```

V_misses_med_LN6=[sigvmm6,
Coef_variacion_carga*sigvmm6]
V_misses_alt_LN6=[sigvma6,
Coef_variacion_carga*sigvma6]

```

```

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

```

```

SE=[Media_Se, Desv_Se]

```

```

Sut=[Valor_medio_sut,Desv_sut]

```

```

if criterio_falla == 1 then

```

```

  [Div_Va_SE_6,
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
D_Va_SE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]

```

```

[Div_Va_SE_6_Cuad, Desv_Div_Va_SE_6_Cuad]=
pot(D_Va_SE6,2)
Division_VaSE6 = [Div_Va_SE_6_Cuad,
Desv_Div_Va_SE_6_Cuad]

```

```

Division_VmSut6 = [0,0]

```

```

Suma6_VmSut_VaSE=Division_VaSE6(1)+Division_VmSut6
(1)

```

```
Desv_Suma6_VmSut_VaSE=sqrt((Division_VaSE6(2)/Division_VaSE6(1))^2+(0)^2)
```

```
ASME_6=[Suma6_VmSut_VaSE,Desv_Suma6_VmSut_VaSE]Cf_ASME_6=ASME_6(2)/ASME_6(1)CRITERIO_F=ASME_6CF_CRITERIO_F=Cf_ASME_6end
```

```
if criterio_falla == 2 then[Div_Va_SE_6,Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)Division_VaSE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]
```

```
Division_VmSut6 = [0,0]
```

```
Suma6_VmSut_VaSE=Division_VaSE6(1)+Division_VmSut6(1)Desv_Suma6_VmSut_VaSE=sqrt((Division_VaSE6(2)/Division_VaSE6(1))^2+(0)^2)
```

```
GERBER_6=[Suma6_VmSut_VaSE,Desv_Suma6_VmSut_VaSE]Cf_GERBER_6=GERBER_6(2)/GERBER_6(1)CRITERIO_F=GERBER_6CF_CRITERIO_F=Cf_GERBER_6end
```

```
if criterio_falla == 3 then[Div_Va_SE_6,Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)Division_Va_SE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]
```

```
Division_VmSut6 = [0,0]
```

```
Suma6_VmSut_VaSE=Division_Va_SE6(1)+Division_VmSut6(1)Desv_Suma6_VmSut_VaSE=sqrt((Division_Va_SE6(2)/Division_Va_SE6(1))^2+(0)^2)GOODMAN_6=[Suma6_VmSut_VaSE,Desv_Suma6_VmSut_VaSE]Cf_GOODMAN_6=GOODMAN_6(2)/GOODMAN_6(1)CRITERIO_F=GOODMAN_6CF_CRITERIO_F=Cf_GOODMAN_6end
```

```
MEDCnormal_6=log(CRITERIO_F(1)-((0.5)*(CF_CRITERIO_F)^2)DESNormal_6=CF_CRITERIO_F
```

```
[Prob_6,Conf_6]=cdfnor("PQ",0,abs(MEDCnormal_6),DESNormal_6)
```

```
DIAMETRO_EJE6=0
```

```
if Prob_6<=1.5*10^-6 & Prob_6>=0.5*10^-6 thenDIAMETRO_EJE6=Diametro_eje6break;end
```

```
enddisp("El diametro del tramo 6 es")disp(DIAMETRO_EJE6)
```

```
for Diametro_eje7=tol.tol:6
```

```
Radio_eje7=Diametro_eje7/2
```

```
Torque7=(W_tangencial3/1000)*((Radio engrane3)-Radio_eje7) // el torque aplicado es la fuerza tangencial por el Radio del engranaje menos el Radio del ejeI7=(%pi)*(((Radio_eje7^2)^4)/4)/inerciaQ7=(2/3)*(Radio_eje7^3)J7=I7*2
```

```
//calculo de esfuerzossigma7=(Momento_max7*Radio_eje7/I7) // esfuerzo flector maximo taotormed7=(Torque7*Radio_eje7/J7) // esfuerzo cortante torsor maximo taotoralt7=(Torque7*Radio_eje7/J7)
```

```
//FACTOR DE TAMAÑO
```

```
if (Diametro_eje7 < 0.11) f_tama7=1endif (Diametro_eje7 >= 0.11 & Diametro_eje7 <= 2) f_tama7=0.879*Diametro_eje7^-0.107endif (Diametro_eje7 >= 3 & Diametro_eje7 <= 10) f_tama7=0.91*(Diametro_eje7^(-0.177))end
```

```
//factor de concentracion de esfuerzo
```

```
h7=Radio_eje7-Radio_eje6relacion7=h7/Radio_muesca6
```

```
//0.1<=h/r<=2.0
```

```
C7_a1=0.926+(1.157*sqrt(h7/Radio_muesca6))-(0.099*(h7/Radio_muesca6))C7_a2=0.012-3.036*sqrt(h7/Radio_muesca6)+0.961*(h7/Radio_muesca6)C7_a3=-0.302+3.977*sqrt(h7/Radio_muesca6)-1.744*(h7/Radio_muesca6)C7_a4=0.365-2.098*sqrt(h7/Radio_muesca6)+0.878*(h7/Radio_muesca6)
```

```
//2.0<=h/r<=20
```

```
C7_a11=1.2+0.860*sqrt(h7/Radio_muesca6)-0.022*(h7/Radio_muesca6)C7_a22=-1.805-0.346*sqrt(h7/Radio_muesca6)-0.038*(h7/Radio_muesca6)C7_a33=2.198-0.486*sqrt(h7/Radio_muesca6)+0.165*(h7/Radio_muesca6)C7_a44=-0.593-0.028*sqrt(h7/Radio_muesca6)-0.106*(h7/Radio_muesca6)
```

```
if (relacion7 >=0.1 & relacion7 <= 2)
```

```
C7_A1=C7_a1C7_A2=C7_a2C7_A3=C7_a3C7_A4=C7_a4Kt7_a=C7_A1+C7_A2*(2*h7/Diametro_eje7)+C7_A3*(2*h7/Diametro_eje7)^2+C7_A4*(2*h7/Diametro_eje7)^3end
```

```
if (relacion7 > 2 & relacion7 <= 20)
```

```
C7_A1=C7_a11C7_A2=C7_a22C7_A3=C7_a33C7_A4=C7_a44
```

```

Kt7_a=C7_A1+C7_A2*(2*h7/Diametro_eje7)+C7_A3*(2*h7/
Diametro_eje7)^2+C7_A4*(2*h7/Diametro_eje7)^3
end
if relacion7 < 0.1 then
    Kt7_a=1
end
if relacion7 > 20 then
    Kt7_a=1
end
KT7_A=Kt7_a

```

//Kt para esfuerzos de flexión (Ktf)

```

//0.1<=h/r<=2.0
C7_f1=0.947+1.206*sqrt(h7/Radio_muesca6)-
0.131*(h7/Radio_muesca6)
C7_f2=0.022-
3.405*sqrt(h7/Radio_muesca6)+0.915*(h7/Radio_muesca6)
C7_f3=0.869+1.777*sqrt(h7/Radio_muesca6)-
0.555*(h7/Radio_muesca6)
C7_f4=-0.810+0.422*sqrt(h7/Radio_muesca6)-
0.260*(h7/Radio_muesca6)

```

```

//2.0<=h/r<=20
C7_f11=1.232+0.832*sqrt(h7/Radio_muesca6)-
0.008*(h7/Radio_muesca6)
C7_f22=-3.813+0.968*sqrt(h7/Radio_muesca6)-
0.260*(h7/Radio_muesca6)
C7_f33=7.423-
4.868*sqrt(h7/Radio_muesca6)+0.869*(h7/Radio_muesca6)
C7_f44=-3.839+3.070*sqrt(h7/Radio_muesca6)-
0.600*(h7/Radio_muesca6)

```

```

if (relacion7 >=0.1 & relacion7 <= 2)
    C7_F1=C7_f1
    C7_F2=C7_f2
    C7_F3=C7_f3
    C7_F4=C7_f4
    Kt7_f=C7_F1+C7_F2*(2*h7/Diametro_eje7)+C7_F3*(2*h7/
    Diametro_eje7)^2+C7_F4*(2*h7/Diametro_eje7)^3
end

```

```

if (relacion7 > 2 & relacion7 <= 20)
    C7_F1=C7_f11
    C7_F2=C7_f22
    C7_F3=C7_f33
    C7_F4=C7_f44
    Kt7_f=C7_F1+C7_F2*(2*h7/Diametro_eje7)+C7_F3*(2*h7/
    Diametro_eje7)^2+C7_F4*(2*h7/Diametro_eje7)^3
end

```

```

if relacion7 < 0.1 then
    Kt7_f=1
end
if relacion7 > 20 then
    Kt7_f=1
end
KT7_F=Kt7_f

```

//Kt para esfuerzos de torsión (Ktf)

```

//0.25<=h/r<=4.0
C7_t1=0.905+0.783*sqrt(h7/Radio_muesca6)-
0.075*(h7/Radio_muesca6)
C7_t2=-0.437-
1.969*sqrt(h7/Radio_muesca6)+0.553*(h7/Radio_muesca6)
C7_t3=1.557+1.073*sqrt(h7/Radio_muesca6)-
0.578*(h7/Radio_muesca6)

```

```

C7_t4=-1.061+0.171*sqrt(h7/Radio_muesca6)-
0.086*(h7/Radio_muesca6)

```

```

if (relacion7 >=0.25 & relacion7 <= 4.0)
    C7_T1=C7_t1
    C7_T2=C7_t2
    C7_T3=C7_t3
    C7_T4=C7_t4
    Kt7_t=C7_T1+C7_T2*(2*h7/Diametro_eje7)+C7_T3*(2*h7/
    Diametro_eje7)^2+C7_T4*(2*h7/Diametro_eje7)^3
end

```

```

if relacion7 < 0.25 then
    Kt7_t=1
end
if relacion7 > 0.25 then
    Kt7_t=1
end

```

KT7_T=Kt7_t

//Factor de concentración de esfuerzo de fatiga (Kf_a - axial), (Kf_f-flexión), (Kf_t-torsión)

```

constante_reemplazo_a7=(2*(KT7_A-
1)/KT7_A)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_a=KT7_A/((1+constante_reemplazo_a7))

```

```

constante_reemplazo_f7=(2*(KT7_F-
1)/KT7_F)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_f=KT7_F/((1+constante_reemplazo_f7))

```

```

constante_reemplazo_t7=(2*(KT7_T-
1)/KT7_T)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_t=KT7_T/((1+constante_reemplazo_t7))

```

//presion por interferencia

```

m3=((((2*Radio_engrane3)^2-
Diametro_eje7^2))*(Diametro_eje7^2))
n3=(2*Radio_engrane3)
PP7=((E*Interferen_maza_eje_max3)/(2*Diametro_eje7^3))*(
m3/n3)

```

```

Am7=[(PP7*f_carga_axial*Kf7_a/f_tama7)
(taotormed7*f_carga_torsion*Kf7_t/f_tama7)
0;(taotormed7*f_carga_torsion*Kf7_t/f_tama7) 0 0;0 0 0]

```

//tensor de esfuerzo medio//

```

[Em7]=spec(Am7)
sigvmm7=((((Em7(1)-Em7(2))^2)+((Em7(2)-
Em7(3))^2)+((Em7(3)-Em7(1))^2))/2^0.5

```

```

Aa7=[(siga7*f_carga_flexion*Kf7_f/f_tama7)
(taotoralt7*f_carga_torsion*Kf7_t/f_tama7) 0;0
(taotoralt7*f_carga_torsion*Kf7_t/f_tama7) 0;0 0 0] //tensor
de esfuerzo alternante

```

```

[Ea7]=spec(Aa7)
sigvma7=((((Ea7(1)-Ea7(2))^2)+((Ea7(2)-
Ea7(3))^2)+((Ea7(3)-Ea7(1))^2))/2^0.5

```

```

V_misses_med_LN7=[sigvmm7,
Coef_variacion_carga*sigvmm7]
V_misses_alt_LN7=[sigvma7,
Coef_variacion_carga*sigvma7]

```

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

```

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

```

```

if criterio_falla == 1 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
D_Va_SE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

[Div_Va_SE_7_Cuad,          Desv_Div_Va_SE_7_Cuad]=
pot(D_Va_SE7,2)
Division_VaSE7          =          [Div_Va_SE_7_Cuad,
Desv_Div_Va_SE_7_Cuad]

Division_VmSut7 = [0,0]

Suma7_VmSut_VaSE=Division_VaSE7(1)+Division_VmSut7
(1)
Desv_Suma7_VmSut_VaSE=sqrt((Division_VaSE7(2)/Divisio
n_VaSE7(1))^2+(0)^2)

ASME_7=[Suma7_VmSut_VaSE,
Desv_Suma7_VmSut_VaSE]
Cf_ASME_7=ASME_7(2)/ASME_7(1)
CRITERIO_F=ASME_7
CF_CRITERIO_F=Cf_ASME_7
end

if criterio_falla == 2 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
Division_VaSE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

Division_VmSut7 = [0,0]

Suma7_VmSut_VaSE=Division_VaSE7(1)+Division_VmSut7
(1)
Desv_Suma7_VmSut_VaSE=sqrt((Division_VaSE7(2)/Divisio
n_VaSE7(1))^2+(0)^2)

GERBER_7=[Suma7_VmSut_VaSE,
Desv_Suma7_VmSut_VaSE]
Cf_GERBER_7=GERBER_7(2)/GERBER_7(1)
CRITERIO_F=GERBER_7
CF_CRITERIO_F=Cf_GERBER_7
end

if criterio_falla == 3 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
Division_Va_SE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

Division_VmSut7 = [0,0]

Suma7_VmSut_VaSE=Division_Va_SE7(1)+Division_VmSut
7(1)
Desv_Suma7_VmSut_VaSE=sqrt((Division_Va_SE7(2)/Divisi
on_Va_SE7(1))^2+(0)^2)
GOODMAN_7=[Suma7_VmSut_VaSE,
Desv_Suma7_VmSut_VaSE]
Cf_GOODMAN_7=GOODMAN_7(2)/GOODMAN_7(1)
CRITERIO_F=GOODMAN_7
CF_CRITERIO_F=Cf_GOODMAN_7
end

MEDCnormal_7=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_7=CF_CRITERIO_F

[Prob_7,Conf_7]=cdfnror("PQ",0,abs(MEDCnormal_7),DESV
normal_7)

DIAMETRO_EJE7=0

if Prob_7<=1.5*10^-6 & Prob_7>=0.5*10^-6 then
DIAMETRO_EJE7=Diametro_eje7
break;
end

end
disp("El diametro del tramo 7 es")
disp(DIAMETRO_EJE7)

for Diametro_eje8=tol:tol:6

Radio_eje8=Diametro_eje8/2
I8=(%pi*(((Radio_eje8*2)^4))/4)/inercia
J8=I8*2

// calculo de esfuerzos
siga8=(Momento_max8*Radio_eje8/I8) // esfuerzo flector
maximo

//FACTOR DE TAMAÑO
if (Diametro_eje8 < 0.11)
f_tama8=1
end
if (Diametro_eje8 >= 0.11 & Diametro_eje8 <= 2)
f_tama8=0.879*Diametro_eje8^-0.107
end
if (Diametro_eje8 >= 3 & Diametro_eje8 <= 10)
f_tama8=0.91*(Diametro_eje8^(-0.157))
end

//factor de concentracion de esfuerzo
h8=Radio_eje8-Radio_eje7
relacion8=h8/Radio_muesca7

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C8_f1=0.947+1.206*sqrt(h8/Radio_muesca7)-
0.131*(h8/Radio_muesca7)
C8_f2=0.022-
3.405*sqrt(h8/Radio_muesca7)+0.915*(h8/Radio_muesca7)
C8_f3=0.869+1.777*sqrt(h8/Radio_muesca7)-
0.555*(h8/Radio_muesca7)
C8_f4=-0.810+0.422*sqrt(h8/Radio_muesca7)-
0.260*(h8/Radio_muesca7)

//2.0<=h/r<=20
C8_f11=1.232+0.832*sqrt(h8/Radio_muesca7)-
0.008*(h8/Radio_muesca7)
C8_f22=-3.813+0.968*sqrt(h8/Radio_muesca7)-
0.260*(h8/Radio_muesca7)
C8_f33=7.423-
4.868*sqrt(h8/Radio_muesca7)+0.869*(h8/Radio_muesca7)
C8_f44=-3.839+3.070*sqrt(h8/Radio_muesca7)-
0.600*(h8/Radio_muesca7)

if (relacion8 >=0.1 & relacion8 <= 2)
C8_F1=C8_f1
C8_F2=C8_f2
C8_F3=C8_f3
C8_F4=C8_f4
Kt8_f=C8_F1+C8_F2*(2*h8/Diametro_eje8)+C8_F3*(2*h8/
Diametro_eje8)^2+C8_F4*(2*h8/Diametro_eje8)^3
end

```

```

if (relacion8 > 2 & relacion8 <= 20)
    C8_F1=C8_f11
    C8_F2=C8_f22
    C8_F3=C8_f33
    C8_F4=C8_f44
    Kt8_f=C8_F1+C8_F2*(2*h8/Diametro_eje8)+C8_F3*(2*h8/
    Diametro_eje8)^2+C8_F4*(2*h8/Diametro_eje8)^3
end

if relacion8 < 0.1 then
    Kt8_f=1
end
if relacion8 > 20 then
    Kt8_f=1
end
KT8_F=Kt8_f

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f8=(2*(KT8_F-
1)/KT8_F)*(RAIZ_DE_A/sqrt(Radio_muesca7))
Kf8_f=KT8_F/((1+constante_reemplazo_f8))

Am8=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio//
[Em8]=spec(Am8)
sigvmm8=(((Em8(1)-Em8(2))^2)+((Em8(2)-
Em8(3))^2)+((Em8(3)-Em8(1))^2)/2)^0.5

Aa8=[(siga8*f_carga_flexion*Kf8_f/f_tama8) 0 0;0 0 0;0 0 0]
//tensor de esfuerzo alternante
[Ea8]=spec(Aa8)
sigvma8=(((Ea8(1)-Ea8(2))^2)+((Ea8(2)-
Ea8(3))^2)+((Ea8(3)-Ea8(1))^2)/2)^0.5

V_misses_med_LN8=[sigvmm8,
Coef_variacion_carga*sigvmm8]
V_misses_alt_LN8=[sigvma8,
Coef_variacion_carga*sigvma8]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
    [Div_Va_SE_8,
    Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)
    D_Va_SE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]

    [Div_Va_SE_8_Cuad,          Desv_Div_Va_SE_8_Cuad]=
    pot(D_Va_SE8,2)
    Division_VaSE8 = [Div_Va_SE_8_Cuad,
    Desv_Div_Va_SE_8_Cuad]

    Division_VmSut8 = [0,0]

    Suma8_VmSut_VaSE=Division_VaSE8(1)+Division_VmSut8
    (1)
    Desv_Suma8_VmSut_VaSE=sqrt((Division_VaSE8(2)/Divisio
    n_VaSE8(1))^2+(0)^2)

    ASME_8=[Suma8_VmSut_VaSE,
    Desv_Suma8_VmSut_VaSE]
    Cf_ASME_8=ASME_8(2)/ASME_8(1)
    CRITERIO_F=ASME_8

```

```

CF_CRITERIO_F=Cf_ASME_8
end

if criterio_falla == 2 then
    [Div_Va_SE_8,
    Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)
    Division_VaSE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]

    Division_VmSut8 = [0,0]

    Suma8_VmSut_VaSE=Division_VaSE8(1)+Division_VmSut8
    (1)
    Desv_Suma8_VmSut_VaSE=sqrt((Division_VaSE8(2)/Divisio
    n_VaSE8(1))^2+(0)^2)

    GERBER_8=[Suma8_VmSut_VaSE,
    Desv_Suma8_VmSut_VaSE]
    Cf_GERBER_8=GERBER_8(2)/GERBER_8(1)
    CRITERIO_F=GERBER_8
    CF_CRITERIO_F=Cf_GERBER_8
end

if criterio_falla == 3 then
    [Div_Va_SE_8,
    Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)
    Division_Va_SE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]

    Division_VmSut8 = [0,0]

    Suma8_VmSut_VaSE=Division_Va_SE8(1)+Division_VmSut
    8(1)
    Desv_Suma8_VmSut_VaSE=sqrt((Division_Va_SE8(2)/Divisi
    on_Va_SE8(1))^2+(0)^2)
    GOODMAN_8=[Suma8_VmSut_VaSE,
    Desv_Suma8_VmSut_VaSE]
    Cf_GOODMAN_8=GOODMAN_8(2)/GOODMAN_8(1)
    CRITERIO_F=GOODMAN_8
    CF_CRITERIO_F=Cf_GOODMAN_8
end

MEDCnormal_8=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_8=CF_CRITERIO_F

[Prob_8,Conf_8]=cdfnorr("PQ",0,abs(MEDCnormal_8),DESV
normal_8)

DIAMETRO_EJE8=0

if Prob_8<=1.5*10^-6 & Prob_8>=0.5*10^-6 then
    DIAMETRO_EJE8=Diametro_eje8
    break;
end
end
disp("El diametro del tramo 8 es")
disp(DIAMETRO_EJE8)

for Diametro_eje9=tol:tol:6

Radio_eje9=Diametro_eje9/2
I9=(%pi*(((Radio_eje9*2)^4))/4)/inercia
Q9=(2/3)*(Radio_eje9^3)
J9=I9*2
// calculo de esfuerzos
siga9=(Momento_max9*Radio_eje9/I9) // esfuerzo flector
maximo

```


//FACTOR DE TAMAÑO

```
if (Diametro_eje9 < 0.11)
  f_tama9=1
end
if (Diametro_eje9 >= 0.11 & Diametro_eje9 <= 2)
  f_tama9=0.879*Diametro_eje9^-0.107
end
if (Diametro_eje9 >= 3 & Diametro_eje9 <= 10)
  f_tama9=0.91*(Diametro_eje9^(-0.157))
end
```

//factor de concentracion de esfuerzo

```
h9=Radio_eje9-Radio_eje8
relacion9=h9/Radio_muesca8
```

//Kt para esfuerzos de flexión (Ktf)

```
//0.1 <= h/r <= 2.0
C9_f1=0.947+1.206*sqrt(h9/Radio_muesca8)-
0.131*(h9/Radio_muesca8)
C9_f2=0.022-
3.405*sqrt(h9/Radio_muesca8)+0.915*(h9/Radio_muesca8)
C9_f3=0.869+1.777*sqrt(h9/Radio_muesca8)-
0.555*(h9/Radio_muesca8)
C9_f4=-0.810+0.422*sqrt(h9/Radio_muesca8)-
0.260*(h9/Radio_muesca8)
```

//2.0 <= h/r <= 20

```
C9_f11=1.232+0.832*sqrt(h9/Radio_muesca8)-
0.008*(h9/Radio_muesca8)
C9_f22=-3.813+0.968*sqrt(h9/Radio_muesca8)-
0.260*(h9/Radio_muesca8)
C9_f33=7.423-
4.868*sqrt(h9/Radio_muesca8)+0.869*(h9/Radio_muesca8)
C9_f44=-3.839+3.070*sqrt(h9/Radio_muesca8)-
0.600*(h9/Radio_muesca8)
```

```
if (relacion9 >= 0.1 & relacion9 <= 2)
  C9_F1=C9_f1
  C9_F2=C9_f2
  C9_F3=C9_f3
  C9_F4=C9_f4
  Kt9_f=C9_F1+C9_F2*(2*h9/Diametro_eje9)+C9_F3*(2*h9/
  Diametro_eje9)^2+C9_F4*(2*h9/Diametro_eje9)^3
end
```

```
if (relacion9 > 2 & relacion9 <= 20)
  C9_F1=C9_f11
  C9_F2=C9_f22
  C9_F3=C9_f33
  C9_F4=C9_f44
  Kt9_f=C9_F1+C9_F2*(2*h9/Diametro_eje9)+C9_F3*(2*h9/
  Diametro_eje9)^2+C9_F4*(2*h9/Diametro_eje9)^3
end
```

if relacion9 < 0.1 then

```
Kt9_f=1
end
if relacion9 > 0.1 then
  Kt9_f=1
end
KT9_F=Kt9_f
```

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

```
constante_reemplazo_f9=(2*(KT9_F-
1)/KT9_F)*(RAIZ_DE_A/sqrt(Radio_muesca8))
Kf9_f=KT9_F/((1+constante_reemplazo_f9))
```

```
Am9=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio//
[Em9]=spec(Am9)
sigvmm9=(((Em9(1)-Em9(2))^2)+((Em9(2)-
Em9(3))^2)+((Em9(3)-Em9(1))^2)/2)^0.5
```

```
Aa9=[(sigma9*f_carga_flexion*Kf9_f/f_tama9) 0 0;0 0 0;0 0 0]
//tensor de esfuerzo alternante
[Ea9]=spec(Aa9)
sigma9=(((Ea9(1)-Ea9(2))^2)+((Ea9(2)-
Ea9(3))^2)+((Ea9(3)-Ea9(1))^2)/2)^0.5
```

```
V_misses_med_LN9=[sigvmm9,
Coef_variacion_carga*sigvmm9]
V_misses_alt_LN9=[sigma9,
Coef_variacion_carga*sigma9]
```

```
[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)
```

```
SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]
```

```
if criterio_falla == 1 then
  [Div_Va_SE_9,
  Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
  D_Va_SE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]
```

```
[Div_Va_SE_9_Cuad, Desv_Div_Va_SE_9_Cuad]=
pot(D_Va_SE9,2)
Division_VaSE9 = [Div_Va_SE_9_Cuad,
Desv_Div_Va_SE_9_Cuad]
```

```
Division_VmSut9 = [0,0]
```

```
Suma9_VmSut_VaSE=Division_VaSE9(1)+Division_VmSut9
(1)
Desv_Suma9_VmSut_VaSE=sqrt((Division_VaSE9(2)/Divisio
n_VaSE9(1))^2+(0)^2)
```

```
ASME_9=[Suma9_VmSut_VaSE,
Desv_Suma9_VmSut_VaSE]
Cf_ASME_9=ASME_9(2)/ASME_9(1)
CRITERIO_F=ASME_9
CF_CRITERIO_F=Cf_ASME_9
end
```

```
if criterio_falla == 2 then
  [Div_Va_SE_9,
  Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
  Division_VaSE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]
```

```
Division_VmSut9 = [0,0]
```

```
Suma9_VmSut_VaSE=Division_VaSE9(1)+Division_VmSut9
(1)
Desv_Suma9_VmSut_VaSE=sqrt((Division_VaSE9(2)/Divisio
n_VaSE9(1))^2+(0)^2)
```

```
GERBER_9=[Suma9_VmSut_VaSE,
Desv_Suma9_VmSut_VaSE]
Cf_GERBER_9=GERBER_9(2)/GERBER_9(1)
CRITERIO_F=GERBER_9
CF_CRITERIO_F=Cf_GERBER_9
end
```

```

if criterio_falla == 3 then
[Div_Va_SE_9,
Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
Division_Va_SE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]

Division_VmSut9 = [0,0]

Suma9_VmSut_VaSE=Division_Va_SE9(1)+Division_VmSut
9(1)
Desv_Suma9_VmSut_VaSE=sqrt((Division_Va_SE9(2)/Divisi
on_Va_SE9(1))^2+(0)^2)
GOODMAN_9=[Suma9_VmSut_VaSE,
Desv_Suma9_VmSut_VaSE]
Cf_GOODMAN_9=GOODMAN_9(2)/GOODMAN_9(1)
CRITERIO_F=GOODMAN_9
CF_CRITERIO_F=Cf_GOODMAN_9
end

MEDCnormal_9=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_9=CF_CRITERIO_F

[Prob_9,Conf_9]=cdfnorr("PQ",0,abs(MEDCnormal_9),DESV
normal_9)

DIAMETRO_EJE9=0

if Prob_9<=1.5*10^-6 & Prob_9>=0.5*10^-6 then
DIAMETRO_EJE9=Diametro_eje9
break;
end
end
disp("El diametro del tramo 9 es")
disp(DIAMETRO_EJE9)

Sut9=0.9*Sut(1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=((sigvma9)-Sut9+(pend*3))/pend //valor de x para el
esfuerzo alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida del eje es :")
disp(ViRe)

```

12. 7_VidaRemanente.sci

```

clc
clear all
if sigvma9 > sigvma8 & sigvma9 > sigvma7 & sigvma9 >
sigvma6 & sigvma9 >sigvma5 & sigvma9 >sigvma4 &
sigvma9 >sigvma3 &sigvma9 >sigvma2 & sigvma9 >sigvma1
then
sigvma=sigvma9
end
if sigvma8 > sigvma9 & sigvma8 > sigvma7 & sigvma8 >
sigvma6 & sigvma8 >sigvma5 & sigvma8 >sigvma4 &
sigvma8 >sigvma3 &sigvma8 >sigvma2 & sigvma8 >sigvma1
then
sigvma=sigvma8
end
if sigvma7 > sigvma9 & sigvma7 > sigvma8 & sigvma7 >
sigvma6 & sigvma7 >sigvma5 & sigvma7 >sigvma4 &

```

```

sigvma7 >sigvma3 &sigvma7 >sigvma2 & sigvma7 >sigvma1
then
sigvma=sigvma7
end
if sigvma6 > sigvma9 & sigvma6 > sigvma8 & sigvma6 >
sigvma7 & sigvma6 >sigvma5 & sigvma6 >sigvma4 &
sigvma6 >sigvma3 &sigvma6 >sigvma2 & sigvma6 >sigvma1
then
sigvma=sigvma6
end
if sigvma5 > sigvma9 & sigvma5 > sigvma8 & sigvma5 >
sigvma7 & sigvma5 >sigvma6 & sigvma5 >sigvma4 &
sigvma5 >sigvma3 &sigvma5 >sigvma2 & sigvma5 >sigvma1
then
sigvma=sigvma5
end
if sigvma4 > sigvma9 & sigvma4 > sigvma8 & sigvma4 >
sigvma7 & sigvma4 >sigvma6 & sigvma4 >sigvma3 &
sigvma4 >sigvma2 &sigvma4 >sigvma1 & sigvma4 >sigvma5
then
sigvma=sigvma4
end
if sigvma3 > sigvma9 & sigvma3 > sigvma8 & sigvma3 >
sigvma7 & sigvma3 >sigvma6 & sigvma3 >sigvma5 &
sigvma3 >sigvma4 &sigvma3 >sigvma2 & sigvma3 >sigvma1
then
sigvma=sigvma3
end
if sigvma2 > sigvma9 & sigvma2 > sigvma8 & sigvma2 >
sigvma7 & sigvma2 >sigvma6 & sigvma2 >sigvma5 &
sigvma2 >sigvma4 &sigvma2 >sigvma3 & sigvma2 >sigvma1
then
sigvma=sigvma2
end
if sigvma1 > sigvma9 & sigvma1 > sigvma8 & sigvma1 >
sigvma7 & sigvma1 >sigvma6 & sigvma1 >sigvma5 &
sigvma1 >sigvma4 &sigvma1 >sigvma3 & sigvma1 >sigvma2
then
sigvma=sigvma9
end

SUT=0.9*Sut(1)
m1=(SUT-SE(1))
m2=(3-6)
pend=m1/m2 // calculo de la pendiente
x=((sigvma)-SUT+(pend*3))/pend //valor de x para el esfuerzo
alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)

```

12_TresEngranes.sci

```

clc
clear all

//Entrada de datos de las sesiones, muescas, ángulos, ubicación
y fuerzas:
exec('12.1_DistanciaSesionesMuecasYUbicacionEngrane.sci',
-1)

//Entrada del ajuste por interferencia para el primer engrane:
exec('11.2_OpcionesInterferenciaUno.sci',-1)

```



```

if interferencia == 1 then
exec('10.3_InterferenciaUno.sci',-1)//Para la opción 1
end

if interferencia == 2 then
exec('10.4_InterferenciaDos.sci',-1)//Para la opción 2
end

if interferencia == 3 then
exec('10.5_InterferenciaTres.sci',-1)//Para la opción 3
end

Interferen_maza_eje_min=Interf_maza_eje_min
Interferen_maza_eje_max=Interf_maza_eje_max

//Entrada del ajuste por interferencia para el segundo engrane:
exec('11.3_OpcionesInterferenciaDos.sci',-1)

if interferencia2 == 1 then
exec('11.4_InterferenciaUno.sci',-1)//Para la opción 1
end

if interferencia2 == 2 then
exec('11.5_InterferenciaDos.sci',-1)//Para la opción 2
end

if interferencia2 == 3 then
exec('11.6_InterferenciaTres.sci',-1)//Para la opción 3
end

Interferen_maza_eje_min2=Interf_maza_eje_min2
Interferen_maza_eje_max2=Interf_maza_eje_max2

//Entrada del ajuste por interferencia para el tercer engrane:
exec('12.2_OpcionesInterferenciaTres.sci',-1)

if interferencia3 == 1 then
exec('12.3_InterferenciaUno.sci',-1)//Para la opción 1
end

if interferencia3 == 2 then
exec('12.4_InterferenciaDos.sci',-1)//Para la opción 2
end

if interferencia3 == 3 then
exec('12.5_InterferenciaTres.sci',-1)//Para la opción 3
end

Interferen_maza_eje_min3=Interf_maza_eje_min3
Interferen_maza_eje_max3=Interf_maza_eje_max3

//Cálculo de diagramas de momentos con sus diagramas:
exec('12.6_TorquesMomentosYGraficas.sci',-1)
//Multiplicación de los factores de Marin Ka por Kd:
exec('10.8_MultiplicacionKaKd.sci',-1)
//Iteraciones de los diámetros:
exec('12.7_IteracionesDeLosDiametros.sci',-1)

13.
1_DistanciaSesionesMuecasYUbicacioesEngrane.s
ci

clc
clear all

```

```

sesion_1=input('Ingrese la distancia de la sesion 1:')
Radio_muesca=input('Ingrese el radio de la muesca de la
sesión 1: ')

c=0
while c==0
c=1
sesion_2=input('Ingrese la distancia de la sesion 2:')
if (sesion_2 <= sesion_1 | sesion_2 >= longitud_eje)
c=0
disp('Distancia de la sesión 2 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca2=input('Ingrese el radio de la muesca de la
sesión 2: ')

c=0
while c==0
c=1
sesion_3=input('Ingrese la distancia de la sesion 3:')
if (sesion_3 <= sesion_2 | sesion_3 <= sesion_1 | sesion_3 >=
longitud_eje)
c=0
disp('Distancia de la sesión 3 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca3=input('Ingrese el radio de la muesca de la
sesión 3: ')

c=0
while c==0
c=1
sesion_4=input('Ingrese la distancia de la sesion 4:')
if (sesion_4 <= sesion_3 | sesion_4 <= sesion_2 | sesion_4
<= sesion_1 | sesion_4 >= longitud_eje)
c=0
disp('Distancia de la sesión 4 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca4=input('Ingrese el radio de la muesca de la
sesión 4: ')

c=0
while c==0
c=1
sesion_5=input('Ingrese la distancia de la sesion 5:')
if (sesion_5 <= sesion_4 | sesion_5 <= sesion_3 | sesion_5
<= sesion_2 | sesion_5 <= sesion_1 | sesion_5 >= longitud_eje)
c=0
disp('Distancia de la sesión 5 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca5=input('Ingrese el radio de la muesca de la
sesión 5: ')

c=0

```

```

while c==0
c=1
sesion_6=input('Ingrese la distancia de la sesion 6:')
if (sesion_6 <= sesion_5 | sesion_6 <= sesion_4 | sesion_6
<= sesion_3 | sesion_6 <= sesion_2 | sesion_6 <= sesion_1 |
sesion_6 >= longitud_eje)
c=0
disp('Distancia de la sesión 6 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca6=input('Ingrese el radio de la muesca de la
sesión 6: ')

c=0
while c==0
c=1
sesion_7=input('Ingrese la distancia de la sesion 7:')
if (sesion_7 <= sesion_6 | sesion_7 <= sesion_5 | sesion_7
<= sesion_4 | sesion_7 <= sesion_3 | sesion_7 <= sesion_2 |
sesion_7 <= sesion_1 | sesion_7 >= longitud_eje)
c=0
disp('Distancia de la sesión 7 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca7=input('Ingrese el radio de la muesca de la
sesión 7: ')

c=0
while c==0
c=1
sesion_8=input('Ingrese la distancia de la sesion 8:')
if (sesion_8 <= sesion_7 | sesion_8 <= sesion_6 | sesion_8
<= sesion_5 | sesion_8 <= sesion_4 | sesion_8 <= sesion_3 |
sesion_8 <= sesion_2 | sesion_8 <= sesion_1 | sesion_8 >=
longitud_eje)
c=0
disp('Distancia de la sesión 8 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca8=input('Ingrese el radio de la muesca de la
sesión 8: ')

c=0
while c==0
c=1
sesion_9=input('Ingrese la distancia de la sesión 9:')
if (sesion_9 <= sesion_8 | sesion_9 <= sesion_7 | sesion_9 <=
sesion_6 | sesion_9 <= sesion_5 | sesion_9 <= sesion_4 |
sesion_9 <= sesion_3 | sesion_9 <= sesion_2 | sesion_9 <=
sesion_1 | sesion_9 >= longitud_eje)
c=0
disp('Distancia de la sesión 9 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca9=input('Ingrese el radio de la muesca de la
sesión 9: ')

```

```

c=0
while c==0
c=1
sesion_10=input('Ingrese la distancia de la sesión 10:')
if (sesion_10 <= sesion_9 | sesion_10 <= sesion_8 | sesion_10
<= sesion_7 | sesion_10 <= sesion_6 | sesion_10 <= sesion_5 |
sesion_10 <= sesion_4 | sesion_10 <= sesion_3 | sesion_10 <=
sesion_2 | sesion_10 <= sesion_1 | sesion_10 >= longitud_eje)
c=0
disp('Distancia de la sesión 10 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca10=input('Ingrese el radio de la muesca de la
sesión 10: ')

```

```

//MÓDULO DE ELASTICIDAD DEL ACERO:
E= 30000

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 1:

```

```

exec('11.1.1_DistanciasFuerzasAnguloUbicacionEngraneUno.
sci',-1)

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 2:

```

```

exec('11.1.2_DistanciasFuerzasAnguloUbicacionEngraneDos.
sci',-1)

```

```

//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 3:

```

```

exec('11.1.3_DistanciasFuerzasAnguloUbicacionEngraneTres.
sci',-1)

```

```

//DISTANCIAS, FUERZAS, ANGULOS Y UBICACION DEL
ENGRANE 4:

```

```

exec('11.1.4_DistanciasFuerzasAnguloUbicacionEngraneCuatr
o.sci',-1)

```

13. 2_OpcionesInterferenciaCuatro.sci

```

clc
clear all

c=0
while c==0
c=1
disp("OPCIONES PARA EL TIPO DE AJUSTE POR
INTERFERENCIA")
disp("1- Ajuste de interferencia localizada H7/P6")
disp("2- Ajuste de impulso medio H7/S6")
disp("3- Ajuste forzado H7/U6")

interferencia4 = input('Seleccione la opción de ajuste por
interferencia deseada para el cuarto engrane: ')
if (interferencia4<1 | interferencia4>3)
c=0
disp('Opción incorrecta. Escriba resume para volver a
ingresar el valor')
pause
end
end

```

13. 3_InterferenciaUno.sci

```
clc
clear all

c=0
while c==0
c=1
tamaño_basico4 = input('Ingrese el tamaño básico:');
p4=0;
if (tamaño_basico4<0 | tamaño_basico4>16)
c=0
disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
pause
end
end
p4=0;

if tamaño_basico4 >=0 & tamaño_basico4 < 0.12 then
p4=tabla14(1,1)
end

if tamaño_basico4 >= 0.12 & tamaño_basico4 < 0.24 then
p4=tabla14(2,1)
end

if tamaño_basico4 >= 0.24 & tamaño_basico4 < 0.40 then
p4=tabla14(3,1)
end

if tamaño_basico4 >= 0.40 & tamaño_basico4 < 0.72 then
p4=tabla14(4,1)
end

if tamaño_basico4 >= 0.72 & tamaño_basico4 < 0.96 then
p4=tabla14(5,1)
end

if tamaño_basico4 >= 0.96 & tamaño_basico4 < 1.20 then
p4=tabla14(6,1)
end

if tamaño_basico4 >= 1.20 & tamaño_basico4 < 1.60 then
p4=tabla14(7,1)
end

if tamaño_basico4 >= 1.60 & tamaño_basico4 < 2.00 then
p4=tabla14(8,1)
end

if tamaño_basico4 >= 2.00 & tamaño_basico4 < 2.60 then
p4=tabla14(9,1)
end

if tamaño_basico4 >= 2.60 & tamaño_basico4 < 3.20 then
p4=tabla14(10,1)
end

if tamaño_basico4 >= 3.20 & tamaño_basico4 < 4.00 then
p4=tabla14(11,1)
end

if tamaño_basico4 >= 4.00 & tamaño_basico4 < 4.80 then
```

```
p4=tabla14(12,1)
end

if tamaño_basico4 >= 4.80 & tamaño_basico4 < 5.60 then
p4=tabla14(13,1)
end

if tamaño_basico4 >= 5.60 & tamaño_basico4 < 6.40 then
p4=tabla14(14,1)
end

if tamaño_basico4 >= 6.40 & tamaño_basico4 < 7.20 then
p4=tabla14(15,1)
end

if tamaño_basico4 >= 7.20 & tamaño_basico4 < 8.00 then
p4=tabla14(16,1)
end

if tamaño_basico4 >= 8.00 & tamaño_basico4 < 9.00 then
p4=tabla14(17,1)
end

if tamaño_basico4 >= 9.00 & tamaño_basico4 < 10.00 then
p4=tabla14(18,1)
end

if tamaño_basico4 >= 10.00 & tamaño_basico4 < 11.20 then
p4=tabla14(19,1)
end

if tamaño_basico4 >= 11.20 & tamaño_basico4 < 12.60 then
p4=tabla14(20,1)
end

if tamaño_basico4 >= 12.60 & tamaño_basico4 < 14.20 then
p4=tabla14(21,1)
end

if tamaño_basico4 >= 14.20 & tamaño_basico4 < 16.00 then
p4=tabla14(22,1)
end

it6_4=0
it7_4=0

if tamaño_basico4>=0 & tamaño_basico4<0.12 then
it6_4=Tabla_13(1,1)
it7_4=Tabla_13(1,2)
end

if tamaño_basico4>=0.12 & tamaño_basico4<0.24 then
it6_4=Tabla_13(2,1)
it7_4=Tabla_13(2,2)
end

if tamaño_basico4>=0.24 & tamaño_basico4<0.40 then
it6_4=Tabla_13(3,1)
it7_4=Tabla_13(3,2)
end

if tamaño_basico4>=0.40 & tamaño_basico4<0.72 then
it6_4=Tabla_13(4,1)
it7_4=Tabla_13(4,2)
end

if tamaño_basico4>=0.72 & tamaño_basico4<1.20 then
it6_4=Tabla_13(5,1)
it7_4=Tabla_13(5,2)
end

if tamaño_basico4>=1.20 & tamaño_basico4<2.00 then
```

```

it6_4=Tabla_13(6,1)
it7_4=Tabla_13(6,2)
end
if tamaño_basico4 >= 2.00 & tamaño_basico4 < 3.20 then
it6_4=Tabla_13(7,1)
it7_4=Tabla_13(7,2)
end
if tamaño_basico4 >= 3.20 & tamaño_basico4 < 4.80 then
it6_4=Tabla_13(8,1)
it7_4=Tabla_13(8,2)
end
if tamaño_basico4 >= 4.80 & tamaño_basico4 < 7.20 then
it6_4=Tabla_13(9,1)
it7_4=Tabla_13(9,2)
end
if tamaño_basico4 >= 7.20 & tamaño_basico4 < 10.00 then
it6_4=Tabla_13(10,1)
it7_4=Tabla_13(10,2)
end
if tamaño_basico4 >= 10.00 & tamaño_basico4 < 12.60 then
it6_4=Tabla_13(11,1)
it7_4=Tabla_13(11,2)
end
if tamaño_basico4 >= 12.60 & tamaño_basico4 < 16.00 then
it6_4=Tabla_13(12,1)
it7_4=Tabla_13(12,2)
end

tolerancia_agujero4 = it7_4
tolerancia_eje4=it6_4
Diametro_max_agujero4=tamaño_basico4 +
tolerancia_agujero4
Diametro_min_agujero4=tamaño_basico4
Diametro_max_eje4= tamaño_basico4 + p4 + tolerancia_eje4
Diametro_min_eje4= tamaño_basico4 + p4

Interf_maza_eje_min4=Diametro_min_eje4-
Diametro_max_agujero4
Interf_maza_eje_max4=Diametro_max_eje4-
Diametro_min_agujero4

```

13. 4_IntrferenciaDos.sci

```

clc
clear all

c=0
while c==0
c=1
tamaño_basico4 = input('Ingrese el tamaño básico:')
s4=0;
if (tamaño_basico4<0 | tamaño_basico4>16)
c=0
disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
pause
end
end
s4=0;

if tamaño_basico4 >=0 & tamaño_basico4 < 0.12 then
s4=tabla14(1,2)
end

if tamaño_basico4 >= 0.12 & tamaño_basico4 < 0.24 then

```

```

s4=tabla14(2,2)
end
if tamaño_basico4 >= 0.24 & tamaño_basico4 < 0.40 then
s4=tabla14(3,2)
end
if tamaño_basico4 >= 0.40 & tamaño_basico4 < 0.72 then
s4=tabla14(4,2)
end
if tamaño_basico4 >= 0.72 & tamaño_basico4 < 0.96 then
s4=tabla14(5,2)
end
if tamaño_basico4 >= 0.96 & tamaño_basico4 < 1.20 then
s4=tabla14(6,2)
end
if tamaño_basico4 >= 1.20 & tamaño_basico4 < 1.60 then
s4=tabla14(7,2)
end
if tamaño_basico4 >= 1.60 & tamaño_basico4 < 2.00 then
s4=tabla14(8,2)
end
if tamaño_basico4 >= 2.00 & tamaño_basico4 < 2.60 then
s4=tabla14(9,2)
end
if tamaño_basico4 >= 2.60 & tamaño_basico4 < 3.20 then
s4=tabla14(10,2)
end
if tamaño_basico4 >= 3.20 & tamaño_basico4 < 4.00 then
s4=tabla14(11,2)
end
if tamaño_basico4 >= 4.00 & tamaño_basico4 < 4.80 then
s4=tabla14(12,2)
end
if tamaño_basico4 >= 4.80 & tamaño_basico4 < 5.60 then
s4=tabla14(13,2)
end
if tamaño_basico4 >= 5.60 & tamaño_basico4 < 6.40 then
s4=tabla14(14,2)
end
if tamaño_basico4 >= 6.40 & tamaño_basico4 < 7.20 then
s4=tabla14(15,2)
end
if tamaño_basico4 >= 7.20 & tamaño_basico4 < 8.00 then
s4=tabla14(16,2)
end
if tamaño_basico4 >= 8.00 & tamaño_basico4 < 9.00 then
s4=tabla14(17,2)
end
if tamaño_basico4 >= 9.00 & tamaño_basico4 < 10.00 then
s4=tabla14(18,2)
end
end

```

```

if tamaño_basico4 >= 10.00 & tamaño_basico4 < 11.20 then
    s4=tabla14(19,2)
end

if tamaño_basico4 >= 11.20 & tamaño_basico4 < 12.60 then
    s4=tabla14(20,2)
end

if tamaño_basico4 >= 12.60 & tamaño_basico4 < 14.20 then
    s4=tabla14(21,2)
end

if tamaño_basico4 >= 14.20 & tamaño_basico4 < 16.00 then
    s4=tabla14(22,2)
end

it6_4=0
it7_4=0

if tamaño_basico4>=0 & tamaño_basico4<0.12 then
    it6_4=Tabla_13(1,1)
    it7_4=Tabla_13(1,2)
end
if tamaño_basico4>=0.12 & tamaño_basico4<0.24 then
    it6_4=Tabla_13(2,1)
    it7_4=Tabla_13(2,2)
end
if tamaño_basico4>=0.24 & tamaño_basico4<0.40 then
    it6_4=Tabla_13(3,1)
    it7_4=Tabla_13(3,2)
end
if tamaño_basico4>=0.40 & tamaño_basico4<0.72 then
    it6_4=Tabla_13(4,1)
    it7_4=Tabla_13(4,2)
end
if tamaño_basico4>=0.72 & tamaño_basico4<1.20 then
    it6_4=Tabla_13(5,1)
    it7_4=Tabla_13(5,2)
end
if tamaño_basico4>=1.20 & tamaño_basico4<2.00 then
    it6_4=Tabla_13(6,1)
    it7_4=Tabla_13(6,2)
end
if tamaño_basico4>=2.00 & tamaño_basico4<3.20 then
    it6_4=Tabla_13(7,1)
    it7_4=Tabla_13(7,2)
end
if tamaño_basico4>=3.20 & tamaño_basico4<4.80 then
    it6_4=Tabla_13(8,1)
    it7_4=Tabla_13(8,2)
end
if tamaño_basico4 >= 4.80 & tamaño_basico4 < 7.20 then
    it6_4=Tabla_13(9,1)
    it7_4=Tabla_13(9,2)
end
if tamaño_basico4 >= 7.20 & tamaño_basico4 < 10.00 then
    it6_4=Tabla_13(10,1)
    it7_4=Tabla_13(10,2)
end
if tamaño_basico4 >= 10.00 & tamaño_basico4 < 12.60 then
    it6_4=Tabla_13(11,1)
    it7_4=Tabla_13(11,2)
end
if tamaño_basico4 >= 12.60 & tamaño_basico4 < 16.00 then
    it6_4=Tabla_13(12,1)

```

```

    it7_4=Tabla_13(12,2)
end

tolerancia_agujero4 = it7_4
tolerancia_eje4=it6_4
Diametro_max_agujero4=tamaño_basico4 + tolerancia_agujero4
Diametro_min_agujero4=tamaño_basico4
Diametro_max_eje4= tamaño_basico4 + s4 + tolerancia_eje4
Diametro_min_eje4= tamaño_basico4 + s4

Interf_maza_eje_min4=Diametro_min_eje4-
Diametro_max_agujero4
Interf_maza_eje_max4=Diametro_max_eje4-
Diametro_min_agujero4

```

13. 5 InterferenciaTres.sci

```

clc
clear all

c=0
while c==0
c=1
tamaño_basico4 = input('Ingrese el tamaño básico:')
u4=0;
if (tamaño_basico4<0 | tamaño_basico4>16)
    c=0
    disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
    pause
end
end
u4=0;

if tamaño_basico4 >=0 & tamaño_basico4 < 0.12 then
    u4=tabla14(1,3)
end

if tamaño_basico4 >= 0.12 & tamaño_basico4 < 0.24 then
    u4=tabla14(2,3)
end

if tamaño_basico4 >= 0.24 & tamaño_basico4 < 0.40 then
    u4=tabla14(3,3)
end

if tamaño_basico4 >= 0.40 & tamaño_basico4 < 0.72 then
    u4=tabla14(4,3)
end

if tamaño_basico4 >= 0.72 & tamaño_basico4 < 0.96 then
    u4=tabla14(5,3)
end

if tamaño_basico4 >= 0.96 & tamaño_basico4 < 1.20 then
    u4=tabla14(6,3)
end

if tamaño_basico4 >= 1.20 & tamaño_basico4 < 1.60 then
    u4=tabla14(7,3)
end

if tamaño_basico4 >= 1.60 & tamaño_basico4 < 2.00 then
    u4=tabla14(8,3)

```

```

end

if tamano_basico4 >= 2.00 & tamano_basico4 < 2.60 then
    u4=tabla14(9,3)
end

if tamano_basico4 >= 2.60 & tamano_basico4 < 3.20 then
    u4=tabla14(10,3)
end

if tamano_basico4 >= 3.20 & tamano_basico4 < 4.00 then
    u4=tabla14(11,3)
end

if tamano_basico4 >= 4.00 & tamano_basico4 < 4.80 then
    u4=tabla14(12,3)
end

if tamano_basico4 >= 4.80 & tamano_basico4 < 5.60 then
    u4=tabla14(13,3)
end

if tamano_basico4 >= 5.60 & tamano_basico4 < 6.40 then
    u4=tabla14(14,3)
end

if tamano_basico4 >= 6.40 & tamano_basico4 < 7.20 then
    u4=tabla14(15,3)
end

if tamano_basico4 >= 7.20 & tamano_basico4 < 8.00 then
    u4=tabla14(16,3)
end

if tamano_basico4 >= 8.00 & tamano_basico4 < 9.00 then
    u4=tabla14(17,3)
end

if tamano_basico4 >= 9.00 & tamano_basico4 < 10.00 then
    u4=tabla14(18,3)
end

if tamano_basico4 >= 10.00 & tamano_basico4 < 11.20 then
    u4=tabla14(19,3)
end

if tamano_basico4 >= 11.20 & tamano_basico4 < 12.60 then
    u4=tabla14(20,3)
end

if tamano_basico4 >= 12.60 & tamano_basico4 < 14.20 then
    u4=tabla14(21,3)
end

if tamano_basico4 >= 14.20 & tamano_basico4 < 16.00 then
    u4=tabla14(22,3)
end

it6_4=0
it7_4=0

if tamano_basico4>=0 & tamano_basico4<0.12 then
    it6_4=Tabla_13(1,1)
    it7_4=Tabla_13(1,2)
end
if tamano_basico4>=0.12 & tamano_basico4<0.24 then
    it6_4=Tabla_13(2,1)
    it7_4=Tabla_13(2,2)
end
if tamano_basico4>=0.24 & tamano_basico4<0.40 then
    it6_4=Tabla_13(3,1)
    it7_4=Tabla_13(3,2)
end
if tamano_basico4>=0.40 & tamano_basico4<0.72 then
    it6_4=Tabla_13(4,1)
    it7_4=Tabla_13(4,2)
end
if tamano_basico4>=0.72 & tamano_basico4<1.20 then
    it6_4=Tabla_13(5,1)
    it7_4=Tabla_13(5,2)
end
if tamano_basico4>=1.20 & tamano_basico4<2.00 then
    it6_4=Tabla_13(6,1)
    it7_4=Tabla_13(6,2)
end
if tamano_basico4>=2.00 & tamano_basico4<3.20 then
    it6_4=Tabla_13(7,1)
    it7_4=Tabla_13(7,2)
end
if tamano_basico4>=3.20 & tamano_basico4<4.80 then
    it6_4=Tabla_13(8,1)
    it7_4=Tabla_13(8,2)
end
if tamano_basico4 >= 4.80 & tamano_basico4 < 7.20 then
    it6_4=Tabla_13(9,1)
    it7_4=Tabla_13(9,2)
end
if tamano_basico4 >= 7.20 & tamano_basico4 < 10.00 then
    it6_4=Tabla_13(10,1)
    it7_4=Tabla_13(10,2)
end
if tamano_basico4 >= 10.00 & tamano_basico4 < 12.60 then
    it6_4=Tabla_13(11,1)
    it7_4=Tabla_13(11,2)
end
if tamano_basico4 >= 12.60 & tamano_basico4 < 16.00 then
    it6_4=Tabla_13(12,1)
    it7_4=Tabla_13(12,2)
end

tolerancia_agujero4 = it7_4
tolerancia_eje4=it6_4
Diametro_max_agujero4=tamano_basico4 +
tolerancia_agujero4
Diametro_min_agujero4=tamano_basico4
Diametro_max_eje4= tamano_basico4 + u4 + tolerancia_eje4
Diametro_min_eje4= tamano_basico4 + u4

Interf_maza_eje_min4=Diametro_min_eje4-
Diametro_max_agujero4
Interf_maza_eje_max4=Diametro_max_eje4-
Diametro_min_agujero4

13. 6 IteracionesDeLosDiametros.sci

clc
clear all
tol=0.001

for Diametro_eje1=tol:tol:6

Radio_eje1=Diametro_eje1/2

```

```

// inercias y momentos polares
I1=(%pi*(((Radio_eje1*2)^4)/4)
J1=I1*2
sigal=(Momento_max1*Radio_eje1/I1)

//FACTOR DE TAMAÑO
if (Diametro_eje1 < 0.11)
  f_tama1=1
end
if (Diametro_eje1 >= 0.11 & Diametro_eje1 <= 2)
  f_tama1=0.879*Diametro_eje1^-0.107
end
if (Diametro_eje1 >= 3 & Diametro_eje1 <= 10)
  f_tama1=0.91*(Diametro_eje1^(-0.157))
end

Am1=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio//
[Em]=spec(Am1)//Funcion para los autovalores
sigvmm1=(((Em(1)-Em(2))^2)+(Em(2)-Em(3))^2)+((Em(3)-Em(1))^2)/2^0.5 //Vmises medio
Aa1=[(sigal*f_carga_flexion/f_tama1) 0 0;0 0 0;0 0 0]
//tensor de esfuerzo alternante
[Ea]=spec(Aa1)//Funcion para los autovalores
sigvma1=(((Ea(1)-Ea(2))^2)+(Ea(2)-Ea(3))^2)+((Ea(3)-Ea(1))^2)/2^0.5 //Vmises alternante

V_misses_med_LN1=[sigvmm1,
Coef_variacion_carga*sigvmm1]
V_misses_alt_LN1=[sigvma1,
Coef_variacion_carga*sigvma1]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
  [Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
D_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

  [Div_Va_SE_1_Cuad,          Desv_Div_Va_SE_1_Cuad]=
pot(D_Va_SE1,2)
  Division_VaSE1 =          [Div_Va_SE_1_Cuad,
Desv_Div_Va_SE_1_Cuad]

  Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1
(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)

ASME_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_ASME_1=ASME_1(2)/ASME_1(1)
CRITERIO_F=ASME_1
CF_CRITERIO_F=Cf_ASME_1
end

if criterio_falla == 2 then
  [Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_VaSE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

  Division_VmSut1 = [0,0]

```

```

Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1
(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)

GERBER_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GERBER_1=GERBER_1(2)/GERBER_1(1)
CRITERIO_F=GERBER_1
CF_CRITERIO_F=Cf_GERBER_1
end

if criterio_falla == 3 then
  [Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

  Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_Va_SE1(1)+Division_VmSut1
(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_Va_SE1(2)/Division_Va_SE1(1))^2+(0)^2)
GOODMAN_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GOODMAN_1=GOODMAN_1(2)/GOODMAN_1(1)
CRITERIO_F=GOODMAN_1
CF_CRITERIO_F=Cf_GOODMAN_1
end

MEDCnormal_1=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_1=CF_CRITERIO_F

[Prob_1,Conf_1]=cdfnor("PQ",0,abs(MEDCnormal_1),DESVnormal_1)

DIAMETRO_EJE1=0

if Prob_1<=1.5*10^-6 & Prob_1>=0.5*10^-6 then
  DIAMETRO_EJE1=Diametro_eje1
  break;
end
end
disp("El diametro del tramo 1 es")
disp(DIAMETRO_EJE1)

for Diametro_eje2=tol:tol:6

Radio_eje2=Diametro_eje2/2

I2=(%pi*(((Radio_eje2*2)^4)/4))/inercia
J2=I2*2

// calculo de esfuerzos
sigal2=(Momento_max2*Radio_eje2/I2) // esfuerzo flector maximo
//FACTOR DE TAMAÑO
if (Diametro_eje2 < 0.11)
  f_tama2=1
end
if (Diametro_eje2 >= 0.11 & Diametro_eje2 <= 2)
  f_tama2=0.879*Diametro_eje2^-0.107
end
if (Diametro_eje2 >= 3 & Diametro_eje2 <= 10)
  f_tama2=0.91*(Diametro_eje2^(-0.157))

```

```

end

//concentracion de esfuerzo
h2=Radio_eje2-Radio_eje1
relacion2=h2/Radio_muesca

//Kt para esfuerzos de flexión (Kf)

//0.1<=h/r<=2.0
C2_f1=0.947+1.206*sqrt(h2/Radio_muesca)-
0.131*(h2/Radio_muesca)
C2_f2=0.022-
3.405*sqrt(h2/Radio_muesca)+0.915*(h2/Radio_muesca)
C2_f3=0.869+1.777*sqrt(h2/Radio_muesca)-
0.555*(h2/Radio_muesca)
C2_f4=-0.810+0.422*sqrt(h2/Radio_muesca)-
0.260*(h2/Radio_muesca)

//2.0<=h/r<=20
C2_f11=1.232+0.832*sqrt(h2/Radio_muesca)-
0.008*(h2/Radio_muesca)
C2_f22=-3.813+0.968*sqrt(h2/Radio_muesca)-
0.260*(h2/Radio_muesca)
C2_f33=7.423-
4.868*sqrt(h2/Radio_muesca)+0.869*(h2/Radio_muesca)
C2_f44=-3.839+3.070*sqrt(h2/Radio_muesca)-
0.600*(h2/Radio_muesca)

if (relacion2 >=0.1 & relacion2 <= 2)
    C2_F1=C2_f1
    C2_F2=C2_f2
    C2_F3=C2_f3
    C2_F4=C2_f4
    Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/
    Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
end

if (relacion2 > 2 & relacion2 <= 20)
    C2_F1=C2_f11
    C2_F2=C2_f22
    C2_F3=C2_f33
    C2_F4=C2_f44
    Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/
    Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
end

if relacion2 < 0.1 then
    Kt2_f=1
end

if relacion2 > 20 then
    Kt2_f=1
end

KT2_F=Kt2_f

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f2=(2*(KT2_F-
1)/KT2_F)*(RAIZ_DE_A/sqrt(Radio_muesca))
Kf2_f=KT2_F/((1+constante_reemplazo_f2))

Am2=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio
[Em2]=spec(Am2)//Funcion para los autovalores
sigvmm2=(((Em2(1)-Em2(2))^2)+(Em2(2)-
Em2(3))^2)+(Em2(3)-Em2(1))^2)/2)^0.5 //Vmises medio

```

```

Aa2=[(sigma2*f_carga_flexion*Kf2_f/f_tama2) 0 0;0 0 0;0 0 0]
//tensor de esfuerzo alternante
[Ea2]=spec(Aa2)//Funcion para los autovalores
sigvma2=(((Ea2(1)-Ea2(2))^2)+(Ea2(2)-
Ea2(3))^2)+(Ea2(3)-Ea2(1))^2)/2)^0.5 //Vmises alternante

V_misses_med_LN2=[sigvmm2,
Coef_variacion_carga*sigvmm2]
V_misses_alt_LN2=[sigvma2,
Coef_variacion_carga*sigvma2]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
    [Div_Va_SE_2,
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
D_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]

[Div_Va_SE_2_Cuad, Desv_Div_Va_SE_2_Cuad]=
pot(D_Va_SE2,2)
Division_VaSE2 = [Div_Va_SE_2_Cuad,
Desv_Div_Va_SE_2_Cuad]

Division_VmSut2 = [0,0]

Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2
(1)
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio
n_VaSE2(1))^2+(0)^2)

ASME_2=[Suma2_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_ASME_2=ASME_2(2)/ASME_2(1)
CRITERIO_F=ASME_2
CF_CRITERIO_F=Cf_ASME_2
end

if criterio_falla == 2 then
    [Div_Va_SE_2,
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
Division_VaSE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]

Division_VmSut2 = [0,0]

Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2
(1)
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio
n_VaSE2(1))^2+(0)^2)

GERBER_2=[Suma2_VmSut_VaSE,
Desv_Suma2_VmSut_VaSE]
Cf_GERBER_2=GERBER_2(2)/GERBER_2(1)
CRITERIO_F=GERBER_2
CF_CRITERIO_F=Cf_GERBER_2
end

if criterio_falla == 3 then
    [Div_Va_SE_2,
Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
Division_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]

Division_VmSut2 = [0,0]

```



```

Suma2_VmSut_VaSE=Division_Va_SE2(1)+Division_VmSut
2(1)
Desv_Suma2_VmSut_VaSE=sqrt((Division_Va_SE2(2)/Divisi
on_Va_SE2(1))^2+(0)^2)
GOODMAN_2=[Suma2_VmSut_VaSE,
Desv_Suma2_VmSut_VaSE]
Cf_GOODMAN_2=GOODMAN_2(2)/GOODMAN_2(1)
CRITERIO_F=GOODMAN_2
CF_CRITERIO_F=Cf_GOODMAN_2
end

MEDCnormal_2=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESNormal_2=CF_CRITERIO_F

[Prob_2,Conf_2]=cdfnor("PQ",0,abs(MEDCnormal_2),DESNormal_2)

DIAMETRO_EJE2=0

if Prob_2<=1.5*10^-6 & Prob_2>=0.5*10^-6 then
    DIAMETRO_EJE2=Diametro_eje2
    break;
end
end

disp("El diametro del tramo 2 es")
disp(DIAMETRO_EJE2)

for Diametro_eje3=tol:tol:6

Radio_eje3=Diametro_eje3/2 // Radio del eje en el tramo del
engrane 1

Torque3=(W_tangencial1/1000)*((Radio_engrane1)-
Radio_eje3) // torque generado por el primer engrane
I3=(%pi)*(((Radio_eje3^2)^4)/4)//momento de inercia
J3=I3*3 // momento polar de inercia
// calculo de esfuerzos para tramo 3
sigma3=(Momento_max3*Radio_eje3/I3) // esfuerzo flexor
maximo
taotormed3=(Torque3*Radio_eje3/J3) // esfuerzo cortante
torsor alternante
taotoralt3=(Torque3*Radio_eje3/J3) // esfuerzo cortante torsor
medio

//FACTOR DE TAMAÑO

if (Diametro_eje3 < 0.11)
    f_tama3=1
end

if (Diametro_eje3 >= 0.11 & Diametro_eje3 <= 2)
    f_tama3=0.879*Diametro_eje3^-0.107
end

if (Diametro_eje3 >= 3 & Diametro_eje3 <= 10)
    f_tama3=0.91*(Diametro_eje3^(-0.157))
end

//factor de concentracion de esfuerzo
h3=Radio_eje3-Radio_eje2
relacion3=h3/Radio_muesca2

//0.1<=h/r<=2.0
C3_a1=0.926+(1.157*sqrt(h3/Radio_muesca2))-
(0.099*(h3/Radio_muesca2))

```

```

C3_a2=0.012-
3.036*sqrt(h3/Radio_muesca2)+0.961*(h3/Radio_muesca2)
C3_a3=-0.302+3.977*sqrt(h3/Radio_muesca2)-
1.744*(h3/Radio_muesca2)
C3_a4=0.365-
2.098*sqrt(h3/Radio_muesca2)+0.878*(h3/Radio_muesca2)

//2.0<=h/r<=20
C3_a11=1.2+0.860*sqrt(h3/Radio_muesca2)-
0.022*(h3/Radio_muesca2)
C3_a22=-1.805-0.346*sqrt(h3/Radio_muesca2)-
0.038*(h3/Radio_muesca2)
C3_a33=2.198-
0.486*sqrt(h3/Radio_muesca2)+0.165*(h3/Radio_muesca2)
C3_a44=-0.593-0.028*sqrt(h3/Radio_muesca2)-
0.106*(h3/Radio_muesca2)

if (relacion3>=0.1 & relacion3 <= 2)
    C3_A1=C3_a1
    C3_A2=C3_a2
    C3_A3=C3_a3
    C3_A4=C3_a4
    Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end

if (relacion3 > 2 & relacion3 <= 20)
    C3_A1=C3_a11
    C3_A2=C3_a22
    C3_A3=C3_a33
    C3_A4=C3_a44
    Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.1 then
    Kt3_a=1
end
if relacion3 > 20 then
    Kt3_a=0
end

KT3_A=Kt3_a

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C3_f1=0.947+1.206*sqrt(h3/Radio_muesca2)-
0.131*(h3/Radio_muesca2)
C3_f2=0.022-
3.405*sqrt(h3/Radio_muesca2)+0.915*(h3/Radio_muesca2)
C3_f3=0.869+1.777*sqrt(h3/Radio_muesca2)-
0.555*(h3/Radio_muesca2)
C3_f4=-0.810+0.422*sqrt(h3/Radio_muesca2)-
0.260*(h3/Radio_muesca2)

//2.0<=h/r<=20
C3_f11=1.232+0.832*sqrt(h3/Radio_muesca2)-
0.008*(h3/Radio_muesca2)
C3_f22=-3.813+0.968*sqrt(h3/Radio_muesca2)-
0.260*(h3/Radio_muesca2)
C3_f33=7.423-
4.868*sqrt(h3/Radio_muesca2)+0.869*(h3/Radio_muesca2)
C3_f44=-3.839+3.070*sqrt(h3/Radio_muesca2)-
0.600*(h3/Radio_muesca2)

if (relacion3 >=0.1 & relacion3 <= 2)

```

```

C3_F1=C3_f1
C3_F2=C3_f2
C3_F3=C3_f3
C3_F4=C3_f4
Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end

if (relacion3 > 2 & relacion3 <= 20)
  C3_F1=C3_f11
  C3_F2=C3_f22
  C3_F3=C3_f33
  C3_F4=C3_f44
  Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
  Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.1 then
  Kt3_f=1
end

if relacion3 > 20 then
  Kt3_f=1
end
KT3_F=Kt3_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C3_t1=0.905+0.783*sqrt(h3/Radio_muesca2)-
0.075*(h3/Radio_muesca2)
C3_t2=-0.437-
1.969*sqrt(h3/Radio_muesca2)+0.553*(h3/Radio_muesca2)
C3_t3=1.557+1.073*sqrt(h3/Radio_muesca2)-
0.578*(h3/Radio_muesca2)
C3_t4=-1.061+0.171*sqrt(h3/Radio_muesca2)-
0.086*(h3/Radio_muesca2)

if (relacion3 >=0.25 & relacion3 <= 4.0)
  C3_T1=C3_t1
  C3_T2=C3_t2
  C3_T3=C3_t3
  C3_T4=C3_t4
  Kt3_t=C3_T1+C3_T2*(2*h3/Diametro_eje3)+C3_T3*(2*h3/
  Diametro_eje3)^2+C3_T4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.25 then
  Kt3_t=1
end

if relacion3 > 4.0 then
  Kt3_t=1
end
KT3_T=Kt3_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)
constante_reemplazo_a3=(2*(KT3_A-
1)/KT3_A)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_a=KT3_A/((1+constante_reemplazo_a3))

constante_reemplazo_f3=(2*(KT3_F-
1)/KT3_F)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_f=KT3_F/((1+constante_reemplazo_f3))

constante_reemplazo_t3=(2*(KT3_T-
1)/KT3_T)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_t=KT3_T/((1+constante_reemplazo_t3))

```

//presion por interferencia

```

m3=(((2*Radio_engrane1)^2-
Diametro_eje3^2))*(Diametro_eje3^2))
n3=(2*Radio_engrane1)
PP3=((E*Interferen_maza_eje_max)/(2*Diametro_eje3^3))*(
m3/n3)

```

```

Am3=[(PP3*f_carga_axial*Kf3_a/f_tama3)
(taotormed3*f_carga_torsion*Kf3_t/f_tama3)
0;(taotormed3*f_carga_torsion*Kf3_t/f_tama3) 0 0;0 0 0]

```

//tensor de esfuerzo medio

```

[Em3]=spec(Am3)
sigvmm3=(((Em3(1)-Em3(2))^2)+(Em3(2)-
Em3(3))^2)+(Em3(3)-Em3(1))^2)/2^0.5//calcula esfuerzo
Von Mises medio

```

```

Aa3=[(siga3*f_carga_flexion*Kf3_f/f_tama3)
(taotoralt3*f_carga_torsion*Kf3_t/f_tama3) 0;0
(taotoralt3*f_carga_torsion*Kf3_t/f_tama3) 0;0 0 0] //tensor
de esfuerzo alternante

```

[Ea3]=spec(Aa3)

```

sigvma3=(((Ea3(1)-Ea3(2))^2)+(Ea3(2)-
Ea3(3))^2)+(Ea3(3)-Ea3(1))^2)/2^0.5//calcula esfuerzo Von
Mises medio

```

```

V_misses_med_LN3=[sigvmm3,
Coef_variacion_carga*sigvmm3]
V_misses_alt_LN3=[sigvma3,
Coef_variacion_carga*sigvma3]

```

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

```

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

```

if criterio_falla == 1 then

```

[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
D_Va_SE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

```

```

[Div_Va_SE_3_Cuad, Desv_Div_Va_SE_3_Cuad]=
pot(D_Va_SE3,2)
Division_VaSE3 = [Div_Va_SE_3_Cuad,
Desv_Div_Va_SE_3_Cuad]

```

```

[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

```

```

[Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]

```

```

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)

```

```

ASME_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_ASME_3=ASME_3(2)/ASME_3(1)
CRITERIO_F=ASME_3
CF_CRITERIO_F=Cf_ASME_3
end

```

if criterio_falla == 2 then

```

[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

```

```

[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

[Div_Vm_Sut_3_Cuad,      Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3      =      [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)

GERBER_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GERBER_3=GERBER_3(2)/GERBER_3(1)
CRITERIO_F=GERBER_3
CF_CRITERIO_F=Cf_GERBER_3
end

if criterio_falla == 3 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]
[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
Division_VmSut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)
GOODMAN_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GOODMAN_3=GOODMAN_3(2)/GOODMAN_3(1)
CRITERIO_F=GOODMAN_3
CF_CRITERIO_F=Cf_GOODMAN_3
end

MEDCnormal_3=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESNormal_3=CF_CRITERIO_F

[Prob_3,Conf_3]=cdfnorr("PQ",0,abs(MEDCnormal_3),DESNormal_3)

DIAMETRO_EJE3=0

if Prob_3<=1.5*10^-6 & Prob_3>=0.5*10^-6 then
    DIAMETRO_EJE3=Diametro_eje3
    break;
end
end
disp("El diametro del tramo 3 es ")
disp(DIAMETRO_EJE3)
for Diametro_eje4=tol:tol:6

Radio_eje4=Diametro_eje4/2
Torque4=(W_tangencial1/1000)*((Radio engrane1)-
Radio_eje3)
I4=(% pi*(((Radio_eje4^2)^4)/4))/inercia
J4=I4*2
// calculo de esfuerzos
sigma4=(Momento_max4*Radio_eje4/I4) // esfuerzo flector
maximo
taotormed4=(Torque4*Radio_eje4/J4) // esfuerzo cortante
torsor maximo
taotalt4=(Torque4*Radio_eje4/J4)

//FACTOR DE TAMAÑO
if (Diametro_eje4 < 0.11)

f_tama4=1
end
if (Diametro_eje4 >= 0.11 & Diametro_eje4 <= 2)
    f_tama4=0.879*Diametro_eje4^-0.107
end
if (Diametro_eje4 >= 3 & Diametro_eje4 <= 10)
    f_tama4=0.91*(Diametro_eje4^(-0.157))
end

//factor de concentracion de esfuerzo
h4=Radio_eje4-Radio_eje3
relacion4=h4/Radio_muesca3

//Kt para esfuerzos de flexión (Ktf)

//0.1 <= h/r <= 2.0
C4_f1=0.947+1.206*sqrt(h4/Radio_muesca3)-
0.131*(h4/Radio_muesca3)
C4_f2=0.022-
3.405*sqrt(h4/Radio_muesca3)+0.915*(h4/Radio_muesca3)
C4_f3=0.869+1.777*sqrt(h4/Radio_muesca3)-
0.555*(h4/Radio_muesca3)
C4_f4=-0.810+0.422*sqrt(h4/Radio_muesca3)-
0.260*(h4/Radio_muesca3)

//2.0 <= h/r <= 20
C4_f11=1.232+0.832*sqrt(h4/Radio_muesca3)-
0.008*(h4/Radio_muesca3)
C4_f22=-3.813+0.968*sqrt(h4/Radio_muesca3)-
0.260*(h4/Radio_muesca3)
C4_f33=7.423-
4.868*sqrt(h4/Radio_muesca3)+0.869*(h4/Radio_muesca3)
C4_f44=-3.839+3.070*sqrt(h4/Radio_muesca3)-
0.600*(h4/Radio_muesca3)

if (relacion4 >=0.1 & relacion4 <= 2)
    C4_F1=C4_f1
    C4_F2=C4_f2
    C4_F3=C4_f3
    C4_F4=C4_f4
    Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if (relacion4 > 2 & relacion4 <= 20)
    C4_F1=C4_f11
    C4_F2=C4_f22
    C4_F3=C4_f33
    C4_F4=C4_f44
    Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if relacion4 < 0.1 then
    Kt4_f=1
end
if relacion4 > 20 then
    Kt4_f=1
end

KT4_F=Kt4_f

//Kt para esfuerzos de torsión (Ktf)

//0.25 <= h/r <= 4.0
C4_t1=0.905+0.783*sqrt(h4/Radio_muesca3)-
0.075*(h4/Radio_muesca3)

```

```

C4_t2=-0.437-
1.969*sqrt(h4/Radio_muesca3)+0.553*(h4/Radio_muesca3)
C4_t3=1.557+1.073*sqrt(h4/Radio_muesca3)-
0.578*(h4/Radio_muesca3)
C4_t4=-1.061+0.171*sqrt(h4/Radio_muesca3)-
0.086*(h4/Radio_muesca3)

```

```

if (relacion4 >=0.25 & relacion4 <= 4.0)
  C4_T1=C4_t1
  C4_T2=C4_t2
  C4_T3=C4_t3
  C4_T4=C4_t4
  Kt4_t=C4_T1+C4_T2*(2*h4/Diametro_eje4)+C4_T3*(2*h4/
  Diametro_eje4)^2+C4_T4*(2*h4/Diametro_eje4)^3
end

```

```

if relacion4 < 0.25 then
  Kt4_t=1
end
if relacion4 > 4.0 then
  Kt4_t=1
end
KT4_T=Kt4_t

```

*//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)*

```

constante_reemplazo_f4=(2*(KT4_F-
1)/KT4_F)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf4_f=KT4_F/((1+constante_reemplazo_f4))

```

```

constante_reemplazo_t4=(2*(KT4_T-
1)/KT4_T)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf4_t=KT4_T/((1+constante_reemplazo_t4))

```

```

Am4=[0 (taotormed4*f_carga_torsion*Kf4_t/f_tama4)
0;(taotormed4*f_carga_torsion*Kf4_t/f_tama4) 0 0;0 0 0]
//tensor de esfuerzo medio
[Em4]=spec(Am4)
sigvmm4=(((Em4(1)-Em4(2))^2)+(Em4(2)-
Em4(3))^2)+((Em4(3)-Em4(1))^2)/2)^0.5//Vmises medio
Aa4=(sig4*f_carga_flexion*Kf4_f/f_tama4)
(taotoralt4*f_carga_torsion*Kf4_t/f_tama4) 0;0
(taotoralt4*f_carga_torsion*Kf4_t/f_tama4) 0;0 0 0] //tensor
de esfuerzo alternante

```

```

[Ea4]=spec(Aa4)
sigvma4=(((Ea4(1)-Ea4(2))^2)+(Ea4(2)-
Ea4(3))^2)+((Ea4(3)-Ea4(1))^2)/2)^0.5//Vmises alternante

```

```

V_misses_med_LN4=[sigvmm4,
Coef_variacion_carga*sigvmm4]
V_misses_alt_LN4=[sigvma4,
Coef_variacion_carga*sigvma4]

```

```

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

```

```

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

```

```

if criterio_falla == 1 then
  [Div_Va_SE_4,
  Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
  D_Va_SE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

```

```

[Div_Va_SE_4_Cuad, Desv_Div_Va_SE_4_Cuad]=
pot(D_Va_SE4,2)

```

```

Division_VaSE4 = [Div_Va_SE_4_Cuad,
Desv_Div_Va_SE_4_Cuad]

```

```

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

```

```

[Div_Vm_Sut_4_Cuad, Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4 = [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

```

```

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

```

```

ASME_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_ASME_4=ASME_4(2)/ASME_4(1)
CRITERIO_F=ASME_4
CF_CRITERIO_F=Cf_ASME_4
end

```

```

if criterio_falla == 2 then
  [Div_Va_SE_4,
  Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
  Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

```

```

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

```

```

[Div_Vm_Sut_4_Cuad, Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4 = [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

```

```

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

```

```

GERBER_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GERBER_4=GERBER_4(2)/GERBER_4(1)
CRITERIO_F=GERBER_4
CF_CRITERIO_F=Cf_GERBER_4
end

```

```

if criterio_falla == 3 then
  [Div_Va_SE_4,
  Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
  Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

```

```

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
Division_VmSut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

```

```

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)
GOODMAN_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GOODMAN_4=GOODMAN_4(2)/GOODMAN_4(1)
CRITERIO_F=GOODMAN_4
CF_CRITERIO_F=Cf_GOODMAN_4
end

```

```

MEDCnormal_4=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_4=CF_CRITERIO_F

```

```

[Prob_4,Conf_4]=cdfnorr("PQ",0,abs(MEDCnormal_4),DESV
normal_4)

```

```

DIAMETRO_EJE4=0

if Prob_4<=1.5*10^-6 & Prob_4>=0.5*10^-6 then
    DIAMETRO_EJE4=Diametro_eje4
    break;
end
end
disp("El diametro del tramo 4 es")
disp(DIAMETRO_EJE4)

for Diametro_eje5=tol:tol:6

Radio_eje5=Diametro_eje5/2

Torque5=(W_tangencial2/1000)*((Radio engrane2)-
Radio_eje5) // el torque aplicado es la fuerza tangencial por
el Radio del engranaje menos el Radio del eje
I5=(%pi*((Radio_eje5^2)^4))/4//ineria
Q5=(2/3)*(Radio_eje5^3)
J5=I5*2

//calculo de esfuerzos
siga5=(Momento_max5*Radio_eje5/I5) // esfuerzo flector
maximo
taotormed5=(Torque5*Radio_eje5/J5) // esfuerzo cortante
torsor maximo
taotoralt5=(Torque5*Radio_eje5/J5)

//FACTOR DE TAMAÑO
if (Diametro_eje5 < 0.11)
    f_tama5=1
end
if (Diametro_eje5 >= 0.11 & Diametro_eje5 <= 2)
    f_tama5=0.879*Diametro_eje5^-0.107
end
if (Diametro_eje5 >= 3 & Diametro_eje5 <= 10)
    f_tama5=0.91*(Diametro_eje5^(-0.157))
end
//factor de concentracion de esfuerzo
h5=Radio_eje5-Radio_eje4
relacion5=h5/Radio_muesca4

//0.1<=h/r<=2.0
C5_a1=          0.926+(1.157*sqrt(h5/Radio_muesca4))-
(0.099*(h5/Radio_muesca4))
C5_a2=0.012-
3.036*sqrt(h5/Radio_muesca4)+0.961*(h5/Radio_muesca4)
C5_a3=-0.302+3.977*sqrt(h5/Radio_muesca4)-
1.744*(h5/Radio_muesca4)
C5_a4=0.365-
2.098*sqrt(h5/Radio_muesca4)+0.878*(h5/Radio_muesca4)

//2.0<=h/r<=20
C5_a11=1.2+0.860*sqrt(h5/Radio_muesca4)-
0.022*(h5/Radio_muesca4)
C5_a22=-1.805-0.346*sqrt(h5/Radio_muesca4)-
0.038*(h5/Radio_muesca4)
C5_a33=2.198-
0.486*sqrt(h5/Radio_muesca4)+0.165*(h5/Radio_muesca4)
C5_a44=-0.593-0.028*sqrt(h5/Radio_muesca4)-
0.106*(h5/Radio_muesca4)

if (relacion5 >=0.1 & relacion5 <= 2)
    C5_A1=C5_a1
    C5_A2=C5_a2
    C5_A3=C5_a3
    C5_A4=C5_a4

```

```

Kt5_a=C5_A1+C5_A2*(2*h5/Diametro_eje5)+C5_A3*(2*h5/
Diametro_eje5)^2+C5_A4*(2*h5/Diametro_eje5)^3
end

if (relacion5 > 2 & relacion5 <= 20)
    C5_A1=C5_a11
    C5_A2=C5_a22
    C5_A3=C5_a33
    C5_A4=C5_a44
Kt5_a=C5_A1+C5_A2*(2*h5/Diametro_eje5)+C5_A3*(2*h5/
Diametro_eje5)^2+C5_A4*(2*h5/Diametro_eje5)^3
end
if relacion5 < 0.1 then
    Kt5_a=1
end

if relacion5 > 20 then
    Kt5_a=1
end
KT5_A=Kt5_a
//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C5_f1=0.947+1.206*sqrt(h5/Radio_muesca4)-
0.131*(h5/Radio_muesca4)
C5_f2=0.022-
3.405*sqrt(h5/Radio_muesca4)+0.915*(h5/Radio_muesca4)
C5_f3=0.869+1.777*sqrt(h5/Radio_muesca4)-
0.555*(h5/Radio_muesca4)
C5_f4=-0.810+0.422*sqrt(h5/Radio_muesca4)-
0.260*(h5/Radio_muesca4)

//2.0<=h/r<=20
C5_f11=1.232+0.832*sqrt(h5/Radio_muesca4)-
0.008*(h5/Radio_muesca4)
C5_f22=-3.813+0.968*sqrt(h5/Radio_muesca4)-
0.260*(h5/Radio_muesca4)
C5_f33=7.423-
4.868*sqrt(h5/Radio_muesca4)+0.869*(h5/Radio_muesca4)
C5_f44=-3.839+3.070*sqrt(h5/Radio_muesca4)-
0.600*(h5/Radio_muesca4)

if (relacion5 >=0.1 & relacion5 <= 2)
    C5_F1=C5_f1
    C5_F2=C5_f2
    C5_F3=C5_f3
    C5_F4=C5_f4
Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

if (relacion5 > 2 & relacion5 <= 20)
    C5_F1=C5_f11
    C5_F2=C5_f22
    C5_F3=C5_f33
    C5_F4=C5_f44
Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

if relacion5 < 0.1 then
    Kt5_f=1
end
if relacion5 > 20 then
    Kt5_f=1
end
KT5_F=Kt5_f

```

```

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C5_t1=0.905+0.783*sqrt(h5/Radio_muesca4)-
0.075*(h5/Radio_muesca4)
C5_t2=-0.437-
1.969*sqrt(h5/Radio_muesca4)+0.553*(h5/Radio_muesca4)
C5_t3=1.557+1.073*sqrt(h5/Radio_muesca4)-
0.578*(h5/Radio_muesca4)
C5_t4=-1.061+0.171*sqrt(h5/Radio_muesca4)-
0.086*(h5/Radio_muesca4)

if (relacion5 >=0.25 & relacion5 <= 4.0)
  C5_T1=C5_t1
  C5_T2=C5_t2
  C5_T3=C5_t3
  C5_T4=C5_t4
  Kt5_t=C5_T1+C5_T2*(2*h5/Diametro_eje5)+C5_T3*(2*h5/
  Diametro_eje5)^2+C5_T4*(2*h5/Diametro_eje5)^3
end
if relacion5 < 0.25 then
  Kt5_t=1
end
if relacion5 > 4.0 then
  Kt5_t=1
end
KT5_T=Kt5_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f flexión), (Kf_t torsión)
constante_reemplazo_a5=(2*(KT5_A-
1)/KT5_A)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_a=KT5_A/((1+constante_reemplazo_a5))

constante_reemplazo_f5=(2*(KT5_F-
1)/KT5_F)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_f=KT5_F/((1+constante_reemplazo_f5))

constante_reemplazo_t5=(2*(KT5_T-
1)/KT5_T)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_t=KT5_T/((1+constante_reemplazo_t5))

//presion por interferencia
m2=(((2*Radio_engrane2)^2-
Diametro_eje5^2)*(Diametro_eje5^2))
n2=(2*Radio_engrane2)
PP5=((E*Interferen_maza_eje_max2)/(2*Diametro_eje5^3))*
(m2/n2)

Am5=[(PP5*f_carga_axial*Kf5_a/f_tama5)
(taotormed5*f_carga_torsion*Kf5_t/f_tama5)
0;(taotormed5*f_carga_torsion*Kf5_t/f_tama5) 0 0;0 0 0]
//tensor de esfuerzo medio//
[Em5]=spec(Am5)
sigvmm5=(((Em5(1)-Em5(2))^2)+((Em5(2)-
Em5(3))^2)+((Em5(3)-Em5(1))^2))/2^0.5

Aa5=[(siga5*f_carga_flexion*Kf5_f/f_tama5)
(taotormed5*f_carga_torsion*Kf5_t/f_tama5) 0;0
(taotormed5*f_carga_torsion*Kf5_t/f_tama5) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea5]=spec(Aa5)
sigvma5=(((Ea5(1)-Ea5(2))^2)+((Ea5(2)-
Ea5(3))^2)+((Ea5(3)-Ea5(1))^2))/2^0.5

```

```

V_misses_med_LN5=[sigvmm5,
Coef_variacion_carga*sigvmm5]
V_misses_alt_LN5=[sigvma5,
Coef_variacion_carga*sigvma5]

```

```
[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)
```

```
SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]
```

```
if criterio_falla == 1 then
```

```
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
D_Va_SE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]
```

```
[Div_Va_SE_5_Cuad, Desv_Div_Va_SE_5_Cuad]=
pot(D_Va_SE5,2)
Division_VaSE5 = [Div_Va_SE_5_Cuad,
Desv_Div_Va_SE_5_Cuad]
```

```
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]
```

```
[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]
```

```
[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
```

```
ASME_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_ASME_5=ASME_5(2)/ASME_5(1)
CRITERIO_F=ASME_5
CF_CRITERIO_F=Cf_ASME_5
end
```

```
if criterio_falla == 2 then
```

```
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]
```

```
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]
```

```
[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]
```

```
[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
```

```
GERBER_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_GERBER_5=GERBER_5(2)/GERBER_5(1)
CRITERIO_F=GERBER_5
CF_CRITERIO_F=Cf_GERBER_5
end
```

```
if criterio_falla == 3 then
```

```
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]
```

```
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
```



```
Division_VmSut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]
```

```
[Suma5_VmSut,Desv_Suma5_VmSut]  
=suma(Division_VaSE5, Division_VmSut5)  
GOODMAN_5=[Suma5_VmSut,Desv_Suma5_VmSut]  
Cf_GOODMAN_5=GOODMAN_5(2)/GOODMAN_5(1)  
CRITERIO_F=GOODMAN_5  
CF_CRITERIO_F=Cf_GOODMAN_5  
end
```

```
MEDCnormal_5=log(CRITERIO_F(1))-  
((0.5)*(CF_CRITERIO_F)^2)  
DESVnormal_5=CF_CRITERIO_F
```

```
[Prob_5,Conf_5]=cdfnor("PQ",0,abs(MEDCnormal_5),DESV  
normal_5)
```

```
DIAMETRO_EJE5=0
```

```
if Prob_5<=1.5*10^-6 & Prob_5>=0.5*10^-6 then  
    DIAMETRO_EJE5=Diametro_eje5  
    break;
```

```
end  
end  
disp("El diametro del tramo 5 es")  
disp(DIAMETRO_EJE5)
```

```
for Diametro_eje6=tol:tol:6
```

```
Radio_eje6=Diametro_eje6/2  
Torque6=(W_tangencial2/1000)*((Radio_engrane2)-  
Radio_eje5)  
I6=(%pi*(((Radio_eje6*2)^4))/4)/inercia  
J6=I6*2  
// calculo de esfuerzos  
sigma6=(Momento_max6*Radio_eje6/I6) // esfuerzo flexor  
maximo  
taotormed6=(Torque6*Radio_eje6/J6) // esfuerzo cortante  
torsor maximo  
taotoralt6=(Torque6*Radio_eje6/J6)
```

```
//FACTOR DE TAMAÑO  
if (Diametro_eje6 < 0.11)  
    f_tama6=1  
end  
if (Diametro_eje6 >= 0.11 & Diametro_eje6 <= 2)  
    f_tama6=0.879*Diametro_eje6^-0.107  
end  
if (Diametro_eje6 >= 3 & Diametro_eje6 <= 10)  
    f_tama6=0.91*(Diametro_eje6^(-0.157))  
end
```

```
//factor de concentracion de esfuerzo  
h6=Radio_eje6-Radio_eje5  
relacion6=h6/Radio_muesca5
```

```
//Kt para esfuerzos de flexión (Ktf)
```

```
//0.1 <= h/r <= 2.0  
C6_f1=0.947+1.206*sqrt(h6/Radio_muesca5)-  
0.131*(h6/Radio_muesca5)  
C6_f2=0.022-  
3.405*sqrt(h6/Radio_muesca5)+0.915*(h6/Radio_muesca5)  
C6_f3=0.869+1.777*sqrt(h6/Radio_muesca5)-  
0.555*(h6/Radio_muesca5)  
C6_f4=-0.810+0.422*sqrt(h6/Radio_muesca5)-  
0.260*(h6/Radio_muesca5)
```

```
//2.0 <= h/r <= 20
```

```
C6_f11=1.232+0.832*sqrt(h6/Radio_muesca5)-  
0.008*(h6/Radio_muesca5)  
C6_f22=-3.813+0.968*sqrt(h6/Radio_muesca5)-  
0.260*(h6/Radio_muesca5)  
C6_f33=7.423-  
4.868*sqrt(h6/Radio_muesca5)+0.869*(h6/Radio_muesca5)  
C6_f44=-3.839+3.070*sqrt(h6/Radio_muesca5)-  
0.600*(h6/Radio_muesca5)
```

```
if (relacion6 >= 0.1 & relacion6 <= 2)
```

```
    C6_F1=C6_f1  
    C6_F2=C6_f2  
    C6_F3=C6_f3  
    C6_F4=C6_f4  
    Kt6_f=C6_F1+C6_F2*(2*h6/Diametro_eje6)+C6_F3*(2*h6/  
Diametro_eje6)^2+C6_F4*(2*h6/Diametro_eje6)^3  
end
```

```
if (relacion6 > 2 & relacion6 <= 20)
```

```
    C6_F1=C6_f11  
    C6_F2=C6_f22  
    C6_F3=C6_f33  
    C6_F4=C6_f44  
    Kt6_f=C6_F1+C6_F2*(2*h6/Diametro_eje6)+C6_F3*(2*h6/  
Diametro_eje6)^2+C6_F4*(2*h6/Diametro_eje6)^3  
end
```

```
if relacion6 < 0.1 then
```

```
    Kt6_f=1  
end  
if relacion6 > 20 then  
    Kt6_f=1  
end  
Kt6_F=Kt6_f
```

```
//Kt para esfuerzos de torsión (Ktf)
```

```
//0.25 <= h/r <= 4.0
```

```
C6_t1=0.905+0.783*sqrt(h6/Radio_muesca5)-  
0.075*(h6/Radio_muesca5)  
C6_t2=-0.437-  
1.969*sqrt(h6/Radio_muesca5)+0.553*(h6/Radio_muesca5)  
C6_t3=1.557+1.073*sqrt(h6/Radio_muesca5)-  
0.578*(h6/Radio_muesca5)  
C6_t4=-1.061+0.171*sqrt(h6/Radio_muesca5)-  
0.086*(h6/Radio_muesca5)
```

```
if (relacion6 >= 0.25 & relacion6 <= 4.0)
```

```
    C6_T1=C6_t1  
    C6_T2=C6_t2  
    C6_T3=C6_t3  
    C6_T4=C6_t4  
    Kt6_t=C6_T1+C6_T2*(2*h6/Diametro_eje6)+C6_T3*(2*h6/  
Diametro_eje6)^2+C6_T4*(2*h6/Diametro_eje6)^3  
end
```

```
if relacion6 < 0.25 then
```

```
    Kt6_t=1  
end  
if relacion6 > 20 then  
    Kt6_t=1  
end  
Kt6_T=Kt6_t
```

```
constante_reemplazo_f6=(2*(Kt6_F-  
1)/Kt6_F)*(RAIZ_DE_A/sqrt(Radio_muesca5))  
Kf6_f=Kt6_F/((1+constante_reemplazo_f6))
```

```

constante_reemplazo_t6=(2*(KT6_T-
1)/KT6_T)*(RAIZ_DE_A/sqrt(Radio_muesca5))
Kf6_t=KT6_T/((1+constante_reemplazo_t6))

```

```

Am6=[0 (taotormed6*f_carga_torsion*Kf6_t/f_tama6)
0;(taotormed6*f_carga_torsion*Kf6_t/f_tama6) 0 0;0 0 0]
//tensor de esfuerzo medio
[Em6]=spec(Am6)
sigvmm6=(((Em6(1)-Em6(2))^2)+(Em6(2)-
Em6(3))^2)+(Em6(3)-Em6(1))^2)/2^0.5

```

```

Aa6=[(siga6*f_carga_flexion*Kf6_f/f_tama6)
(taotort6*f_carga_torsion*Kf6_t/f_tama6) 0;0
(taotort6*f_carga_torsion*Kf6_t/f_tama6) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea6]=spec(Aa6)
sigvma6=(((Ea6(1)-Ea6(2))^2)+(Ea6(2)-
Ea6(3))^2)+(Ea6(3)-Ea6(1))^2)/2^0.5

```

```

V_misses_med_LN6=[sigvmm6,
Coef_variacion_carga*sigvmm6]
V_misses_alt_LN6=[sigvma6,
Coef_variacion_carga*sigvma6]

```

```
[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)
```

```
SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]
```

```

if criterio_falla == 1 then
[Div_Va_SE_6,
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
D_Va_SE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]

```

```

[Div_Va_SE_6_Cuad, Desv_Div_Va_SE_6_Cuad]=
pot(D_Va_SE6,2)
Division_VaSE6 = [Div_Va_SE_6_Cuad,
Desv_Div_Va_SE_6_Cuad]

```

```
Division_VmSut6 = [0,0]
```

```

Suma6_VmSut_VaSE=Division_VaSE6(1)+Division_VmSut6
(1)
Desv_Suma6_VmSut_VaSE=sqrt((Division_VaSE6(2)/Divisio
n_VaSE6(1))^2+(0)^2)

```

```

ASME_6=[Suma6_VmSut_VaSE,
Desv_Suma6_VmSut_VaSE]
Cf_ASME_6=ASME_6(2)/ASME_6(1)
CRITERIO_F=ASME_6
CF_CRITERIO_F=Cf_ASME_6
end

```

```

if criterio_falla == 2 then
[Div_Va_SE_6,
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
Division_VaSE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]

```

```
Division_VmSut6 = [0,0]
```

```

Suma6_VmSut_VaSE=Division_VaSE6(1)+Division_VmSut6
(1)
Desv_Suma6_VmSut_VaSE=sqrt((Division_VaSE6(2)/Divisio
n_VaSE6(1))^2+(0)^2)

```

```

GERBER_6=[Suma6_VmSut_VaSE,
Desv_Suma6_VmSut_VaSE]
Cf_GERBER_6=GERBER_6(2)/GERBER_6(1)
CRITERIO_F=GERBER_6
CF_CRITERIO_F=Cf_GERBER_6
end

```

```

if criterio_falla == 3 then
[Div_Va_SE_6,
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
Division_Va_SE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]

```

```
Division_VmSut6 = [0,0]
```

```

Suma6_VmSut_VaSE=Division_Va_SE6(1)+Division_VmSut
6(1)
Desv_Suma6_VmSut_VaSE=sqrt((Division_Va_SE6(2)/Divisi
on_Va_SE6(1))^2+(0)^2)
GOODMAN_6=[Suma6_VmSut_VaSE,
Desv_Suma6_VmSut_VaSE]
Cf_GOODMAN_6=GOODMAN_6(2)/GOODMAN_6(1)
CRITERIO_F=GOODMAN_6
CF_CRITERIO_F=Cf_GOODMAN_6
end

```

```

MEDCnormal_6=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_6=CF_CRITERIO_F

```

```
[Prob_6,Conf_6]=cdfnor("PQ",0,abs(MEDCnormal_6),DESV
normal_6)
```

```
DIAMETRO_EJE6=0
```

```

if Prob_6<=1.5*10^-6 & Prob_6>=0.5*10^-6 then
DIAMETRO_EJE6=Diametro_eje6
break;
end

```

```

end
disp("El diametro del tramo 6 es")
disp(DIAMETRO_EJE6)

```

```
for Diametro_eje7=tol:tol:6
```

```
Radio_eje7=Diametro_eje7/2
```

```

Torque7=(W_tangencial3/1000)*((Radio engrane3)-
Radio_eje7) // el torque aplicado es la fuerza tangencial por
el Radio del engranaje menos el Radio del eje
I7=(%pi*(((Radio_eje7^2)^4))/4)/inercia
Q7=(2/3)*(Radio_eje7^3)
J7=I7*2

```

```

//calculo de esfuerzos
siga7=(Momento_max7*Radio_eje7/I7) // esfuerzo flexor
maximo
taotormed7=(Torque7*Radio_eje7/J7) // esfuerzo cortante
torsor maximo
taotort7=(Torque7*Radio_eje7/J7)

```

```
//FACTOR DE TAMAÑO
```

```

if (Diametro_eje7 < 0.11)
f_tama7=1
end
if (Diametro_eje7 >= 0.11 & Diametro_eje7 <= 2)

```



```

f_tama7=0.879*Diametro_eje7^0.107
end
if (Diametro_eje7 >= 3 & Diametro_eje7 <= 10)
f_tama7=0.91*(Diametro_eje7^(-0.177))
end

//factor de concentracion de esfuerzo
h7=Radio_eje7-Radio_eje6
relacion7=h7/Radio_muesca6

//0.1<=h/r<=2.0
C7_a1=0.926+(1.157*sqrt(h7/Radio_muesca6))-
(0.099*(h7/Radio_muesca6))
C7_a2=0.012-
3.036*sqrt(h7/Radio_muesca6)+0.961*(h7/Radio_muesca6)
C7_a3=-0.302+3.977*sqrt(h7/Radio_muesca6)-
1.744*(h7/Radio_muesca6)
C7_a4=0.365-
2.098*sqrt(h7/Radio_muesca6)+0.878*(h7/Radio_muesca6)

//2.0<=h/r<=20
C7_a11=1.2+0.860*sqrt(h7/Radio_muesca6)-
0.022*(h7/Radio_muesca6)
C7_a22=-1.805-0.346*sqrt(h7/Radio_muesca6)-
0.038*(h7/Radio_muesca6)
C7_a33=2.198-
0.486*sqrt(h7/Radio_muesca6)+0.165*(h7/Radio_muesca6)
C7_a44=-0.593-0.028*sqrt(h7/Radio_muesca6)-
0.106*(h7/Radio_muesca6)

if (relacion7 >=0.1 & relacion7 <= 2)
C7_A1=C7_a1
C7_A2=C7_a2
C7_A3=C7_a3
C7_A4=C7_a4
Kt10_a=C7_A1+C7_A2*(2*h7/Diametro_eje7)+C7_A3*(2*h
7/Diametro_eje7)^2+C7_A4*(2*h7/Diametro_eje7)^3
end

if (relacion7 > 2 & relacion7 <= 20)
C7_A1=C7_a11
C7_A2=C7_a22
C7_A3=C7_a33
C7_A4=C7_a44
Kt10_a=C7_A1+C7_A2*(2*h7/Diametro_eje7)+C7_A3*(2*h
7/Diametro_eje7)^2+C7_A4*(2*h7/Diametro_eje7)^3
end
if relacion7 < 0.1 then
Kt10_a=1
end
if relacion7 > 20 then
Kt10_a=1
end
KT7_A=Kt10_a

//Kt para esfuerzos de flexión (Kf)

//0.1<=h/r<=2.0
C7_f1=0.947+1.206*sqrt(h7/Radio_muesca6)-
0.131*(h7/Radio_muesca6)
C7_f2=0.022-
3.405*sqrt(h7/Radio_muesca6)+0.915*(h7/Radio_muesca6)
C7_f3=0.869+1.777*sqrt(h7/Radio_muesca6)-
0.555*(h7/Radio_muesca6)
C7_f4=-0.810+0.422*sqrt(h7/Radio_muesca6)-
0.260*(h7/Radio_muesca6)

//2.0<=h/r<=20
C7_f11=1.232+0.832*sqrt(h7/Radio_muesca6)-
0.008*(h7/Radio_muesca6)
C7_f22=-3.813+0.968*sqrt(h7/Radio_muesca6)-
0.260*(h7/Radio_muesca6)
C7_f33=7.423-
4.868*sqrt(h7/Radio_muesca6)+0.869*(h7/Radio_muesca6)
C7_f44=-3.839+3.070*sqrt(h7/Radio_muesca6)-
0.600*(h7/Radio_muesca6)

if (relacion7 >=0.1 & relacion7 <= 2)
C7_F1=C7_f1
C7_F2=C7_f2
C7_F3=C7_f3
C7_F4=C7_f4
Kt10_f=C7_F1+C7_F2*(2*h7/Diametro_eje7)+C7_F3*(2*h7/
Diametro_eje7)^2+C7_F4*(2*h7/Diametro_eje7)^3
end

if (relacion7 > 2 & relacion7 <= 20)
C7_F1=C7_f11
C7_F2=C7_f22
C7_F3=C7_f33
C7_F4=C7_f44
Kt10_f=C7_F1+C7_F2*(2*h7/Diametro_eje7)+C7_F3*(2*h7/
Diametro_eje7)^2+C7_F4*(2*h7/Diametro_eje7)^3
end
if relacion7 < 0.1 then
Kt10_f=1
end
if relacion7 > 20 then
Kt10_f=1
end
KT7_F=Kt10_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C7_t1=0.905+0.783*sqrt(h7/Radio_muesca6)-
0.075*(h7/Radio_muesca6)
C7_t2=-0.437-
1.969*sqrt(h7/Radio_muesca6)+0.553*(h7/Radio_muesca6)
C7_t3=1.557+1.073*sqrt(h7/Radio_muesca6)-
0.578*(h7/Radio_muesca6)
C7_t4=-1.061+0.171*sqrt(h7/Radio_muesca6)-
0.086*(h7/Radio_muesca6)

if (relacion7 >=0.25 & relacion7 <= 4.0)
C7_T1=C7_t1
C7_T2=C7_t2
C7_T3=C7_t3
C7_T4=C7_t4
Kt10_t=C7_T1+C7_T2*(2*h7/Diametro_eje7)+C7_T3*(2*h7/
Diametro_eje7)^2+C7_T4*(2*h7/Diametro_eje7)^3
end

if relacion7 < 0.25 then
Kt10_t=1
end
if relacion7 > 0.25 then
Kt10_t=1
end
KT7_T=Kt10_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

```

```

constante_reemplazo_a7=(2*(KT7_A-
1)/KT7_A)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_a=KT7_A/((1+constante_reemplazo_a7))

constante_reemplazo_f7=(2*(KT7_F-
1)/KT7_F)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_f=KT7_F/((1+constante_reemplazo_f7))

constante_reemplazo_t7=(2*(KT7_T-
1)/KT7_T)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_t=KT7_T/((1+constante_reemplazo_t7))
//presion por interferencia
m3=(((2*Radio_engrane3)^2-
Diametro_eje7^2)*(Diametro_eje7^2))
n3=(2*Radio_engrane3)
PP7=((E*Interferen_maza_eje_max3)/(2*Diametro_eje7^3))*(
m3/n3)

Am7=[(PP7*f_carga_axial*Kf7_a/f_tama7)
(taotormed7*f_carga_torsion*Kf7_t/f_tama7)
0;(taotormed7*f_carga_torsion*Kf7_t/f_tama7) 0 0;0 0 0]
//tensor de esfuerzo medio//
[Em7]=spec(Am7)
sigvmm7=(((Em7(1)-Em7(2))^2)+((Em7(2)-
Em7(3))^2)+((Em7(3)-Em7(1))^2))/2^0.5

Aa7=[(siga7*f_carga_flexion*Kf7_f/f_tama7)
(taotort7*f_carga_torsion*Kf7_t/f_tama7) 0;0
(taotort7*f_carga_torsion*Kf7_t/f_tama7) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea7]=spec(Aa7)
sigvma7=(((Ea7(1)-Ea7(2))^2)+((Ea7(2)-
Ea7(3))^2)+((Ea7(3)-Ea7(1))^2))/2^0.5

V_misses_med_LN7=[sigvmm7,
Coef_variacion_carga*sigvmm7]
V_misses_alt_LN7=[sigvma7,
Coef_variacion_carga*sigvma7]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
D_Va_SE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

[Div_Va_SE_7_Cuad,          Desv_Div_Va_SE_7_Cuad]=
pot(D_Va_SE7,2)
Division_VaSE7 = [Div_Va_SE_7_Cuad,
Desv_Div_Va_SE_7_Cuad]

[Div_Vm_Sut_7,
Desv_Div_Vm_Sut_7]=div(V_misses_med_LN7, Sut)
D_Vm_Sut7 = [Div_Vm_Sut_7, Desv_Div_Vm_Sut_7]

[Div_Vm_Sut_7_Cuad,          Desv_Div_Vm_Sut_7_Cuad]=
pot(D_Vm_Sut7,2)
Division_VmSut7 = [Div_Vm_Sut_7_Cuad,
Desv_Div_Vm_Sut_7_Cuad]

[Suma7_VmSut,Desv_Suma7_VmSut]
=suma(Division_VaSE7, Division_VmSut7)

ASME_7=[Suma7_VmSut,Desv_Suma7_VmSut]

```

```

Cf_ASME_7=ASME_7(2)/ASME_7(1)
CRITERIO_F=ASME_7
CF_CRITERIO_F=Cf_ASME_7
end

if criterio_falla == 2 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
Division_VaSE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

[Div_Vm_Sut_7,
Desv_Div_Vm_Sut_7]=div(V_misses_med_LN7, Sut)
D_Vm_Sut7 = [Div_Vm_Sut_7, Desv_Div_Vm_Sut_7]

[Div_Vm_Sut_7_Cuad,          Desv_Div_Vm_Sut_7_Cuad]=
pot(D_Vm_Sut7,2)
Division_VmSut7 = [Div_Vm_Sut_7_Cuad,
Desv_Div_Vm_Sut_7_Cuad]

[Suma7_VmSut,Desv_Suma7_VmSut]
=suma(Division_VaSE7, Division_VmSut7)

GERBER_7=[Suma7_VmSut,Desv_Suma7_VmSut]
Cf_GERBER_7=GERBER_7(2)/GERBER_7(1)
CRITERIO_F=GERBER_7
CF_CRITERIO_F=Cf_GERBER_7
end

if criterio_falla == 3 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
Division_VaSE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]
[Div_Vm_Sut_7,
Desv_Div_Vm_Sut_7]=div(V_misses_med_LN7, Sut)
Division_VmSut7 = [Div_Vm_Sut_7, Desv_Div_Vm_Sut_7]

[Suma7_VmSut,Desv_Suma7_VmSut]
=suma(Division_VaSE7, Division_VmSut7)
GOODMAN_7=[Suma7_VmSut,Desv_Suma7_VmSut]
Cf_GOODMAN_7=GOODMAN_7(2)/GOODMAN_7(1)
CRITERIO_F=GOODMAN_7
CF_CRITERIO_F=Cf_GOODMAN_7
end
MEDCnormal_7=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_7=CF_CRITERIO_F

[Prob_7,Conf_7]=cdfnor("PQ",0,abs(MEDCnormal_7),DESV
normal_7)

DIAMETRO_EJE7=0

if Prob_7<=1.5*10^-6 & Prob_7>=0.5*10^-6 then
DIAMETRO_EJE7=Diametro_eje7
break;
end

end
disp("El diametro del tramo 7 es")
disp(DIAMETRO_EJE7)

for Diametro_eje8=tol:tol:6

Radio_eje8=Diametro_eje8/2
I8=(%pi*(((Radio_eje8*2)^4))/4)/inercia
J8=I8*2

```

```

// calculo de esfuerzos
siga8=(Momento_max8*Radio_eje8/I8) // esfuerzo flector
maximo

//FACTOR DE TAMAÑO
if (Diametro_eje8 < 0.11)
    f_tama8=1
end
if (Diametro_eje8 >= 0.11 & Diametro_eje8 <= 2)
    f_tama8=0.879*Diametro_eje8^-0.107
end
if (Diametro_eje8 >= 3 & Diametro_eje8 <= 10)
    f_tama8=0.91*(Diametro_eje8^(-0.157))
end

//factor de concentracion de esfuerzo
h8=Radio_eje8-Radio_eje7
relacion8=h8/Radio_muesca7

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C8_f1=0.947+1.206*sqrt(h8/Radio_muesca7)-
0.131*(h8/Radio_muesca7)
C8_f2=0.022-
3.405*sqrt(h8/Radio_muesca7)+0.915*(h8/Radio_muesca7)
C8_f3=0.869+1.777*sqrt(h8/Radio_muesca7)-
0.555*(h8/Radio_muesca7)
C8_f4=-0.810+0.422*sqrt(h8/Radio_muesca7)-
0.260*(h8/Radio_muesca7)

//2.0<=h/r<=20
C8_f11=1.232+0.832*sqrt(h8/Radio_muesca7)-
0.008*(h8/Radio_muesca7)
C8_f22=-3.813+0.968*sqrt(h8/Radio_muesca7)-
0.260*(h8/Radio_muesca7)
C8_f33=7.423-
4.868*sqrt(h8/Radio_muesca7)+0.869*(h8/Radio_muesca7)
C8_f44=-3.839+3.070*sqrt(h8/Radio_muesca7)-
0.600*(h8/Radio_muesca7)

if (relacion8 >=0.1 & relacion8 <= 2)
    C8_F1=C8_f1
    C8_F2=C8_f2
    C8_F3=C8_f3
    C8_F4=C8_f4
    Kt8_f=C8_F1+C8_F2*(2*h8/Diametro_eje8)+C8_F3*(2*h8/
    Diametro_eje8)^2+C8_F4*(2*h8/Diametro_eje8)^3
end

if (relacion8 > 2 & relacion8 <= 20)
    C8_F1=C8_f11
    C8_F2=C8_f22
    C8_F3=C8_f33
    C8_F4=C8_f44
    Kt8_f=C8_F1+C8_F2*(2*h8/Diametro_eje8)+C8_F3*(2*h8/
    Diametro_eje8)^2+C8_F4*(2*h8/Diametro_eje8)^3
end

if relacion8 < 0.1 then
    Kt8_f=1
end
if relacion8 > 20 then
    Kt8_f=1
end
Kt8_F=Kt8_f
//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C8_t1=0.905+0.783*sqrt(h8/Radio_muesca7)-
0.075*(h8/Radio_muesca7)
C8_t2=-0.437-
1.969*sqrt(h8/Radio_muesca7)+0.553*(h8/Radio_muesca7)
C8_t3=1.557+1.073*sqrt(h8/Radio_muesca7)-
0.578*(h8/Radio_muesca7)
C8_t4=-1.061+0.171*sqrt(h8/Radio_muesca7)-
0.086*(h8/Radio_muesca7)

if (relacion8 >=0.25 & relacion8 <= 4.0)
    C8_T1=C8_t1
    C8_T2=C8_t2
    C8_T3=C8_t3
    C8_T4=C8_t4
    Kt8_t=C8_T1+C8_T2*(2*h8/Diametro_eje8)+C8_T3*(2*h8/
    Diametro_eje8)^2+C8_T4*(2*h8/Diametro_eje8)^3
end

if relacion8 < 0.25 then
    Kt8_t=1
end
if relacion8 > 4.0 then
    Kt8_t=1
end
Kt8_T=Kt8_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f- flexión), (Kf_t-torsión)

constante_reemplazo_f8=(2*(Kt8_F-
1)/Kt8_F)*(RAIZ_DE_A/sqrt(Radio_muesca7))
constante_reemplazo_t8=(2*(Kt8_T-
1)/Kt8_T)*(RAIZ_DE_A/sqrt(Radio_muesca7))
Kf8_f=Kt8_F/((1+constante_reemplazo_f8))
Kf8_t=Kt8_T/((1+constante_reemplazo_t8))

Torque8=Torque7
//Esfuerzo cortante torsor alternante:
taotormed8=(Torque8*Radio_eje8/J8)
//Esfuerzo cortante torsor medio:
taotoralt8=(Torque8*Radio_eje8/J8)

Am8=[0 (taotormed8*f_carga_torsion*Kf8_t/f_tama8)
0;(taotormed8*f_carga_torsion*Kf8_t/f_tama8) 0 0;0 0 0]
//tensor de esfuerzo medio//
[Em8]=spec(Am8)
sigvmm8=(((Em8(1)-Em8(2))^2)+((Em8(2)-
Em8(3))^2)+((Em8(3)-Em8(1))^2))/2^0.5

Aa8=[(siga8*f_carga_flexion*Kf8_f/f_tama8)
(taotoralt8*f_carga_torsion*Kf8_t/f_tama8) 0;0 0 0]
[ Ea8]=spec(Aa8)
sigvma8=(((Ea8(1)-Ea8(2))^2)+((Ea8(2)-
Ea8(3))^2)+((Ea8(3)-Ea8(1))^2))/2^0.5

V_misses_med_LN8=[sigvmm8,
Coef_variacion_carga*sigvmm8]
V_misses_alt_LN8=[sigvma8,
Coef_variacion_carga*sigvma8]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

```

```

if criterio_falla == 1 then
[Div_Va_SE_8,
Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)
D_Va_SE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]

[Div_Va_SE_8_Cuad,          Desv_Div_Va_SE_8_Cuad]=
pot(D_Va_SE8,2)
Division_VaSE8            =          [Div_Va_SE_8_Cuad,
Desv_Div_Va_SE_8_Cuad]

[Div_Vm_Sut_8,
Desv_Div_Vm_Sut_8]=div(V_misses_med_LN8, Sut)
D_Vm_Sut8 = [Div_Vm_Sut_8, Desv_Div_Vm_Sut_8]

[Div_Vm_Sut_8_Cuad,          Desv_Div_Vm_Sut_8_Cuad]=
pot(D_Vm_Sut8,2)
Division_VmSut8            =          [Div_Vm_Sut_8_Cuad,
Desv_Div_Vm_Sut_8_Cuad]

[Suma8_VmSut,Desv_Suma8_VmSut]
=suma(Division_VaSE8, Division_VmSut8)

ASME_8=[Suma8_VmSut,Desv_Suma8_VmSut]
Cf_ASME_8=ASME_8(2)/ASME_8(1)
CRITERIO_F=ASME_8
CF_CRITERIO_F=Cf_ASME_8
end

if criterio_falla == 2 then
[Div_Va_SE_8,
Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)
Division_VaSE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]

[Div_Vm_Sut_8,
Desv_Div_Vm_Sut_8]=div(V_misses_med_LN8, Sut)
D_Vm_Sut8 = [Div_Vm_Sut_8, Desv_Div_Vm_Sut_8]

[Div_Vm_Sut_8_Cuad,          Desv_Div_Vm_Sut_8_Cuad]=
pot(D_Vm_Sut8,2)
Division_VmSut8            =          [Div_Vm_Sut_8_Cuad,
Desv_Div_Vm_Sut_8_Cuad]

[Suma8_VmSut,Desv_Suma8_VmSut]
=suma(Division_VaSE8, Division_VmSut8)

GERBER_8=[Suma8_VmSut,Desv_Suma8_VmSut]
Cf_GERBER_8=GERBER_8(2)/GERBER_8(1)
CRITERIO_F=GERBER_8
CF_CRITERIO_F=Cf_GERBER_8
end

if criterio_falla == 3 then
[Div_Va_SE_8,
Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)
Division_VaSE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]
[Div_Vm_Sut_8,
Desv_Div_Vm_Sut_8]=div(V_misses_med_LN8, Sut)
Division_VmSut8 = [Div_Vm_Sut_8, Desv_Div_Vm_Sut_8]

[Suma8_VmSut,Desv_Suma8_VmSut]
=suma(Division_VaSE8, Division_VmSut8)
GOODMAN_8=[Suma8_VmSut,Desv_Suma8_VmSut]
Cf_GOODMAN_8=GOODMAN_8(2)/GOODMAN_8(1)
CRITERIO_F=GOODMAN_8
CF_CRITERIO_F=Cf_GOODMAN_8
end

MEDCnormal_8=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_8=CF_CRITERIO_F

[Prob_8,Conf_8]=cdfnor("PQ",0,abs(MEDCnormal_8),DESV
normal_8)

DIAMETRO_EJE8=0

if Prob_8<=1.5*10^-6 & Prob_8>=0.5*10^-6 then
    DIAMETRO_EJE8=Diametro_eje8
    break;
end
end
disp("El diametro del tramo 8 es")
disp(DIAMETRO_EJE8)

for Diametro_eje9=tol:tol:6

Radio_eje9=Diametro_eje9/2
I9=(%pi*(((Radio_eje9*2)^4))/4)//inerxia
Q9=(2/3)*(Radio_eje9^3)
J9=I9*2
// calculo de esfuerzos
sigma9=(Momento_max9*Radio_eje9/I9) // esfuerzo flexor
maximo
Torque9=(W_tangencial4/1000)*((Radio_engrane4)-
Radio_eje9)
taotormed9=(Torque9*Radio_eje9/J9)
//Esfuerzo cortante torsor medio:
taotoralt9=(Torque9*Radio_eje9/J9)

//FACTOR DE TAMAÑO

if (Diametro_eje9 < 0.11)
    f_tama9=1
end
if (Diametro_eje9 >= 0.11 & Diametro_eje9 <= 2)
    f_tama9=0.879*Diametro_eje9^-0.107
end
if (Diametro_eje9 >= 3 & Diametro_eje9 <= 10)
    f_tama9=0.91*(Diametro_eje9^(-0.157))
end

//factor de concentracion de esfuerzo
h9=Radio_eje9-Radio_eje8
relacion9=h9/Radio_muesca8

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C9_f1=0.947+1.206*sqrt(h9/Radio_muesca8)-
0.131*(h9/Radio_muesca8)
C9_f2=0.022-
3.405*sqrt(h9/Radio_muesca8)+0.915*(h9/Radio_muesca8)
C9_f3=0.869+1.777*sqrt(h9/Radio_muesca8)-
0.555*(h9/Radio_muesca8)
C9_f4=-0.810+0.422*sqrt(h9/Radio_muesca8)-
0.260*(h9/Radio_muesca8)

//2.0<=h/r<=20
C9_f11=1.232+0.832*sqrt(h9/Radio_muesca8)-
0.008*(h9/Radio_muesca8)
C9_f22=-3.813+0.968*sqrt(h9/Radio_muesca8)-
0.260*(h9/Radio_muesca8)

```

```

C9_f33=7.423-
4.868*sqrt(h9/Radio_muesca8)+0.869*(h9/Radio_muesca8)
C9_f44=-3.839+3.070*sqrt(h9/Radio_muesca8)-
0.600*(h9/Radio_muesca8)

```

```

if (relacion9 >=0.1 & relacion9 <= 2)
  C9_F1=C9_f1
  C9_F2=C9_f2
  C9_F3=C9_f3
  C9_F4=C9_f4
  Kt9_f=C9_F1+C9_F2*(2*h9/Diametro_eje9)+C9_F3*(2*h9/
  Diametro_eje9)^2+C9_F4*(2*h9/Diametro_eje9)^3
end

```

```

if (relacion9 > 2 & relacion9 <= 20)
  C9_F1=C9_f11
  C9_F2=C9_f22
  C9_F3=C9_f33
  C9_F4=C9_f44
  Kt9_f=C9_F1+C9_F2*(2*h9/Diametro_eje9)+C9_F3*(2*h9/
  Diametro_eje9)^2+C9_F4*(2*h9/Diametro_eje9)^3
end

```

```

if relacion9 < 0.1 then
  Kt9_f=1
end
if relacion9 > 0.1 then
  Kt9_f=1
end
KT9_F=Kt9_f

```

//Kt para esfuerzos de torsión (Ktf)

```

//0.25<=h/r<=4.0
C9_t1=0.905+0.783*sqrt(h9/Radio_muesca8)-
0.075*(h9/Radio_muesca8)
C9_t2=-0.437-
1.969*sqrt(h9/Radio_muesca8)+0.553*(h9/Radio_muesca8)
C9_t3=1.557+1.073*sqrt(h9/Radio_muesca8)-
0.578*(h9/Radio_muesca8)
C9_t4=-1.061+0.171*sqrt(h9/Radio_muesca8)-
0.086*(h9/Radio_muesca8)

```

```

if (relacion9 >=0.25 & relacion9 <= 4.0)
  C9_T1=C9_t1
  C9_T2=C9_t2
  C9_T3=C9_t3
  C9_T4=C9_t4
  Kt9_t=C9_T1+C9_T2*(2*h9/Diametro_eje9)+C9_T3*(2*h9/
  Diametro_eje9)^2+C9_T4*(2*h9/Diametro_eje9)^3
end

```

```

if relacion9 < 0.25 then
  Kt9_t=1
end
if relacion9 > 4.0 then
  Kt9_t=1
end
KT9_T=Kt9_t

```

//constantes para carga axial

```

//0.1<=h/r<=2.0
C9_a1= 0.926+(1.157*sqrt(h9/Radio_muesca8))-
(0.099*(h9/Radio_muesca8))
C9_a2=0.012-
3.036*sqrt(h9/Radio_muesca8)+0.961*(h9/Radio_muesca8)
C9_a3=-0.302+3.977*sqrt(h9/Radio_muesca8)-
1.744*(h9/Radio_muesca8)

```

```

C9_a4=0.365-
2.098*sqrt(h9/Radio_muesca8)+0.878*(h9/Radio_muesca8)

```

//2.0<=h/r<=20

```

C9_a11=1.2+0.860*sqrt(h9/Radio_muesca8)-
0.022*(h9/Radio_muesca8)
C9_a22=-1.805-0.346*sqrt(h9/Radio_muesca8)-
0.038*(h9/Radio_muesca8)
C9_a33=2.198-
0.486*sqrt(h9/Radio_muesca8)+0.165*(h9/Radio_muesca8)
C9_a44=-0.593-0.028*sqrt(h9/Radio_muesca8)-
0.106*(h9/Radio_muesca8)

```

```

if (relacion9 >=0.1 & relacion9 <= 2)

```

```

  C9_A1=C9_a1
  C9_A2=C9_a2
  C9_A3=C9_a3
  C9_A4=C9_a4
  Kt9_a=C9_A1+C9_A2*(2*h9/Diametro_eje9)+C9_A3*(2*h9/
  Diametro_eje9)^2+C9_A4*(2*h9/Diametro_eje9)^3
end

```

```

if (relacion9 > 2 & relacion9 <= 20)

```

```

  C9_A1=C9_a11
  C9_A2=C9_a22
  C9_A3=C9_a33
  C9_A4=C9_a44
  Kt9_a=C9_A1+C9_A2*(2*h9/Diametro_eje9)+C9_A3*(2*h9/
  Diametro_eje9)^2+C9_A4*(2*h9/Diametro_eje9)^3
end

```

```

if relacion9 < 0.1 then

```

```

  Kt9_a=1
end

```

```

if relacion9 > 20 then

```

```

  Kt9_a=1
end

```

```

KT9_A=Kt9_a

```

//Factor de concentración de esfuerzo de fatiga (Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

```

constante_reemplazo_f9=(2*(KT9_F-
1)/KT9_F)*(RAIZ_DE_A/sqrt(Radio_muesca8))
Kf9_f=KT9_F/((1+constante_reemplazo_f9))
constante_reemplazo_t9=(2*(KT9_T-
1)/KT9_T)*(RAIZ_DE_A/sqrt(Radio_muesca8))
Kf9_t=KT9_T/((1+constante_reemplazo_t9))
constante_reemplazo_a9=(2*(KT9_A-
1)/KT9_A)*(RAIZ_DE_A/sqrt(Radio_muesca8))
Kf9_a=KT9_A/((1+constante_reemplazo_a9))

```

//CÁLCULO DE LA PRESIÓN POR INTERFERENCIA

```

m4=(((2*Radio engrane4)^2-
Diametro_eje9^2))*(Diametro_eje9^2))
n4=(2*Radio engrane4)
PP9=((E*Interferen_maza_eje_max4)/(2*Diametro_eje9^3))*
(m4/n4)

```

```

Am9=[(PP9*f_carga_axial*Kf9_a/f_tama9)
(taotormed9*f_carga_torsion*Kf9_t/f_tama9)
0;(taotormed9*f_carga_torsion*Kf9_t/f_tama9) 0 0;0 0 0]
//tensor de esfuerzo medio//
[Em9]=spec(Am9)
sigvmm9=(((Em9(1)-Em9(2))^2)+((Em9(2)-
Em9(3))^2)+((Em9(3)-Em9(1))^2))/2^0.5

```

```

Aa9=[(sigma9*f_carga_flexion*Kf9_f/f_tama9)
(taotoralt9*f_carga_torsion*Kf9_t/f_tama9) 0;0
(taotoralt9*f_carga_torsion*Kf9_t/f_tama9) 0;0 0 0] //Tensor
de esfuerzos alternantes
[Ea9]=spec(Aa9)
sigvma9=(((Ea9(1)-Ea9(2))^2)+((Ea9(2)-
Ea9(3))^2)+((Ea9(3)-Ea9(1))^2)/2)^0.5

V_misses_med_LN9=[sigvmm9,
Coef_variacion_carga*sigvmm9]
V_misses_alt_LN9=[sigvma9,
Coef_variacion_carga*sigvma9]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_9,
Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
D_Va_SE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]

[Div_Va_SE_9_Cuad,          Desv_Div_Va_SE_9_Cuad]=
pot(D_Va_SE9,2)
Division_VaSE9           =          [Div_Va_SE_9_Cuad,
Desv_Div_Va_SE_9_Cuad]

[Div_Vm_Sut_9,
Desv_Div_Vm_Sut_9]=div(V_misses_med_LN9, Sut)
D_Vm_Sut9 = [Div_Vm_Sut_9, Desv_Div_Vm_Sut_9]

[Div_Vm_Sut_9_Cuad,          Desv_Div_Vm_Sut_9_Cuad]=
pot(D_Vm_Sut9,2)
Division_VmSut9           =          [Div_Vm_Sut_9_Cuad,
Desv_Div_Vm_Sut_9_Cuad]

[Suma9_VmSut,Desv_Suma9_VmSut]
=suma(Division_VaSE9, Division_VmSut9)

ASME_9=[Suma9_VmSut,Desv_Suma9_VmSut]
Cf_ASME_9=ASME_9(2)/ASME_9(1)
CRITERIO_F=ASME_9
CF_CRITERIO_F=Cf_ASME_9
end

if criterio_falla == 2 then
[Div_Va_SE_9,
Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
Division_VaSE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]

[Div_Vm_Sut_9,
Desv_Div_Vm_Sut_9]=div(V_misses_med_LN9, Sut)
D_Vm_Sut9 = [Div_Vm_Sut_9, Desv_Div_Vm_Sut_9]

[Div_Vm_Sut_9_Cuad,          Desv_Div_Vm_Sut_9_Cuad]=
pot(D_Vm_Sut9,2)
Division_VmSut9           =          [Div_Vm_Sut_9_Cuad,
Desv_Div_Vm_Sut_9_Cuad]

[Suma9_VmSut,Desv_Suma9_VmSut]
=suma(Division_VaSE9, Division_VmSut9)

GERBER_9=[Suma9_VmSut,Desv_Suma9_VmSut]
Cf_GERBER_9=GERBER_9(2)/GERBER_9(1)
CRITERIO_F=GERBER_9
CF_CRITERIO_F=Cf_GERBER_9

```

```

end

if criterio_falla == 3 then
[Div_Va_SE_9,
Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
Division_VaSE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]
[Div_Vm_Sut_9,
Desv_Div_Vm_Sut_9]=div(V_misses_med_LN9, Sut)
Division_VmSut9 = [Div_Vm_Sut_9, Desv_Div_Vm_Sut_9]

[Suma9_VmSut,Desv_Suma9_VmSut]
=suma(Division_VaSE9, Division_VmSut9)
GOODMAN_9=[Suma9_VmSut,Desv_Suma9_VmSut]
Cf_GOODMAN_9=GOODMAN_9(2)/GOODMAN_9(1)
CRITERIO_F=GOODMAN_9
CF_CRITERIO_F=Cf_GOODMAN_9
end

MEDCnormal_9=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_9=CF_CRITERIO_F

[Prob_9,Conf_9]=cdfnor("PQ",0,abs(MEDCnormal_9),DESV
normal_9)

DIAMETRO_EJE9=0

if Prob_9<=1.5*10^-6 & Prob_9>=0.5*10^-6 then
    DIAMETRO_EJE9=Diametro_eje9
    break;
end
end
disp("El diametro del tramo 9 es")
disp(DIAMETRO_EJE9)

//ITERACIÓN DEL DÉCIMO DIÁMETRO:
for Diametro_eje10=tol:tol:6

Radio_eje10=Diametro_eje10/2
//MOMENTO DE INERCIA:
I10=(%pi*(((Radio_eje10*2)^4)/4)
//MOMENTO DE ÁREA:
Q10=(2/3)*(Radio_eje10^3)
//MOMENTO POLAR DE INERCIA:
J10=I10*2
//CÁLCULO DE ESFUERZOS:
//Esfuerzo flector máximo:
sigma10=(Momento_max10*Radio_eje10/I10)

//FACTOR DE TAMAÑO
if (Diametro_eje10 < 0.11)
    f_tama10=1
end
if (Diametro_eje10 >= 0.11 & Diametro_eje10 <= 2)
    f_tama10=0.879*Diametro_eje10^-0.107
end
if (Diametro_eje10 >= 3 & Diametro_eje10 <= 10)
    f_tama10=0.91*(Diametro_eje10^(-0.157))
end

//Factor de concentración de esfuerzo
h10=Radio_eje10-Radio_eje9
relacion10=h10/Radio_muesca10

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON:

```


//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON PARA FLEXIÓN:

//0.1 <= h/r <= 2.0

C7_f1=0.947+1.206*sqrt(h10/Radio_muesca9)-
0.131*(h10/Radio_muesca9)

C7_f2=0.022-

3.405*sqrt(h10/Radio_muesca9)+0.915*(h10/Radio_muesca9)

C7_f3=0.869+1.777*sqrt(h10/Radio_muesca9)-
0.555*(h10/Radio_muesca9)

C7_f4=-0.810+0.422*sqrt(h10/Radio_muesca9)-
0.260*(h10/Radio_muesca9)

//2.0 <= h/r <= 20

C7_f11=1.232+0.832*sqrt(h10/Radio_muesca9)-
0.008*(h10/Radio_muesca9)

C7_f22=-3.813+0.968*sqrt(h10/Radio_muesca9)-
0.260*(h10/Radio_muesca9)

C7_f33=7.423-

4.868*sqrt(h10/Radio_muesca9)+0.869*(h10/Radio_muesca9)

C7_f44=-3.839+3.070*sqrt(h10/Radio_muesca9)-
0.600*(h10/Radio_muesca9)

if (relacion10 >= 0.1 & relacion10 <= 2)

C7_F1=C7_f1

C7_F2=C7_f2

C7_F3=C7_f3

C7_F4=C7_f4

Kt10_f=C7_F1+C7_F2*(2*h10/Diametro_eje10)+C7_F3*(2*
h10/Diametro_eje10)^2+C7_F4*(2*h10/Diametro_eje10)^3
end

if (relacion10 > 2 & relacion10 <= 20)

C7_F1=C7_f11

C7_F2=C7_f22

C7_F3=C7_f33

C7_F4=C7_f44

Kt10_f=C7_F1+C7_F2*(2*h10/Diametro_eje10)+C7_F3*(2*
h10/Diametro_eje10)^2+C7_F4*(2*h10/Diametro_eje10)^3
end

if relacion10 < 0.1 then

Kt10_f=1

end

if relacion10 > 20 then

Kt10_f=1

end

KT10_F=Kt10_f

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f10=(2*(KT10_F-
1)/KT10_F)*(RAIZ_DE_A/sqrt(Radio_muesca10))
Kf10_f=KT10_F/((1+constante_reemplazo_f10))

//TENSOR DE ESFUERZOS MEDIOS:

Am10=[0 0 0; 0 0 0; 0 0 0]

//CÁLCULO DE AUTOVALORES:

[Em10]=spec(Am10)

//CRITERIO DE VON-MISSES:

sigvmm10=(((Em10(1)-Em10(2))^2)+((Em10(2)-
Em10(3))^2)+((Em10(3)-Em10(1))^2))/2)^0.5

//TENSOR DE ESFUERZOS ALTERNANTES:

Aa10=[(sigma10*f_carga_flexion*Kf10_f/f_tama10) 0 0; 0 0 0; 0
0 0]

//CÁLCULO DE AUTOVALORES:

[Ea10]=spec(Aa10)

//CRITERIO DE VON-MISSES:

sigvma10=(((Ea10(1)-Ea10(2))^2)+((Ea10(2)-
Ea10(3))^2)+((Ea10(3)-Ea10(1))^2))/2)^0.5

V_misses_med_LN10=[sigvmm10,

Coef_variacion_carga*sigvmm10]

V_misses_alt_LN10=[sigvma10,

Coef_variacion_carga*sigvma10]

//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:

if criterio_falla == 1 then

[Div_Va_SE_10,

Desv_Div_Va_SE_10]=div(V_misses_alt_LN10, SE)

D_Va_SE10 = [Div_Va_SE_10, Desv_Div_Va_SE_10]

[Div_Va_SE_10_Cuad, Desv_Div_Va_SE_10_Cuad]=
pot(D_Va_SE10,2)

Division_VaSE10 = [Div_Va_SE_10_Cuad,
Desv_Div_Va_SE_10_Cuad]

Division_VmSut10 = [0,0]

Suma10_VmSut_VaSE=Division_VaSE10(1)+Division_VmS
ut10(1)

Desv_Suma10_VmSut_VaSE=sqrt((Division_VaSE10(2)/Divi
sion_VaSE10(1))^2+(0)^2)

ASME_10=[Suma10_VmSut_VaSE,

Desv_Suma10_VmSut_VaSE]

Cf_ASME_10=ASME_10(2)/ASME_10(1)

CRITERIO_F=ASME_10

CF_CRITERIO_F=Cf_ASME_10

end

if criterio_falla == 2 then

[Div_Va_SE_10,

Desv_Div_Va_SE_10]=div(V_misses_alt_LN10, SE)

Division_VaSE10 = [Div_Va_SE_10, Desv_Div_Va_SE_10]

Division_VmSut10 = [0,0]

Suma10_VmSut_VaSE=Division_VaSE10(1)+Division_VmS
ut10(1)

Desv_Suma10_VmSut_VaSE=sqrt((Division_VaSE10(2)/Divi
sion_VaSE10(1))^2+(0)^2)

GERBER_10=[Suma10_VmSut_VaSE,

Desv_Suma10_VmSut_VaSE]

Cf_GERBER_10=GERBER_10(2)/GERBER_10(1)

CRITERIO_F=GERBER_10

CF_CRITERIO_F=Cf_GERBER_10

end

if criterio_falla == 3 then

[Div_Va_SE_10,

Desv_Div_Va_SE_10]=div(V_misses_alt_LN10, SE)

Division_Va_SE10 = [Div_Va_SE_10, Desv_Div_Va_SE_10]

Division_VmSut10 = [0,0]

Suma10_VmSut_VaSE=Division_Va_SE10(1)+Division_VmS
ut10(1)

```

Desv_Suma10_VmSut_VaSE=sqrt((Division_Va_SE10(2)/Division_Va_SE10(1))^2+(0)^2)
GOODMAN_10=[Suma10_VmSut_VaSE,
Desv_Suma10_VmSut_VaSE]
Cf_GOODMAN_10=GOODMAN_10(2)/GOODMAN_10(1)
CRITERIO_F=GOODMAN_10
CF_CRITERIO_F=Cf_GOODMAN_10
end

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A LA DISTRIBUCIÓN NORMAL
MEDCnormal_10=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_10=CF_CRITERIO_F

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_10,Conf_10]=cdfnorf("PQ",0,abs(MEDCnormal_10),DESVnormal_10)

DIAMETRO_EJE10=0

if Prob_10<=1.5*10^-6 & Prob_10>=0.5*10^-6 then
    DIAMETRO_EJE10=Diametro_eje10
    break;
end
end
disp("El diametro del tramo 10 es")
disp(DIAMETRO_EJE10)

//ITERACIÓN DEL UNDÉCIMO DIÁMETRO:
for Diametro_eje11=tol:tol:6

Radio_eje11=Diametro_eje11/2
//MOMENTO DE INERCIA:
I11=(%pi*(((Radio_eje11*2)^4))/4)
//MOMENTO DE ÁREA:
Q11=(2/3)*(Radio_eje11^3)
//MOMENTO POLAR DE INERCIA:
J11=I11*2
//CÁLCULO DE ESFUERZOS:
//Esfuerzo flector máximo:
sigal1=(Momento_max11*Radio_eje11/I11)

//FACTOR DE TAMAÑO
if (Diametro_eje11 < 0.11)
    f_tama11=1
end
if (Diametro_eje11 >= 0.11 & Diametro_eje11 <= 2)
    f_tama11=0.879*Diametro_eje11^-0.107
end
if (Diametro_eje11 >= 3 & Diametro_eje11 <= 10)
    f_tama11=0.91*(Diametro_eje11^(-0.157))
end

//Factor de concentración de esfuerzo
h11=Radio_eje11-Radio_eje10
relacion11=h11/Radio_muesca10

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE PETERSON:
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE PETERSON PARA FLEXIÓN:
//0.1<=h/r<=2.0
C7_f1=0.947+1.206*sqrt(h11/Radio_muesca10)-
0.131*(h11/Radio_muesca10)

C7_f2=0.022-
3.405*sqrt(h11/Radio_muesca10)+0.915*(h11/Radio_muesca10)
C7_f3=0.869+1.777*sqrt(h11/Radio_muesca10)-
0.555*(h11/Radio_muesca10)
C7_f4=-0.810+0.422*sqrt(h11/Radio_muesca10)-
0.260*(h11/Radio_muesca10)

//2.0<=h/r<=20
C7_f11=1.232+0.832*sqrt(h11/Radio_muesca10)-
0.008*(h11/Radio_muesca10)
C7_f22=-3.813+0.968*sqrt(h11/Radio_muesca10)-
0.260*(h11/Radio_muesca10)
C7_f33=7.423-
4.868*sqrt(h11/Radio_muesca10)+0.869*(h11/Radio_muesca10)
C7_f44=-3.839+3.070*sqrt(h11/Radio_muesca10)-
0.600*(h11/Radio_muesca10)

if (relacion11 >=0.1 & relacion11 <= 2)
    C7_F1=C7_f1
    C7_F2=C7_f2
    C7_F3=C7_f3
    C7_F4=C7_f4
    Kt11_f=C7_F1+C7_F2*(2*h11/Diametro_eje11)+C7_F3*(2*
h11/Diametro_eje11)^2+C7_F4*(2*h11/Diametro_eje11)^3
end

if (relacion11 > 2 & relacion11 <= 20)
    C7_F1=C7_f11
    C7_F2=C7_f22
    C7_F3=C7_f33
    C7_F4=C7_f44
    Kt11_f=C7_F1+C7_F2*(2*h11/Diametro_eje11)+C7_F3*(2*
h11/Diametro_eje11)^2+C7_F4*(2*h11/Diametro_eje11)^3
end

if relacion11 < 0.1 then
    Kt11_f=1
end

if relacion11 > 20 then
    Kt11_f=1
end

KT11_F=Kt11_f

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f11=(2*(KT11_F-
1)/KT11_F)*(RAIZ_DE_A/sqrt(Radio_muesca10))
Kf11_f=KT11_F/((1+constante_reemplazo_f11))

//TENSOR DE ESFUERZOS MEDIOS:
Am11=[0 0 0;0 0 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Em11]=spec(Am11)
//CRITERIO DE VON-MISES:
sigvmm11=(((Em11(1)-Em11(2))^2)+((Em11(2)-
Em11(3))^2)+((Em11(3)-Em11(1))^2))/2^0.5

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa11=[(sigal1*f_carga_flexion*Kf11_f/f_tama11) 0 0;0 0 0;0
0 0]
//CÁLCULO DE AUTOVALORES:
[Ea11]=spec(Aa11)

```



```

//CRITERIO DE VON-MISSES:
sigvma11=(((Ea11(1)-Ea11(2))^2)+((Ea11(2)-
Ea11(3))^2)+((Ea11(3)-Ea11(1))^2)/2)^0.5

V_misses_med_LN11=[sigvmm11,
Coef_variacion_carga*sigvmm11]
V_misses_alt_LN11=[sigvma11,
Coef_variacion_carga*sigvma11]

//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:
if criterio_falla == 1 then
[Div_Va_SE_11,
Desv_Div_Va_SE_11]=div(V_misses_alt_LN11, SE)
D_Va_SE11 = [Div_Va_SE_11, Desv_Div_Va_SE_11]

[Div_Va_SE_11_Cuad,      Desv_Div_Va_SE_11_Cuad]=
pot(D_Va_SE11,2)
Division_VaSE11      =      [Div_Va_SE_11_Cuad,
Desv_Div_Va_SE_11_Cuad]

Division_VmSut11 = [0,0]

Suma11_VmSut_VaSE=Division_VaSE11(1)+Division_VmS
ut11(1)
Desv_Suma11_VmSut_VaSE=sqrt((Division_VaSE11(2)/Divi
sion_VaSE11(1))^2+(0)^2)

ASME_11=[Suma11_VmSut_VaSE,
Desv_Suma11_VmSut_VaSE]
Cf_ASME_11=ASME_11(2)/ASME_11(1)
CRITERIO_F=ASME_11
CF_CRITERIO_F=Cf_ASME_11
end

if criterio_falla == 2 then
[Div_Va_SE_11,
Desv_Div_Va_SE_11]=div(V_misses_alt_LN11, SE)
Division_VaSE11 = [Div_Va_SE_11, Desv_Div_Va_SE_11]

Division_VmSut11 = [0,0]

Suma11_VmSut_VaSE=Division_VaSE11(1)+Division_VmS
ut11(1)
Desv_Suma11_VmSut_VaSE=sqrt((Division_VaSE11(2)/Divi
sion_VaSE11(1))^2+(0)^2)

GERBER_11=[Suma11_VmSut_VaSE,
Desv_Suma11_VmSut_VaSE]
Cf_GERBER_11=GERBER_11(2)/GERBER_11(1)
CRITERIO_F=GERBER_11
CF_CRITERIO_F=Cf_GERBER_11
end

if criterio_falla == 3 then
[Div_Va_SE_11,
Desv_Div_Va_SE_11]=div(V_misses_alt_LN11, SE)
Division_Va_SE11 = [Div_Va_SE_11, Desv_Div_Va_SE_11]

Division_VmSut11 = [0,0]

Suma11_VmSut_VaSE=Division_Va_SE11(1)+Division_Vm
Sut11(1)
Desv_Suma11_VmSut_VaSE=sqrt((Division_Va_SE11(2)/Di
vision_Va_SE11(1))^2+(0)^2)
GOODMAN_11=[Suma11_VmSut_VaSE,
Desv_Suma11_VmSut_VaSE]
Cf_GOODMAN_11=GOODMAN_11(2)/GOODMAN_11(1)

```

```

CRITERIO_F=GOODMAN_11
CF_CRITERIO_F=Cf_GOODMAN_11
end

```

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A LA DISTRIBUCIÓN NORMAL

```

MEDCnormal_11=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_11=CF_CRITERIO_F

```

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:

```

[Prob_11,Conf_11]=cdfnor("PQ",0,abs(MEDCnormal_11),DE
SVnormal_11)

```

```
DIAMETRO_EJE11=0
```

```
if Prob_11<=1.5*10^-6 & Prob_11>=0.5*10^-6 then
```

```
DIAMETRO_EJE11=Diametro_eje11
```

```
break;
```

```
end
```

```
end
```

```
disp("El diametro del tramo 11 es")
```

```
disp(DIAMETRO_EJE11)
```

```
Sut9=0.9*Sut(1)
```

```
m1=(Sut9-SE(1))
```

```
m22=(3-6)
```

```
pend=m1/m22 // calculo de la pendiente
```

```
x=((sigvma11)-Sut9+(pend*3))/pend //valor de x para el
esfuerzo alterante
```

```
CiTo=10^x //REVISAR VARIABLE "x".
```

```
ViRe=CiTo
```

```
disp("la vida del eje es :")
```

```
disp(ViRe)
```

13. 7_TorquesMomentosYGraficas.sci

```
clc
```

```
clear all
```

```
W_tangencial11=W_tangencial1/1000
```

```
W_radial11=W_radial1/1000
```

```
W_tangencial22=W_tangencial2/1000
```

```
W_radial22=W_radial2/1000
```

```
W_tangencial33=W_tangencial3/1000
```

```
W_radial33=W_radial3/1000
```

```
W_tangencial44=W_tangencial4/1000
```

```
W_radial44=W_radial4/1000
```

```
if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2) then
```

```
Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial22*lon
gitud_eje)+(W_radial33*longitud_eje)+(W_radial44*longitud
_eje)-(W_radial11*distancia_1)-(W_radial22*distancia_2)-
(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
```

```
Momento_1xy=Reaccion_Axy*distancia_1
```

```
Momento_2xy=(Reaccion_Axy*distancia_2)-
```

```
(W_radial11*distancia_2)+(W_radial11*distancia_1)
```

```
Momento_3xy=(Reaccion_Axy*distancia_3)-
```

```
(W_radial11*distancia_3)-
```

```
(W_radial22*distancia_3)+(W_radial11*distancia_1)+(W_radial22*distancia_2)
```

```
Momento_4xy=(Reaccion_Axy*distancia_4)-  
(W_radial11*distancia_4)-(W_radial22*distancia_4)-  
(W_radial33*distancia_4)+(W_radial22*distancia_2)+(W_radial11*distancia_1)+(W_radial33*distancia_3)
```

```
pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)  
pendiente_2xy=(Momento_2xy-  
Momento_1xy)/(distancia_2-distancia_1)  
pendiente_3xy=(Momento_3xy-  
Momento_2xy)/(distancia_3-distancia_2)  
pendiente_4xy=(Momento_4xy-  
Momento_3xy)/(distancia_4-distancia_3)  
pendiente_5xy=(0-Momento_4xy)/(longitud_eje-  
distancia_4)
```

```
Momento_max1=0  
Momento_max2=0  
Momento_parte1_max3=0  
Momento_parte2_max3=0  
Momento_max4=0  
Momento_parte1_max5=0  
Momento_parte2_max5=0  
Momento_max6=0  
Momento_parte1_max7=0  
Momento_parte2_max7=0  
Momento_max8=0  
Momento_parte1_max9=0  
Momento_parte2_max9=0  
Momento_max10=0  
Momento_max11=0
```

```
for x1=0:0.1:sesion_1  
Momento_1XY=pendiente_1xy*(x1-0)+0  
if Momento_1XY>Momento_max1 then  
Momento_max1=Momento_1XY  
Xmax_1=x1  
end  
end  
for x2=sesion_1:0.1:sesion_2  
Momento_2XY=pendiente_1xy*(x2-0)+0  
if Momento_2XY>Momento_max2 then  
Momento_max2=Momento_2XY  
Xmax_2=x2  
end  
end  
for parte1_x3=sesion_2:0.1:distancia_1  
Momento_parte1_3XY=pendiente_1xy*(parte1_x3-0)+0  
if Momento_parte1_3XY>Momento_parte1_max3 then  
Momento_parte1_max3=Momento_parte1_3XY  
Xmax_parte1_3=parte1_x3  
end  
end  
for parte2_x3=distancia_1:0.1:sesion_3  
Momento_parte2_3XY=pendiente_2xy*(parte2_x3-  
distancia_1)+Momento_1xy  
if Momento_parte2_3XY>Momento_parte2_max3 then  
Momento_parte2_max3=Momento_parte2_3XY  
Xmax_parte2_3=parte2_x3  
end  
end  
if Momento_parte2_max3 > Momento_parte1_max3 then  
Momento_max3=Momento_parte2_max3  
Xmax_3= Xmax_parte2_3  
end  
if Momento_parte1_max3 > Momento_parte2_max3 then
```

```
Momento_max3=Momento_parte1_max3  
Xmax_3=Xmax_parte1_3  
end  
if Momento_parte1_max3 == Momento_parte2_max3 then  
Momento_max3=Momento_parte1_max3  
Xmax_3=Xmax_parte1_3  
end  
for x4=sesion_3:0.1:sesion_4  
Momento_4XY=pendiente_2xy*(x4-  
distancia_1)+Momento_1xy  
if Momento_4XY>Momento_max4 then  
Momento_max4=Momento_4XY  
Xmax_4=x4  
end  
end  
for parte1_x5=sesion_4:0.1:distancia_2  
Momento_parte1_5XY=pendiente_2xy*(parte1_x5-  
distancia_1)+Momento_1xy  
if Momento_parte1_5XY>Momento_parte1_max5 then  
Momento_parte1_max5=Momento_parte1_5XY  
Xmax_parte1_5=parte1_x5  
end  
end  
for parte2_x5=distancia_2:0.1:sesion_5  
Momento_parte2_5XY=pendiente_3xy*(parte2_x5-  
distancia_2)+Momento_2xy  
if Momento_parte2_5XY>Momento_parte2_max5 then  
Momento_parte2_max5=Momento_parte2_5XY  
Xmax_parte2_5=parte2_x5  
end  
end  
if Momento_parte2_max5 > Momento_parte1_max5  
then  
Momento_max5=Momento_parte2_max5  
Xmax_5= Xmax_parte2_5  
end  
if Momento_parte1_max5 > Momento_parte2_max5 then  
Momento_max5=Momento_parte1_max5  
Xmax_5=Xmax_parte1_5  
end  
end  
if Momento_parte1_max5 == Momento_parte2_max5 then  
Momento_max5=Momento_parte1_max5  
Xmax_5=Xmax_parte1_5  
end  
end  
for x6=sesion_5:0.1:sesion_6  
Momento_6XY=pendiente_3xy*(x6-  
distancia_2)+Momento_2xy  
if Momento_6XY>Momento_max6 then  
Momento_max6=Momento_6XY  
Xmax_6=x6  
end  
end  
for parte1_x7=sesion_6:0.1:distancia_3  
Momento_parte1_7XY=pendiente_3xy*(parte1_x7-  
distancia_2)+Momento_2xy  
if Momento_parte1_7XY>Momento_parte1_max7 then  
Momento_parte1_max7=Momento_parte1_7XY  
Xmax_parte1_7=parte1_x7  
end  
end  
for parte2_x7=distancia_3:0.1:sesion_7  
Momento_parte2_7XY=pendiente_4xy*(parte2_x7-  
distancia_3)+Momento_3xy  
if Momento_parte2_7XY>Momento_parte2_max7 then  
Momento_parte2_max7=Momento_parte2_7XY  
Xmax_parte2_7=parte2_x7
```

```

end
end
if Momento_parte2_max7 > Momento_parte1_max7
then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=pendiente_4xy*(x8-
distancia_3)+Momento_3xy
if Momento_8XY>Momento_max8 then
Momento_max8=Momento_8XY
Xmax_8=x8
end
end

for parte1_x9=sesion_8:0.1:sesion_9
Momento_parte1_9XY=pendiente_4xy*(parte1_x9-
distancia_3)+Momento_3xy
if Momento_parte1_9XY>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9XY
Xmax_parte1_9=parte1_x9
end
end

for parte2_x9=sesion_8:0.1:sesion_9
Momento_parte2_9XY=pendiente_5xy*(parte2_x9-
longitud_eje)+0
if Momento_parte2_9XY>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9XY
Xmax_parte2_9=parte2_x9
end
end

if Momento_parte2_max9 > Momento_parte1_max9
then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end

for x10=sesion_9:0.1:sesion_10
Momento_10XY=pendiente_5xy*(x10-longitud_eje)+0
if Momento_10XY>Momento_max10 then
Momento_max10=Momento_10XY
Xmax_10=x10
end
end
for x11=sesion_10:0.1:longitud_eje
Momento_11XY=pendiente_5xy*(x11-longitud_eje)+0
if Momento_11XY>Momento_max11 then
Momento_max11=Momento_11XY
Xmax_11=x11
end
end

```

```

end
end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 == ubicacion_3
& ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2) then

Reaccion_Axy=((W_radial22*longitud_eje)+(W_radial33*lon
gitud_eje)+(W_radial44*longitud_eje)-
(W_radial22*distancia_2)-(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_2
Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial22*distancia_3)+(W_radial22*distancia_2)
Momento_3xy=(Reaccion_Axy*distancia_4)-
(W_radial22*distancia_4)-
(W_radial33*distancia_4)+(W_radial22*distancia_2)+(W_radi
al33*distancia_3)
Reaccion_Axz=W_radial11-
((W_radial11*distancia_1)/longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_1
pendiente_1xy=(Momento_1xy-0)/(distancia_2-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_2)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_4-distancia_3)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_4)
pendiente_1xz=(Momento_1xz-0)/(distancia_1-0)
pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
distancia_1)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_max11=0

for x1=0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end

```

```

end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_1xy*(parte2_x3-
0)+0)
Momento_parte2_3XZ=(pendiente_2xz*(parte2_x3-
longitud_eje)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_1xy*(x4-0)+0)
Momento_4XZ=(pendiente_2xz*(x4-longitud_eje)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_1xy*(parte1_x5-
0)+0)
Momento_parte1_5XZ=(pendiente_2xz*(parte1_x5-
longitud_eje)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_2)+Momento_1xy)
Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
longitud_eje)+0)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_2xy*(x6-
distancia_2)+Momento_1xy)
Momento_6XZ=(pendiente_2xz*(x6-longitud_eje)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_2)+Momento_1xy)
Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
longitud_eje)+0)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_3)+Momento_2xy)
Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7

```

```

Xmax_7=Xmax_parte1_7
end
  for x8=sesion_7:0.1:sesion_8
    Momento_8XY=(pendiente_3xy*(x8-
distancia_3)+Momento_2xy)
    Momento_8XZ=(pendiente_2xz*(x8-longitud_eje)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
  if Momento_8>Momento_max8 then
    Momento_max8=Momento_8
    Xmax_8=x8
  end
end

  for parte1_x9=sesion_8:0.1:distancia_4
    Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_3)+Momento_2xy)
    Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
  if Momento_parte1_9>Momento_parte1_max9 then
    Momento_parte1_max9=Momento_parte1_9
    Xmax_parte1_9=parte1_x9
  end
end
  for parte2_x9=distancia_4:0.1:sesion_9
    Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
longitud_eje)+0)
    Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
  if Momento_parte2_9>Momento_parte2_max9 then
    Momento_parte2_max9=Momento_parte2_9
    Xmax_parte2_9=parte2_x9
  end
end
  if Momento_parte2_max9 > Momento_parte1_max9 then
    Momento_max9=Momento_parte2_max9
    Xmax_9= Xmax_parte2_9
end
  if Momento_parte1_max9 > Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end
  if Momento_parte1_max9 == Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end
  for x10=sesion_9:0.1:sesion_10
    Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)
    Momento_10XZ=(pendiente_2xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
  if Momento_10>Momento_max10 then
    Momento_max10=Momento_10
    Xmax_10=x10
  end
end
  for x11=sesion_10:0.1:longitud_eje
    Momento_11XY=(pendiente_4xy*(x11-longitud_eje)+0)
    Momento_11XZ=(pendiente_2xz*(x11-longitud_eje)+0)

```

```

Momento_11=sqrt(((Momento_11XY)^2)+((Momento_11XZ)
^2))
  if Momento_11>Momento_max11 then
    Momento_max11=Momento_11
    Xmax_11=x11
  end
end
end

```

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2) then

```

```

Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial33*lon
gitud_eje)+(W_radial44*longitud_eje)-
(W_radial11*distancia_1)-(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
  Momento_1xy=Reaccion_Axy*distancia_1
  Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)+(W_radial11*distancia_1)
  Momento_3xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)-
(W_radial33*distancia_4)+(W_radial11*distancia_1)+(W_radi
al33*distancia_3)
  Reaccion_Axz=W_radial22-
((W_radial22*distancia_2)/longitud_eje)
  Momento_1xz=Reaccion_Axz*distancia_2
  pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
  pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_1)
  pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_4-distancia_3)
  pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_4)
  pendiente_1xz=(Momento_1xz-0)/(distancia_2-0)
  pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
distancia_2)

```

```

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_max11=0

```

```

  for x1=0:0:0.1:sesion_1
    Momento_1XY=(pendiente_1xy*(x1-0)+0)
    Momento_1XZ=(pendiente_1xz*(x1-0)+0)

```

```

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
  if Momento_1>Momento_max1 then
    Momento_max1=Momento_1
    Xmax_1=x1
  end
end

```

```

    for x2=sesion_1:0.1:sesion_2
    Momento_2XY=(pendiente_1xy*(x2-0)+0)
    Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    if Momento_2>Momento_max2 then
        Momento_max2=Momento_2
        Xmax_2=x2
    end
end
for parte1_x3=sesion_2:0.1:distancia_1
    Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
    Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
    if Momento_parte1_3>Momento_parte1_max3 then
        Momento_parte1_max3=Momento_parte1_3
        Xmax_parte1_3=parte1_x3
    end
end
for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
    Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
    if Momento_parte2_3>Momento_parte2_max3 then
        Momento_parte2_max3=Momento_parte2_3
        Xmax_parte2_3=parte2_x3
    end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
    Momento_4XZ=(pendiente_1xz*(x4-0)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    if Momento_4>Momento_max4 then
        Momento_max4=Momento_4
        Xmax_4=x4
    end
end
for parte1_x5=sesion_4:0.1:distancia_2
    Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
    Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
    if Momento_parte1_5>Momento_parte1_max5 then
        Momento_parte1_max5=Momento_parte1_5
        Xmax_parte1_5=parte1_x5
    end
end
for parte2_x5=distancia_2:0.1:sesion_5
    Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_1)+Momento_1xy)
    Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
longitud_eje)+0)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
    if Momento_parte2_5>Momento_parte2_max5 then
        Momento_parte2_max5=Momento_parte2_5
        Xmax_parte2_5=parte2_x5
    end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
    Momento_max5=Momento_parte2_max5
    Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
    Momento_6XY=(pendiente_2xy*(x6-
distancia_1)+Momento_1xy)
    Momento_6XZ=(pendiente_2xz*(x6-longitud_eje)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    if Momento_6>Momento_max6 then
        Momento_max6=Momento_6
        Xmax_6=x6
    end
end
for parte1_x7=sesion_6:0.1:distancia_3
    Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_1)+Momento_1xy)
    Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
longitud_eje)+0)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
    if Momento_parte1_7>Momento_parte1_max7 then
        Momento_parte1_max7=Momento_parte1_7
        Xmax_parte1_7=parte1_x7
    end
end
for parte2_x7=distancia_3:0.1:sesion_7
    Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_3)+Momento_2xy)
    Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
    if Momento_parte2_7>Momento_parte2_max7 then
        Momento_parte2_max7=Momento_parte2_7
        Xmax_parte2_7=parte2_x7
    end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
    Momento_max7=Momento_parte2_max7
    Xmax_7= Xmax_parte2_7
end

```



```

end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_3xy*(x8-
distancia_3)+Momento_2xy)
Momento_8XZ=(pendiente_2xz*(x8-longitud_eje)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_3)+Momento_2xy)
Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)
Momento_10XZ=(pendiente_2xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10

```

```

Xmax_10=x10
end
for x11=sesion_10:0.1:longitud_eje
Momento_11XY=(pendiente_4xy*(x11-longitud_eje)+0)
Momento_11XZ=(pendiente_2xz*(x11-longitud_eje)+0)
Momento_11=sqrt(((Momento_11XY)^2)+((Momento_11XZ)
^2))
if Momento_11>Momento_max11 then
Momento_max11=Momento_11
Xmax_11=x11
end
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2) then
Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial22*lon
gitud_eje)+(W_radial44*longitud_eje)-
(W_radial11*distancia_1)-(W_radial22*distancia_2)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)-
(W_radial22*distancia_4)+(W_radial11*distancia_1)+(W_radi
al22*distancia_2)
Reaccion_Axz=W_radial33-
((W_radial33*distancia_3)/longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_3
pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_4-distancia_2)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_4)
pendiente_1xz=(Momento_1xz-0)/(distancia_3-0)
pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
distancia_3)
Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_max11=0
for x1=0:0.0:1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))

```

```

    if Momento_1>Momento_max1 then
        Momento_max1=Momento_1
        Xmax_1=x1
    end
end
for x2=sesion_1:0.1:sesion_2
    Momento_2XY=(pendiente_1xy*(x2-0)+0)
    Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
    Momento_max2=Momento_2
    Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
    Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
    Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
    Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
    Momento_4XZ=(pendiente_1xz*(x4-0)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
    Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
    Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
    Momento_parte1_max5=Momento_parte1_5
    Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
    Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
distancia_2)+Momento_2xy)
    Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
    Momento_parte2_max5=Momento_parte2_5
    Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
    Momento_max5=Momento_parte2_max5
    Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
    Momento_6XY=(pendiente_3xy*(x6-
distancia_2)+Momento_2xy)
    Momento_6XZ=(pendiente_1xz*(x6-0)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
    Momento_max6=Momento_6
    Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
    Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
distancia_2)+Momento_2xy)
    Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
    Momento_parte1_max7=Momento_parte1_7
    Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
    Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_2)+Momento_2xy)
    Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
    Momento_parte2_max7=Momento_parte2_7
    Xmax_parte2_7=parte2_x7
end
end

```



```

end
  if Momento_parte2_max7 > Momento_parte1_max7 then
    Momento_max7=Momento_parte2_max7
    Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
  Momento_max7=Momento_parte1_max7
  Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
  Momento_max7=Momento_parte1_max7
  Xmax_7=Xmax_parte1_7
end
  for x8=sesion_7:0.1:sesion_8
    Momento_8XY=(pendiente_3xy*(x8-
  distancia_2)+Momento_2xy)
    Momento_8XZ=(pendiente_2xz*(x8-longitud_eje)+0)

Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
  if Momento_8>Momento_max8 then
    Momento_max8=Momento_8
    Xmax_8=x8
  end
end

  for parte1_x9=sesion_8:0.1:distancia_4
    Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
  distancia_2)+Momento_2xy)
    Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
  longitud_eje)+0)

Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
  if Momento_parte1_9>Momento_parte1_max9 then
    Momento_parte1_max9=Momento_parte1_9
    Xmax_parte1_9=parte1_x9
  end
end
  for parte2_x9=distancia_4:0.1:sesion_9
    Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
  longitud_eje)+0)
    Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
  longitud_eje)+0)

Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
  if Momento_parte2_9>Momento_parte2_max9 then
    Momento_parte2_max9=Momento_parte2_9
    Xmax_parte2_9=parte2_x9
  end
end
  if Momento_parte2_max9 > Momento_parte1_max9 then
    Momento_max9=Momento_parte2_max9
    Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
  Momento_max9=Momento_parte1_max9
  Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
  Momento_max9=Momento_parte1_max9
  Xmax_9=Xmax_parte1_9
end
  for x10=sesion_9:0.1:sesion_10
    Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)
    Momento_10XZ=(pendiente_2xz*(x10-longitud_eje)+0)

```

```

Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
  if Momento_10>Momento_max10 then
    Momento_max10=Momento_10
    Xmax_10=x10
  end
end
  for x11=sesion_10:0.1:longitud_eje
    Momento_11XY=(pendiente_4xy*(x11-longitud_eje)+0)
    Momento_11XZ=(pendiente_2xz*(x11-longitud_eje)+0)

Momento_11=sqrt(((Momento_11XY)^2)+((Momento_11XZ)
^2))
  if Momento_11>Momento_max11 then
    Momento_max11=Momento_11
    Xmax_11=x11
  end
end
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2) then

Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial22*lon
gitud_eje)+(W_radial33*longitud_eje)-
(W_radial11*distancia_1)-(W_radial22*distancia_2)-
(W_radial33*distancia_3))/(longitud_eje)
  Momento_1xy=Reaccion_Axy*distancia_1
  Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
  Momento_3xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)-
(W_radial22*distancia_3)+(W_radial11*distancia_1)+(W_radi
al22*distancia_2)
  Reaccion_Axz=W_radial44-
((W_radial44*distancia_4)/longitud_eje)
  Momento_1xz=Reaccion_Axz*distancia_4
  pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
  pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
  pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_3-distancia_2)
  pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_3)
  pendiente_1xz=(Momento_1xz-0)/(distancia_4-0)
  pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
distancia_4)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_max11=0

```

```

for x1=0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_2)+Momento_2xy)
Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_2xy*(x4-
distancia_2+Momento_2xy))
Momento_4XZ=(pendiente_1xz*(x4-0)+0)

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end

```

```

end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_2)+Momento_2xy)
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)

Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
distancia_3)+Momento_3xy)
Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)

Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_3xy*(x6-
distancia_3)+Momento_3xy)
Momento_6XZ=(pendiente_1xz*(x6-0)+0)

Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
distancia_3)+Momento_3xy)
Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)

Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_4xy*(parte2_x7-
longitud_eje+0))
Momento_parte2_7XZ=(pendiente_1xz*(parte2_x7-0)+0)

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))

```

```

if Momento_parte2_7>Momento_parte2_max7 then
    Momento_parte2_max7=Momento_parte2_7
    Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
    Momento_max7=Momento_parte2_max7
    Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
    Momento_8XY=(pendiente_4xy*(x8-longitud_eje)+0)
    Momento_8XZ=(pendiente_1xz*(x8-0)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
    Momento_max8=Momento_8
    Xmax_8=x8
end
end

for parte1_x9=sesion_8:0.1:distancia_4
    Momento_parte1_9XY=(pendiente_4xy*(parte1_x9-
longitud_eje)+0)
    Momento_parte1_9XZ=(pendiente_1xz*(parte1_x9-0)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
    Momento_parte1_max9=Momento_parte1_9
    Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
    Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
longitud_eje)+0)
    Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
    Momento_parte2_max9=Momento_parte2_9
    Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
    Momento_max9=Momento_parte2_max9
    Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
    Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)

```

```

    Momento_10XZ=(pendiente_2xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
    Momento_max10=Momento_10
    Xmax_10=x10
end
end
for x11=sesion_10:0.1:longitud_eje
    Momento_11XY=(pendiente_4xy*(x11-longitud_eje)+0)
    Momento_11XZ=(pendiente_2xz*(x11-longitud_eje)+0)
Momento_11=sqrt(((Momento_11XY)^2)+((Momento_11XZ)
^2))
if Momento_11>Momento_max11 then
    Momento_max11=Momento_11
    Xmax_11=x11
end
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2) then
Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial22*lon
gitud_eje)-(W_radial11*distancia_1)-
(W_radial22*distancia_2))/(longitud_eje)
    Momento_1xy=Reaccion_Axy*distancia_1
    Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
Reaccion_Axz=((W_radial33*longitud_eje)+(W_radial44*lon
gitud_eje)-(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
    Momento_1xz=Reaccion_Axz*distancia_3
    Momento_2xz=(Reaccion_Axz*distancia_4)-
(W_radial33*distancia_4)+(W_radial33*distancia_3)
    pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
    pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
    pendiente_3xy=(0-Momento_2xy)/(longitud_eje-
distancia_2)
    pendiente_1xz=(Momento_1xz-0)/(distancia_3-0)
    pendiente_2xz=(Momento_2xy-
Momento_1xy)/(distancia_4-distancia_3)
    pendiente_3xz=(0-Momento_1xz)/(longitud_eje-
distancia_4)
Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_max11=0

```

```

for x1=0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_2)+Momento_2xy)
Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_2xy*(x4-
distancia_2)+Momento_2xy)
Momento_4XZ=(pendiente_1xz*(x4-0)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end

```

```

end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_2)+Momento_2xy)
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
longitud_eje)+0)
Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_3xy*(x6-longitud_eje)+0)
Momento_6XZ=(pendiente_1xz*(x6-0)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
longitud_eje)+0)
Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
distancia_3)+Momento_1xz)

```

```

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
    if Momento_parte2_7>Momento_parte2_max7 then
        Momento_parte2_max7=Momento_parte2_7
        Xmax_parte2_7=parte2_x7
    end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
    Momento_max7=Momento_parte2_max7
    Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
    for x8=sesion_7:0.1:sesion_8
        Momento_8XY=(pendiente_3xy*(x8-longitud_eje)+0)
        Momento_8XZ=(pendiente_2xz*(x8-
distancia_3)+Momento_1xz)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    if Momento_8>Momento_max8 then
        Momento_max8=Momento_8
        Xmax_8=x8
    end
end
    for parte1_x9=sesion_8:0.1:distancia_4
        Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
longitud_eje)+0)
        Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
distancia_3)+Momento_1xz)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
    if Momento_parte1_9>Momento_parte1_max9 then
        Momento_parte1_max9=Momento_parte1_9
        Xmax_parte1_9=parte1_x9
    end
end
    for parte2_x9=distancia_4:0.1:sesion_9
        Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
longitud_eje)+0)
        Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
    if Momento_parte2_9>Momento_parte2_max9 then
        Momento_parte2_max9=Momento_parte2_9
        Xmax_parte2_9=parte2_x9
    end
end
    if Momento_parte2_max9 > Momento_parte1_max9 then
        Momento_max9=Momento_parte2_max9
        Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9

```

```

Xmax_9=Xmax_parte1_9
end
    for x10=sesion_9:0.1:sesion_10
        Momento_10XY=(pendiente_3xy*(x10-longitud_eje)+0)
        Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
    if Momento_10>Momento_max10 then
        Momento_max10=Momento_10
        Xmax_10=x10
    end
end
    for x11=sesion_10:0.1:longitud_eje
        Momento_11XY=(pendiente_3xy*(x11-longitud_eje)+0)
        Momento_11XZ=(pendiente_3xz*(x11-longitud_eje)+0)
Momento_11=sqrt(((Momento_11XY)^2)+((Momento_11XZ)
^2))
    if Momento_11>Momento_max11 then
        Momento_max11=Momento_11
        Xmax_11=x11
    end
end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2) then
Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial44*lon
gitud_eje)-(W_radial11*distancia_1)-
(W_radial44*distancia_4))/(longitud_eje)
    Momento_1xy=Reaccion_Axy*distancia_1
    Momento_2xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)+(W_radial11*distancia_1)
Reaccion_Axz=((W_radial22*longitud_eje)+(W_radial33*lon
gitud_eje)-(W_radial22*distancia_2)-
(W_radial33*distancia_3))/(longitud_eje)
    Momento_1xz=Reaccion_Axz*distancia_2
    Momento_2xz=(Reaccion_Axz*distancia_3)-
(W_radial22*distancia_3)+(W_radial22*distancia_2)
    pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
    pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_4-distancia_1)
    pendiente_3xy=(0-Momento_2xy)/(longitud_eje-
distancia_4)
    pendiente_1xz=(Momento_1xz-0)/(distancia_2-0)
    pendiente_2xz=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_2)
    pendiente_3xz=(0-Momento_1xz)/(longitud_eje-
distancia_3)
Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0

```

```

Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_max11=0

for x1=0.0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-distancia_4)+Momento_2xy)
Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_2xy*(x4-distancia_4)+Momento_2xy)
Momento_4XZ=(pendiente_1xz*(x4-0)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-distancia_4)+Momento_2xy)
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-distancia_4)+Momento_2xy)
Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-distancia_3)+Momento_2xz)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_3xy*(x6-longitud_eje)+0)
Momento_6XZ=(pendiente_1xz*(x6-0)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-distancia_4)+Momento_2xy)
Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-distancia_3)+Momento_2xz)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7

```



```

Momento_parte2_7XY=(pendiente_2xy*(parte2_x7-
distancia_4)+Momento_2xy)
Momento_parte2_7XZ=(pendiente_3xz*(parte2_x7-
longitud_eje)+0)

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_2xy*(x8-
distancia_4)+Momento_2xy)
Momento_8XZ=(pendiente_3xz*(x8-longitud_eje)+0)

Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_2xy*(parte1_x9-
distancia_4)+Momento_2xy)
Momento_parte1_9XZ=(pendiente_3xz*(parte1_x9-
longitud_eje)+0)

Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)

Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9

```

```

Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_3xy*(x10-longitud_eje)+0)
Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)

Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for x11=sesion_10:0.1:longitud_eje
Momento_11XY=(pendiente_3xy*(x11-longitud_eje)+0)
Momento_11XZ=(pendiente_3xz*(x11-longitud_eje)+0)

Momento_11=sqrt(((Momento_11XY)^2)+((Momento_11XZ)
^2))
if Momento_11>Momento_max11 then
Momento_max11=Momento_11
Xmax_11=x11
end
end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2) then

Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial33*lon
gitud_eje)-(W_radial11*distancia_1)-
(W_radial33*distancia_3))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)+(W_radial11*distancia_1)

Reaccion_Axz=((W_radial22*longitud_eje)+(W_radial44*lon
gitud_eje)-(W_radial22*distancia_2)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_2
Momento_2xz=(Reaccion_Axy*distancia_4)-
(W_radial22*distancia_4)+(W_radial22*distancia_2)
pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_1)
pendiente_3xy=(0-Momento_2xy)/(longitud_eje-
distancia_3)
pendiente_1xz=(Momento_1xz-0)/(distancia_2-0)
pendiente_2xz=(Momento_2xy-
Momento_1xy)/(distancia_4-distancia_2)
pendiente_3xz=(0-Momento_1xz)/(longitud_eje-
distancia_4)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0

```

```

Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_max11=0

for x1=0.0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-distancia_3)+Momento_2xy)
Momento_parte2_3XZ=(pendiente_2xz*(parte2_x3-0)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_2xy*(x4-distancia_3)+Momento_2xy)
Momento_4XZ=(pendiente_1xz*(x4-0)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-distancia_3)+Momento_2xy)
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-distancia_3)+Momento_2xy)
Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-distancia_4)+Momento_2xz)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5=Xmax_parte2_5
end
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_3xy*(x6-longitud_eje)+0)
Momento_6XZ=(pendiente_1xz*(x6-0)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-distancia_3)+Momento_2xy)
Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-distancia_4)+Momento_2xz)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then

```



```

Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
distancia_4)+Momento_2xz)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_3xy*(x8-longitud_eje)+0)
Momento_8XZ=(pendiente_2xz*(x8-
distancia_4)+Momento_2xz)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
distancia_4)+Momento_2xz)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then

```

```

Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_3xy*(x10-longitud_eje)+0)
Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for x11=sesion_10:0.1:longitud_eje
Momento_11XY=(pendiente_3xy*(x11-longitud_eje)+0)
Momento_11XZ=(pendiente_3xz*(x11-longitud_eje)+0)
Momento_11=sqrt(((Momento_11XY)^2)+((Momento_11XZ)
^2))
if Momento_11>Momento_max11 then
Momento_max11=Momento_11
Xmax_11=x11
end
end
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2) then
x=0:0.1:longitud_eje
function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_2xy*(x-distancia_1)+Momento_1xy
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_2xy*(x-distancia_1)+Momento_1xy
elseif x>=distancia_2 & x < sesion_5 then
s=pendiente_3xy*(x-distancia_2)+Momento_2xy
elseif x>=sesion_5 & x < sesion_6 then
s=pendiente_3xy*(x-distancia_2)+Momento_2xy
elseif x>=sesion_6 & x < distancia_3 then
s=pendiente_3xy*(x-distancia_2)+Momento_2xy
elseif x>=distancia_3 & x < sesion_7 then
s=pendiente_4xy*(x-distancia_3)+Momento_3xy
elseif x>=sesion_7 & x < sesion_8 then
s=pendiente_4xy*(x-distancia_3)+Momento_3xy
elseif x>=sesion_8 & x < distancia_4 then
s=pendiente_4xy*(x-distancia_3)+Momento_3xy

```

```

elseif x>=distancia_4 & x < sesion_9 then
    s=pendiente_5xy*(x-longitud_eje)+0
elseif x>=sesion_9 & x < sesion_10 then
    s=pendiente_5xy*(x-longitud_eje)+0
elseif x>=sesion_10 & x < longitud_eje then
    s=pendiente_5xy*(x-longitud_eje)+0
else
    s=0
end

endfunction

plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 == ubicacion_3
& ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2) then
    x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_3 & x < sesion_4 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_1xy*(x-0)+0)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=pendiente_3xy*(x-distancia_3)+Momento_2xy
    elseif x>=sesion_7 & x < sesion_8 then
        s=pendiente_3xy*(x-distancia_3)+Momento_2xy
    elseif x>=sesion_8 & x < distancia_4 then
        s=pendiente_3xy*(x-distancia_3)+Momento_2xy
    elseif x>=distancia_4 & x < sesion_9 then
        s=pendiente_4xy*(x-longitud_eje)+0
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_4xy*(x-longitud_eje)+0)

    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    else
        s=0
    end

endfunction

x=0.0:0.1:longitud_eje
function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then

```

```

        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=pendiente_5xy*(x-longitud_eje)+0
    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    else
        s=0
    end

endfunction

x=0.0:0.1:longitud_eje
function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then
        s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
    elseif x>=distancia_1 & x < sesion_3 then
        s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
    elseif x>=sesion_3 & x < sesion_4 then
        s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x < distancia_2 then
        s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
    elseif x>=distancia_2 & x < sesion_5 then
        s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < distancia_3 then
        s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
    elseif x>=distancia_3 & x < sesion_7 then
        s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))

```

```

elseif x>=sesion_7 & x < sesion_8 then
    s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
    s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < longitud_eje then
    s=sqrt(((Momento_11XY)^2)+((Momento_11XZ)^2))
else
    s=0
end

```

endfunction

```

plot(x, MomentoXY1, "r")
plot(x, MomentoXZ1, "b")
plot(x, Momento_Resultante, "g")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2) then

```

```

    x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    else
        s=0
    end

```

end

endfunction

```

x=0.0:0.1:longitud_eje
function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_1 & x < sesion_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_2 & x < distancia_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    else
        s=0
    end

```

endfunction

```

x=0.0:0.1:longitud_eje
function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
    s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x < distancia_2 then

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then

```

```

        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
        elseif x>=sesion_6 & x < distancia_3 then

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)
^2))
        elseif x>=distancia_3 & x < sesion_7 then

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)
^2))
        elseif x>=sesion_7 & x < sesion_8 then
            s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
        elseif x>=sesion_8 & x < distancia_4 then

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)
^2))
        elseif x>=distancia_4 & x < sesion_9 then

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)
^2))
        elseif x>=sesion_9 & x < sesion_10 then
            s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
        elseif x>=sesion_10 & x < longitud_eje then
            s=sqrt(((Momento_11XY)^2)+((Momento_11XZ)^2))
        else
            s=0
        end

endfunction
plot(x, MomentoXY1, "r")
plot(x, MomentoXZ1, "b")
plot(x, Momento_Resultante, "g")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")

end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2) then
x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=sesion_8 & x < distancia_4 then

```

```

        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    else
        s=0
    end
endfunction

```

endfunction

```

x=0.0:0.1:longitud_eje
function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_1 & x < sesion_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_2 & x < distancia_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    else
        s=0
    end
endfunction

```

endfunction

```

x=0.0:0.1:longitud_eje
function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)
^2))
        elseif x>=distancia_1 & x < sesion_3 then

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)
^2))
        elseif x>=sesion_3 & x < sesion_4 then
            s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))

```

```

        elseif x>=sesion_4 & x < distancia_2 then
s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)
^2))
        elseif x>=distancia_2 & x < sesion_5 then
s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)
^2))
        elseif x>=sesion_5 & x < sesion_6 then
            s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
        elseif x>=sesion_6 & x < distancia_3 then
s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)
^2))
        elseif x>=distancia_3 & x < sesion_7 then
s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)
^2))
        elseif x>=sesion_7 & x < sesion_8 then
            s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
        elseif x>=sesion_8 & x < distancia_4 then
s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)
^2))
        elseif x>=distancia_4 & x < sesion_9 then
s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)
^2))
        elseif x>=sesion_9 & x < sesion_10 then
            s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
        elseif x>=sesion_10 & x < longitud_eje then
            s=sqrt(((Momento_11XY)^2)+((Momento_11XZ)^2))
        else
            s=0
        end
endfunction

```

```

plot(x, MomentoXY1, "r")
plot(x, MomentoXZ1, "b")
plot(x, Momento_Resultante, "g")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2) then
x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xy*(x-distancia_2+Momento_2xy))
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xy*(x-distancia_2+Momento_2xy))
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xy*(x-distancia_2+Momento_2xy))
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)
    elseif x>=sesion_5 & x < sesion_6 then

```

```

        s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_4xy*(x-longitud_eje+0))
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_4xy*(x-longitud_eje+0))
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_4xy*(x-longitud_eje+0))
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_4xy*(x-longitud_eje+0))
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_4xy*(x-longitud_eje+0))
    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_4xy*(x-longitud_eje+0))
    else
        s=0
    end
endfunction

```

endfunction

```

x=0.0:0.1:longitud_eje
function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_1 & x < sesion_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_2 & x < distancia_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    else
        s=0
    end
endfunction

```

endfunction

```

x=0.0:0.1:longitud_eje
function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then

```

```

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
    elseif x>=distancia_1 & x< sesion_3 then

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
    elseif x>=sesion_3 & x< sesion_4 then
        s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x< distancia_2 then

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
    elseif x>=distancia_2 & x< sesion_5 then

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
    elseif x>=sesion_5 & x< sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x< distancia_3 then

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
    elseif x>=distancia_3 & x< sesion_7 then

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
    elseif x>=sesion_7 & x< sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x< distancia_4 then

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
    elseif x>=distancia_4 & x< sesion_9 then

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
    elseif x>=sesion_9 & x< sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x< longitud_eje then
        s=sqrt(((Momento_11XY)^2)+((Momento_11XZ)^2))
    else
        s=0
    end

```

endfunction

```

plot(x, MomentoXY1, "r")
plot(x, MomentoXZ1, "b")
plot(x, Momento_Resultante, "g")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")

```

end

```

if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2) then
x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x< sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x< sesion_2 then

```

```

        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x< distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x< sesion_3 then
        s=(pendiente_2xy*(x-distancia_2+Momento_2xy))
    elseif x>=sesion_3 & x< sesion_4 then
        s=(pendiente_2xy*(x-distancia_2+Momento_2xy))
    elseif x>=sesion_4 & x< distancia_2 then
        s=(pendiente_2xy*(x-distancia_2+Momento_2xy))
    elseif x>=distancia_2 & x< sesion_5 then
        s=(pendiente_3xy*(x-longitud_eje+0))
    elseif x>=sesion_5 & x< sesion_6 then
        s=(pendiente_3xy*(x-longitud_eje+0))
    elseif x>=sesion_6 & x< distancia_3 then
        s=(pendiente_3xy*(x-longitud_eje+0))
    elseif x>=distancia_3 & x< sesion_7 then
        s=(pendiente_3xy*(x-longitud_eje+0))
    elseif x>=sesion_7 & x< sesion_8 then
        s=(pendiente_3xy*(x-longitud_eje+0))
    elseif x>=sesion_8 & x< distancia_4 then
        s=(pendiente_3xy*(x-longitud_eje+0))
    elseif x>=distancia_4 & x< sesion_9 then
        s=(pendiente_3xy*(x-longitud_eje+0))
    elseif x>=sesion_9 & x< sesion_10 then
        s=(pendiente_3xy*(x-longitud_eje+0))
    elseif x>=sesion_10 & x< longitud_eje then
        s=(pendiente_3xy*(x-longitud_eje+0))
    else
        s=0
    end

```

endfunction

```

x=0.0:0.1:longitud_eje
function s=MomentoXZ1(x)
    if x>=0 & x< sesion_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_1 & x< sesion_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_2 & x< distancia_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_1 & x< sesion_3 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_3 & x< sesion_4 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_4 & x< distancia_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_2 & x< sesion_5 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_5 & x< sesion_6 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_6 & x< distancia_3 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_3 & x< sesion_7 then
        s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
    elseif x>=sesion_7 & x< sesion_8 then
        s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
    elseif x>=sesion_8 & x< distancia_4 then
        s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
    elseif x>=distancia_4 & x< sesion_9 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x< sesion_10 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x< longitud_eje then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    else
        s=0
    end

```



```

end
endfunction

x=0.0:0.1:longitud_eje
function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then
        s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
    elseif x>=distancia_1 & x < sesion_3 then
        s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
    elseif x>=sesion_3 & x < sesion_4 then
        s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x < distancia_2 then
        s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
    elseif x>=distancia_2 & x < sesion_5 then
        s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < distancia_3 then
        s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
    elseif x>=distancia_3 & x < sesion_7 then
        s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
    elseif x>=sesion_7 & x < sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x < distancia_4 then
        s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
    elseif x>=distancia_4 & x < sesion_9 then
        s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
    elseif x>=sesion_9 & x < sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x < longitud_eje then
        s=sqrt(((Momento_11XY)^2)+((Momento_11XZ)^2))
    else
        s=0
    end
endfunction

plot(x, MomentoXY1, "r")
plot(x, MomentoXZ1, "b")
plot(x, Momento_Resultante, "g")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")

```

```

end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2) then
x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xy*(x-distancia_4)+Momento_2xy)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xy*(x-distancia_4)+Momento_2xy)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xy*(x-distancia_4)+Momento_2xy)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xy*(x-distancia_4)+Momento_2xy)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_3xy*(x6-longitud_eje)+0)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xy*(x-distancia_4)+Momento_2xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_2xy*(x-distancia_4)+Momento_2xy)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_2xy*(x-distancia_4)+Momento_2xy)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_2xy*(x-distancia_4)+Momento_2xy)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_3xy*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_3xy*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < longitud_eje then
        s=(pendiente_3xy*(x-longitud_eje)+0)
    else
        s=0
    end
endfunction

x=0.0:0.1:longitud_eje
function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_1 & x < sesion_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_2 & x < distancia_1 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xz*(x-distancia_3)+Momento_2xz)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_1xz*(x-0)+0)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xz*(parte1_x7-
distancia_3)+Momento_2xz)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=sesion_7 & x < sesion_8 then

```

```

s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < longitud_eje then
s=(pendiente_3xz*(x-longitud_eje)+0)
else
s=0
end

endfunction

x=0.0:0.1:longitud_eje
function s=Momento_Resultante(x)
if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then
s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then
s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then
s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then
s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then
s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then
s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
elseif x>=sesion_7 & x < sesion_8 then
s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then
s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then
s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < longitud_eje then
s=sqrt(((Momento_11XY)^2)+((Momento_11XZ)^2))
else
s=0
end
end

```

endfunction

```

plot(x, MomentoXY1, "r")
plot(x, MomentoXZ1, "b")
plot(x, Momento_Resultante, "g")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")

```

end

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2) then

```

```

x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_2xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_2xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_4 & x < distancia_2 then
s=(pendiente_2xy*(x-distancia_3)+Momento_2xy)
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_3xy*(x-longitud_eje)+0)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xy*(x-distancia_3)+Momento_2xy)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_3xy*(x-longitud_eje)+0)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_3xy*(x-longitud_eje)+0)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_3xy*(x-longitud_eje)+0)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_3xy*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_3xy*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < longitud_eje then
s=(pendiente_3xy*(x-longitud_eje)+0)
else
s=0
end
end

```

endfunction

```

x=0.0:0.1:longitud_eje
function s=MomentoXZ1(x)
if x>=0 & x < sesion_1 then
s=(pendiente_1xz*(x-0)+0)
elseif x>=sesion_1 & x < sesion_2 then
s=(pendiente_1xz*(x-0)+0)
elseif x>=sesion_2 & x < distancia_1 then
s=(pendiente_1xz*(x-0)+0)
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_1xz*(x-0)+0)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_1xz*(x-0)+0)

```



```

elseif x>=sesion_4 & x < distancia_2 then
    s=(pendiente_1xz*(x-0)+0)
elseif x>=distancia_2 & x < sesion_5 then
    s=(pendiente_2xz*(x-distancia_3)+Momento_2xz)
elseif x>=sesion_5 & x < sesion_6 then
    s=(pendiente_1xz*(x-0)+0)
elseif x>=sesion_6 & x < distancia_3 then
    s=(pendiente_2xz*(x-distancia_4)+Momento_2xz)
elseif x>=distancia_3 & x < sesion_7 then
    s=(pendiente_2xz*(x-distancia_4)+Momento_2xz)
elseif x>=sesion_7 & x < sesion_8 then
    s=(pendiente_2xz*(x-distancia_4)+Momento_2xz)
elseif x>=sesion_8 & x < distancia_4 then
    s=(pendiente_2xz*(x-distancia_4)+Momento_2xz)
elseif x>=distancia_4 & x < sesion_9 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < longitud_eje then
    s=(pendiente_3xz*(x-longitud_eje)+0)
else
    s=0
end
endfunction

```

endfunction

```

x=0:0:0.1:longitud_eje
function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then
        s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
    elseif x>=distancia_1 & x < sesion_3 then
        s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
    elseif x>=sesion_3 & x < sesion_4 then
        s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x < distancia_2 then
        s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
    elseif x>=distancia_2 & x < sesion_5 then
        s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < distancia_3 then
        s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
    elseif x>=distancia_3 & x < sesion_7 then
        s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
    elseif x>=sesion_7 & x < sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x < distancia_4 then
        s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
    elseif x>=distancia_4 & x < sesion_9 then

```

```

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
    elseif x>=sesion_9 & x < sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x < longitud_eje then
        s=sqrt(((Momento_11XY)^2)+((Momento_11XZ)^2))
    else
        s=0
    end
endfunction

```

```

endfunction
plot(x, MomentoXY1, "r")
plot(x, MomentoXZ1, "b")
plot(x, Momento_Resultante, "g")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

end

13. 8_VidaRemanente.sci

```

clc
clear all
if sigvma11 > sigvma10 & sigvma11 > sigvma9 & sigvma11 > sigvma8 & sigvma11 > sigvma7 & sigvma11 > sigvma6 & sigvma11 > sigvma5 & sigvma11 > sigvma4 & sigvma11 > sigvma3 & sigvma11 > sigvma2 & sigvma11 > sigvma1 then
    sigvma=sigvma11
end
if sigvma10 > sigvma11 & sigvma10 > sigvma9 & sigvma10 > sigvma8 & sigvma10 > sigvma7 & sigvma10 > sigvma6 & sigvma10 > sigvma5 & sigvma10 > sigvma4 & sigvma10 > sigvma3 & sigvma10 > sigvma2 & sigvma10 > sigvma1 then
    sigvma=sigvma10
end
if sigvma9 > sigvma11 & sigvma9 > sigvma10 & sigvma9 > sigvma8 & sigvma9 > sigvma7 & sigvma9 > sigvma6 & sigvma9 > sigvma5 & sigvma9 > sigvma4 & sigvma9 > sigvma3 & sigvma9 > sigvma2 & sigvma9 > sigvma1 then
    sigvma=sigvma9
end
if sigvma8 > sigvma11 & sigvma8 > sigvma10 & sigvma8 > sigvma9 & sigvma8 > sigvma7 & sigvma8 > sigvma6 & sigvma8 > sigvma5 & sigvma8 > sigvma4 & sigvma8 > sigvma3 & sigvma8 > sigvma2 & sigvma8 > sigvma1 then
    sigvma=sigvma8
end
if sigvma7 > sigvma11 & sigvma7 > sigvma10 & sigvma7 > sigvma9 & sigvma7 > sigvma8 & sigvma7 > sigvma6 & sigvma7 > sigvma5 & sigvma7 > sigvma4 & sigvma7 > sigvma3 & sigvma7 > sigvma2 & sigvma7 > sigvma1 then
    sigvma=sigvma7
end
if sigvma6 > sigvma11 & sigvma6 > sigvma10 & sigvma6 > sigvma9 & sigvma6 > sigvma8 & sigvma6 > sigvma7 & sigvma6 > sigvma5 & sigvma6 > sigvma4 & sigvma6 > sigvma3 & sigvma6 > sigvma2 & sigvma6 > sigvma1 then
    sigvma=sigvma6
end
if sigvma5 > sigvma11 & sigvma5 > sigvma10 & sigvma5 > sigvma9 & sigvma5 > sigvma8 & sigvma5 > sigvma7 & sigvma5 > sigvma6 & sigvma5 > sigvma4 & sigvma5 > sigvma3 & sigvma5 > sigvma2 & sigvma5 > sigvma1 then

```

```

sigvma5 > sigvma6 & sigvma5 > sigvma4 & sigvma5 > sigvma3
& sigvma5 > sigvma2 & sigvma5 > sigvma1 then
sigvma=sigvma5
end
if sigvma4 > sigvma11 & sigvma4 > sigvma10 & sigvma4 >
sigvma9 & sigvma4 > sigvma8 & sigvma4 > sigvma7 &
sigvma4 > sigvma6 & sigvma4 > sigvma5 & sigvma4 > sigvma3
& sigvma4 > sigvma2 & sigvma4 > sigvma1 then
sigvma=sigvma4
end
if sigvma3 > sigvma11 & sigvma3 > sigvma10 & sigvma3 >
sigvma9 & sigvma3 > sigvma8 & sigvma3 > sigvma7 &
sigvma3 > sigvma6 & sigvma3 > sigvma5 & sigvma3 > sigvma4
& sigvma3 > sigvma2 & sigvma3 > sigvma1 then
sigvma=sigvma3
end
if sigvma2 > sigvma11 & sigvma2 > sigvma10 & sigvma2 >
sigvma9 & sigvma2 > sigvma8 & sigvma2 > sigvma7 &
sigvma2 > sigvma6 & sigvma2 > sigvma5 & sigvma2 > sigvma4
& sigvma2 > sigvma2 & sigvma2 > sigvma3 then
sigvma=sigvma2
end
if sigvma1 > sigvma11 & sigvma1 > sigvma10 & sigvma1 >
sigvma9 & sigvma1 > sigvma8 & sigvma1 > sigvma7 &
sigvma1 > sigvma6 & sigvma1 > sigvma5 & sigvma1 > sigvma4
& sigvma1 > sigvma3 & sigvma1 > sigvma2 then
sigvma=sigvma1
end
SUT=0.9*Sut (1)
m1=(SUT-SE(1))
m2=(3-6)
pend=m1/m2 // calculo de la pendiente
x=((sigvma)-SUT+(pend*3))/pend //valor de x para el esfuerzo
alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)

```

13_CuatroEngranes.sci

```

clc
clear all

//Entrada de datos de las sesiones, muescas, ángulos, ubicación
y fuerzas:
exec('13.1_DistanciaSesionesMuecasYUbicacionEngrane.sci',
-1)

//Entrada del ajuste por interferencia para el primer engrane:
exec('11.2_OpcionesInterferenciaUno.sci',-1)

if interferencia == 1 then
exec('10.3_InterferenciaUno.sci',-1) //Para la opción 1
end

if interferencia == 2 then
exec('10.4_InterferenciaDos.sci',-1) // Para la opción 2
end

if interferencia == 3 then
exec('10.5_InterferenciaTres.sci',-1) //Para la opción 3
end

Interferen_maza_eje_min=Interf_maza_eje_min

```

```

Interferen_maza_eje_max=Interf_maza_eje_max

//Entrada del ajuste por interferencia para el segundo
engrane:
exec('11.3_OpcionesInterferenciaDos.sci',-1)

if interferencia2 == 1 then
exec('11.4_InterferenciaUno.sci',-1) //Para la opción 1
end

if interferencia2 == 2 then
exec('11.5_InterferenciaDos.sci',-1) // Para la opción 2
end

if interferencia2 == 3 then
exec('11.6_InterferenciaTres.sci',-1) //Para la opción 3
end

Interferen_maza_eje_min2=Interf_maza_eje_min2
Interferen_maza_eje_max2=Interf_maza_eje_max2

//Entrada del ajuste por interferencia para el tercer engrane:
exec('12.2_OpcionesInterferenciaTres.sci',-1)

if interferencia3 == 1 then
exec('12.3_InterferenciaUno.sci',-1) //Para la opción 1
end

if interferencia3 == 2 then
exec('12.4_InterferenciaDos.sci',-1) // Para la opción 2
end

if interferencia3 == 3 then
exec('12.5_InterferenciaTres.sci',-1) //Para la opción 3
end

Interferen_maza_eje_min3=Interf_maza_eje_min3
Interferen_maza_eje_max3=Interf_maza_eje_max3

//Entrada del ajuste por interferencia para el tercer engrane:
exec('13.2_OpcionesInterferenciaCuatro.sci',-1)

if interferencia4 == 1 then
exec('13.3_InterferenciaUno.sci',-1) //Para la opción 1
end

if interferencia4 == 2 then
exec('13.4_InterferenciaDos.sci',-1) // Para la opción 2
end

if interferencia4 == 3 then
exec('13.5_InterferenciaTres.sci',-1) //Para la opción 3
end

Interferen_maza_eje_min4=Interf_maza_eje_min4
Interferen_maza_eje_max4=Interf_maza_eje_max4

//Cálculo de diagramas de momentos con sus diagramas:
exec('13.7_TorquesMomentosYGraficas.sci',-1)
//Multiplicación de los factores de Marin Ka por Kd:
exec('10.8_MultiplicacionKaKd.sci',-1)
//Iteraciones de los diámetros:
exec('13.6_IteracionesDeLosDiametros.sci',-1)

```

14. 1_DitanciasSesionesMuecasYUbicacionEngrane.sci

```

clc
clear all

sesion_1=input('Ingrese la distancia de la sesion 1:')
Radio_muesca=input('Ingrese el radio de la muesca de la
sesión 1:')

c=0
while c==0
c=1
sesion_2=input('Ingrese la distancia de la sesion 2:')
if (sesion_2 <= sesion_1| sesion_2 >= longitud_eje)
c=0
disp('Distancia de la sesión 2 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca2=input('Ingrese el radio de la muesca de la
sesión 2:')

c=0
while c==0
c=1
sesion_3=input('Ingrese la distancia de la sesion 3:')
if (sesion_3 <= sesion_2 | sesion_3 <= sesion_1 | sesion_3
>= longitud_eje)
c=0
disp('Distancia de la sesión 3 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca3=input('Ingrese el radio de la muesca de la
sesión 3:')

c=0
while c==0
c=1
sesion_4=input('Ingrese la distancia de la sesion 4:')
if (sesion_4 <= sesion_3 | sesion_4 <= sesion_2 | sesion_4
<= sesion_1| sesion_4 >= longitud_eje)
c=0
disp('Distancia de la sesión 4 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca4=input('Ingrese el radio de la muesca de la
sesión 4:')

c=0
while c==0
c=1
sesion_5=input('Ingrese la distancia de la sesion 5:')
if (sesion_5 <= sesion_4 | sesion_5 <= sesion_3 | sesion_5
<= sesion_2 | sesion_5 <= sesion_1| sesion_5 >= longitud_eje)
c=0
disp('Distancia de la sesión 5 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca5=input('Ingrese el radio de la muesca de la
sesión 5:')

```

```

c=0
while c==0
c=1
sesion_6=input('Ingrese la distancia de la sesion 6:')
if (sesion_6 <= sesion_5 | sesion_6 <= sesion_4 | sesion_6
<= sesion_3 | sesion_6 <= sesion_2 | sesion_6 <= sesion_1|
sesion_6 >= longitud_eje)
c=0
disp('Distancia de la sesión 6 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca6=input('Ingrese el radio de la muesca de la
sesión 6:')

```

```

c=0
while c==0
c=1
sesion_7=input('Ingrese la distancia de la sesion 7:')
if (sesion_7 <= sesion_6 | sesion_7 <= sesion_5 | sesion_7 <=
sesion_4 | sesion_7 <= sesion_3 | sesion_7 <= sesion_2 |
sesion_7 <= sesion_1 | sesion_7 >= longitud_eje)
c=0
disp('Distancia de la sesión 7 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca7=input('Ingrese el radio de la muesca de la
sesión 7:')

```

```

c=0
while c==0
c=1
sesion_8=input('Ingrese la distancia de la sesion 8:')
if (sesion_8 <= sesion_7 | sesion_8 <= sesion_6 | sesion_8 <=
sesion_5 | sesion_8 <= sesion_4 | sesion_8 <= sesion_3 |
sesion_8 <= sesion_2 | sesion_8 <= sesion_1 | sesion_8 >=
longitud_eje)
c=0
disp('Distancia de la sesión 8 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca8=input('Ingrese el radio de la muesca de la
sesión 8:')

```

```

c=0
while c==0
c=1
sesion_9=input('Ingrese la distancia de la sesión 9:')
if (sesion_9 <= sesion_8 | sesion_9 <= sesion_7 | sesion_9
<= sesion_6 | sesion_9 <= sesion_5 | sesion_9 <= sesion_4 |
sesion_9 <= sesion_3 | sesion_9 <= sesion_2 |sesion_9 <=
sesion_1 | sesion_9 >= longitud_eje)
c=0
disp('Distancia de la sesión 9 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end

```

```
Radio_muesca9=input('Ingrese el radio de la muesca de la
sesión 9: ')

```

```
c=0
while c==0
c=1
sesion_10=input('Ingrese la distancia de la sesión 10:')
if (sesion_10 <= sesion_9 | sesion_10 <= sesion_8 |
sesion_10 <= sesion_7 | sesion_10 <= sesion_6 | sesion_10 <=
sesion_5 | sesion_10 <= sesion_4 | sesion_10 <= sesion_3 |
sesion_10 <= sesion_2 | sesion_10 <= sesion_1 | sesion_10 >=
longitud_eje)
c=0
disp('Distancia de la sesión 10 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca10=input('Ingrese el radio de la muesca de la
sesión 10: ')

```

```
c=0
while c==0
c=1
sesion_11=input('Ingrese la distancia de la sesión 11:')
if (sesion_11 <= sesion_10 | sesion_11 <= sesion_9 |
sesion_11 <= sesion_8 | sesion_11 <= sesion_7 | sesion_11 <=
sesion_6 | sesion_11 <= sesion_5 | sesion_11 <= sesion_4 |
sesion_11 <= sesion_3 | sesion_11 <= sesion_2 | sesion_11 <=
sesion_1 | sesion_11 >= longitud_eje)
c=0
disp('Distancia de la sesión 11 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca11=input('Ingrese el radio de la muesca de la
sesión 11: ')

```

```
c=0
while c==0
c=1
sesion_12=input('Ingrese la distancia de la sesión 12:')
if (sesion_12 <= sesion_11 | sesion_12 <= sesion_10 |
sesion_12 <= sesion_9 | sesion_12 <= sesion_8 | sesion_12 <=
sesion_7 | sesion_12 <= sesion_6 | sesion_12 <= sesion_5 |
sesion_12 <= sesion_4 | sesion_12 <= sesion_3 | sesion_12 <=
sesion_2 | sesion_12 <= sesion_1 | sesion_12 >= longitud_eje)
c=0
disp('Distancia de la sesión 12 incorrecta. Escriba resume
para volver a ingresar el valor')
pause
end
end
Radio_muesca12=input('Ingrese el radio de la muesca de la
sesión 12: ')

```

```
E= 30000

```

```
//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 1:
exec('11.1.1_DistanciasFuerzasAnguloUbicacionEngraneUno.
sci',-1)

```

```
//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 2:
exec('11.1.2_DistanciasFuerzasAnguloUbicacionEngraneDos.
sci',-1)

```

```
//DISTANCIAS, FUERZAS, ÁNGULOS Y UBICACIÓN DEL
ENGRANE 3:
exec('11.1.3_DistanciasFuerzasAnguloUbicacionEngraneTres.
sci',-1)

```

```
//DISTANCIAS, FUERZAS, ANGULOS Y UBICACION DEL
ENGRANE 4:
exec('11.1.4_DistanciasFuerzasAnguloUbicacionEngraneCuatr
o.sci',-1)

```

```
//DISTANCIAS, FUERZAS, ANGULOS Y UBICACION DEL
ENGRANE 5:
exec('11.1.5_DistanciasFuerzasAnguloUbicacionEngraneCinc
o.sci',-1)

```

14. 2_OpcionesInterferenciaCinco.sci

```
clc
clear all
c=0
while c==0
c=1
disp("OPCIONES PARA EL TIPO DE AJUSTE POR
INTERFERENCIA")
disp("1- Ajuste de interferencia localizada H7/P6")
disp("2- Ajuste de impulso medio H7/S6")
disp("3- Ajuste forzado H7/U6")

```

```
interferencia5 = input('Seleccione la opción de ajuste por
interferencia deseada para el quinto engrane: ')
if (interferencia5<1 | interferencia5>3)
c=0
disp('Opción incorrecta. Escriba resume para volver a
ingresar el valor')
pause
end
end

```

14. 3_InterferenciaUno.sci

```
clc
clear all
c=0
while c==0
c=1
tamaño_basico5 = input('Ingrese el tamaño básico:')
if (tamaño_basico5<0 | tamaño_basico5>16)
c=0
disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
pause
end
end
p5=0;

```

```
if tamaño_basico5 >=0 & tamaño_basico5 < 0.12 then
p5=tabla14(1,1)
end

```

```

if tamano_basico5 >= 0.12 & tamano_basico5 < 0.24 then
  p5=tabla14(2,1)
end

if tamano_basico5 >= 0.24 & tamano_basico5 < 0.40 then
  p5=tabla14(3,1)
end

if tamano_basico5 >= 0.40 & tamano_basico5 < 0.72 then
  p5=tabla14(4,1)
end

if tamano_basico5 >= 0.72 & tamano_basico5 < 0.96 then
  p5=tabla14(5,1)
end

if tamano_basico5 >= 0.96 & tamano_basico5 < 1.20 then
  p5=tabla14(6,1)
end

if tamano_basico5 >= 1.20 & tamano_basico5 < 1.60 then
  p5=tabla14(7,1)
end

if tamano_basico5 >= 1.60 & tamano_basico5 < 2.00 then
  p5=tabla14(8,1)
end

if tamano_basico5 >= 2.00 & tamano_basico5 < 2.60 then
  p5=tabla14(9,1)
end

if tamano_basico5 >= 2.60 & tamano_basico5 < 3.20 then
  p5=tabla14(10,1)
end

if tamano_basico5 >= 3.20 & tamano_basico5 < 4.00 then
  p5=tabla14(11,1)
end

if tamano_basico5 >= 4.00 & tamano_basico5 < 4.80 then
  p5=tabla14(12,1)
end

if tamano_basico5 >= 4.80 & tamano_basico5 < 5.60 then
  p5=tabla14(13,1)
end

if tamano_basico5 >= 5.60 & tamano_basico5 < 6.40 then
  p5=tabla14(14,1)
end

if tamano_basico5 >= 6.40 & tamano_basico5 < 7.20 then
  p5=tabla14(15,1)
end

if tamano_basico5 >= 7.20 & tamano_basico5 < 8.00 then
  p5=tabla14(16,1)
end

if tamano_basico5 >= 8.00 & tamano_basico5 < 9.00 then
  p5=tabla14(17,1)
end

if tamano_basico5 >= 9.00 & tamano_basico5 < 10.00 then
  p5=tabla14(18,1)

```

```

end

if tamano_basico5 >= 10.00 & tamano_basico5 < 11.20 then
  p5=tabla14(19,1)
end

if tamano_basico5 >= 11.20 & tamano_basico5 < 12.60 then
  p5=tabla14(20,1)
end

if tamano_basico5 >= 12.60 & tamano_basico5 < 14.20 then
  p5=tabla14(21,1)
end

if tamano_basico5 >= 14.20 & tamano_basico5 < 16.00 then
  p5=tabla14(22,1)
end

it6_5=0
it7_5=0

if tamano_basico5>=0 & tamano_basico5<0.12 then
  it6_5=Tabla_13(1,1)
  it7_5=Tabla_13(1,2)
end

if tamano_basico5>=0.12 & tamano_basico5<0.24 then
  it6_5=Tabla_13(2,1)
  it7_5=Tabla_13(2,2)
end

if tamano_basico5>=0.24 & tamano_basico5<0.40 then
  it6_5=Tabla_13(3,1)
  it7_5=Tabla_13(3,2)
end

if tamano_basico5>=0.40 & tamano_basico5<0.72 then
  it6_5=Tabla_13(4,1)
  it7_5=Tabla_13(4,2)
end

if tamano_basico5>=0.72 & tamano_basico5<1.20 then
  it6_5=Tabla_13(5,1)
  it7_5=Tabla_13(5,2)
end

if tamano_basico5>=1.20 & tamano_basico5<2.00 then
  it6_5=Tabla_13(6,1)
  it7_5=Tabla_13(6,2)
end

if tamano_basico5>=2.00 & tamano_basico5<3.20 then
  it6_5=Tabla_13(7,1)
  it7_5=Tabla_13(7,2)
end

if tamano_basico5>=3.20 & tamano_basico5<4.80 then
  it6_5=Tabla_13(8,1)
  it7_5=Tabla_13(8,2)
end

if tamano_basico5 >= 4.80 & tamano_basico5 < 7.20 then
  it6_5=Tabla_13(9,1)
  it7_5=Tabla_13(9,2)
end

if tamano_basico5 >= 7.20 & tamano_basico5 < 10.00 then
  it6_5=Tabla_13(10,1)
  it7_5=Tabla_13(10,2)
end

if tamano_basico5 >= 10.00 & tamano_basico5 < 12.60 then
  it6_5=Tabla_13(11,1)
  it7_5=Tabla_13(11,2)
end

if tamano_basico5 >= 12.60 & tamano_basico5 < 16.00 then
  it6_5=Tabla_13(12,1)

```

```

it7_5=Tabla_13(12,2)
end

tolerancia_agujero5=it7_5
tolerancia_eje5=it6_5
Diametro_max_agujero5=tamano_basico5 +
tolerancia_agujero5
Diametro_min_agujero5=tamano_basico5
Diametro_max_eje5= tamano_basico5 + p5 + tolerancia_eje5
Diametro_min_eje5= tamano_basico5 + p5

Interf_maza_eje_min5=Diametro_min_eje5-
Diametro_max_agujero5
Interf_maza_eje_max5=Diametro_max_eje5-
Diametro_min_agujero5

```

14. 4 InterferenciaDos.sci

```

clc
clear all
c=0
while c==0
c=1
tamano_basico5 = input('Ingrese el tamaño básico:')
if (tamano_basico5<0 | tamano_basico5>16)
c=0
disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ingresar el valor')
pause
end
end
s5=0;

if tamano_basico5 >=0 & tamano_basico5 < 0.12 then
s5=tabla14(1,2)
end

if tamano_basico5 >= 0.12 & tamano_basico5 < 0.24 then
s5=tabla14(2,2)
end

if tamano_basico5 >= 0.24 & tamano_basico5 < 0.40 then
s5=tabla14(3,2)
end

if tamano_basico5 >= 0.40 & tamano_basico5 < 0.72 then
s5=tabla14(4,2)
end

if tamano_basico5 >= 0.72 & tamano_basico5 < 0.96 then
s5=tabla14(5,2)
end

if tamano_basico5 >= 0.96 & tamano_basico5 < 1.20 then
s5=tabla14(6,2)
end

if tamano_basico5 >= 1.20 & tamano_basico5 < 1.60 then
s5=tabla14(7,2)
end

if tamano_basico5 >= 1.60 & tamano_basico5 < 2.00 then
s5=tabla14(8,2)
end

```

```

if tamano_basico5 >= 2.00 & tamano_basico5 < 2.60 then
s5=tabla14(9,2)
end

if tamano_basico5 >= 2.60 & tamano_basico5 < 3.20 then
s5=tabla14(10,2)
end

if tamano_basico5 >= 3.20 & tamano_basico5 < 4.00 then
s5=tabla14(11,2)
end

if tamano_basico5 >= 4.00 & tamano_basico5 < 4.80 then
s5=tabla14(12,2)
end

if tamano_basico5 >= 4.80 & tamano_basico5 < 5.60 then
s5=tabla14(13,2)
end

if tamano_basico5 >= 5.60 & tamano_basico5 < 6.40 then
s5=tabla14(14,2)
end

if tamano_basico5 >= 6.40 & tamano_basico5 < 7.20 then
s5=tabla14(15,2)
end

if tamano_basico5 >= 7.20 & tamano_basico5 < 8.00 then
s5=tabla14(16,2)
end

if tamano_basico5 >= 8.00 & tamano_basico5 < 9.00 then
s5=tabla14(17,2)
end

if tamano_basico5 >= 9.00 & tamano_basico5 < 10.00 then
s5=tabla14(18,2)
end

if tamano_basico5 >= 10.00 & tamano_basico5 < 11.20 then
s5=tabla14(19,2)
end

if tamano_basico5 >= 11.20 & tamano_basico5 < 12.60 then
s5=tabla14(20,2)
end

if tamano_basico5 >= 12.60 & tamano_basico5 < 14.20 then
s5=tabla14(21,2)
end

if tamano_basico5 >= 14.20 & tamano_basico5 < 16.00 then
s5=tabla14(22,2)
end

it6_5=0
it7_5=0

if tamano_basico5>=0 & tamano_basico5<0.12 then
it6_5=Tabla_13(1,1)
it7_5=Tabla_13(1,2)
end
if tamano_basico5>=0.12 & tamano_basico5<0.24 then
it6_5=Tabla_13(2,1)
it7_5=Tabla_13(2,2)
end
end

```

```

if tamaño_basico5 >= 0.24 & tamaño_basico5 < 0.40 then
    it6_5=Tabla_13(3,1)
    it7_5=Tabla_13(3,2)
end
if tamaño_basico5 >= 0.40 & tamaño_basico5 < 0.72 then
    it6_5=Tabla_13(4,1)
    it7_5=Tabla_13(4,2)
end
if tamaño_basico5 >= 0.72 & tamaño_basico5 < 1.20 then
    it6_5=Tabla_13(5,1)
    it7_5=Tabla_13(5,2)
end
if tamaño_basico5 >= 1.20 & tamaño_basico5 < 2.00 then
    it6_5=Tabla_13(6,1)
    it7_5=Tabla_13(6,2)
end
if tamaño_basico5 >= 2.00 & tamaño_basico5 < 3.20 then
    it6_5=Tabla_13(7,1)
    it7_5=Tabla_13(7,2)
end
if tamaño_basico5 >= 3.20 & tamaño_basico5 < 4.80 then
    it6_5=Tabla_13(8,1)
    it7_5=Tabla_13(8,2)
end
if tamaño_basico5 >= 4.80 & tamaño_basico5 < 7.20 then
    it6_5=Tabla_13(9,1)
    it7_5=Tabla_13(9,2)
end
if tamaño_basico5 >= 7.20 & tamaño_basico5 < 10.00 then
    it6_5=Tabla_13(10,1)
    it7_5=Tabla_13(10,2)
end
if tamaño_basico5 >= 10.00 & tamaño_basico5 < 12.60 then
    it6_5=Tabla_13(11,1)
    it7_5=Tabla_13(11,2)
end
if tamaño_basico5 >= 12.60 & tamaño_basico5 < 16.00 then
    it6_5=Tabla_13(12,1)
    it7_5=Tabla_13(12,2)
end

tolerancia_agujero5 = it7_5
tolerancia_eje5 = it6_5
Diametro_max_agujero5 = tamaño_basico5 +
tolerancia_agujero5
Diametro_min_agujero5 = tamaño_basico5 -
Diametro_max_eje5 = tamaño_basico5 + s5 + tolerancia_eje5
Diametro_min_eje5 = tamaño_basico5 + s5

Interf_maza_eje_min5 = Diametro_min_eje5 -
Diametro_max_agujero5
Interf_maza_eje_max5 = Diametro_max_eje5 -
Diametro_min_agujero5

```

14. 5_InterferenciaTres.sci

```

clc
clear all
c=0
while c==0
    c=1
    tamaño_basico5 = input('Ingrese el tamaño básico:')
    u5=0;
    if (tamaño_basico5 < 0 | tamaño_basico5 > 16)
        c=0

```

```

        disp('Opción de tamaño básico incorrecto. Escriba resume
para volver a ver el menu anterior')
        pause
    end
end

if tamaño_basico5 >= 0 & tamaño_basico5 < 0.12 then
    u5=tabla14(1,3)
end

if tamaño_basico5 >= 0.12 & tamaño_basico5 < 0.24 then
    u5=tabla14(2,3)
end

if tamaño_basico5 >= 0.24 & tamaño_basico5 < 0.40 then
    u5=tabla14(3,3)
end

if tamaño_basico5 >= 0.40 & tamaño_basico5 < 0.72 then
    u5=tabla14(4,3)
end

if tamaño_basico5 >= 0.72 & tamaño_basico5 < 0.96 then
    u5=tabla14(5,3)
end

if tamaño_basico5 >= 0.96 & tamaño_basico5 < 1.20 then
    u5=tabla14(6,3)
end

if tamaño_basico5 >= 1.20 & tamaño_basico5 < 1.60 then
    u5=tabla14(7,3)
end

if tamaño_basico5 >= 1.60 & tamaño_basico5 < 2.00 then
    u5=tabla14(8,3)
end

if tamaño_basico5 >= 2.00 & tamaño_basico5 < 2.60 then
    u5=tabla14(9,3)
end

if tamaño_basico5 >= 2.60 & tamaño_basico5 < 3.20 then
    u5=tabla14(10,3)
end

if tamaño_basico5 >= 3.20 & tamaño_basico5 < 4.00 then
    u5=tabla14(11,3)
end

if tamaño_basico5 >= 4.00 & tamaño_basico5 < 4.80 then
    u5=tabla14(12,3)
end

if tamaño_basico5 >= 4.80 & tamaño_basico5 < 5.60 then
    u5=tabla14(13,3)
end

if tamaño_basico5 >= 5.60 & tamaño_basico5 < 6.40 then
    u5=tabla14(14,3)
end

if tamaño_basico5 >= 6.40 & tamaño_basico5 < 7.20 then
    u5=tabla14(15,3)
end

if tamaño_basico5 >= 7.20 & tamaño_basico5 < 8.00 then

```



```

u5=tabla14(16,3)
end

if tamaño_basico5 >= 8.00 & tamaño_basico5 < 9.00 then
u5=tabla14(17,3)
end

if tamaño_basico5 >= 9.00 & tamaño_basico5 < 10.00 then
u5=tabla14(18,3)
end

if tamaño_basico5 >= 10.00 & tamaño_basico5 < 11.20 then
u5=tabla14(19,3)
end

if tamaño_basico5 >= 11.20 & tamaño_basico5 < 12.60 then
u5=tabla14(20,3)
end

if tamaño_basico5 >= 12.60 & tamaño_basico5 < 14.20 then
u5=tabla14(21,3)
end

if tamaño_basico5 >= 14.20 & tamaño_basico5 < 16.00 then
u5=tabla14(22,3)
end

it6_5=0
it7_5=0

if tamaño_basico5>=0 & tamaño_basico5<0.12 then
it6_5=Tabla_13(1,1)
it7_5=Tabla_13(1,2)
end
if tamaño_basico5>=0.12 & tamaño_basico5<0.24 then
it6_5=Tabla_13(2,1)
it7_5=Tabla_13(2,2)
end
if tamaño_basico5>=0.24 & tamaño_basico5<0.40 then
it6_5=Tabla_13(3,1)
it7_5=Tabla_13(3,2)
end
if tamaño_basico5>=0.40 & tamaño_basico5<0.72 then
it6_5=Tabla_13(4,1)
it7_5=Tabla_13(4,2)
end
if tamaño_basico5>=0.72 & tamaño_basico5<1.20 then
it6_5=Tabla_13(5,1)
it7_5=Tabla_13(5,2)
end
if tamaño_basico5>=1.20 & tamaño_basico5<2.00 then
it6_5=Tabla_13(6,1)
it7_5=Tabla_13(6,2)
end
if tamaño_basico5>=2.00 & tamaño_basico5<3.20 then
it6_5=Tabla_13(7,1)
it7_5=Tabla_13(7,2)
end
if tamaño_basico5>=3.20 & tamaño_basico5<4.80 then
it6_5=Tabla_13(8,1)
it7_5=Tabla_13(8,2)
end
if tamaño_basico5 >= 4.80 & tamaño_basico5 < 7.20 then
it6_5=Tabla_13(9,1)
it7_5=Tabla_13(9,2)
end
if tamaño_basico5 >= 7.20 & tamaño_basico5 < 10.00 then

```

```

it6_5=Tabla_13(10,1)
it7_5=Tabla_13(10,2)
end
if tamaño_basico5 >= 10.00 & tamaño_basico5 < 12.60 then
it6_5=Tabla_13(11,1)
it7_5=Tabla_13(11,2)
end
if tamaño_basico5 >= 12.60 & tamaño_basico5 < 16.00 then
it6_5=Tabla_13(12,1)
it7_5=Tabla_13(12,2)
end

tolerancia_agujero5 = it7_5
tolerancia_eje5=it6_5
Diametro_max_agujero5=tamaño_basico5 +
tolerancia_agujero5
Diametro_min_agujero5=tamaño_basico5
Diametro_max_eje5= tamaño_basico5 + u5 + tolerancia_eje5
Diametro_min_eje5= tamaño_basico5 + u5

Interf_maza_eje_min5=Diametro_min_eje5-
Diametro_max_agujero5
Interf_maza_eje_max5=Diametro_max_eje5-
Diametro_min_agujero5

```

14. 6_TorquesMomentosYGrafica.sci

```

clc
clear all

W_tangencial11=W_tangencial1/1000
W_radial11=W_radial1/1000
W_tangencial22=W_tangencial2/1000
W_radial22=W_radial2/1000
W_tangencial33=W_tangencial3/1000
W_radial33=W_radial3/1000
W_tangencial44=W_tangencial4/1000
W_radial44=W_radial4/1000
W_tangencial55=W_tangencial5/1000
W_radial55=W_radial5/1000

if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial22*lon
gitud_eje)+(W_radial33*longitud_eje)+(W_radial44*longitud
_eje)+(W_radial55*longitud_eje)-(W_radial11*distancia_1)-
(W_radial22*distancia_2)-(W_radial33*distancia_3)-
(W_radial44*distancia_4)-
(W_radial55*distancia_5))/(longitud_eje)

Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)-
(W_radial22*distancia_3)+(W_radial11*distancia_1)+(W_radi
al22*distancia_2)
Momento_4xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)-(W_radial22*distancia_4)-

```



```

(W_radial33*distancia_4)+(W_radial22*distancia_2)+(W_radial11*distancia_1)+(W_radial33*distancia_3)
Momento_5xy=(Reaccion_Axy*distancia_5)-
(W_radial11*distancia_5)-(W_radial22*distancia_5)-
(W_radial33*distancia_5)-
(W_radial44*distancia_5)+(W_radial22*distancia_2)+(W_radial11*distancia_1)+(W_radial33*distancia_3)+(W_radial44*distancia_4)

pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_3-distancia_2)
pendiente_4xy=(Momento_4xy-
Momento_3xy)/(distancia_4-distancia_3)
pendiente_5xy=(Momento_5xy-
Momento_4xy)/(distancia_5-distancia_4)
pendiente_6xy=(0-Momentos_5xy)/(longitud_eje-
distancia_5)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

for x1=0:0.1:sesion_1
Momento_1XY=pendiente_1xy*(x1-0)+0
if Momento_1XY>Momento_max1 then
Momento_max1=Momento_1XY
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=pendiente_1xy*(x2-0)+0
if Momento_2XY>Momento_max2 then
Momento_max2=Momento_2XY
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=pendiente_1xy*(parte1_x3-0)+0
if Momento_parte1_3XY>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3XY
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy
if Momento_parte2_3XY>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3XY
Xmax_parte2_3=parte2_x3
end

end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=pendiente_2xy*(x4-
distancia_1)+Momento_1xy
if Momento_4XY>Momento_max4 then
Momento_max4=Momento_4XY
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy
if Momento_parte1_5XY>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5XY
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=pendiente_3xy*(parte2_x5-
distancia_2)+Momento_2xy
if Momento_parte2_5XY>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5XY
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5
then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=pendiente_3xy*(x6-
distancia_2)+Momento_2xy
if Momento_6XY>Momento_max6 then
Momento_max6=Momento_6XY
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=pendiente_3xy*(parte1_x7-
distancia_2)+Momento_2xy
if Momento_parte1_7XY>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7XY
Xmax_parte1_7=parte1_x7
end
end

```

```

end
  for parte2_x7=distancia_3:0.1:sesion_7
    Momento_parte2_7XY=pendiente_4xy*(parte2_x7-
distancia_3)+Momento_3xy
    if Momento_parte2_7XY>Momento_parte2_max7 then
      Momento_parte2_max7=Momento_parte2_7XY
      Xmax_parte2_7=parte2_x7
    end
  end
end
  if Momento_parte2_max7 > Momento_parte1_max7
then
  Momento_max7=Momento_parte2_max7
  Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
  Momento_max7=Momento_parte1_max7
  Xmax_7=Xmax_parte1_7
end
end
  if Momento_parte1_max7 == Momento_parte2_max7 then
  Momento_max7=Momento_parte1_max7
  Xmax_7=Xmax_parte1_7
end
end
  for x8=sesion_7:0.1:sesion_8
    Momento_8XY=pendiente_4xy*(x8-
distancia_3)+Momento_3xy
    if Momento_8XY>Momento_max8 then
      Momento_max8=Momento_8XY
      Xmax_8=x8
    end
  end
end
  for parte1_x9=sesion_8:0.1:distancia_4
    Momento_parte1_9XY=pendiente_4xy*(parte1_x9-
distancia_3)+Momento_3xy
    if Momento_parte1_9XY>Momento_parte1_max9 then
      Momento_parte1_max9=Momento_parte1_9XY
      Xmax_parte1_9=parte1_x9
    end
  end
end
  for parte2_x9=distancia_4:0.1:sesion_9
    Momento_parte2_9XY=pendiente_5xy*(parte2_x9-
distancia_4)+Momento_4xy
    if Momento_parte2_9XY>Momento_parte2_max9 then
      Momento_parte2_max9=Momento_parte2_9XY
      Xmax_parte2_9=parte2_x9
    end
  end
end
  if Momento_parte2_max9 > Momento_parte1_max9
then
  Momento_max9=Momento_parte2_max9
  Xmax_9= Xmax_parte2_9
end
end
  if Momento_parte1_max9 > Momento_parte2_max9 then
  Momento_max9=Momento_parte1_max9
  Xmax_9=Xmax_parte1_9
end
end
  if Momento_parte1_max9 == Momento_parte2_max9 then
  Momento_max9=Momento_parte1_max9
  Xmax_9=Xmax_parte1_9
end
end
  for x10=sesion_9:0.1:sesion_10
    Momento_10XY=pendiente_5xy*(x10-
distancia_4)+Momento_4xy
    if Momento_10XY>Momento_max10 then
      Momento_max10=Momento_10XY
      Xmax_10=x10
    end
  end
end

```

```

  for parte1_x11=sesion_10:0.1:distancia_5
    Momento_parte1_11XY=pendiente_5xy*(parte1_x11-
distancia_4)+Momento_4xy
    if Momento_parte1_11XY>Momento_parte1_max11
then
      Momento_parte1_max11=Momento_parte1_11XY
      Xmax_parte1_11=parte1_x11
    end
  end
end
  for parte2_x11=distancia_5:0.1:sesion_11
    Momento_parte2_11XY=pendiente_6xy*(parte2_x11-
distancia_5)+Momento_5xy
    if Momento_parte2_11XY>Momento_parte2_max11
then
      Momento_parte2_max11=Momento_parte2_11XY
      Xmax_parte2_11=parte2_x11
    end
  end
end
  if Momento_parte2_max11 >
Momento_parte1_max11 then
    Momento_max11=Momento_parte2_max11
    Xmax_11= Xmax_parte2_11
  end
end
  if Momento_parte1_max11 > Momento_parte2_max11 then
    Momento_max11=Momento_parte1_max11
    Xmax_11=Xmax_parte1_11
  end
end
  if Momento_parte1_max11 == Momento_parte2_max11
then
    Momento_max11=Momento_parte1_max11
    Xmax_11=Xmax_parte1_11
  end
end
  for x12=sesion_11:0.1:sesion_12
    Momento_12XY=pendiente_6xy*(x12-longitud_eje)+0
    if Momento_12XY>Momento_max12 then
      Momento_max12=Momento_12XY
      Xmax_12=x12
    end
  end
end
  for x13=sesion_12:0.1:longitud_eje
    Momento_13XY=pendiente_6xy*(x13-longitud_eje)+0
    if Momento_13XY>Momento_max13 then
      Momento_max13=Momento_13XY
      Xmax_13=x13
    end
  end
end
end
  if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 ==
ubicacion_4) then
Reaccion_Axy=((W_radial55*longitud_eje)+(W_radial22*lon
gitud_eje)+(W_radial33*longitud_eje)+(W_radial44*longitud
_eje)-(W_radial55*distancia_5)-(W_radial22*distancia_2)-
(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
  Momento_1xy=Reaccion_Axy*distancia_2
  Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial22*distancia_3)+(W_radial22*distancia_2)
  Momento_3xy=(Reaccion_Axy*distancia_4)-
(W_radial22*distancia_4)-

```

```

(W_radial33*distancia_4)+(W_radial22*distancia_2)+(W_radial33*distancia_3)
Momento_4xy=(Reaccion_Axy*distancia_5)-
(W_radial22*distancia_5)-(W_radial33*distancia_5)-
(W_radial44*distancia_5)+(W_radial33*distancia_3)+(W_radial22*distancia_2)+(W_radial44*distancia_4)
Reaccion_Axz=W_radial11-
((W_radial11*distancia_1)/longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_1

pendiente_1xy=(Momento_1xy-0)/(distancia_2-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_2)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_4-distancia_3)
pendiente_4xy=(Momento_4xy-
Momento_3xy)/(distancia_5-distancia_4)
pendiente_5xy=(0-Momento_4xy)/(longitud_eje-
distancia_5)
pendiente_1xz=(Momento_1xz-0)/(distancia_1-0)
pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
distancia_1)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

for x1=0:0:1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0:1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0:1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

```

```

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0:1:sesion_3
Momento_parte2_3XY=(pendiente_1xy*(parte2_x3-
0)+0)
Momento_parte2_3XZ=(pendiente_2xz*(parte2_x3-
longitud_eje)+0)

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0:1:sesion_4
Momento_4XY=(pendiente_1xy*(x4-0)+0)
Momento_4XZ=(pendiente_2xz*(x4-longitud_eje)+0)

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0:1:distancia_2
Momento_parte1_5XY=(pendiente_1xy*(parte1_x5-
0)+0)
Momento_parte1_5XZ=(pendiente_2xz*(parte1_x5-
longitud_eje)+0)

Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0:1:sesion_5
Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_2)+Momento_1xy)
Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
longitud_eje)+0)

Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end

```

```

end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_2xy*(x6-
distancia_2)+Momento_1xy)
Momento_6XZ=(pendiente_2xz*(x6-longitud_eje)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_2)+Momento_1xy)
Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
longitud_eje)+0)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_3)+Momento_2xy)
Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_3xy*(x8-
distancia_3)+Momento_2xy)
Momento_8XZ=(pendiente_2xz*(x8-longitud_eje)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_3)+Momento_2xy)
Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
distancia_4)+Momento_3xy)
Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_4xy*(x10-
distancia_4)+Momento_3xy)
Momento_10XZ=(pendiente_2xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
distancia_4)+Momento_3xy)
Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))

```

```

    if Momento_parte1_11>Momento_parte1_max11 then
        Momento_parte1_max11=Momento_parte1_11
        Xmax_parte1_11=parte1_x11
    end
end
for parte2_x11=distancia_5:0.1:sesion_11
    Momento_parte2_11XY=(pendiente_5xy*(parte2_x11-
longitud_eje)+0)
    Momento_parte2_11XZ=(pendiente_2xz*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
    if Momento_parte2_11>Momento_parte2_max11 then
        Momento_parte2_max11=Momento_parte2_11
        Xmax_parte2_11=parte2_x11
    end
end

    if Momento_parte2_max11 > Momento_parte1_max11
then
        Momento_max11=Momento_parte2_max11
        Xmax_11= Xmax_parte2_11
    end
    if Momento_parte1_max11 > Momento_parte2_max11 then
        Momento_max11=Momento_parte1_max11
        Xmax_11=Xmax_parte1_11
    end
    if Momento_parte1_max11 == Momento_parte2_max11
then
        Momento_max11=Momento_parte1_max11
        Xmax_11=Xmax_parte1_11
    end
    for x12=sesion_11:0.1:sesion_12
        Momento_12XY=(pendiente_5xy*(x12-longitud_eje)+0)
        Momento_12XZ=(pendiente_2xz*(x12-longitud_eje)+0)
Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
    if Momento_12>Momento_max12 then
        Momento_max12=Momento_12
        Xmax_12=x12
    end
end
    for x13=sesion_12:0.1:longitud_eje
        Momento_13XY=(pendiente_5xy*(x13-longitud_eje)+0)
        Momento_13XZ=(pendiente_2xz*(x13-longitud_eje)+0)
Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
    if Momento_13>Momento_max13 then
        Momento_max13=Momento_13
        Xmax_13=x13
    end
end
end

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

Reaccion_Axy=((W_radial55*longitud_eje)+(W_radial11*lon
gitud_eje)+(W_radial33*longitud_eje)+(W_radial44*longitud

```

```

_eje)-(W_radial55*distancia_5)-(W_radial11*distancia_1)-
(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
    Momento_1xy=Reaccion_Axy*distancia_1
    Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)+(W_radial11*distancia_1)
    Momento_3xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)-
(W_radial33*distancia_4)+(W_radial11*distancia_1)+(W_radi
al33*distancia_3)
    Momento_4xy=(Reaccion_Axy*distancia_5)-
(W_radial11*distancia_5)-(W_radial33*distancia_5)-
(W_radial44*distancia_5)+(W_radial33*distancia_3)+(W_radi
al11*distancia_1)+(W_radial44*distancia_4)
    Reaccion_Axz=W_radial22-
((W_radial22*distancia_2)/longitud_eje)
    Momento_1xz=Reaccion_Axz*distancia_2
    pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
    pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_1)
    pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_4-distancia_3)
    pendiente_4xy=(Momento_4xy-
Momento_3xy)/(distancia_5-distancia_4)
    pendiente_5xy=(0-Momento_4xy)/(longitud_eje-
distancia_5)
    pendiente_1xz=(Momento_1xz-0)/(distancia_2-0)
    pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
distancia_2)

    Momento_max1=0
    Momento_max2=0
    Momento_parte1_max3=0
    Momento_parte2_max3=0
    Momento_max4=0
    Momento_parte1_max5=0
    Momento_parte2_max5=0
    Momento_max6=0
    Momento_parte1_max7=0
    Momento_parte2_max7=0
    Momento_max8=0
    Momento_parte1_max9=0
    Momento_parte2_max9=0
    Momento_max10=0
    Momento_parte1_max11=0
    Momento_parte2_max11=0
    Momento_max12=0
    Momento_max13=0
for x1=0:0:0.1:sesion_1
    Momento_1XY=(pendiente_1xy*(x1-0)+0)
    Momento_1XZ=(pendiente_1xz*(x1-0)+0)
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    if Momento_1>Momento_max1 then
        Momento_max1=Momento_1
        Xmax_1=x1
    end
end
    for x2=sesion_1:0.1:sesion_2
        Momento_2XY=(pendiente_1xy*(x2-0)+0)
        Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    if Momento_2>Momento_max2 then
        Momento_max2=Momento_2
        Xmax_2=x2
    end
end

```

```

end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
Momento_4XZ=(pendiente_1xz*(x4-0)+0)

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)

Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_1)+Momento_1xy)
Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
longitud_eje)+0)

Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_2xy*(x6-
distancia_1)+Momento_1xy)
Momento_6XZ=(pendiente_2xz*(x6-longitud_eje)+0)

Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_1)+Momento_1xy)
Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
longitud_eje)+0)

Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_3)+Momento_2xy)
Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
longitud_eje)+0)

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7

```

```

Xmax_7=Xmax_parte1_7
end
  for x8=sesion_7:0.1:sesion_8
    Momento_8XY=(pendiente_3xy*(x8-
distancia_3)+Momento_2xy)
    Momento_8XZ=(pendiente_2xz*(x8-longitud_eje)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
  if Momento_8>Momento_max8 then
    Momento_max8=Momento_8
    Xmax_8=x8
  end
end

  for parte1_x9=sesion_8:0.1:distancia_4
    Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_3)+Momento_2xy)
    Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
  if Momento_parte1_9>Momento_parte1_max9 then
    Momento_parte1_max9=Momento_parte1_9
    Xmax_parte1_9=parte1_x9
  end
end

  for parte2_x9=distancia_4:0.1:sesion_9
    Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
distancia_4)+Momento_3xy)
    Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
  if Momento_parte2_9>Momento_parte2_max9 then
    Momento_parte2_max9=Momento_parte2_9
    Xmax_parte2_9=parte2_x9
  end
end

  if Momento_parte2_max9 > Momento_parte1_max9
then
    Momento_max9=Momento_parte2_max9
    Xmax_9= Xmax_parte2_9
end
  if Momento_parte1_max9 > Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end
  if Momento_parte1_max9 == Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end

  for x10=sesion_9:0.1:sesion_10
    Momento_10XY=(pendiente_4xy*(x10-
distancia_4)+Momento_3xy)
    Momento_10XZ=(pendiente_2xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
  if Momento_10>Momento_max10 then
    Momento_max10=Momento_10
    Xmax_10=x10
  end
end
  for parte1_x11=sesion_10:0.1:distancia_5

```

```

    Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
distancia_4)+Momento_3xy)
    Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
  if Momento_parte1_11>Momento_parte1_max11 then
    Momento_parte1_max11=Momento_parte1_11
    Xmax_parte1_11=parte1_x11
  end
end

  for parte2_x11=distancia_5:0.1:sesion_11
    Momento_parte2_11XY=(pendiente_5xy*(parte2_x11-
longitud_eje)+0)
    Momento_parte2_11XZ=(pendiente_2xz*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
  if Momento_parte2_11>Momento_parte2_max11 then
    Momento_parte2_max11=Momento_parte2_11
    Xmax_parte2_11=parte2_x11
  end
end

  if Momento_parte2_max11 > Momento_parte1_max11
then
    Momento_max11=Momento_parte2_max11
    Xmax_11= Xmax_parte2_11
end
  if Momento_parte1_max11 > Momento_parte2_max11 then
    Momento_max11=Momento_parte1_max11
    Xmax_11=Xmax_parte1_11
end
  if Momento_parte1_max11 == Momento_parte2_max11
then
    Momento_max11=Momento_parte1_max11
    Xmax_11=Xmax_parte1_11
end

  for x12=sesion_11:0.1:sesion_12
    Momento_12XY=(pendiente_5xy*(x12-longitud_eje)+0)
    Momento_12XZ=(pendiente_2xz*(x12-longitud_eje)+0)
Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
  if Momento_12>Momento_max12 then
    Momento_max12=Momento_12
    Xmax_12=x12
  end
end

  for x13=sesion_12:0.1:longitud_eje
    Momento_13XY=(pendiente_5xy*(x13-longitud_eje)+0)
    Momento_13XZ=(pendiente_2xz*(x13-longitud_eje)+0)
Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
  if Momento_13>Momento_max13 then
    Momento_max13=Momento_13
    Xmax_13=x13
  end
end
end

if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 ==

```



```
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 ==  
ubicacion_4) then
```

```
Reaccion_Axy=((W_radial55*longitud_eje)+(W_radial11*lon  
gitud_eje)+(W_radial22*longitud_eje)+(W_radial44*longitud  
_eje)-(W_radial55*distancia_5)-(W_radial11*distancia_1)-  
(W_radial22*distancia_2)-  
(W_radial44*distancia_4))/(longitud_eje)  
Momento_1xy=Reaccion_Axy*distancia_1  
Momento_2xy=(Reaccion_Axy*distancia_2)-  
(W_radial11*distancia_2)+(W_radial11*distancia_1)  
Momento_3xy=(Reaccion_Axy*distancia_4)-  
(W_radial11*distancia_4)-  
(W_radial22*distancia_4)+(W_radial11*distancia_1)+(W_radi  
al22*distancia_2)  
Momento_4xy=(Reaccion_Axy*distancia_5)-  
(W_radial11*distancia_5)-(W_radial22*distancia_5)-  
(W_radial44*distancia_5)+(W_radial22*distancia_2)+(W_radi  
al11*distancia_1)+(W_radial44*distancia_4)  
Reaccion_Axz=W_radial33-  
((W_radial33*distancia_3)/longitud_eje)  
Momento_1xz=Reaccion_Axz*distancia_3  
pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)  
pendiente_2xy=(Momento_2xy-  
Momento_1xy)/(distancia_2-distancia_1)  
pendiente_3xy=(Momento_3xy-  
Momento_2xy)/(distancia_4-distancia_2)  
pendiente_4xy=(Momento_4xy-  
Momento_3xy)/(distancia_5-distancia_4)  
pendiente_5xy=(0-Momento_4xy)/(longitud_eje-  
distancia_5)  
pendiente_1xz=(Momento_1xz-0)/(distancia_3-0)  
pendiente_2xz=(0-Momento_1xz)/(longitud_eje-  
distancia_3)
```

```
Momento_max1=0  
Momento_max2=0  
Momento_parte1_max3=0  
Momento_parte2_max3=0  
Momento_max4=0  
Momento_parte1_max5=0  
Momento_parte2_max5=0  
Momento_max6=0  
Momento_parte1_max7=0  
Momento_parte2_max7=0  
Momento_max8=0  
Momento_parte1_max9=0  
Momento_parte2_max9=0  
Momento_max10=0  
Momento_parte1_max11=0  
Momento_parte2_max11=0  
Momento_max12=0  
Momento_max13=0
```

```
for x1=0:0:1:sesion_1  
Momento_1XY=(pendiente_1xy*(x1-0)+0)  
Momento_1XZ=(pendiente_1xz*(x1-0)+0)
```

```
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))  
if Momento_1>Momento_max1 then  
Momento_max1=Momento_1  
Xmax_1=x1  
end  
end  
for x2=sesion_1:0:1:sesion_2  
Momento_2XY=(pendiente_1xy*(x2-0)+0)  
Momento_2XZ=(pendiente_1xz*(x2-0)+0)
```

```
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))  
if Momento_2>Momento_max2 then  
Momento_max2=Momento_2  
Xmax_2=x2  
end
```

```
end  
for parte1_x3=sesion_2:0:1:distancia_1  
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-  
0)+0)  
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
```

```
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom  
ento_parte1_3XZ)^2))  
if Momento_parte1_3>Momento_parte1_max3 then  
Momento_parte1_max3=Momento_parte1_3  
Xmax_parte1_3=parte1_x3  
end
```

```
end  
for parte2_x3=distancia_1:0:1:sesion_3  
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-  
distancia_1)+Momento_1xy)  
Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
```

```
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom  
ento_parte2_3XZ)^2))  
if Momento_parte2_3>Momento_parte2_max3 then  
Momento_parte2_max3=Momento_parte2_3  
Xmax_parte2_3=parte2_x3  
end
```

```
end  
if Momento_parte2_max3 > Momento_parte1_max3 then  
Momento_max3=Momento_parte2_max3  
Xmax_3= Xmax_parte2_3  
end
```

```
if Momento_parte1_max3 > Momento_parte2_max3 then  
Momento_max3=Momento_parte1_max3  
Xmax_3=Xmax_parte1_3  
end
```

```
if Momento_parte1_max3 == Momento_parte2_max3 then  
Momento_max3=Momento_parte1_max3  
Xmax_3=Xmax_parte1_3  
end
```

```
for x4=sesion_3:0:1:sesion_4  
Momento_4XY=(pendiente_2xy*(x4-  
distancia_1)+Momento_1xy)  
Momento_4XZ=(pendiente_1xz*(x4-0)+0)
```

```
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))  
if Momento_4>Momento_max4 then  
Momento_max4=Momento_4  
Xmax_4=x4  
end
```

```
end  
for parte1_x5=sesion_4:0:1:distancia_2  
Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-  
distancia_1)+Momento_1xy)  
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
```

```
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom  
ento_parte1_5XZ)^2))  
if Momento_parte1_5>Momento_parte1_max5 then  
Momento_parte1_max5=Momento_parte1_5  
Xmax_parte1_5=parte1_x5  
end
```

```
end  
for parte2_x5=distancia_2:0:1:sesion_5
```



```

Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
distancia_2)+Momento_2xy)
Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)

Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_3xy*(x6-
distancia_2)+Momento_2xy)
Momento_6XZ=(pendiente_1xz*(x6-0)+0)

Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
distancia_2)+Momento_2xy)
Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)

Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_2)+Momento_2xy)
Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
longitud_eje)+0)

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end

if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_3xy*(x8-
distancia_2)+Momento_2xy)
Momento_8XZ=(pendiente_2xz*(x8-longitud_eje)+0)

Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_2)+Momento_2xy)
Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
longitud_eje)+0)

Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
distancia_4)+Momento_3xy)
Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
longitud_eje)+0)

Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_4xy*(x10-
distancia_4)+Momento_3xy)
Momento_10XZ=(pendiente_2xz*(x10-longitud_eje)+0)

Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
end

```

```

    for parte1_x11=sesion_10:0.1:distancia_5
      Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
distancia_4)+Momento_3xy)
      Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
longitud_eje)+0)

Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
      if Momento_parte1_11>Momento_parte1_max11 then
        Momento_parte1_max11=Momento_parte1_11
        Xmax_parte1_11=parte1_x11
      end
    end
    for parte2_x11=distancia_5:0.1:sesion_11
      Momento_parte2_11XY=(pendiente_5xy*(parte2_x11-
longitud_eje)+0)
      Momento_parte2_11XZ=(pendiente_2xz*(parte2_x11-
longitud_eje)+0)

Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
      if Momento_parte2_11>Momento_parte2_max11 then
        Momento_parte2_max11=Momento_parte2_11
        Xmax_parte2_11=parte2_x11
      end
    end
    if Momento_parte2_max11 > Momento_parte1_max11
then
      Momento_max11=Momento_parte2_max11
      Xmax_11= Xmax_parte2_11
    end
    if Momento_parte1_max11 > Momento_parte2_max11 then
      Momento_max11=Momento_parte1_max11
      Xmax_11=Xmax_parte1_11
    end
    if Momento_parte1_max11 == Momento_parte2_max11
then
      Momento_max11=Momento_parte1_max11
      Xmax_11=Xmax_parte1_11
    end
    for x12=sesion_11:0.1:sesion_12
      Momento_12XY=(pendiente_5xy*(x12-longitud_eje)+0)
      Momento_12XZ=(pendiente_2xz*(x12-longitud_eje)+0)

Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
      if Momento_12>Momento_max12 then
        Momento_max12=Momento_12
        Xmax_12=x12
      end
    end
    for x13=sesion_12:0.1:longitud_eje
      Momento_13XY=(pendiente_5xy*(x13-longitud_eje)+0)
      Momento_13XZ=(pendiente_2xz*(x13-longitud_eje)+0)

Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
      if Momento_13>Momento_max13 then
        Momento_max13=Momento_13
        Xmax_13=x13
      end
    end
  end
  end
  if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>

```

```

ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

```

```

Reaccion_Axy=((W_radial55*longitud_eje)+(W_radial11*lon
gitud_eje)+(W_radial33*longitud_eje)+(W_radial22*longitud
_eje)-(W_radial55*distancia_5)-(W_radial11*distancia_1)-
(W_radial33*distancia_3)-
(W_radial22*distancia_2))/(longitud_eje)

```

```

      Momento_1xy=Reaccion_Axy*distancia_1
      Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
      Momento_3xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)-
(W_radial22*distancia_3)+(W_radial11*distancia_1)+(W_radi
al22*distancia_2)
      Momento_4xy=(Reaccion_Axy*distancia_5)-
(W_radial11*distancia_5)-(W_radial22*distancia_5)-
(W_radial33*distancia_5)+(W_radial22*distancia_2)+(W_radi
al11*distancia_1)+(W_radial33*distancia_3)
      Reaccion_Axz=W_radial44-
((W_radial44*distancia_4)/longitud_eje)
      Momento_1xz=Reaccion_Axz*distancia_4
      pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
      pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
      pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_3-distancia_2)
      pendiente_4xy=(Momento_4xy-
Momento_3xy)/(distancia_5-distancia_3)
      pendiente_5xy=(0-Momento_4xy)/(longitud_eje-
distancia_5)
      pendiente_1xz=(Momento_1xz-0)/(distancia_4-0)
      pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
distancia_4)

```

```

      Momento_max1=0
      Momento_max2=0
      Momento_parte1_max3=0
      Momento_parte2_max3=0
      Momento_max4=0
      Momento_parte1_max5=0
      Momento_parte2_max5=0
      Momento_max6=0
      Momento_parte1_max7=0
      Momento_parte2_max7=0
      Momento_max8=0
      Momento_parte1_max9=0
      Momento_parte2_max9=0
      Momento_max10=0
      Momento_parte1_max11=0
      Momento_parte2_max11=0
      Momento_max12=0
      Momento_max13=0

```

```

for x1=0:0.1:sesion_1
  Momento_1XY=(pendiente_1xy*(x1-0)+0)
  Momento_1XZ=(pendiente_1xz*(x1-0)+0)

```

```

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
  if Momento_1>Momento_max1 then
    Momento_max1=Momento_1
    Xmax_1=x1
  end
end
  for x2=sesion_1:0.1:sesion_2
    Momento_2XY=(pendiente_1xy*(x2-0)+0)

```

```

Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
    Momento_max2=Momento_2
    Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
    Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
    Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_2)+Momento_2xy)
    Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_2xy*(x4-
distancia_2+Momento_2xy))
    Momento_4XZ=(pendiente_1xz*(x4-0)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
    Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_2)+Momento_2xy)
    Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
    Momento_parte1_max5=Momento_parte1_5
    Xmax_parte1_5=parte1_x5
end
end
end
for parte2_x5=distancia_2:0.1:sesion_5
    Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
distancia_3)+Momento_3xy)
    Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
    Momento_parte2_max5=Momento_parte2_5
    Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
    Momento_max5=Momento_parte2_max5
    Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
    Momento_6XY=(pendiente_3xy*(x6-
distancia_3)+Momento_3xy)
    Momento_6XZ=(pendiente_1xz*(x6-0)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
    Momento_max6=Momento_6
    Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
    Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
distancia_3)+Momento_3xy)
    Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
    Momento_parte1_max7=Momento_parte1_7
    Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
    Momento_parte2_7XY=(pendiente_4xy*(parte2_x7-
distancia_5)+Momento_4xy)
    Momento_parte2_7XZ=(pendiente_1xz*(parte2_x7-0)+0)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
    Momento_parte2_max7=Momento_parte2_7
    Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
    Momento_max7=Momento_parte2_max7
    Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
end

```

```

if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_4xy*(x8-
distancia_5)+Momento_4xy)
Momento_8XZ=(pendiente_1xz*(x8-0)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end

for parte1_x9=sesion_9:0.1:distancia_4
Momento_parte1_9XY=(pendiente_4xy*(parte1_x9-
distancia_5)+Momento_4xy)
Momento_parte1_9XZ=(pendiente_1xz*(parte1_x9-0)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end

for parte2_x9=distancia_4:0.1:sesion_10
Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
distancia_5)+Momento_4xy)
Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end

if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end

if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end

if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end

for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_4xy*(x10-
distancia_5)+Momento_4xy)
Momento_10XZ=(pendiente_2xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end

for parte1_x11=sesion_10:0.1:distancia_5

```

```

Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
distancia_5)+Momento_4xy)
Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end

for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_5xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_2xz*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end

if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end

if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end

if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end

for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_5xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_2xz*(x12-longitud_eje)+0)
Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end

for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_5xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_2xz*(x13-longitud_eje)+0)
Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then
Momento_max13=Momento_13
Xmax_13=x13
end
end

if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 <>

```

```
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 <>
ubicacion_4) then
```

```
Reaccion_Axy=((W_radial22*longitud_eje)+(W_radial11*lon
gitud_eje)+(W_radial33*longitud_eje)+(W_radial44*longitud
_eje)-(W_radial22*distancia_2)-(W_radial11*distancia_1)-
(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)-
(W_radial22*distancia_3)+(W_radial11*distancia_1)+(W_radi
al22*distancia_2)
Momento_4xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)-(W_radial22*distancia_4)-
(W_radial33*distancia_4)+(W_radial22*distancia_2)+(W_radi
al11*distancia_1)+(W_radial33*distancia_3)
Reaccion_Axz=W_radial55-
((W_radial55*distancia_5)/longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_5
pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_3-distancia_2)
pendiente_4xy=(Momento_4xy-
Momento_3xy)/(distancia_4-distancia_3)
pendiente_5xy=(0-Momento_4xy)/(longitud_eje-
distancia_4)
pendiente_1xz=(Momento_1xz-0)/(distancia_5-0)
pendiente_2xz=(0-Momento_1xz)/(longitud_eje-
distancia_5)
```

```
Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0
```

```
for x1=0.0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)
```

```
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)
```

```
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
```

```
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_2)+Momento_2xy)
Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
```

```
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
```

```
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_2xy*(x4-
distancia_2+Momento_2xy))
Momento_4XZ=(pendiente_1xz*(x4-0)+0)
```

```
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_2)+Momento_2xy)
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
```

```
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
```

```

Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
distancia_3)+Momento_3xy)
Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_3xy*(x6-
distancia_3)+Momento_3xy)
Momento_6XZ=(pendiente_1xz*(x6-0)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:sesion_7
Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
distancia_3)+Momento_3xy)
Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_4xy*(parte2_x7-
distancia_4)+Momento_4xy)
Momento_parte2_7XZ=(pendiente_1xz*(parte2_x7-0)+0)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_4xy*(x8-
distancia_4)+Momento_4xy)
Momento_8XZ=(pendiente_1xz*(x8-0)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_9:0.1:sesion_10
Momento_parte1_9XY=(pendiente_4xy*(parte1_x9-
distancia_4)+Momento_4xy)
Momento_parte1_9XZ=(pendiente_1xz*(parte1_x9-0)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_10
Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-0)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)
Momento_10XZ=(pendiente_2xz*(x10-0)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:sesion_11
Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
longitud_eje)+0)

```



```

Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
0)+0)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_5xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_2xz*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end
if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_5xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_2xz*(x12-longitud_eje)+0)
Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_5xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_2xz*(x13-longitud_eje)+0)
Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then
Momento_max13=Momento_13
Xmax_13=x13
end
end
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

```

Reaccion_Axy=((W_radial55*longitud_eje)+(W_radial33*lon
gitud_eje)+(W_radial44*longitud_eje)-
(W_radial55*distancia_5)-(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_3
Momento_2xy=(Reaccion_Axy*distancia_4)-
(W_radial33*distancia_4)+(W_radial33*distancia_3)
Momento_3xy=(Reaccion_Axy*distancia_5)-
(W_radial33*distancia_5)-
(W_radial44*distancia_5)+(W_radial33*distancia_3)+(W_radi
al44*distancia_4)
Reaccion_Axz=((W_radial11*longitud_eje)+(W_radial22*lon
gitud_eje)-(W_radial11*distancia_1)-
(W_radial22*distancia_2))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_1
Momento_2xz=(Reaccion_Axz*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
pendiente_1xy=(Momento_1xy-0)/(distancia_3-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_4-distancia_3)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_5-distancia_4)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_5)
pendiente_1xz=(Momento_1xz-0)/(distancia_1-0)
pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_2-distancia_1)
pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_2)
Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0
for x1=0:0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))

```

```

    if Momento_2>Momento_max2 then
        Momento_max2=Momento_2
        Xmax_2=x2
    end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_1xy*(parte2_x3-
0)+0)
    Momento_parte2_3XZ=(pendiente_2xz*(parte2_x3-
distancia_1)+Momento_1xz)

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_1xy*(x4-0)+0)
    Momento_4XZ=(pendiente_2xz*(x4-
distancia_1)+Momento_1xz)

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
    Momento_parte1_5XY=(pendiente_1xy*(parte1_x5-
0)+0)
    Momento_parte1_5XZ=(pendiente_2xz*(parte1_x5-
distancia_1)+Momento_1xz)

Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
    Momento_parte1_max5=Momento_parte1_5
    Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
    Momento_parte2_5XY=(pendiente_1xy*(parte2_x5-
0)+0)
    Momento_parte2_5XZ=(pendiente_3xz*(parte2_x5-
longitud_eje)+0)

Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
    Momento_parte2_max5=Momento_parte2_5
    Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
    Momento_max5=Momento_parte2_max5
    Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
    Momento_6XY=(pendiente_1xy*(x6-0)+0)
    Momento_6XZ=(pendiente_3xz*(x6-longitud_eje)+0)

Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
    Momento_max6=Momento_6
    Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
    Momento_parte1_7XY=(pendiente_1xy*(parte1_x7-
0)+0)
    Momento_parte1_7XZ=(pendiente_3xz*(parte1_x7-
longitud_eje)+0)

Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
    Momento_parte1_max7=Momento_parte1_7
    Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
    Momento_parte2_7XY=(pendiente_2xy*(parte2_x7-
distancia_3)+Momento_1xy)
    Momento_parte2_7XZ=(pendiente_3xz*(parte2_x7-
longitud_eje)+0)

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
    Momento_parte2_max7=Momento_parte2_7
    Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
    Momento_max7=Momento_parte2_max7
    Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end

```



```

end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_2xy*(x8-
distancia_3)+Momento_1xy)
Momento_8XZ=(pendiente_3xz*(x8-longitud_eje)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_2xy*(parte1_x9-
distancia_3)+Momento_1xy)
Momento_parte1_9XZ=(pendiente_3xz*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
distancia_4)+Momento_2xy)
Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_3xy*(x10-
distancia_4)+Momento_2xy)
Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end

for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_3xy*(parte1_x11-
distancia_4)+Momento_2xy)
Momento_parte1_11XZ=(pendiente_3xz*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end
if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)
Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)
Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then
Momento_max13=Momento_13
Xmax_13=x13
end
end
end
end

```

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

```

Reaccion_Axy=((W_radial55*longitud_eje)+(W_radial11*lon
gitud_eje)+(W_radial44*longitud_eje)-
(W_radial55*distancia_5)-(W_radial11*distancia_1)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_5)-
(W_radial11*distancia_5)-
(W_radial44*distancia_5)+(W_radial11*distancia_1)+(W_radi
al44*distancia_4)

```

```

Reaccion_Axz=((W_radial22*longitud_eje)+(W_radial33*lon
gitud_eje)-(W_radial22*distancia_2)-
(W_radial33*distancia_3))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_2
Momento_2xz=(Reaccion_Axz*distancia_3)-
(W_radial22*distancia_3)+(W_radial22*distancia_2)

pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_4-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_5-distancia_4)
pendiente_4xy=(0-Momentos_3xy)/(longitud_eje-
distancia_5)
pendiente_1xz=(Momento_1xz-0)/(distancia_2-0)
pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_3-distancia_2)
pendiente_3xz=(0-Momentos_2xz)/(longitud_eje-
distancia_3)

```

```

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

```

```

for x1=0.0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)

```

```

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end

```

```

end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)

```

```

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

```

```

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end

```

```

for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)

```

```

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end

```

```

if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
end

```

```

if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
end

```

```

for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
Momento_4XZ=(pendiente_1xz*(x4-0)+0)

```

```

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end

```

```

for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)

```

```

Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
end

```

```

        Xmax_parte1_5=parte1_x5
    end
end
    for parte2_x5=distancia_2:0.1:sesion_5
        Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_1)+Momento_1xy)
        Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
distancia_2)+Momento_1xz)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
    if Momento_parte2_5>Momento_parte2_max5 then
        Momento_parte2_max5=Momento_parte2_5
        Xmax_parte2_5=parte2_x5
    end
end
    if Momento_parte2_max5 > Momento_parte1_max5 then
        Momento_max5=Momento_parte2_max5
        Xmax_5= Xmax_parte2_5
end
    if Momento_parte1_max5 > Momento_parte2_max5 then
        Momento_max5=Momento_parte1_max5
        Xmax_5=Xmax_parte1_5
end
    if Momento_parte1_max5 == Momento_parte2_max5 then
        Momento_max5=Momento_parte1_max5
        Xmax_5=Xmax_parte1_5
end
    for x6=sesion_5:0.1:sesion_6
        Momento_6XY=(pendiente_2xy*(x6-
distancia_1)+Momento_1xy)
        Momento_6XZ=(pendiente_2xz*(x6-
distancia_2)+Momento_1xz)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    if Momento_6>Momento_max6 then
        Momento_max6=Momento_6
        Xmax_6=x6
    end
end
    for parte1_x7=sesion_6:0.1:distancia_3
        Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_1)+Momento_1xy)
        Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
distancia_2)+Momento_1xz)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
    if Momento_parte1_7>Momento_parte1_max7 then
        Momento_parte1_max7=Momento_parte1_7
        Xmax_parte1_7=parte1_x7
    end
end
    for parte2_x7=distancia_3:0.1:sesion_7
        Momento_parte2_7XY=(pendiente_2xy*(parte2_x7-
distancia_1)+Momento_1xy)
        Momento_parte2_7XZ=(pendiente_3xz*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
    if Momento_parte2_7>Momento_parte2_max7 then
        Momento_parte2_max7=Momento_parte2_7
        Xmax_parte2_7=parte2_x7
    end
end
    if Momento_parte2_max7 > Momento_parte1_max7 then
        Momento_max7=Momento_parte2_max7
        Xmax_7= Xmax_parte2_7
    end
end
    if Momento_parte1_max7 > Momento_parte2_max7 then
        Momento_max7=Momento_parte1_max7
        Xmax_7=Xmax_parte1_7
    end
end
    if Momento_parte1_max7 == Momento_parte2_max7 then
        Momento_max7=Momento_parte1_max7
        Xmax_7=Xmax_parte1_7
    end
end
    for x8=sesion_7:0.1:sesion_8
        Momento_8XY=(pendiente_2xy*(x8-
distancia_1)+Momento_1xy)
        Momento_8XZ=(pendiente_3xz*(x8-longitud_eje)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    if Momento_8>Momento_max8 then
        Momento_max8=Momento_8
        Xmax_8=x8
    end
end
    for parte1_x9=sesion_8:0.1:distancia_4
        Momento_parte1_9XY=(pendiente_2xy*(parte1_x9-
distancia_1)+Momento_1xy)
        Momento_parte1_9XZ=(pendiente_3xz*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
    if Momento_parte1_9>Momento_parte1_max9 then
        Momento_parte1_max9=Momento_parte1_9
        Xmax_parte1_9=parte1_x9
    end
end
    for parte2_x9=distancia_4:0.1:sesion_9
        Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
distancia_4)+Momento_2xy)
        Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
    if Momento_parte2_9>Momento_parte2_max9 then
        Momento_parte2_max9=Momento_parte2_9
        Xmax_parte2_9=parte2_x9
    end
end
    if Momento_parte2_max9 > Momento_parte1_max9 then
        Momento_max9=Momento_parte2_max9
        Xmax_9= Xmax_parte2_9
    end
end
    if Momento_parte1_max9 > Momento_parte2_max9 then
        Momento_max9=Momento_parte1_max9
        Xmax_9=Xmax_parte1_9
    end
end
    if Momento_parte1_max9 == Momento_parte2_max9 then
        Momento_max9=Momento_parte1_max9
        Xmax_9=Xmax_parte1_9
    end
end
    for x10=sesion_9:0.1:sesion_10
        Momento_10XY=(pendiente_3xy*(x10-
distancia_4)+Momento_2xy)
        Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)

```

```

Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_3xy*(parte1_x11-
distancia_4)+Momento_2xy)
Momento_parte1_11XZ=(pendiente_3xz*(parte1_x11-
longitud_eje)+0)

Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)

Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end
if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)

Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)

Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then

```

```

Momento_max13=Momento_13
Xmax_13=x13
end
end

if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

Reaccion_Axy=((W_radial55*longitud_eje)+(W_radial11*lon
gitud_eje)+(W_radial22*longitud_eje)-
(W_radial55*distancia_5)-(W_radial11*distancia_1)-
(W_radial22*distancia_2))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_5)-
(W_radial11*distancia_5)-
(W_radial22*distancia_5)+(W_radial11*distancia_1)+(W_radi
al22*distancia_2)

Reaccion_Axz=((W_radial33*longitud_eje)+(W_radial44*lon
gitud_eje)-(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_3
Momento_2xz=(Reaccion_Axz*distancia_4)-
(W_radial33*distancia_4)+(W_radial33*distancia_3)

pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_5-distancia_2)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_5)
pendiente_1xz=(Momento_1xz-0)/(distancia_3-0)
pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_4-distancia_3)
pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_4)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

for x1=0.0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)

```

```

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
  if Momento_1>Momento_max1 then
    Momento_max1=Momento_1
    Xmax_1=x1
  end
end
  for x2=sesion_1:0.1:sesion_2
    Momento_2XY=(pendiente_1xy*(x2-0)+0)
    Momento_2XZ=(pendiente_1xz*(x2-0)+0)

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
  if Momento_2>Momento_max2 then
    Momento_max2=Momento_2
    Xmax_2=x2
  end
end
  for parte1_x3=sesion_2:0.1:distancia_1
    Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
    Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
  if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
  end
end
  for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
    Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
  if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
  end
end
  if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
  end
end
  if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
  end
end
  if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
  end
end
  for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
    Momento_4XZ=(pendiente_1xz*(x4-0)+0)

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
  if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
  end
end
  for parte1_x5=sesion_4:0.1:distancia_2
    Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
    Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)

Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
  if Momento_parte1_5>Momento_parte1_max5 then
    Momento_parte1_max5=Momento_parte1_5
    Xmax_parte1_5=parte1_x5
  end
end
  for parte2_x5=distancia_2:0.1:sesion_5
    Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
distancia_2)+Momento_2xy)
    Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)

Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
  if Momento_parte2_5>Momento_parte2_max5 then
    Momento_parte2_max5=Momento_parte2_5
    Xmax_parte2_5=parte2_x5
  end
end
  if Momento_parte2_max5 > Momento_parte1_max5 then
    Momento_max5=Momento_parte2_max5
    Xmax_5= Xmax_parte2_5
  end
end
  if Momento_parte1_max5 > Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
  end
end
  if Momento_parte1_max5 == Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
  end
end
  for x6=sesion_5:0.1:sesion_6
    Momento_6XY=(pendiente_3xy*(x6-
distancia_2)+Momento_2xy)
    Momento_6XZ=(pendiente_1xz*(x6-0)+0)

Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
  if Momento_6>Momento_max6 then
    Momento_max6=Momento_6
    Xmax_6=x6
  end
end
  for parte1_x7=sesion_6:0.1:distancia_3
    Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
distancia_2)+Momento_2xy)
    Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)

Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
  if Momento_parte1_7>Momento_parte1_max7 then
    Momento_parte1_max7=Momento_parte1_7
    Xmax_parte1_7=parte1_x7
  end
end
  for parte2_x7=distancia_3:0.1:sesion_7
    Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_2)+Momento_2xy)
    Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
distancia_3)+Momento_1xz)

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
  if Momento_parte2_7>Momento_parte2_max7 then
    Momento_parte2_max7=Momento_parte2_7
    Xmax_parte2_7=parte2_x7
  end
end

```

```

end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_3xy*(x8-
distancia_2)+Momento_2xy)
Momento_8XZ=(pendiente_2xz*(x8-
distancia_3)+Momento_1xz)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_2)+Momento_2xy)
Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
distancia_3)+Momento_1xz)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
distancia_2)+Momento_2xy)
Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_3xy*(x10-
distancia_2)+Momento_2xy)
Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_3xy*(parte1_x11-
distancia_2)+Momento_2xy)
Momento_parte1_11XZ=(pendiente_3xz*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end
if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)
Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)
Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))

```



```

if Momento_13>Momento_max13 then
    Momento_max13=Momento_13
    Xmax_13=x13
end
end
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

```

Reaccion_Axy=((W_radial33*longitud_eje)+(W_radial11*lon
gitud_eje)+(W_radial22*longitud_eje)-
(W_radial33*distancia_3)-(W_radial11*distancia_1)-
(W_radial22*distancia_2))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)-
(W_radial22*distancia_3)+(W_radial11*distancia_1)+(W_radi
al22*distancia_2)

```

```

Reaccion_Axz=((W_radial44*longitud_eje)+(W_radial55*lon
gitud_eje)-(W_radial44*distancia_4)-
(W_radial55*distancia_5))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_4
Momento_2xz=(Reaccion_Axz*distancia_5)-
(W_radial44*distancia_5)+(W_radial44*distancia_4)

```

```

pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_3-distancia_2)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_3)
pendiente_1xz=(Momento_1xz-0)/(distancia_4-0)
pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_5-distancia_4)
pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_5)

```

```

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

```

```

for x1=0:0.1:sesion_1
    Momento_1XY=(pendiente_1xy*(x1-0)+0)

```

```

Momento_1XZ=(pendiente_1xz*(x1-0)+0)

```

```

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
    Momento_max1=Momento_1
    Xmax_1=x1
end
end

```

```

for x2=sesion_1:0.1:sesion_2
    Momento_2XY=(pendiente_1xy*(x2-0)+0)
    Momento_2XZ=(pendiente_1xz*(x2-0)+0)

```

```

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
    Momento_max2=Momento_2
    Xmax_2=x2
end
end

```

```

for parte1_x3=sesion_2:0.1:distancia_1
    Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
    Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

```

```

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))

```

```

if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
end
end

```

```

for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
    Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)

```

```

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))

```

```

if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
end
end

```

```

if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
end
end

```

```

if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
end

```

```

if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
end

```

```

for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
    Momento_4XZ=(pendiente_1xz*(x4-0)+0)

```

```

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
end
end

```

```

for parte1_x5=sesion_4:0.1:distancia_2

```

```

Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)

Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
distancia_2)+Momento_2xy)
Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)

Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_3xy*(x6-
distancia_2)+Momento_2xy)
Momento_6XZ=(pendiente_1xz*(x6-0)+0)

Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
distancia_2)+Momento_2xy)
Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)

Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_4xy*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7XZ=(pendiente_1xz*(parte2_x7-0)+0)

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_4xy*(x8-longitud_eje)+0)
Momento_8XZ=(pendiente_1xz*(x8-0)+0)

Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_4xy*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9XZ=(pendiente_1xz*(parte1_x9-0)+0)

Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
distancia_4)+Momento_1xz)

Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)
Momento_10XZ=(pendiente_2xz*(x10-
distancia_4)+Momento_1xz)

```



```

Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
distancia_4)+Momento_1xz)

Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)

Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end
if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)

Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)

Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then

```

```

Momento_max13=Momento_13
Xmax_13=x13
end
end
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

Reaccion_Axy=((W_radial44*longitud_eje)+(W_radial11*lon
gitud_eje)+(W_radial22*longitud_eje)-
(W_radial44*distancia_4)-(W_radial11*distancia_1)-
(W_radial22*distancia_2))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_2)-
(W_radial11*distancia_2)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)-
(W_radial22*distancia_4)+(W_radial11*distancia_1)+(W_radi
al22*distancia_2)

Reaccion_Axz=((W_radial33*longitud_eje)+(W_radial55*lon
gitud_eje)-(W_radial33*distancia_3)-
(W_radial55*distancia_5))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_3
Momento_2xz=(Reaccion_Axz*distancia_5)-
(W_radial33*distancia_5)+(W_radial33*distancia_3)

pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_2-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_4-distancia_2)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_4)
pendiente_1xz=(Momento_1xz-0)/(distancia_3-0)
pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_5-distancia_3)
pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_5)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

for x1=0:0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)

```

```

Momento_1XZ=(pendiente_1xz*(x1-0)+0)
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
    Momento_max1=Momento_1
    Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
    Momento_2XY=(pendiente_1xy*(x2-0)+0)
    Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
    Momento_max2=Momento_2
    Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
    Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
    Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
    Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
end
end
if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
end
if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
end
for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
    Momento_4XZ=(pendiente_1xz*(x4-0)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
    Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
    Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
    Momento_parte1_max5=Momento_parte1_5
    Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
    Momento_parte2_5XY=(pendiente_3xy*(parte2_x5-
distancia_2)+Momento_2xy)
    Momento_parte2_5XZ=(pendiente_1xz*(parte2_x5-0)+0)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
    Momento_parte2_max5=Momento_parte2_5
    Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
    Momento_max5=Momento_parte2_max5
    Xmax_5= Xmax_parte2_5
end
end
if Momento_parte1_max5 > Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
    Momento_6XY=(pendiente_3xy*(x6-
distancia_2)+Momento_2xy)
    Momento_6XZ=(pendiente_1xz*(x6-0)+0)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
    Momento_max6=Momento_6
    Xmax_6=x6
end
end
end
for parte1_x7=sesion_6:0.1:distancia_3
    Momento_parte1_7XY=(pendiente_3xy*(parte1_x7-
distancia_2)+Momento_2xy)
    Momento_parte1_7XZ=(pendiente_1xz*(parte1_x7-0)+0)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
    Momento_parte1_max7=Momento_parte1_7
    Xmax_parte1_7=parte1_x7
end
end
end
for parte2_x7=distancia_3:0.1:sesion_7
    Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_2)+Momento_2xy)
    Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
distancia_3)+Momento_1xz)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then

```

```

    Momento_parte2_max7=Momento_parte2_7
    Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_3xy*(x8-
distancia_2)+Momento_2xy)
Momento_8XZ=(pendiente_2xz*(x8-
distancia_3)+Momento_1xz)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_2)+Momento_2xy)
Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
distancia_3)+Momento_1xz)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
distancia_3)+Momento_1xz)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)
Momento_10XZ=(pendiente_2xz*(x10-
distancia_3)+Momento_1xz)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
distancia_3)+Momento_1xz)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end
if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)
Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)

```

```

Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
    if Momento_13>Momento_max13 then
        Momento_max13=Momento_13
        Xmax_13=x13
    end
end
end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

```

Reaccion_Axy=((W_radial22*longitud_eje)+(W_radial44*lon
gitud_eje)+(W_radial55*longitud_eje)-
(W_radial22*distancia_2)-(W_radial44*distancia_4)-
(W_radial55*distancia_5))/(longitud_eje)
    Momento_1xy=Reaccion_Axy*distancia_2
    Momento_2xy=(Reaccion_Axy*distancia_4)-
(W_radial22*distancia_4)+(W_radial22*distancia_2)
    Momento_3xy=(Reaccion_Axy*distancia_5)-
(W_radial22*distancia_5)-
(W_radial44*distancia_5)+(W_radial22*distancia_2)+(W_radi
al44*distancia_4)

```

```

Reaccion_Axz=((W_radial11*longitud_eje)+(W_radial33*lon
gitud_eje)-(W_radial11*distancia_1)-
(W_radial33*distancia_3))/(longitud_eje)
    Momento_1xz=Reaccion_Axz*distancia_1
    Momento_2xz=(Reaccion_Axz*distancia_3)-
(W_radial11*distancia_3)+(W_radial11*distancia_1)

```

```

    pendiente_1xy=(Momento_1xy-0)/(distancia_2-0)
    pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_4-distancia_2)
    pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_5-distancia_4)
    pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_5)
    pendiente_1xz=(Momento_1xz-0)/(distancia_1-0)
    pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_3-distancia_1)
    pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_3)

```

```

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

```

```

for x1=0.0:0.1:sesion_1
    Momento_1XY=(pendiente_1xy*(x1-0)+0)
    Momento_1XZ=(pendiente_1xz*(x1-0)+0)

```

```

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    if Momento_1>Momento_max1 then
        Momento_max1=Momento_1
        Xmax_1=x1
    end
end
    for x2=sesion_1:0.1:sesion_2
        Momento_2XY=(pendiente_1xy*(x2-0)+0)
        Momento_2XZ=(pendiente_1xz*(x2-0)+0)

```

```

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    if Momento_2>Momento_max2 then
        Momento_max2=Momento_2
        Xmax_2=x2
    end
end
    for parte1_x3=sesion_2:0.1:distancia_1
        Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
        Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

```

```

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
    if Momento_parte1_3>Momento_parte1_max3 then
        Momento_parte1_max3=Momento_parte1_3
        Xmax_parte1_3=parte1_x3
    end
end
    for parte2_x3=distancia_1:0.1:sesion_3
        Momento_parte2_3XY=(pendiente_1xy*(parte2_x3-
0)+0)
        Momento_parte2_3XZ=(pendiente_2xz*(parte2_x3-
distancia_1)+Momento_1xz)

```

```

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
    if Momento_parte2_3>Momento_parte2_max3 then
        Momento_parte2_max3=Momento_parte2_3
        Xmax_parte2_3=parte2_x3
    end
end
    if Momento_parte2_max3 > Momento_parte1_max3 then
        Momento_max3=Momento_parte2_max3
        Xmax_3= Xmax_parte2_3
    end

```

```

    if Momento_parte1_max3 > Momento_parte2_max3 then
        Momento_max3=Momento_parte1_max3
        Xmax_3=Xmax_parte1_3
    end
end
    if Momento_parte1_max3 == Momento_parte2_max3 then
        Momento_max3=Momento_parte1_max3
        Xmax_3=Xmax_parte1_3
    end
end
    for x4=sesion_3:0.1:sesion_4
        Momento_4XY=(pendiente_1xy*(x4-0)+0)
        Momento_4XZ=(pendiente_2xz*(x4-
distancia_1)+Momento_1xz)

```

```

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    if Momento_4>Momento_max4 then
        Momento_max4=Momento_4
        Xmax_4=x4
    end

```

```

    end
end
    for parte1_x5=sesion_4:0.1:distancia_2
        Momento_parte1_5XY=(pendiente_1xy*(parte1_x5-
0)+0)
        Momento_parte1_5XZ=(pendiente_2xz*(parte1_x5-
distancia_1)+Momento_1xz)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
        if Momento_parte1_5>Momento_parte1_max5 then
            Momento_parte1_max5=Momento_parte1_5
            Xmax_parte1_5=parte1_x5
        end
    end
    for parte2_x5=distancia_2:0.1:sesion_5
        Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_2)+Momento_1xy)
        Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
distancia_1)+Momento_1xz)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
        if Momento_parte2_5>Momento_parte2_max5 then
            Momento_parte2_max5=Momento_parte2_5
            Xmax_parte2_5=parte2_x5
        end
    end
    if Momento_parte2_max5 > Momento_parte1_max5 then
        Momento_max5=Momento_parte2_max5
        Xmax_5= Xmax_parte2_5
    end
    if Momento_parte1_max5 > Momento_parte2_max5 then
        Momento_max5=Momento_parte1_max5
        Xmax_5=Xmax_parte1_5
    end
    if Momento_parte1_max5 == Momento_parte2_max5 then
        Momento_max5=Momento_parte1_max5
        Xmax_5=Xmax_parte1_5
    end
    for x6=sesion_5:0.1:sesion_6
        Momento_6XY=(pendiente_2xy*(x6-
distancia_2)+Momento_1xy)
        Momento_6XZ=(pendiente_2xz*(x6-
distancia_1)+Momento_1xz)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
        if Momento_6>Momento_max6 then
            Momento_max6=Momento_6
            Xmax_6=x6
        end
    end
    for parte1_x7=sesion_6:0.1:distancia_3
        Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_2)+Momento_1xy)
        Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
distancia_1)+Momento_1xz)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
        if Momento_parte1_7>Momento_parte1_max7 then
            Momento_parte1_max7=Momento_parte1_7
            Xmax_parte1_7=parte1_x7
        end
    end
    for parte2_x7=distancia_3:0.1:sesion_7
        Momento_parte2_7XY=(pendiente_2xy*(parte2_x7-
distancia_2)+Momento_1xy)
        Momento_parte2_7XZ=(pendiente_3xz*(parte2_x7-
longitud_eje)+0)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
        if Momento_parte2_7>Momento_parte2_max7 then
            Momento_parte2_max7=Momento_parte2_7
            Xmax_parte2_7=parte2_x7
        end
    end
    if Momento_parte2_max7 > Momento_parte1_max7 then
        Momento_max7=Momento_parte2_max7
        Xmax_7= Xmax_parte2_7
    end
    if Momento_parte1_max7 > Momento_parte2_max7 then
        Momento_max7=Momento_parte1_max7
        Xmax_7=Xmax_parte1_7
    end
    if Momento_parte1_max7 == Momento_parte2_max7 then
        Momento_max7=Momento_parte1_max7
        Xmax_7=Xmax_parte1_7
    end
    for x8=sesion_7:0.1:sesion_8
        Momento_8XY=(pendiente_2xy*(x8-
distancia_2)+Momento_1xy)
        Momento_8XZ=(pendiente_3xz*(x8-longitud_eje)+0)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
        if Momento_8>Momento_max8 then
            Momento_max8=Momento_8
            Xmax_8=x8
        end
    end
    for parte1_x9=sesion_8:0.1:distancia_4
        Momento_parte1_9XY=(pendiente_2xy*(parte1_x9-
distancia_2)+Momento_1xy)
        Momento_parte1_9XZ=(pendiente_3xz*(parte1_x9-
longitud_eje)+0)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
        if Momento_parte1_9>Momento_parte1_max9 then
            Momento_parte1_max9=Momento_parte1_9
            Xmax_parte1_9=parte1_x9
        end
    end
    for parte2_x9=distancia_4:0.1:sesion_9
        Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
distancia_4)+Momento_2xy)
        Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
        if Momento_parte2_9>Momento_parte2_max9 then
            Momento_parte2_max9=Momento_parte2_9
            Xmax_parte2_9=parte2_x9
        end
    end
    if Momento_parte2_max9 > Momento_parte1_max9 then
        Momento_max9=Momento_parte2_max9
        Xmax_9= Xmax_parte2_9
    end
    if Momento_parte1_max9 > Momento_parte2_max9 then
        Momento_max9=Momento_parte1_max9
    end

```

```

Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_3xy*(x10-
distancia_4)+Momento_2xy)
Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_3xy*(parte1_x11-
distancia_4)+Momento_2xy)
Momento_parte1_11XZ=(pendiente_3xz*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end
if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)
Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12

```

```

Xmax_12=x12
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)
Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then
Momento_max13=Momento_13
Xmax_13=x13
end
end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 <>
ubicacion_4) then
Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial33*lon
gitud_eje)+(W_radial55*longitud_eje)-
(W_radial11*distancia_1)-(W_radial33*distancia_3)-
(W_radial55*distancia_5))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_5)-
(W_radial11*distancia_5)-
(W_radial33*distancia_5)+(W_radial11*distancia_1)+(W_radi
al33*distancia_3)
Reaccion_Axz=((W_radial22*longitud_eje)+(W_radial44*lon
gitud_eje)-(W_radial22*distancia_2)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_2
Momento_2xz=(Reaccion_Axz*distancia_4)-
(W_radial22*distancia_4)+(W_radial22*distancia_2)
pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_5-distancia_3)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_5)
pendiente_1xz=(Momento_1xz-0)/(distancia_2-0)
pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_4-distancia_2)
pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_4)
Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0

```



```

Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

for x1=0.0:0.1:sesion_1
  Momento_1XY=(pendiente_1xy*(x1-0)+0)
  Momento_1XZ=(pendiente_1xz*(x1-0)+0)

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
  if Momento_1>Momento_max1 then
    Momento_max1=Momento_1
    Xmax_1=x1
  end
end

for x2=sesion_1:0.1:sesion_2
  Momento_2XY=(pendiente_1xy*(x2-0)+0)
  Momento_2XZ=(pendiente_1xz*(x2-0)+0)

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
  if Momento_2>Momento_max2 then
    Momento_max2=Momento_2
    Xmax_2=x2
  end
end

for parte1_x3=sesion_2:0.1:distancia_1
  Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
  Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
  if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
  end
end

for parte2_x3=distancia_1:0.1:sesion_3
  Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
  Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
  if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
  end
end

if Momento_parte2_max3 > Momento_parte1_max3 then
  Momento_max3=Momento_parte2_max3
  Xmax_3= Xmax_parte2_3
end

if Momento_parte1_max3 > Momento_parte2_max3 then
  Momento_max3=Momento_parte1_max3
  Xmax_3=Xmax_parte1_3
end

if Momento_parte1_max3 == Momento_parte2_max3 then
  Momento_max3=Momento_parte1_max3
  Xmax_3=Xmax_parte1_3
end

for x4=sesion_3:0.1:sesion_4
  Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
  Momento_4XZ=(pendiente_1xz*(x4-0)+0)

Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
  if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
  end
end

for parte1_x5=sesion_4:0.1:distancia_2
  Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
  Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)

Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
  if Momento_parte1_5>Momento_parte1_max5 then
    Momento_parte1_max5=Momento_parte1_5
    Xmax_parte1_5=parte1_x5
  end
end

for parte2_x5=distancia_2:0.1:sesion_5
  Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_1)+Momento_1xy)
  Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
distancia_2)+Momento_1xz)

Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
  if Momento_parte2_5>Momento_parte2_max5 then
    Momento_parte2_max5=Momento_parte2_5
    Xmax_parte2_5=parte2_x5
  end
end

if Momento_parte2_max5 > Momento_parte1_max5 then
  Momento_max5=Momento_parte2_max5
  Xmax_5= Xmax_parte2_5
end

if Momento_parte1_max5 > Momento_parte2_max5 then
  Momento_max5=Momento_parte1_max5
  Xmax_5=Xmax_parte1_5
end

if Momento_parte1_max5 == Momento_parte2_max5 then
  Momento_max5=Momento_parte1_max5
  Xmax_5=Xmax_parte1_5
end

for x6=sesion_5:0.1:sesion_6
  Momento_6XY=(pendiente_2xy*(x6-
distancia_1)+Momento_1xy)
  Momento_6XZ=(pendiente_2xz*(x6-
distancia_2)+Momento_1xz)

Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
  if Momento_6>Momento_max6 then
    Momento_max6=Momento_6
    Xmax_6=x6
  end
end

for parte1_x7=sesion_6:0.1:distancia_3
  Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_1)+Momento_1xy)
  Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
distancia_2)+Momento_1xz)

Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
  if Momento_parte1_7>Momento_parte1_max7 then
    Momento_parte1_max7=Momento_parte1_7

```

```

        Xmax_parte1_7=parte1_x7
    end
end

    for parte2_x7=distancia_3:0.1:sesion_7
        Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_3)+Momento_2xy)
        Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
distancia_2)+Momento_1xz)

Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
        if Momento_parte2_7>Momento_parte2_max7 then
            Momento_parte2_max7=Momento_parte2_7
            Xmax_parte2_7=parte2_x7
        end
    end
    if Momento_parte2_max7 > Momento_parte1_max7 then
        Momento_max7=Momento_parte2_max7
        Xmax_7= Xmax_parte2_7
    end
end
if Momento_parte1_max7 > Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
    for x8=sesion_7:0.1:sesion_8
        Momento_8XY=(pendiente_3xy*(x8-
distancia_3)+Momento_2xy)
        Momento_8XZ=(pendiente_2xz*(x8-
distancia_2)+Momento_1xz)

Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
        if Momento_8>Momento_max8 then

            Momento_max8=Momento_8
            Xmax_8=x8
        end
    end
    for parte1_x9=sesion_8:0.1:distancia_4
        Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_3)+Momento_2xy)
        Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
distancia_2)+Momento_1xz)

Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
        if Momento_parte1_9>Momento_parte1_max9 then
            Momento_parte1_max9=Momento_parte1_9
            Xmax_parte1_9=parte1_x9
        end
    end
    for parte2_x9=distancia_4:0.1:sesion_9
        Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
distancia_3)+Momento_2xy)
        Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)

Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
        if Momento_parte2_9>Momento_parte2_max9 then
            Momento_parte2_max9=Momento_parte2_9
            Xmax_parte2_9=parte2_x9
        end
    end

        Xmax_parte1_7=parte1_x7
    end
end

    if Momento_parte2_max9 > Momento_parte1_max9 then
        Momento_max9=Momento_parte2_max9
        Xmax_9= Xmax_parte2_9
    end
end
    if Momento_parte1_max9 > Momento_parte2_max9 then
        Momento_max9=Momento_parte1_max9
        Xmax_9=Xmax_parte1_9
    end
end
    if Momento_parte1_max9 == Momento_parte2_max9 then
        Momento_max9=Momento_parte1_max9
        Xmax_9=Xmax_parte1_9
    end
end
    for x10=sesion_9:0.1:sesion_10
        Momento_10XY=(pendiente_3xy*(x10-
distancia_3)+Momento_2xy)
        Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)

Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
        if Momento_10>Momento_max10 then
            Momento_max10=Momento_10
            Xmax_10=x10
        end
    end
end
    for parte1_x11=sesion_10:0.1:distancia_5
        Momento_parte1_11XY=(pendiente_3xy*(parte1_x11-
distancia_3)+Momento_2xy)
        Momento_parte1_11XZ=(pendiente_3xz*(parte1_x11-
longitud_eje)+0)

Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
        if Momento_parte1_11>Momento_parte1_max11 then
            Momento_parte1_max11=Momento_parte1_11
            Xmax_parte1_11=parte1_x11
        end
    end
end
    for parte2_x11=distancia_5:0.1:sesion_11
        Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
        Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)

Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
        if Momento_parte2_11>Momento_parte2_max11 then
            Momento_parte2_max11=Momento_parte2_11
            Xmax_parte2_11=parte2_x11
        end
    end
end
    if Momento_parte2_max11 > Momento_parte1_max11
then
        Momento_max11=Momento_parte2_max11
        Xmax_11= Xmax_parte2_11
    end
end
    if Momento_parte1_max11 > Momento_parte2_max11 then
        Momento_max11=Momento_parte1_max11
        Xmax_11=Xmax_parte1_11
    end
end
    if Momento_parte1_max11 == Momento_parte2_max11
then
        Momento_max11=Momento_parte1_max11
        Xmax_11=Xmax_parte1_11
    end
end

```



```

end
  for x12=sesion_11:0.1:sesion_12
    Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
    Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)

Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
    if Momento_12>Momento_max12 then
      Momento_max12=Momento_12
      Xmax_12=x12
    end
  end
  for x13=sesion_12:0.1:longitud_eje
    Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
    Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)

Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
    if Momento_13>Momento_max13 then
      Momento_max13=Momento_13
      Xmax_13=x13
    end
  end
end

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

Reaccion_Axy=((W_radial22*longitud_eje)+(W_radial33*lon
gitud_eje)+(W_radial44*longitud_eje)-
(W_radial22*distancia_2)-(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
  Momento_1xy=Reaccion_Axy*distancia_2
  Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial22*distancia_3)+(W_radial22*distancia_2)
  Momento_3xy=(Reaccion_Axy*distancia_4)-
(W_radial22*distancia_4)-
(W_radial33*distancia_4)+(W_radial22*distancia_2)+(W_radi
al33*distancia_3)

Reaccion_Axz=((W_radial11*longitud_eje)+(W_radial55*lon
gitud_eje)-(W_radial11*distancia_1)-
(W_radial55*distancia_5))/(longitud_eje)
  Momento_1xz=Reaccion_Axz*distancia_1
  Momento_2xz=(Reaccion_Axz*distancia_5)-
(W_radial11*distancia_5)+(W_radial11*distancia_1)

  pendiente_1xy=(Momento_1xy-0)/(distancia_2-0)
  pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_2)
  pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_4-distancia_3)
  pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_4)
  pendiente_1xz=(Momento_1xz-0)/(distancia_1-0)
  pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_5-distancia_1)
  pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_5)

  Momento_max1=0
  Momento_max2=0

```

```

Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

for x1=0:0:0.1:sesion_1
  Momento_1XY=(pendiente_1xy*(x1-0)+0)
  Momento_1XZ=(pendiente_1xz*(x1-0)+0)

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
  if Momento_1>Momento_max1 then
    Momento_max1=Momento_1
    Xmax_1=x1
  end
end
  for x2=sesion_1:0.1:sesion_2
    Momento_2XY=(pendiente_1xy*(x2-0)+0)
    Momento_2XZ=(pendiente_1xz*(x2-0)+0)

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
  if Momento_2>Momento_max2 then
    Momento_max2=Momento_2
    Xmax_2=x2
  end
end
  for parte1_x3=sesion_2:0.1:distancia_1
    Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
    Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
  if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
  end
end
  for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_1xy*(parte2_x3-
0)+0)
    Momento_parte2_3XZ=(pendiente_2xz*(parte2_x3-
distancia_1)+Momento_1xz)

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
  if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
  end
end
  if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
  end
end
  if Momento_parte1_max3 > Momento_parte2_max3 then

```

```

Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_1xy*(x4-0)+0)
Momento_4XZ=(pendiente_2xz*(x4-
distancia_1)+Momento_1xz)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_1xy*(parte1_x5-
0)+0)
Momento_parte1_5XZ=(pendiente_2xz*(parte1_x5-
distancia_1)+Momento_1xz)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_2)+Momento_1xy)
Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
distancia_1)+Momento_1xz)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_2xy*(x6-
distancia_2)+Momento_1xy)
Momento_6XZ=(pendiente_2xz*(x6-
distancia_1)+Momento_1xz)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_2)+Momento_1xy)
Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
distancia_1)+Momento_1xz)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_3)+Momento_2xy)
Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
distancia_1)+Momento_1xz)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_3xy*(x8-
distancia_3)+Momento_2xy)
Momento_8XZ=(pendiente_2xz*(x8-
distancia_1)+Momento_1xz)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
end
for parte1_x9=sesion_8:0.1:distancia_4
Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_3)+Momento_2xy)
Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
distancia_1)+Momento_1xz)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
end
for parte2_x9=distancia_4:0.1:sesion_9

```

```

Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
distancia_1)+Momento_1xz)

Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end

if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end

for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)
Momento_10XZ=(pendiente_2xz*(x10-
distancia_1)+Momento_1xz)

Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
distancia_1)+Momento_1xz)

Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)

Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end

if Momento_parte2_max11 > Momento_parte1_max11
then

```

```

Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)

Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)

Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then
Momento_max13=Momento_13
Xmax_13=x13
end
end

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

Reaccion_Axy=((W_radial22*longitud_eje)+(W_radial33*lon
gitud_eje)+(W_radial55*longitud_eje)-
(W_radial22*distancia_2)-(W_radial33*distancia_3)-
(W_radial55*distancia_5))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_2
Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial22*distancia_3)+(W_radial22*distancia_2)
Momento_3xy=(Reaccion_Axy*distancia_5)-
(W_radial22*distancia_5)-
(W_radial33*distancia_5)+(W_radial22*distancia_2)+(W_radi
al33*distancia_3)

Reaccion_Axz=((W_radial11*longitud_eje)+(W_radial44*lon
gitud_eje)-(W_radial11*distancia_1)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_1
Momento_2xz=(Reaccion_Axz*distancia_4)-
(W_radial11*distancia_4)+(W_radial11*distancia_1)

pendiente_1xy=(Momento_1xy-0)/(distancia_2-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_2)

```

```

pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_5-distancia_3)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_5)
pendiente_1xz=(Momento_1xz-0)/(distancia_1-0)
pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_4-distancia_1)
pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_4)

```

```

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

```

```

for x1=0.0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)
Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)
Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)
Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))
if Momento_parte1_3>Momento_parte1_max3 then
Momento_parte1_max3=Momento_parte1_3
Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
Momento_parte2_3XY=(pendiente_1xy*(parte2_x3-
0)+0)
Momento_parte2_3XZ=(pendiente_2xz*(parte2_x3-
distancia_1)+Momento_1xz)

```

```

Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
Momento_parte2_max3=Momento_parte2_3
Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
Momento_max3=Momento_parte2_max3
Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
Momento_max3=Momento_parte1_max3
Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0.1:sesion_4
Momento_4XY=(pendiente_1xy*(x4-0)+0)
Momento_4XZ=(pendiente_2xz*(x4-
distancia_1)+Momento_1xz)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
Momento_max4=Momento_4
Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
Momento_parte1_5XY=(pendiente_1xy*(parte1_x5-
0)+0)
Momento_parte1_5XZ=(pendiente_2xz*(parte1_x5-
distancia_1)+Momento_1xz)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
Momento_parte1_max5=Momento_parte1_5
Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_2)+Momento_1xy)
Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
distancia_1)+Momento_1xz)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
Momento_parte2_max5=Momento_parte2_5
Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
Momento_max5=Momento_parte2_max5
Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5
Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
Momento_max5=Momento_parte1_max5

```

```

Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
Momento_6XY=(pendiente_2xy*(x6-
distancia_2)+Momento_1xy)
Momento_6XZ=(pendiente_2xz*(x6-
distancia_1)+Momento_1xz)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
Momento_max6=Momento_6
Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_2)+Momento_1xy)
Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
distancia_1)+Momento_1xz)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
Momento_parte1_max7=Momento_parte1_7
Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_3)+Momento_2xy)
Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
distancia_1)+Momento_1xz)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
Momento_parte2_max7=Momento_parte2_7
Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
Momento_max7=Momento_parte2_max7
Xmax_7= Xmax_parte2_7
end
end
if Momento_parte1_max7 > Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end
if Momento_parte1_max7 == Momento_parte2_max7 then
Momento_max7=Momento_parte1_max7
Xmax_7=Xmax_parte1_7
end
end
for x8=sesion_7:0.1:sesion_8
Momento_8XY=(pendiente_3xy*(x8-
distancia_3)+Momento_2xy)
Momento_8XZ=(pendiente_2xz*(x8-
distancia_1)+Momento_1xz)
Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
Momento_max8=Momento_8
Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4

```

```

Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_3)+Momento_2xy)
Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
distancia_1)+Momento_1xz)
Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
Momento_parte1_max9=Momento_parte1_9
Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
Momento_parte2_9XY=(pendiente_3xy*(parte2_x9-
distancia_3)+Momento_2xy)
Momento_parte2_9XZ=(pendiente_3xz*(parte2_x9-
longitud_eje)+0)
Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
Momento_parte2_max9=Momento_parte2_9
Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
Momento_max9=Momento_parte2_max9
Xmax_9= Xmax_parte2_9
end
end
if Momento_parte1_max9 > Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
if Momento_parte1_max9 == Momento_parte2_max9 then
Momento_max9=Momento_parte1_max9
Xmax_9=Xmax_parte1_9
end
end
for x10=sesion_9:0.1:sesion_10
Momento_10XY=(pendiente_3xy*(x10-
distancia_3)+Momento_2xy)
Momento_10XZ=(pendiente_3xz*(x10-longitud_eje)+0)
Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
Momento_max10=Momento_10
Xmax_10=x10
end
end
end
for parte1_x11=sesion_10:0.1:distancia_5
Momento_parte1_11XY=(pendiente_3xy*(parte1_x11-
distancia_3)+Momento_2xy)
Momento_parte1_11XZ=(pendiente_3xz*(parte1_x11-
longitud_eje)+0)
Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
Momento_parte1_max11=Momento_parte1_11
Xmax_parte1_11=parte1_x11
end
end
end
for parte2_x11=distancia_5:0.1:sesion_11
Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)

```

```

Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)

Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
Momento_parte2_max11=Momento_parte2_11
Xmax_parte2_11=parte2_x11
end
end

if Momento_parte2_max11 > Momento_parte1_max11
then
Momento_max11=Momento_parte2_max11
Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
if Momento_parte1_max11 == Momento_parte2_max11
then
Momento_max11=Momento_parte1_max11
Xmax_11=Xmax_parte1_11
end
for x12=sesion_11:0.1:sesion_12
Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)

Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
Momento_max12=Momento_12
Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)

Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then
Momento_max13=Momento_13
Xmax_13=x13
end
end
end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

Reaccion_Axy=((W_radial11*longitud_eje)+(W_radial33*lon
gitud_eje)+(W_radial44*longitud_eje)-
(W_radial11*distancia_1)-(W_radial33*distancia_3)-
(W_radial44*distancia_4))/(longitud_eje)
Momento_1xy=Reaccion_Axy*distancia_1
Momento_2xy=(Reaccion_Axy*distancia_3)-
(W_radial11*distancia_3)+(W_radial11*distancia_1)
Momento_3xy=(Reaccion_Axy*distancia_4)-
(W_radial11*distancia_4)-
(W_radial33*distancia_4)+(W_radial11*distancia_1)+(W_radi
al33*distancia_3)

```

```

Reaccion_Axz=((W_radial22*longitud_eje)+(W_radial55*lon
gitud_eje)-(W_radial22*distancia_2)-
(W_radial55*distancia_5))/(longitud_eje)
Momento_1xz=Reaccion_Axz*distancia_2
Momento_2xz=(Reaccion_Axz*distancia_5)-
(W_radial22*distancia_5)+(W_radial22*distancia_2)

pendiente_1xy=(Momento_1xy-0)/(distancia_1-0)
pendiente_2xy=(Momento_2xy-
Momento_1xy)/(distancia_3-distancia_1)
pendiente_3xy=(Momento_3xy-
Momento_2xy)/(distancia_4-distancia_3)
pendiente_4xy=(0-Momento_3xy)/(longitud_eje-
distancia_4)
pendiente_1xz=(Momento_1xz-0)/(distancia_2-0)
pendiente_2xz=(Momento_2xz-
Momento_1xz)/(distancia_5-distancia_2)
pendiente_3xz=(0-Momento_2xz)/(longitud_eje-
distancia_5)

Momento_max1=0
Momento_max2=0
Momento_parte1_max3=0
Momento_parte2_max3=0
Momento_max4=0
Momento_parte1_max5=0
Momento_parte2_max5=0
Momento_max6=0
Momento_parte1_max7=0
Momento_parte2_max7=0
Momento_max8=0
Momento_parte1_max9=0
Momento_parte2_max9=0
Momento_max10=0
Momento_parte1_max11=0
Momento_parte2_max11=0
Momento_max12=0
Momento_max13=0

for x1=0:0:0.1:sesion_1
Momento_1XY=(pendiente_1xy*(x1-0)+0)
Momento_1XZ=(pendiente_1xz*(x1-0)+0)

Momento_1=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
if Momento_1>Momento_max1 then
Momento_max1=Momento_1
Xmax_1=x1
end
end
for x2=sesion_1:0.1:sesion_2
Momento_2XY=(pendiente_1xy*(x2-0)+0)
Momento_2XZ=(pendiente_1xz*(x2-0)+0)

Momento_2=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
if Momento_2>Momento_max2 then
Momento_max2=Momento_2
Xmax_2=x2
end
end
for parte1_x3=sesion_2:0.1:distancia_1
Momento_parte1_3XY=(pendiente_1xy*(parte1_x3-
0)+0)
Momento_parte1_3XZ=(pendiente_1xz*(parte1_x3-0)+0)

Momento_parte1_3=sqrt(((Momento_parte1_3XY)^2)+((Mom
ento_parte1_3XZ)^2))

```



```

if Momento_parte1_3>Momento_parte1_max3 then
    Momento_parte1_max3=Momento_parte1_3
    Xmax_parte1_3=parte1_x3
end
end
for parte2_x3=distancia_1:0.1:sesion_3
    Momento_parte2_3XY=(pendiente_2xy*(parte2_x3-
distancia_1)+Momento_1xy)
    Momento_parte2_3XZ=(pendiente_1xz*(parte2_x3-0)+0)
Momento_parte2_3=sqrt(((Momento_parte2_3XY)^2)+((Mom
ento_parte2_3XZ)^2))
if Momento_parte2_3>Momento_parte2_max3 then
    Momento_parte2_max3=Momento_parte2_3
    Xmax_parte2_3=parte2_x3
end
end
if Momento_parte2_max3 > Momento_parte1_max3 then
    Momento_max3=Momento_parte2_max3
    Xmax_3= Xmax_parte2_3
end
if Momento_parte1_max3 > Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
if Momento_parte1_max3 == Momento_parte2_max3 then
    Momento_max3=Momento_parte1_max3
    Xmax_3=Xmax_parte1_3
end
for x4=sesion_3:0.1:sesion_4
    Momento_4XY=(pendiente_2xy*(x4-
distancia_1)+Momento_1xy)
    Momento_4XZ=(pendiente_1xz*(x4-0)+0)
Momento_4=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
if Momento_4>Momento_max4 then
    Momento_max4=Momento_4
    Xmax_4=x4
end
end
for parte1_x5=sesion_4:0.1:distancia_2
    Momento_parte1_5XY=(pendiente_2xy*(parte1_x5-
distancia_1)+Momento_1xy)
    Momento_parte1_5XZ=(pendiente_1xz*(parte1_x5-0)+0)
Momento_parte1_5=sqrt(((Momento_parte1_5XY)^2)+((Mom
ento_parte1_5XZ)^2))
if Momento_parte1_5>Momento_parte1_max5 then
    Momento_parte1_max5=Momento_parte1_5
    Xmax_parte1_5=parte1_x5
end
end
for parte2_x5=distancia_2:0.1:sesion_5
    Momento_parte2_5XY=(pendiente_2xy*(parte2_x5-
distancia_1)+Momento_1xy)
    Momento_parte2_5XZ=(pendiente_2xz*(parte2_x5-
distancia_2)+Momento_1xz)
Momento_parte2_5=sqrt(((Momento_parte2_5XY)^2)+((Mom
ento_parte2_5XZ)^2))
if Momento_parte2_5>Momento_parte2_max5 then
    Momento_parte2_max5=Momento_parte2_5
    Xmax_parte2_5=parte2_x5
end
end
if Momento_parte2_max5 > Momento_parte1_max5 then
    Momento_max5=Momento_parte2_max5

```

```

Xmax_5= Xmax_parte2_5
end
if Momento_parte1_max5 > Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
if Momento_parte1_max5 == Momento_parte2_max5 then
    Momento_max5=Momento_parte1_max5
    Xmax_5=Xmax_parte1_5
end
for x6=sesion_5:0.1:sesion_6
    Momento_6XY=(pendiente_2xy*(x6-
distancia_1)+Momento_1xy)
    Momento_6XZ=(pendiente_2xz*(x6-
distancia_2)+Momento_1xz)
Momento_6=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
if Momento_6>Momento_max6 then
    Momento_max6=Momento_6
    Xmax_6=x6
end
end
for parte1_x7=sesion_6:0.1:distancia_3
    Momento_parte1_7XY=(pendiente_2xy*(parte1_x7-
distancia_1)+Momento_1xy)
    Momento_parte1_7XZ=(pendiente_2xz*(parte1_x7-
distancia_2)+Momento_1xz)
Momento_parte1_7=sqrt(((Momento_parte1_7XY)^2)+((Mom
ento_parte1_7XZ)^2))
if Momento_parte1_7>Momento_parte1_max7 then
    Momento_parte1_max7=Momento_parte1_7
    Xmax_parte1_7=parte1_x7
end
end
for parte2_x7=distancia_3:0.1:sesion_7
    Momento_parte2_7XY=(pendiente_3xy*(parte2_x7-
distancia_3)+Momento_2xy)
    Momento_parte2_7XZ=(pendiente_2xz*(parte2_x7-
distancia_2)+Momento_1xz)
Momento_parte2_7=sqrt(((Momento_parte2_7XY)^2)+((Mom
ento_parte2_7XZ)^2))
if Momento_parte2_7>Momento_parte2_max7 then
    Momento_parte2_max7=Momento_parte2_7
    Xmax_parte2_7=parte2_x7
end
end
if Momento_parte2_max7 > Momento_parte1_max7 then
    Momento_max7=Momento_parte2_max7
    Xmax_7= Xmax_parte2_7
end
if Momento_parte1_max7 > Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
if Momento_parte1_max7 == Momento_parte2_max7 then
    Momento_max7=Momento_parte1_max7
    Xmax_7=Xmax_parte1_7
end
for x8=sesion_7:0.1:sesion_8
    Momento_8XY=(pendiente_3xy*(x8-
distancia_3)+Momento_2xy)
    Momento_8XZ=(pendiente_2xz*(x8-
distancia_2)+Momento_1xz)

```

```

Momento_8=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
if Momento_8>Momento_max8 then
    Momento_max8=Momento_8
    Xmax_8=x8
end
end
for parte1_x9=sesion_8:0.1:distancia_4
    Momento_parte1_9XY=(pendiente_3xy*(parte1_x9-
distancia_3)+Momento_2xy)
    Momento_parte1_9XZ=(pendiente_2xz*(parte1_x9-
distancia_2)+Momento_1xz)

Momento_parte1_9=sqrt(((Momento_parte1_9XY)^2)+((Mom
ento_parte1_9XZ)^2))
if Momento_parte1_9>Momento_parte1_max9 then
    Momento_parte1_max9=Momento_parte1_9
    Xmax_parte1_9=parte1_x9
end
end
for parte2_x9=distancia_4:0.1:sesion_9
    Momento_parte2_9XY=(pendiente_4xy*(parte2_x9-
longitud_eje)+0)
    Momento_parte2_9XZ=(pendiente_2xz*(parte2_x9-
distancia_2)+Momento_1xz)

Momento_parte2_9=sqrt(((Momento_parte2_9XY)^2)+((Mom
ento_parte2_9XZ)^2))
if Momento_parte2_9>Momento_parte2_max9 then
    Momento_parte2_max9=Momento_parte2_9
    Xmax_parte2_9=parte2_x9
end
end
if Momento_parte2_max9 > Momento_parte1_max9 then
    Momento_max9=Momento_parte2_max9
    Xmax_9= Xmax_parte2_9
end
if Momento_parte1_max9 > Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end
if Momento_parte1_max9 == Momento_parte2_max9 then
    Momento_max9=Momento_parte1_max9
    Xmax_9=Xmax_parte1_9
end
end
for x10=sesion_9:0.1:sesion_10
    Momento_10XY=(pendiente_4xy*(x10-longitud_eje)+0)
    Momento_10XZ=(pendiente_2xz*(x10-
distancia_2)+Momento_1xz)

Momento_10=sqrt(((Momento_10XY)^2)+((Momento_10XZ)
^2))
if Momento_10>Momento_max10 then
    Momento_max10=Momento_10
    Xmax_10=x10
end
end
for parte1_x11=sesion_10:0.1:distancia_5
    Momento_parte1_11XY=(pendiente_4xy*(parte1_x11-
longitud_eje)+0)
    Momento_parte1_11XZ=(pendiente_2xz*(parte1_x11-
distancia_2)+Momento_1xz)

Momento_parte1_11=sqrt(((Momento_parte1_11XY)^2)+((M
omento_parte1_11XZ)^2))
if Momento_parte1_11>Momento_parte1_max11 then
    Momento_parte1_max11=Momento_parte1_11
    Xmax_parte1_11=parte1_x11
end
end
for parte2_x11=distancia_5:0.1:sesion_11
    Momento_parte2_11XY=(pendiente_4xy*(parte2_x11-
longitud_eje)+0)
    Momento_parte2_11XZ=(pendiente_3xz*(parte2_x11-
longitud_eje)+0)

Momento_parte2_11=sqrt(((Momento_parte2_11XY)^2)+((M
omento_parte2_11XZ)^2))
if Momento_parte2_11>Momento_parte2_max11 then
    Momento_parte2_max11=Momento_parte2_11
    Xmax_parte2_11=parte2_x11
end
end
if Momento_parte2_max11 > Momento_parte1_max11
then
    Momento_max11=Momento_parte2_max11
    Xmax_11= Xmax_parte2_11
end
if Momento_parte1_max11 > Momento_parte2_max11 then
    Momento_max11=Momento_parte1_max11
    Xmax_11=Xmax_parte1_11
end
if Momento_parte1_max11 == Momento_parte2_max11
then
    Momento_max11=Momento_parte1_max11
    Xmax_11=Xmax_parte1_11
end
end
for x12=sesion_11:0.1:sesion_12
    Momento_12XY=(pendiente_4xy*(x12-longitud_eje)+0)
    Momento_12XZ=(pendiente_3xz*(x12-longitud_eje)+0)

Momento_12=sqrt(((Momento_12XY)^2)+((Momento_12XZ)
^2))
if Momento_12>Momento_max12 then
    Momento_max12=Momento_12
    Xmax_12=x12
end
end
for x13=sesion_12:0.1:longitud_eje
    Momento_13XY=(pendiente_4xy*(x13-longitud_eje)+0)
    Momento_13XZ=(pendiente_3xz*(x13-longitud_eje)+0)

Momento_13=sqrt(((Momento_13XY)^2)+((Momento_13XZ)
^2))
if Momento_13>Momento_max13 then
    Momento_max13=Momento_13
    Xmax_13=x13
end
end
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 ==
ubicacion_4) then
x=0:0.0.1:longitud_eje
function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then
    s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then

```



```

s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_2xy*(x-distancia_1)+Momento_1xy
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_2xy*(x-distancia_1)+Momento_1xy
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_2xy*(x-distancia_1)+Momento_1xy
elseif x>=distancia_2 & x < sesion_5 then
s=pendiente_3xy*(x-distancia_2)+Momento_2xy
elseif x>=sesion_5 & x < sesion_6 then
s=pendiente_3xy*(x-distancia_2)+Momento_2xy
elseif x>=sesion_6 & x < distancia_3 then
s=pendiente_3xy*(x-distancia_2)+Momento_2xy
elseif x>=distancia_3 & x < sesion_7 then
s=pendiente_4xy*(x-distancia_3)+Momento_3xy
elseif x>=sesion_7 & x < sesion_8 then
s=pendiente_4xy*(x-distancia_3)+Momento_3xy
elseif x>=sesion_8 & x < distancia_4 then
s=pendiente_4xy*(x-distancia_3)+Momento_3xy
elseif x>=distancia_4 & x < sesion_9 then
s=pendiente_5xy*(x-distancia_4)+Momento_4xy
elseif x>=sesion_9 & x < sesion_10 then
s=pendiente_5xy*(x-distancia_4)+Momento_4xy
elseif x>=sesion_10 & x < distancia_5 then
s=pendiente_5xy*(x-distancia_4)+Momento_4xy
elseif x>=distancia_5 & x < sesion_11 then
s=pendiente_6xy*(x-distancia_5)+Momento_5xy
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_6xy*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_6xy*(x-longitud_eje)+0
else
s=0
end

```

endfunction

```

plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")

```

end

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 ==
ubicacion_3 & ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

```

x=0:0.1:longitud_eje

```

```

function s=MomentoXY1(x)

```

```

if x>=0 & x < sesion_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_1xy*(x-0)+0

```

```

elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_5xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=(pendiente_5xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
s=(pendiente_5xy*(x-longitud_eje)+0)
else
s=0
end

```

endfunction

```

function s=MomentoXZ1(x)

```

```

if x>=0 & x < sesion_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_4 & x < distancia_2 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_2xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_2xz*(x-longitud_eje)+0

```

```

else
    s=0
end
endfunction

function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then
        s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
    elseif x>=distancia_1 & x < sesion_3 then
        s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
    elseif x>=sesion_3 & x < sesion_4 then
        s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x < distancia_2 then
        s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
    elseif x>=distancia_2 & x < sesion_5 then
        s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < distancia_3 then
        s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
    elseif x>=distancia_3 & x < sesion_7 then
        s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
    elseif x>=sesion_7 & x < sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x < distancia_4 then
        s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
    elseif x>=distancia_4 & x < sesion_9 then
        s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
    elseif x>=sesion_9 & x < sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x < distancia_5 then
        s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
    elseif x>=distancia_5 & x < sesion_11 then
        s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
    elseif x>=sesion_11 & x < sesion_12 then
        s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
    elseif x>=sesion_12 & x < longitud_eje then
        s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
    else
        s=0
    end
end

```

endfunction

```

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")

```

end

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

x=0:0.1:longitud_eje

function s=MomentoXY1(x)

```

if x>=0 & x < sesion_1 then
    s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
    s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
    s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_3 & x < sesion_4 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_4 & x < distancia_2 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_2 & x < sesion_5 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_5 & x < sesion_6 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_6 & x < distancia_3 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_3 & x < sesion_7 then
    s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_7 & x < sesion_8 then
    s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_8 & x < distancia_4 then
    s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=distancia_4 & x < sesion_9 then
    s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=sesion_9 & x < sesion_10 then
    s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=sesion_10 & x < distancia_5 then
    s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=distancia_5 & x < sesion_11 then
    s=(pendiente_5xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
    s=(pendiente_5xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
    s=(pendiente_5xy*(x-longitud_eje)+0)
else
    s=0
end

```

endfunction

function s=MomentoXZ1(x)

```

if x>=0 & x < sesion_1 then

```

```

s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_2xy*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_2xy*(x-longitud_eje)+0
else
s=0
end

```

endfunction

function s=Momento_Resultante(x)

```

if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then

```

```

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then

```

```

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then

```

```

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then

```

```

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then

```

```

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then

```

```

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
elseif x>=sesion_7 & x < sesion_8 then
s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then

```

```

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then

```

```

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < distancia_5 then

```

```

s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
elseif x>=distancia_5 & x < sesion_11 then

```

```

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
elseif x>=sesion_11 & x < sesion_12 then
s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
elseif x>=sesion_12 & x < longitud_eje then
s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
else
s=0
end

```

endfunction

plot(x, Momento_Resultante, "g")

plot(x, MomentoXY1, "b")

plot(x, MomentoXZ1, "r")

xlabel("Distancia (pulg)")

ylabel("Momento resultante (Klb*pulg)")

a=gca();

a.grid=[1,2]

title("Diagrama de Momento")

end

```

if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

x=0.0:0.1:longitud_eje

function s=MomentoXY1(x)

```

if x>=0 & x < sesion_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_3 & x < sesion_4 then

```

```

s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_4 & x < distancia_2 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_4xy*(x-distancia_4)+Momento_3xy)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_5xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=(pendiente_5xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
s=(pendiente_5xy*(x-longitud_eje)+0)
else
s=0
end
endfunction

function s=MomentoXZ1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_2 & x < sesion_5 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_5 & x < sesion_6 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_6 & x < distancia_3 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_2xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_2xz*(x-longitud_eje)+0
else
s=0
end
endfunction

function s=Momento_Resultante(x)
if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then
s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then
s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then
s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then
s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then
s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then
s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
elseif x>=sesion_7 & x < sesion_8 then
s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then
s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then
s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < distancia_5 then
s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
elseif x>=distancia_5 & x < sesion_11 then
s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
elseif x>=sesion_11 & x < sesion_12 then
s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
elseif x>=sesion_12 & x < longitud_eje then
s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
else
s=0
end
endfunction

```

```

end

endfunction

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")

end

if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 <>
ubicacion_4) then
x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_2xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_2xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_4 & x < distancia_2 then
s=(pendiente_2xy*(x-distancia_2)+Momento_2xy)
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_4xy*(x-distancia_5)+Momento_4xy)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_4xy*(x-distancia_5)+Momento_4xy)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_4xy*(x-distancia_5)+Momento_4xy)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_4xy*(x-distancia_5)+Momento_4xy)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_4xy*(x-distancia_5)+Momento_4xy)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_4xy*(x-distancia_5)+Momento_4xy)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_5xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=(pendiente_5xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
s=(pendiente_5xy*(x-longitud_eje)+0)
else
s=0
end
endfunction
function s=MomentoXZ1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then

```

```

s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_2 & x < sesion_5 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_5 & x < sesion_6 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_6 & x < distancia_3 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_3 & x < sesion_7 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_7 & x < sesion_8 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_8 & x < distancia_4 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_2xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_2xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_2xz*(x-longitud_eje)+0
else
s=0
end
endfunction
function s=Momento_Resultante(x)
if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then
s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then
s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then
s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then
s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then
s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then

```



```

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
    elseif x>=sesion_7 & x < sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x < distancia_4 then

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
    elseif x>=distancia_4 & x < sesion_9 then

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
    elseif x>=sesion_9 & x < sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x < distancia_5 then

s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
    elseif x>=distancia_5 & x < sesion_11 then

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
    elseif x>=sesion_11 & x < sesion_12 then
        s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
    elseif x>=sesion_12 & x < longitud_eje then
        s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
    else
        s=0
    end
endfunction

```

endfunction

```

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

```

if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 <>
ubicacion_4) then
x=0.0:0.1:longitud_eje

```

```

function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_2xy)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_2xy)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_2xy)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)

```

```

    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_3xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_4xy*(x-distancia_4)+Momento_4xy)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_4xy*(x-distancia_4)+Momento_4xy)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < distancia_5 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=distancia_5 & x < sesion_11 then
        s=(pendiente_5xy*(x-longitud_eje)+0)
    elseif x>=sesion_11 & x < sesion_12 then
        s=(pendiente_5xy*(x-longitud_eje)+0)
    elseif x>=sesion_12 & x < longitud_eje then
        s=(pendiente_5xy*(x-longitud_eje)+0)
    else
        s=0
    end
endfunction

```

endfunction

```

function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_3 & x < sesion_4 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_4 & x < distancia_2 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_2 & x < sesion_5 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_5 & x < sesion_6 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_6 & x < distancia_3 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_3 & x < sesion_7 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_7 & x < sesion_8 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_8 & x < distancia_4 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_2xz*(x-0)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_2xz*(x-0)+0)
    elseif x>=sesion_10 & x < distancia_5 then
        s=(pendiente_2xz*(x-0)+0)
    elseif x>=distancia_5 & x < sesion_11 then
        s=(pendiente_2xz*(x-longitud_eje)+0)
    elseif x>=sesion_11 & x < sesion_12 then
        s=pendiente_2xz*(x-longitud_eje)+0
    elseif x>=sesion_12 & x < longitud_eje then
        s=pendiente_2xz*(x-longitud_eje)+0
    else
        s=0
    end
endfunction
function s=Momento_Resultante(x)

```

```

if x>=0 & x < sesion_1 then
    s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
    s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^
^2))
    elseif x>=distancia_1 & x< sesion_3 then

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^
^2))
    elseif x>=sesion_3 & x < sesion_4 then
        s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x < distancia_2 then

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^
^2))
    elseif x>=distancia_2 & x < sesion_5 then

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^
^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < distancia_3 then

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^
^2))
    elseif x>=distancia_3 & x < sesion_7 then

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^
^2))
    elseif x>=sesion_7 & x < sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x < distancia_4 then

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^
^2))
    elseif x>=distancia_4 & x < sesion_9 then

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^
^2))
    elseif x>=sesion_9 & x < sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x < distancia_5 then

s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11X
Z)^2))
    elseif x>=distancia_5 & x < sesion_11 then

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11X
Z)^2))
    elseif x>=sesion_11 & x < sesion_12 then
        s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
    elseif x>=sesion_12 & x < longitud_eje then
        s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
    else
        s=0
    end
endfunction

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")

```

```

a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end
if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 ==
ubicacion_4) then
x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x< sesion_3 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_3 & x < sesion_4 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_4 & x < distancia_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_2 & x < sesion_5 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_5 & x < sesion_6 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_6 & x < distancia_3 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_2xy*(x-distancia_3)+Momento_1xy)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_2xy*(x-distancia_3)+Momento_1xy)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_2xy*(x-distancia_3)+Momento_1xy)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
    elseif x>=sesion_10 & x < distancia_5 then
        s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
    elseif x>=distancia_5 & x < sesion_11 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_11 & x < sesion_12 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_12 & x < longitud_eje then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    else
        s=0
    end
endfunction

function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_1 & x< sesion_3 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)

```

```

elseif x>=distancia_2 & x < sesion_5 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_5 & x < sesion_6 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_6 & x < distancia_3 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=distancia_3 & x < sesion_7 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_7 & x < sesion_8 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_8 & x < distancia_4 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=distancia_4 & x < sesion_9 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
    s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
    s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
    s=pendiente_3xz*(x-longitud_eje)+0
else
    s=0
end
endfunction

```

```
function s=Momento_Resultante(x)
```

```

if x>=0 & x < sesion_1 then
    s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
    s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
    elseif x>=distancia_1 & x < sesion_3 then

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
    elseif x>=sesion_3 & x < sesion_4 then
        s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
    elseif x>=sesion_4 & x < distancia_2 then

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
    elseif x>=distancia_2 & x < sesion_5 then

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < distancia_3 then

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
    elseif x>=distancia_3 & x < sesion_7 then

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
    elseif x>=sesion_7 & x < sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x < distancia_4 then

```

```

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
    elseif x>=distancia_4 & x < sesion_9 then

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
    elseif x>=sesion_9 & x < sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x < distancia_5 then

s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
    elseif x>=distancia_5 & x < sesion_11 then

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
    elseif x>=sesion_11 & x < sesion_12 then
        s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
    elseif x>=sesion_12 & x < longitud_eje then
        s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
    else
        s=0
    end
endfunction

```

```
endfunction
```

```

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

```

x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)

```



```

elseif x>=sesion_8 & x < distancia_4 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_4 & x < sesion_9 then
    s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
elseif x>=sesion_9 & x < sesion_10 then
    s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
elseif x>=sesion_10 & x < distancia_5 then
    s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
elseif x>=distancia_5 & x < sesion_11 then
    s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
    s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
    s=(pendiente_4xy*(x-longitud_eje)+0)
else
    s=0
end

endfunction
function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_3 & x < sesion_4 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_4 & x < distancia_2 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
    elseif x>=distancia_3 & x < sesion_7 then
        s=pendiente_3xz*(x-longitud_eje)+0
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < distancia_5 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=distancia_5 & x < sesion_11 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=sesion_11 & x < sesion_12 then
        s=pendiente_3xz*(x-longitud_eje)+0
    elseif x>=sesion_12 & x < longitud_eje then
        s=pendiente_3xz*(x-longitud_eje)+0
    else
        s=0
    end

endfunction
function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then
        s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
        elseif x>=distancia_1 & x < sesion_3 then
            s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
            elseif x>=sesion_3 & x < sesion_4 then
                s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
            elseif x>=sesion_4 & x < distancia_2 then
                s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
                elseif x>=distancia_2 & x < sesion_5 then
                    s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
                    elseif x>=sesion_5 & x < sesion_6 then
                        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
                    elseif x>=sesion_6 & x < distancia_3 then
                        s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
                        elseif x>=distancia_3 & x < sesion_7 then
                            s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
                            elseif x>=sesion_7 & x < sesion_8 then
                                s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
                            elseif x>=sesion_8 & x < distancia_4 then
                                s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
                                elseif x>=distancia_4 & x < sesion_9 then
                                    s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
                                    elseif x>=sesion_9 & x < sesion_10 then
                                        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
                                    elseif x>=sesion_10 & x < distancia_5 then
                                        s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
                                        elseif x>=distancia_5 & x < sesion_11 then
                                            s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
                                            elseif x>=sesion_11 & x < sesion_12 then
                                                s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
                                            elseif x>=sesion_12 & x < longitud_eje then
                                                s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
                                            else
                                                s=0
                                            end
                                        endfunction
                                        plot(x, Momento_Resultante, "g")
                                        plot(x, MomentoXZ1, "b")
                                        plot(x, MomentoXY1, "r")
                                        xlabel("Distancia (pulg)")
                                        ylabel("Momento resultante (Klb*pulg)")
                                        a=gca();
                                        a.grid=[1,2]
                                        title("Diagrama de Momento")
                                        end

```

```

if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

```

```

x=0.0:0.1:longitud_eje

```

```

function s=MomentoXY1(x)

```

```

if x>=0 & x < sesion_1 then

```

```

s=pendiente_1xy*(x-0)+0

```

```

elseif x>=sesion_1 & x < sesion_2 then

```

```

s=pendiente_1xy*(x-0)+0

```

```

elseif x>=sesion_2 & x < distancia_1 then

```

```

s=pendiente_1xy*(x-0)+0

```

```

elseif x>=distancia_1 & x < sesion_3 then

```

```

s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)

```

```

elseif x>=sesion_3 & x < sesion_4 then

```

```

s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)

```

```

elseif x>=sesion_4 & x < distancia_2 then

```

```

s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)

```

```

elseif x>=distancia_2 & x < sesion_5 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=sesion_5 & x < sesion_6 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=sesion_6 & x < distancia_3 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=distancia_3 & x < sesion_7 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=sesion_7 & x < sesion_8 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=sesion_8 & x < distancia_4 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=distancia_4 & x < sesion_9 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=sesion_9 & x < sesion_10 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=sesion_10 & x < distancia_5 then

```

```

s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)

```

```

elseif x>=distancia_5 & x < sesion_11 then

```

```

s=(pendiente_4xy*(x-longitud_eje)+0)

```

```

elseif x>=sesion_11 & x < sesion_12 then

```

```

s=(pendiente_4xy*(x-longitud_eje)+0)

```

```

elseif x>=sesion_12 & x < longitud_eje then

```

```

s=(pendiente_4xy*(x-longitud_eje)+0)

```

```

else

```

```

s=0

```

```

end

```

```

endfunction

```

```

function s=MomentoXZ1(x)

```

```

if x>=0 & x < sesion_1 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=sesion_1 & x < sesion_2 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=sesion_2 & x < distancia_1 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=distancia_1 & x < sesion_3 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=sesion_3 & x < sesion_4 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=sesion_4 & x < distancia_2 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=distancia_2 & x < sesion_5 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=sesion_5 & x < sesion_6 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=sesion_6 & x < distancia_3 then

```

```

s=pendiente_1xz*(x-0)+0

```

```

elseif x>=distancia_3 & x < sesion_7 then

```

```

s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)

```

```

elseif x>=sesion_7 & x < sesion_8 then

```

```

s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)

```

```

elseif x>=sesion_8 & x < distancia_4 then

```

```

s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)

```

```

elseif x>=distancia_4 & x < sesion_9 then

```

```

s=(pendiente_3xz*(x-longitud_eje)+0)

```

```

elseif x>=sesion_9 & x < sesion_10 then

```

```

s=(pendiente_3xz*(x-longitud_eje)+0)

```

```

elseif x>=sesion_10 & x < distancia_5 then

```

```

s=(pendiente_3xz*(x-longitud_eje)+0)

```

```

elseif x>=distancia_5 & x < sesion_11 then

```

```

s=(pendiente_3xz*(x-longitud_eje)+0)

```

```

elseif x>=sesion_11 & x < sesion_12 then

```

```

s=pendiente_3xz*(x-longitud_eje)+0

```

```

elseif x>=sesion_12 & x < longitud_eje then

```

```

s=pendiente_3xz*(x-longitud_eje)+0

```

```

else

```

```

s=0

```

```

end

```

```

endfunction

```

```

function s=Momento_Resultante(x)

```

```

if x>=0 & x < sesion_1 then

```

```

s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))

```

```

elseif x>=sesion_1 & x < sesion_2 then

```

```

s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))

```

```

elseif x>=sesion_2 & x < distancia_1 then

```

```

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))

```

```

elseif x>=distancia_1 & x < sesion_3 then

```

```

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))

```

```

elseif x>=sesion_3 & x < sesion_4 then

```

```

s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))

```

```

elseif x>=sesion_4 & x < distancia_2 then

```

```

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))

```

```

elseif x>=distancia_2 & x < sesion_5 then

```

```

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))

```

```

elseif x>=sesion_5 & x < sesion_6 then

```

```

s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))

```

```

elseif x>=sesion_6 & x < distancia_3 then

```

```

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))

```

```

elseif x>=distancia_3 & x < sesion_7 then

```

```

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))

```

```

elseif x>=sesion_7 & x < sesion_8 then

```

```

s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))

```

```

elseif x>=sesion_8 & x < distancia_4 then

```

```

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))

```

```

elseif x>=distancia_4 & x < sesion_9 then

```

```

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))

```

```

elseif x>=sesion_9 & x < sesion_10 then

```

```

        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
        elseif x>=sesion_10 & x < distancia_5 then

s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
        elseif x>=distancia_5 & x < sesion_11 then

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
        elseif x>=sesion_11 & x < sesion_12 then
            s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
        elseif x>=sesion_12 & x < longitud_eje then
            s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
        else
            s=0
        end
endfunction

```

endfunction

```

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

```

if (ubicacion_1 == ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 ==
ubicacion_4) then

```

```

x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_10 & x < distancia_5 then

```

```

        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=distancia_5 & x < sesion_11 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_11 & x < sesion_12 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_12 & x < longitud_eje then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    else
        s=0
    end
endfunction

```

endfunction

```

function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_3 & x < sesion_4 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_4 & x < distancia_2 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_2 & x < sesion_5 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_5 & x < sesion_6 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_6 & x < distancia_3 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_3 & x < sesion_7 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_7 & x < sesion_8 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_8 & x < distancia_4 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_2xz*(x-distancia_4)+Momento_1xz)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_2xz*(x-distancia_4)+Momento_1xz)
    elseif x>=sesion_10 & x < distancia_5 then
        s=(pendiente_2xz*(x-distancia_4)+Momento_1xz)
    elseif x>=distancia_5 & x < sesion_11 then
        s=(pendiente_3xz*(x-longitud_eje)+0)
    elseif x>=sesion_11 & x < sesion_12 then
        s=pendiente_3xz*(x-longitud_eje)+0
    elseif x>=sesion_12 & x < longitud_eje then
        s=pendiente_3xz*(x-longitud_eje)+0
    else
        s=0
    end
endfunction

```

endfunction

```

function s=Momento_Resultante(x)
    if x>=0 & x < sesion_1 then
        s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
    elseif x>=sesion_1 & x < sesion_2 then
        s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
    elseif x>=sesion_2 & x < distancia_1 then

```

```

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
        elseif x>=distancia_1 & x < sesion_3 then

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))

```

```

elseif x>=sesion_3 & x < sesion_4 then
    s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)
^2))
    elseif x>=distancia_2 & x < sesion_5 then

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)
^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < distancia_3 then

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)
^2))
    elseif x>=distancia_3 & x < sesion_7 then

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)
^2))
    elseif x>=sesion_7 & x < sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x < distancia_4 then

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)
^2))
    elseif x>=distancia_4 & x < sesion_9 then

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)
^2))
    elseif x>=sesion_9 & x < sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x < distancia_5 then

s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11X
Z)^2))
    elseif x>=distancia_5 & x < sesion_11 then

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11X
Z)^2))
    elseif x>=sesion_11 & x < sesion_12 then
        s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
    elseif x>=sesion_12 & x < longitud_eje then
        s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
    else
        s=0
    end
endfunction

```

endfunction

```

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

```

if (ubicacion_1 == ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 <>
ubicacion_4) then
x=0.0:0.1:longitud_eje

```

```

function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then
    s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
    s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
    s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_3 & x < sesion_4 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_4 & x < distancia_2 then
    s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_2 & x < sesion_5 then
    s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_5 & x < sesion_6 then
    s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_6 & x < distancia_3 then
    s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=distancia_3 & x < sesion_7 then
    s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_7 & x < sesion_8 then
    s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=sesion_8 & x < distancia_4 then
    s=(pendiente_3xy*(x-distancia_2)+Momento_2xy)
elseif x>=distancia_4 & x < sesion_9 then
    s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
    s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
    s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
    s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
    s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
    s=(pendiente_4xy*(x-longitud_eje)+0)
else
    s=0
end
endfunction

```

endfunction

```

function s=MomentoXZ1(x)
if x>=0 & x < sesion_1 then
    s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
    s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
    s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
    s=pendiente_1xz*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
    s=pendiente_1xz*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
    s=pendiente_1xz*(x-0)+0
elseif x>=distancia_2 & x < sesion_5 then
    s=pendiente_1xz*(x-0)+0
elseif x>=sesion_5 & x < sesion_6 then
    s=pendiente_1xz*(x-0)+0
elseif x>=sesion_6 & x < distancia_3 then
    s=pendiente_1xz*(x-0)+0
elseif x>=distancia_3 & x < sesion_7 then
    s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
elseif x>=sesion_7 & x < sesion_8 then
    s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
elseif x>=sesion_8 & x < distancia_4 then

```

```

s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_2xz*(x-distancia_3)+Momento_1xz)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_3xz*(x-longitud_eje)+0
else
s=0
end
endfunction
function s=Momento_Resultante(x)
if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then
s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then
s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then
s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then
s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then
s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then
s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
elseif x>=sesion_7 & x < sesion_8 then
s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then
s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then
s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < distancia_5 then
s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))

```

```

elseif x>=distancia_5 & x < sesion_11 then
s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
elseif x>=sesion_11 & x < sesion_12 then
s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
elseif x>=sesion_12 & x < longitud_eje then
s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
else
s=0
end
endfunction
plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end
if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 ==
ubicacion_4) then
x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_3xy*(x-distancia_4)+Momento_2xy)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then

```



```

s=(pendiente_4xy*(x-longitud_eje)+0)
else
s=0
end

```

```
endfunction
```

```

function s=MomentoXZ1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_4 & x < distancia_2 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=distancia_3 & x < sesion_7 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_7 & x < sesion_8 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_8 & x < distancia_4 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=distancia_4 & x < sesion_9 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_9 & x < sesion_10 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_10 & x < distancia_5 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_3xz*(x-longitud_eje)+0
else
s=0
end
endfunction

```

```

function s=Momento_Resultante(x)
if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then
s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then
s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then

```

```

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then

```

```

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then

```

```

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then

```

```

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
elseif x>=sesion_7 & x < sesion_8 then
s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then

```

```

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then

```

```

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < distancia_5 then

```

```

s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
elseif x>=distancia_5 & x < sesion_11 then

```

```

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
elseif x>=sesion_11 & x < sesion_12 then
s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
elseif x>=sesion_12 & x < longitud_eje then
s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
else
s=0
end

```

```
endfunction
```

```

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

```

```

x=0:0.01:longitud_eje
function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then

```

```

s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_4 & x < distancia_2 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
s=(pendiente_4xy*(x-longitud_eje)+0)
else
s=0
end
endfunction

function s=MomentoXZ1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=distancia_4 & x < sesion_9 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_9 & x < sesion_10 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_10 & x < sesion_12 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_3xz*(x-longitud_eje)+0
else
s=0
end
endfunction

function s=Momento_Resultante(x)
if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then
s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then
s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then
s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then
s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then
s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then
s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
elseif x>=sesion_7 & x < sesion_8 then
s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then
s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then
s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < distancia_5 then
s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
elseif x>=distancia_5 & x < sesion_11 then

```

```

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11X
Z)^2))
elseif x>=sesion_11 & x < sesion_12 then
s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
elseif x>=sesion_12 & x < longitud_eje then
s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
else
s=0
end

```

endfunction

```

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 <> ubicacion_1 & ubicacion_4 ==
ubicacion_2 & ubicacion_5 == ubicacion_1 & ubicacion_5 <>
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 <>
ubicacion_4) then
x=0.0:0.1:longitud_eje

```

function s=MomentoXY1(x)

```

if x>=0 & x < sesion_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
s=(pendiente_4xy*(x-longitud_eje)+0)

```

```

else
s=0
end

```

endfunction

function s=MomentoXZ1(x)

```

if x>=0 & x < sesion_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_4 & x < distancia_2 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_3xz*(x-longitud_eje)+0
else
s=0
end

```

endfunction

function s=Momento_Resultante(x)

```

if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then

```

```

s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)
^2))
elseif x>=distancia_1 & x < sesion_3 then

```

```

s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)
^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then

```



```

s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)
^2))
    elseif x>=distancia_2 & x < sesion_5 then

s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)
^2))
    elseif x>=sesion_5 & x < sesion_6 then
        s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
    elseif x>=sesion_6 & x < distancia_3 then

s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)
^2))
    elseif x>=distancia_3 & x < sesion_7 then

s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)
^2))
    elseif x>=sesion_7 & x < sesion_8 then
        s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
    elseif x>=sesion_8 & x < distancia_4 then

s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)
^2))
    elseif x>=distancia_4 & x < sesion_9 then

s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)
^2))
    elseif x>=sesion_9 & x < sesion_10 then
        s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
    elseif x>=sesion_10 & x < distancia_5 then

s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11X
Z)^2))
    elseif x>=distancia_5 & x < sesion_11 then

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11X
Z)^2))
    elseif x>=sesion_11 & x < sesion_12 then
        s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
    elseif x>=sesion_12 & x < longitud_eje then
        s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
    else
        s=0
    end
endfunction

```

endfunction

```

plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

```

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 == ubicacion_3
& ubicacion_1 <> ubicacion_3 & ubicacion_4 <>
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 == ubicacion_3 & ubicacion_5 <>
ubicacion_4) then

```

```

x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
    if x>=0 & x < sesion_1 then

```

```

        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_3 & x < sesion_4 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=sesion_4 & x < distancia_2 then
        s=pendiente_1xy*(x-0)+0
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xy*(x-distancia_2)+Momento_1xy)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=distancia_4 & x < sesion_9 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=sesion_9 & x < sesion_10 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=sesion_10 & x < distancia_5 then
        s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
    elseif x>=distancia_5 & x < sesion_11 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_11 & x < sesion_12 then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    elseif x>=sesion_12 & x < longitud_eje then
        s=(pendiente_4xy*(x-longitud_eje)+0)
    else
        s=0
    end
endfunction

```

endfunction

```

function s=MomentoXZ1(x)
    if x>=0 & x < sesion_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_1 & x < sesion_2 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=sesion_2 & x < distancia_1 then
        s=pendiente_1xz*(x-0)+0
    elseif x>=distancia_1 & x < sesion_3 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=sesion_3 & x < sesion_4 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=sesion_4 & x < distancia_2 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=distancia_2 & x < sesion_5 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=sesion_5 & x < sesion_6 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=sesion_6 & x < distancia_3 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=distancia_3 & x < sesion_7 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=sesion_7 & x < sesion_8 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=sesion_8 & x < distancia_4 then
        s=(pendiente_2xz*(x-distancia_1)+Momento_1xz)
    elseif x>=distancia_4 & x < sesion_9 then

```

```

s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_3xz*(x-longitud_eje)+0
else
s=0
end

endfunction

function s=Momento_Resultante(x)
if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then
s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then
s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then
s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))
elseif x>=distancia_2 & x < sesion_5 then
s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then
s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then
s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
elseif x>=sesion_7 & x < sesion_8 then
s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then
s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then
s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < distancia_5 then
s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
elseif x>=distancia_5 & x < sesion_11 then

```

```

s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
elseif x>=sesion_11 & x < sesion_12 then
s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
elseif x>=sesion_12 & x < longitud_eje then
s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
else
s=0
end

endfunction
plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

if (ubicacion_1 <> ubicacion_2 & ubicacion_2 <> ubicacion_3
& ubicacion_1 == ubicacion_3 & ubicacion_4 ==
ubicacion_3 & ubicacion_4 == ubicacion_1 & ubicacion_4 <>
ubicacion_2 & ubicacion_5 <> ubicacion_1 & ubicacion_5 ==
ubicacion_2 & ubicacion_5 <> ubicacion_3 & ubicacion_5 <>
ubicacion_4) then
x=0.0:0.1:longitud_eje
function s=MomentoXY1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xy*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xy*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_3 & x < sesion_4 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_4 & x < distancia_2 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xy*(x-distancia_1)+Momento_1xy)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_3xy*(x-distancia_3)+Momento_2xy)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=(pendiente_4xy*(x-longitud_eje)+0)
elseif x>=sesion_12 & x < longitud_eje then
s=(pendiente_4xy*(x-longitud_eje)+0)
else

```

```

s=0
end

endfunction

function s=MomentoXZ1(x)
if x>=0 & x < sesion_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_1 & x < sesion_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_2 & x < distancia_1 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_1 & x < sesion_3 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_3 & x < sesion_4 then
s=pendiente_1xz*(x-0)+0
elseif x>=sesion_4 & x < distancia_2 then
s=pendiente_1xz*(x-0)+0
elseif x>=distancia_2 & x < sesion_5 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_5 & x < sesion_6 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_6 & x < distancia_3 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=distancia_3 & x < sesion_7 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_7 & x < sesion_8 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_8 & x < distancia_4 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=distancia_4 & x < sesion_9 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_9 & x < sesion_10 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=sesion_10 & x < distancia_5 then
s=(pendiente_2xz*(x-distancia_2)+Momento_1xz)
elseif x>=distancia_5 & x < sesion_11 then
s=(pendiente_3xz*(x-longitud_eje)+0)
elseif x>=sesion_11 & x < sesion_12 then
s=pendiente_3xz*(x-longitud_eje)+0
elseif x>=sesion_12 & x < longitud_eje then
s=pendiente_3xz*(x-longitud_eje)+0
else
s=0
end
endfunction

function s=Momento_Resultante(x)
if x>=0 & x < sesion_1 then
s=sqrt(((Momento_1XY)^2)+((Momento_1XZ)^2))
elseif x>=sesion_1 & x < sesion_2 then
s=sqrt(((Momento_2XY)^2)+((Momento_2XZ)^2))
elseif x>=sesion_2 & x < distancia_1 then
s=sqrt(((Momento_parte1_3XY)^2)+((Momento_parte1_3XZ)^2))
elseif x>=distancia_1 & x < sesion_3 then
s=sqrt(((Momento_parte2_3XY)^2)+((Momento_parte2_3XZ)^2))
elseif x>=sesion_3 & x < sesion_4 then
s=sqrt(((Momento_4XY)^2)+((Momento_4XZ)^2))
elseif x>=sesion_4 & x < distancia_2 then
s=sqrt(((Momento_parte1_5XY)^2)+((Momento_parte1_5XZ)^2))

```

```

elseif x>=distancia_2 & x < sesion_5 then
s=sqrt(((Momento_parte2_5XY)^2)+((Momento_parte2_5XZ)^2))
elseif x>=sesion_5 & x < sesion_6 then
s=sqrt(((Momento_6XY)^2)+((Momento_6XZ)^2))
elseif x>=sesion_6 & x < distancia_3 then
s=sqrt(((Momento_parte1_7XY)^2)+((Momento_parte1_7XZ)^2))
elseif x>=distancia_3 & x < sesion_7 then
s=sqrt(((Momento_parte2_7XY)^2)+((Momento_parte2_7XZ)^2))
elseif x>=sesion_7 & x < sesion_8 then
s=sqrt(((Momento_8XY)^2)+((Momento_8XZ)^2))
elseif x>=sesion_8 & x < distancia_4 then
s=sqrt(((Momento_parte1_9XY)^2)+((Momento_parte1_9XZ)^2))
elseif x>=distancia_4 & x < sesion_9 then
s=sqrt(((Momento_parte2_9XY)^2)+((Momento_parte2_9XZ)^2))
elseif x>=sesion_9 & x < sesion_10 then
s=sqrt(((Momento_10XY)^2)+((Momento_10XZ)^2))
elseif x>=sesion_10 & x < distancia_5 then
s=sqrt(((Momento_parte1_11XY)^2)+((Momento_parte1_11XZ)^2))
elseif x>=distancia_5 & x < sesion_11 then
s=sqrt(((Momento_parte2_11XY)^2)+((Momento_parte2_11XZ)^2))
elseif x>=sesion_11 & x < sesion_12 then
s=sqrt(((Momento_12XY)^2)+((Momento_12XZ)^2))
elseif x>=sesion_12 & x < longitud_eje then
s=sqrt(((Momento_13XY)^2)+((Momento_13XZ)^2))
else
s=0
end
endfunction
plot(x, Momento_Resultante, "g")
plot(x, MomentoXZ1, "b")
plot(x, MomentoXY1, "r")
xlabel("Distancia (pulg)")
ylabel("Momento resultante (Klb*pulg)")
a=gca();
a.grid=[1,2]
title("Diagrama de Momento")
end

```

14. 7 IteracionesDeLosDiametros.sci

```

clc
clear all
tol=0.001

for Diametro_eje1=tol:tol:6

Radio_eje1=Diametro_eje1/2

// inercias y momentos polares
I1=(%pi*(((Radio_eje1^2)^4))/4)

```

```

J1=I1*2
sigal=(Momento_max1*Radio_eje1/I1)

//FACTOR DE TAMAÑO
if (Diametro_eje1 < 0.11)
    f_tama1=1
end
if (Diametro_eje1 >= 0.11 & Diametro_eje1 <= 2)
    f_tama1=0.879*Diametro_eje1^-0.107
end
if (Diametro_eje1 >= 3 & Diametro_eje1 <= 10)
    f_tama1=0.91*(Diametro_eje1^(-0.157))
end

Am1=[0 0 0;0 0 0;0 0 0] //tensor de esfuerzo medio//
[Em]=spec(Am1)//Funcion para los autovalores
sigvmm1=(((Em(1)-Em(2))^2)+((Em(2)-Em(3))^2)+((Em(3)-Em(1))^2))/2)^0.5 //Vmises medio
Aa1=[(sigal*f_carga_flexion/f_tama1) 0 0;0 0 0;0 0 0]
//tensor de esfuerzo alternante
[Ea]=spec(Aa1)//Funcion para los autovalores
sigvma1=(((Ea(1)-Ea(2))^2)+((Ea(2)-Ea(3))^2)+((Ea(3)-Ea(1))^2))/2)^0.5 //Vmises alternante

V_misses_med_LN1=[sigvmm1,
Coef_variacion_carga*sigvmm1]
V_misses_alt_LN1=[sigvma1,
Coef_variacion_carga*sigvma1]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
    [Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
D_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

    [Div_Va_SE_1_Cuad,          Desv_Div_Va_SE_1_Cuad]=
    pot(D_Va_SE1,2)
    Division_VaSE1 = [Div_Va_SE_1_Cuad,
Desv_Div_Va_SE_1_Cuad]

    Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)

ASME_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_ASME_1=ASME_1(2)/ASME_1(1)
CRITERIO_F=ASME_1
CF_CRITERIO_F=Cf_ASME_1
end

if criterio_falla == 2 then
    [Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_VaSE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

    Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_VaSE1(1)+Division_VmSut1(1)

```

```

Desv_Suma1_VmSut_VaSE=sqrt((Division_VaSE1(2)/Division_VaSE1(1))^2+(0)^2)

GERBER_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GERBER_1=GERBER_1(2)/GERBER_1(1)
CRITERIO_F=GERBER_1
CF_CRITERIO_F=Cf_GERBER_1
end

if criterio_falla == 3 then
    [Div_Va_SE_1,
Desv_Div_Va_SE_1]=div(V_misses_alt_LN1, SE)
Division_Va_SE1 = [Div_Va_SE_1, Desv_Div_Va_SE_1]

    Division_VmSut1 = [0,0]

Suma1_VmSut_VaSE=Division_Va_SE1(1)+Division_VmSut1(1)
Desv_Suma1_VmSut_VaSE=sqrt((Division_Va_SE1(2)/Division_Va_SE1(1))^2+(0)^2)
GOODMAN_1=[Suma1_VmSut_VaSE,
Desv_Suma1_VmSut_VaSE]
Cf_GOODMAN_1=GOODMAN_1(2)/GOODMAN_1(1)
CRITERIO_F=GOODMAN_1
CF_CRITERIO_F=Cf_GOODMAN_1
end

MEDCnormal_1=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_1=CF_CRITERIO_F

[Prob_1,Conf_1]=cdfnor("PQ",0,abs(MEDCnormal_1),DESVnormal_1)

DIAMETRO_EJE1=0

if Prob_1<=1.5*10^-6 & Prob_1>=0.5*10^-6 then
    DIAMETRO_EJE1=Diametro_eje1
    break;
end
disp("El diametro del tramo 1 es")
disp(DIAMETRO_EJE1)

for Diametro_eje2=tol:tol:6
    Radio_eje2=Diametro_eje2/2

    I2=(%pi*(((Radio_eje2*2)^4))/4)/inercia
    J2=I2*2

    // calculo de esfuerzos
    siga2=(Momento_max2*Radio_eje2/I2) // esfuerzo flector maximo
    //FACTOR DE TAMAÑO
    if (Diametro_eje2 < 0.11)
        f_tama2=1
    end
    if (Diametro_eje2 >= 0.11 & Diametro_eje2 <= 2)
        f_tama2=0.879*Diametro_eje2^-0.107
    end
    if (Diametro_eje2 >= 3 & Diametro_eje2 <= 10)
        f_tama2=0.91*(Diametro_eje2^(-0.157))
    end

    //concentracion de esfuerzo

```

```
h2=Radio_eje2-Radio_eje1
relacion2=h2/Radio_muesca
```

```
//Kt para esfuerzos de flexión (Ktf)
```

```
//0.1 <= h/r <= 2.0
```

```
C2_f1=0.947+1.206*sqrt(h2/Radio_muesca)-
0.131*(h2/Radio_muesca)
C2_f2=0.022-
3.405*sqrt(h2/Radio_muesca)+0.915*(h2/Radio_muesca)
C2_f3=0.869+1.777*sqrt(h2/Radio_muesca)-
0.555*(h2/Radio_muesca)
C2_f4=-0.810+0.422*sqrt(h2/Radio_muesca)-
0.260*(h2/Radio_muesca)
```

```
//2.0 <= h/r <= 20
```

```
C2_f11=1.232+0.832*sqrt(h2/Radio_muesca)-
0.008*(h2/Radio_muesca)
C2_f22=-3.813+0.968*sqrt(h2/Radio_muesca)-
0.260*(h2/Radio_muesca)
C2_f33=7.423-
4.868*sqrt(h2/Radio_muesca)+0.869*(h2/Radio_muesca)
C2_f44=-3.839+3.070*sqrt(h2/Radio_muesca)-
0.600*(h2/Radio_muesca)
```

```
if (relacion2 >= 0.1 & relacion2 <= 2)
```

```
    C2_F1=C2_f1
    C2_F2=C2_f2
    C2_F3=C2_f3
    C2_F4=C2_f4
```

```
    Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/
Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
```

```
end
```

```
if (relacion2 > 2 & relacion2 <= 20)
```

```
    C2_F1=C2_f11
    C2_F2=C2_f22
    C2_F3=C2_f33
    C2_F4=C2_f44
```

```
    Kt2_f=C2_F1+C2_F2*(2*h2/Diametro_eje2)+C2_F3*(2*h2/
Diametro_eje2)^2+C2_F4*(2*h2/Diametro_eje2)^3
```

```
end
```

```
if relacion2 < 0.1 then
```

```
    Kt2_f=1
```

```
end
```

```
if relacion2 > 20 then
```

```
    Kt2_f=1
```

```
end
```

```
KT2_F=Kt2_f
```

```
//Factor de concentración de esfuerzo de fatiga (Kf_a - axial),
(Kf_f- flexión), (Kf_t-torsión)
```

```
constante_reemplazo_f2=(2*(KT2_F-
1)/KT2_F)*(RAIZ_DE_A/sqrt(Radio_muesca))
Kf2_f=KT2_F/((1+constante_reemplazo_f2))
```

```
Am2=[0 0 0; 0 0 0; 0 0 0] //tensor de esfuerzo medio
```

```
[Em2]=spec(Am2)//Funcion para los autovalores
```

```
sigvmm2=(((Em2(1)-Em2(2))^2)+((Em2(2)-
Em2(3))^2)+((Em2(3)-Em2(1))^2))/2)^0.5 //Vmises medio
```

```
Aa2=[(sigma2*f_carga_flexion*Kf2_f/f_tama2) 0 0; 0 0; 0 0 0]
```

```
//tensor de esfuerzo alternante
```

```
[Ea2]=spec(Aa2)//Funcion para los autovalores
```

```
sigvma2=(((Ea2(1)-Ea2(2))^2)+((Ea2(2)-
Ea2(3))^2)+((Ea2(3)-Ea2(1))^2))/2)^0.5 //Vmises alternante
```

```
V_misses_med_LN2=[sigvmm2,
Coef_variacion_carga*sigvmm2]
```

```
V_misses_alt_LN2=[sigvma2,
Coef_variacion_carga*sigvma2]
```

```
[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)
```

```
SE=[Media_Se, Desv_Se]
```

```
Sut=[Valor_medio_sut,Desv_sut]
```

```
if criterio_falla == 1 then
```

```
    [Div_Va_SE_2,
```

```
    Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
```

```
    D_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]
```

```
    [Div_Va_SE_2_Cuad, Desv_Div_Va_SE_2_Cuad]=
pot(D_Va_SE2,2)
```

```
    Division_VaSE2 = [Div_Va_SE_2_Cuad,
Desv_Div_Va_SE_2_Cuad]
```

```
Division_VmSut2 = [0,0]
```

```
Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2
(1)
```

```
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio
n_VaSE2(1))^2+(0)^2)
```

```
ASME_2=[Suma2_VmSut_VaSE,
```

```
Desv_Suma1_VmSut_VaSE]
```

```
Cf_ASME_2=ASME_2(2)/ASME_2(1)
```

```
CRITERIO_F=ASME_2
```

```
CF_CRITERIO_F=Cf_ASME_2
```

```
end
```

```
if criterio_falla == 2 then
```

```
    [Div_Va_SE_2,
```

```
    Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
```

```
    Division_VaSE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]
```

```
Division_VmSut2 = [0,0]
```

```
Suma2_VmSut_VaSE=Division_VaSE2(1)+Division_VmSut2
(1)
```

```
Desv_Suma2_VmSut_VaSE=sqrt((Division_VaSE2(2)/Divisio
n_VaSE2(1))^2+(0)^2)
```

```
GERBER_2=[Suma2_VmSut_VaSE,
```

```
Desv_Suma2_VmSut_VaSE]
```

```
Cf_GERBER_2=GERBER_2(2)/GERBER_2(1)
```

```
CRITERIO_F=GERBER_2
```

```
CF_CRITERIO_F=Cf_GERBER_2
```

```
end
```

```
if criterio_falla == 3 then
```

```
    [Div_Va_SE_2,
```

```
    Desv_Div_Va_SE_2]=div(V_misses_alt_LN2, SE)
```

```
    Division_Va_SE2 = [Div_Va_SE_2, Desv_Div_Va_SE_2]
```

```
Division_VmSut2 = [0,0]
```

```
Suma2_VmSut_VaSE=Division_Va_SE2(1)+Division_VmSut
2(1)
```

```
Desv_Suma2_VmSut_VaSE=sqrt((Division_Va_SE2(2)/Divisi
on_Va_SE2(1))^2+(0)^2)
```

```

GOODMAN_2=[Suma2_VmSut_VaSE,
Desv_Suma2_VmSut_VaSE]
Cf_GOODMAN_2=GOODMAN_2(2)/GOODMAN_2(1)
CRITERIO_F=GOODMAN_2
CF_CRITERIO_F=Cf_GOODMAN_2
end

MEDCnormal_2=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_2=CF_CRITERIO_F

[Prob_2,Conf_2]=cdfnorr("PQ",0,abs(MEDCnormal_2),DESV
normal_2)

DIAMETRO_EJE2=0

if Prob_2<=1.5*10^-6 & Prob_2>=0.5*10^-6 then
    DIAMETRO_EJE2=Diametro_eje2
    break;
end
end

disp("El diametro del tramo 2 es")
disp(DIAMETRO_EJE2)

for Diametro_eje3=tol:tol:6

Radio_eje3=Diametro_eje3/2 // Radio del eje en el tramo del
engrane 1

Torque3=W_tangencial1*((Radio engrane1)-Radio_eje3) //
torque generado por el primer engrane
I3=(%pi*(((Radio_eje3^2)^4)/4))/momento de inercia
J3=I3*3 // momento polar de inercia
// calculo de esfuerzos para tramo 3
siga3=(Momento_max3*Radio_eje3/I3) // esfuerzo flector
maximo
taotormed3=(Torque3*Radio_eje3/J3) // esfuerzo cortante
torsor alternante
taotoralt3=(Torque3*Radio_eje3/J3) // esfuerzo cortante torsor
medio

//FACTOR DE TAMAÑO

if (Diametro_eje3 < 0.11)
    f_tama3=1
end

if (Diametro_eje3 >= 0.11 & Diametro_eje3 <= 2)
    f_tama3=0.879*Diametro_eje3^-0.107
end

if (Diametro_eje3 >= 3 & Diametro_eje3 <= 10)
    f_tama3=0.91*(Diametro_eje3^(-0.157))
end

//factor de concentracion de esfuerzo
h3=Radio_eje3-Radio_eje2
relacion3=h3/Radio_muesca2

//0.1<=h/r<=2.0
C3_a1=0.926+(1.157*sqrt(h3/Radio_muesca2))-
(0.099*(h3/Radio_muesca2))
C3_a2=0.012-
3.036*sqrt(h3/Radio_muesca2)+0.961*(h3/Radio_muesca2)
C3_a3=-0.302+3.977*sqrt(h3/Radio_muesca2)-
1.744*(h3/Radio_muesca2)

```

```

C3_a4=0.365-
2.098*sqrt(h3/Radio_muesca2)+0.878*(h3/Radio_muesca2)

//2.0<=h/r<=20
C3_a11=1.2+0.860*sqrt(h3/Radio_muesca2)-
0.022*(h3/Radio_muesca2)
C3_a22=-1.805-0.346*sqrt(h3/Radio_muesca2)-
0.038*(h3/Radio_muesca2)
C3_a33=2.198-
0.486*sqrt(h3/Radio_muesca2)+0.165*(h3/Radio_muesca2)
C3_a44=-0.593-0.028*sqrt(h3/Radio_muesca2)-
0.106*(h3/Radio_muesca2)

if (relacion3>=0.1 & relacion3 <= 2)
    C3_A1=C3_a1
    C3_A2=C3_a2
    C3_A3=C3_a3
    C3_A4=C3_a4
    Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end

if (relacion3 > 2 & relacion3 <= 20)
    C3_A1=C3_a11
    C3_A2=C3_a22
    C3_A3=C3_a33
    C3_A4=C3_a44
    Kt3_a=C3_A1+C3_A2*(2*h3/Diametro_eje3)+C3_A3*(2*h3/
Diametro_eje3)^2+C3_A4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.1 then
    Kt3_a=1
end
if relacion3 > 20 then
    Kt3_a=0
end

KT3_A=Kt3_a

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C3_f1=0.947+1.206*sqrt(h3/Radio_muesca2)-
0.131*(h3/Radio_muesca2)
C3_f2=0.022-
3.405*sqrt(h3/Radio_muesca2)+0.915*(h3/Radio_muesca2)
C3_f3=0.869+1.777*sqrt(h3/Radio_muesca2)-
0.555*(h3/Radio_muesca2)
C3_f4=-0.810+0.422*sqrt(h3/Radio_muesca2)-
0.260*(h3/Radio_muesca2)

//2.0<=h/r<=20
C3_f11=1.232+0.832*sqrt(h3/Radio_muesca2)-
0.008*(h3/Radio_muesca2)
C3_f22=-3.813+0.968*sqrt(h3/Radio_muesca2)-
0.260*(h3/Radio_muesca2)
C3_f33=7.423-
4.868*sqrt(h3/Radio_muesca2)+0.869*(h3/Radio_muesca2)
C3_f44=-3.839+3.070*sqrt(h3/Radio_muesca2)-
0.600*(h3/Radio_muesca2)

if (relacion3 >=0.1 & relacion3 <= 2)
    C3_F1=C3_f1
    C3_F2=C3_f2
    C3_F3=C3_f3
    C3_F4=C3_f4

```



```

Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end

if (relacion3 > 2 & relacion3 <= 20)
    C3_F1=C3_f11
    C3_F2=C3_f22
    C3_F3=C3_f33
    C3_F4=C3_f44
    Kt3_f=C3_F1+C3_F2*(2*h3/Diametro_eje3)+C3_F3*(2*h3/
    Diametro_eje3)^2+C3_F4*(2*h3/Diametro_eje3)^3
end

if relacion3 < 0.1 then
    Kt3_f=1
end

if relacion3 > 20 then
    Kt3_f=1
end
KT3_F=Kt3_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C3_t1=0.905+0.783*sqrt(h3/Radio_muesca2)-
0.075*(h3/Radio_muesca2)
C3_t2=-0.437-
1.969*sqrt(h3/Radio_muesca2)+0.553*(h3/Radio_muesca2)
C3_t3=1.557+1.073*sqrt(h3/Radio_muesca2)-
0.578*(h3/Radio_muesca2)
C3_t4=-1.061+0.171*sqrt(h3/Radio_muesca2)-
0.086*(h3/Radio_muesca2)

if (relacion3 >=0.25 & relacion3 <= 4.0)
    C3_T1=C3_t1
    C3_T2=C3_t2
    C3_T3=C3_t3
    C3_T4=C3_t4
    Kt3_t=C3_T1+C3_T2*(2*h3/Diametro_eje3)+C3_T3*(2*h3/
    Diametro_eje3)^2+C3_T4*(2*h3/Diametro_eje3)^3
end
if relacion3 < 0.25 then
    Kt3_t=1
end
if relacion3 > 4.0 then
    Kt3_t=1
end
KT3_T=Kt3_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f- flexión), (Kf_t-torsión)
constante_reemplazo_a3=(2*(KT3_A-
1)/KT3_A)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_a=KT3_A/((1+constante_reemplazo_a3))

constante_reemplazo_f3=(2*(KT3_F-
1)/KT3_F)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_f=KT3_F/((1+constante_reemplazo_f3))

constante_reemplazo_t3=(2*(KT3_T-
1)/KT3_T)*(RAIZ_DE_A/sqrt(Radio_muesca2))
Kf3_t=KT3_T/((1+constante_reemplazo_t3))

//presion por interferencia
m3=(((2*Radio_engrane1)^2-
Diametro_eje3^2))*(Diametro_eje3^2)

```

```

n3=(2*Radio_engrane1)
PP3=(E*Interferen_maza_eje_max)/(2*Diametro_eje3^3)*(
m3/n3)

Am3=[(PP3*f_carga_axial*Kf3_a/f_tama3)
(taotormed3*f_carga_torsion*Kf3_t/f_tama3)
0;(taotormed3*f_carga_torsion*Kf3_t/f_tama3) 0 0;0 0 0]
//tensor de esfuerzo medio
[Em3]=spec(Am3)
sigvmm3=(((Em3(1)-Em3(2))^2)+(Em3(2)-
Em3(3))^2)+(Em3(3)-Em3(1))^2)/2^0.5//calculo esfuerzo
Von Mises medio
Aa3=[(siga3*f_carga_flexion*Kf3_f/f_tama3)
(taotoralt3*f_carga_torsion*Kf3_t/f_tama3) 0;0
(taotoralt3*f_carga_torsion*Kf3_t/f_tama3) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea3]=spec(Aa3)
sigvma3=(((Ea3(1)-Ea3(2))^2)+(Ea3(2)-
Ea3(3))^2)+(Ea3(3)-Ea3(1))^2)/2^0.5//calculo esfuerzo Von
Mises medio

V_misses_med_LN3=[sigvmm3,
Coef_variacion_carga*sigvmm3]
V_misses_alt_LN3=[sigvma3,
Coef_variacion_carga*sigvma3]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
    [Div_Va_SE_3,
    Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
    D_Va_SE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

    [Div_Va_SE_3_Cuad, Desv_Div_Va_SE_3_Cuad]=
    pot(D_Va_SE3,2)
    Division_VaSE3 = [Div_Va_SE_3_Cuad,
    Desv_Div_Va_SE_3_Cuad]

    [Div_Vm_Sut_3,
    Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
    D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

    [Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
    pot(D_Vm_Sut3,2)
    Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
    Desv_Div_Vm_Sut_3_Cuad]

    [Suma3_VmSut,Desv_Suma3_VmSut]
    =suma(Division_VaSE3, Division_VmSut3)

    ASME_3=[Suma3_VmSut,Desv_Suma3_VmSut]
    Cf_ASME_3=ASME_3(2)/ASME_3(1)
    CRITERIO_F=ASME_3
    CF_CRITERIO_F=Cf_ASME_3
end

if criterio_falla == 2 then
    [Div_Va_SE_3,
    Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
    Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]

    [Div_Vm_Sut_3,
    Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
    D_Vm_Sut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

```

```

[Div_Vm_Sut_3_Cuad, Desv_Div_Vm_Sut_3_Cuad]=
pot(D_Vm_Sut3,2)
Division_VmSut3 = [Div_Vm_Sut_3_Cuad,
Desv_Div_Vm_Sut_3_Cuad]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)

GERBER_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GERBER_3=GERBER_3(2)/GERBER_3(1)
CRITERIO_F=GERBER_3
CF_CRITERIO_F=Cf_GERBER_3
end

if criterio_falla == 3 then
[Div_Va_SE_3,
Desv_Div_Va_SE_3]=div(V_misses_alt_LN3, SE)
Division_VaSE3 = [Div_Va_SE_3, Desv_Div_Va_SE_3]
[Div_Vm_Sut_3,
Desv_Div_Vm_Sut_3]=div(V_misses_med_LN3, Sut)
Division_VmSut3 = [Div_Vm_Sut_3, Desv_Div_Vm_Sut_3]

[Suma3_VmSut,Desv_Suma3_VmSut]
=suma(Division_VaSE3, Division_VmSut3)
GOODMAN_3=[Suma3_VmSut,Desv_Suma3_VmSut]
Cf_GOODMAN_3=GOODMAN_3(2)/GOODMAN_3(1)
CRITERIO_F=GOODMAN_3
CF_CRITERIO_F=Cf_GOODMAN_3
end

MEDCnormal_3=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESNormal_3=CF_CRITERIO_F

[Prob_3,Conf_3]=cdfnorr("PQ",0,abs(MEDCnormal_3),DESNormal_3)

DIAMETRO_EJE3=0

if Prob_3<=1.5*10^-6 & Prob_3>=0.5*10^-6 then
DIAMETRO_EJE3=Diametro_eje3
break;
end
end
disp("El diametro del tramo 3 es ")
disp(DIAMETRO_EJE3)
for Diametro_eje4=tol:tol:6

Radio_eje4=Diametro_eje4/2
Torque4=W_tangencial1*((Radio engrane1)-Radio_eje3)
I4=(%pi*(((Radio_eje4*2)^4)/4))/inercia
J4=I4*2
// calculo de esfuerzos
siga4=(Momento_max4*Radio_eje4/I4) // esfuerzo flexor
maximo
taotormed4=(Torque4*Radio_eje4/J4) // esfuerzo cortante
torsor maximo
taotortal4=(Torque4*Radio_eje4/J4)

//FACTOR DE TAMAÑO
if (Diametro_eje4 < 0.11)
f_tama4=1
end
if (Diametro_eje4 >= 0.11 & Diametro_eje4 <= 2)
f_tama4=0.879*Diametro_eje4^-0.107
end

if (Diametro_eje4 >= 3 & Diametro_eje4 <= 10)
f_tama4=0.91*(Diametro_eje4^(-0.157))
end

//factor de concentracion de esfuerzo
h4=Radio_eje4-Radio_eje3
relacion4=h4/Radio_muesca3

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C4_f1=0.947+1.206*sqrt(h4/Radio_muesca3)-
0.131*(h4/Radio_muesca3)
C4_f2=0.022-
3.405*sqrt(h4/Radio_muesca3)+0.915*(h4/Radio_muesca3)
C4_f3=0.869+1.777*sqrt(h4/Radio_muesca3)-
0.555*(h4/Radio_muesca3)
C4_f4=-0.810+0.422*sqrt(h4/Radio_muesca3)-
0.260*(h4/Radio_muesca3)

//2.0<=h/r<=20
C4_f11=1.232+0.832*sqrt(h4/Radio_muesca3)-
0.008*(h4/Radio_muesca3)
C4_f22=-3.813+0.968*sqrt(h4/Radio_muesca3)-
0.260*(h4/Radio_muesca3)
C4_f33=7.423-
4.868*sqrt(h4/Radio_muesca3)+0.869*(h4/Radio_muesca3)
C4_f44=-3.839+3.070*sqrt(h4/Radio_muesca3)-
0.600*(h4/Radio_muesca3)

if (relacion4 >=0.1 & relacion4 <= 2)
C4_F1=C4_f1
C4_F2=C4_f2
C4_F3=C4_f3
C4_F4=C4_f4
Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if (relacion4 > 2 & relacion4 <= 20)
C4_F1=C4_f11
C4_F2=C4_f22
C4_F3=C4_f33
C4_F4=C4_f44
Kt4_f=C4_F1+C4_F2*(2*h4/Diametro_eje4)+C4_F3*(2*h4/
Diametro_eje4)^2+C4_F4*(2*h4/Diametro_eje4)^3
end

if relacion4 < 0.1 then
Kt4_f=1
end
if relacion4 > 20 then
Kt4_f=1
end

KT4_F=Kt4_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C4_t1=0.905+0.783*sqrt(h4/Radio_muesca3)-
0.075*(h4/Radio_muesca3)
C4_t2=-0.437-
1.969*sqrt(h4/Radio_muesca3)+0.553*(h4/Radio_muesca3)
C4_t3=1.557+1.073*sqrt(h4/Radio_muesca3)-
0.578*(h4/Radio_muesca3)

```



```

C4_t4=-1.061+0.171*sqrt(h4/Radio_muesca3)-
0.086*(h4/Radio_muesca3)

if (relacion4 >=0.25 & relacion4 <= 4.0)
    C4_T1=C4_t1
    C4_T2=C4_t2
    C4_T3=C4_t3
    C4_T4=C4_t4
Kt4_t=C4_T1+C4_T2*(2*h4/Diametro_eje4)+C4_T3*(2*h4/
Diametro_eje4)^2+C4_T4*(2*h4/Diametro_eje4)^3
end

if relacion4 < 0.25 then
Kt4_t=1
end
if relacion4 > 4.0 then
Kt4_t=1
end
KT4_T=Kt4_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f4=(2*(KT4_F-
1)/KT4_F)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf4_f=KT4_F/((1+constante_reemplazo_f4))

constante_reemplazo_t4=(2*(KT4_T-
1)/KT4_T)*(RAIZ_DE_A/sqrt(Radio_muesca3))
Kf4_t=KT4_T/((1+constante_reemplazo_t4))

Am4=[0 (taotormed4*f_carga_torsion*Kf4_t/f_tama4)
0;(taotormed4*f_carga_torsion*Kf4_t/f_tama4) 0 0;0 0 0]
//tensor de esfuerzo medio
[Em4]=spec(Am4)
sigvmm4=(((Em4(1)-Em4(2))^2)+(Em4(2)-
Em4(3))^2)+((Em4(3)-Em4(1))^2)/2)^0.5//Vmises medio
Aa4=[(siga4*f_carga_flexion*Kf4_f/f_tama4)
(taotort4*f_carga_torsion*Kf4_t/f_tama4) 0;0]
(taotort4*f_carga_torsion*Kf4_t/f_tama4) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea4]=spec(Aa4)
sigvma4=(((Ea4(1)-Ea4(2))^2)+(Ea4(2)-
Ea4(3))^2)+((Ea4(3)-Ea4(1))^2)/2)^0.5//Vmises alternante

V_misses_med_LN4=[sigvmm4,
Coef_variacion_carga*sigvmm4]
V_misses_alt_LN4=[sigvma4,
Coef_variacion_carga*sigvma4]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
D_Va_SE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

[Div_Va_SE_4_Cuad, Desv_Div_Va_SE_4_Cuad]=
pot(D_Va_SE4,2)
Division_VaSE4 = [Div_Va_SE_4_Cuad,
Desv_Div_Va_SE_4_Cuad]

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)

```

```

D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Div_Vm_Sut_4_Cuad, Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4 = [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

ASME_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_ASME_4=ASME_4(2)/ASME_4(1)
CRITERIO_F=ASME_4
CF_CRITERIO_F=Cf_ASME_4
end

if criterio_falla == 2 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]

[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
D_Vm_Sut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Div_Vm_Sut_4_Cuad, Desv_Div_Vm_Sut_4_Cuad]=
pot(D_Vm_Sut4,2)
Division_VmSut4 = [Div_Vm_Sut_4_Cuad,
Desv_Div_Vm_Sut_4_Cuad]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)

GERBER_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GERBER_4=GERBER_4(2)/GERBER_4(1)
CRITERIO_F=GERBER_4
CF_CRITERIO_F=Cf_GERBER_4
end

if criterio_falla == 3 then
[Div_Va_SE_4,
Desv_Div_Va_SE_4]=div(V_misses_alt_LN4, SE)
Division_VaSE4 = [Div_Va_SE_4, Desv_Div_Va_SE_4]
[Div_Vm_Sut_4,
Desv_Div_Vm_Sut_4]=div(V_misses_med_LN4, Sut)
Division_VmSut4 = [Div_Vm_Sut_4, Desv_Div_Vm_Sut_4]

[Suma4_VmSut,Desv_Suma4_VmSut]
=suma(Division_VaSE4, Division_VmSut4)
GOODMAN_4=[Suma4_VmSut,Desv_Suma4_VmSut]
Cf_GOODMAN_4=GOODMAN_4(2)/GOODMAN_4(1)
CRITERIO_F=GOODMAN_4
CF_CRITERIO_F=Cf_GOODMAN_4
end

MEDCnormal_4=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_4=CF_CRITERIO_F

[Prob_4,Conf_4]=cdfnor("PQ",0,abs(MEDCnormal_4),DESV
normal_4)

DIAMETRO_EJE4=0

if Prob_4<=1.5*10^-6 & Prob_4>=0.5*10^-6 then
DIAMETRO_EJE4=Diametro_eje4
break;

```

```

end
end
disp("El diametro del tramo 4 es")
disp(DIAMETRO_EJE4)

for Diametro_eje5=tol.tol:6

Radio_eje5=Diametro_eje5/2

Torque5=W_tangencial2*((Radio engrane2)-Radio_eje5) // el
torque aplicado es la fuerza tangencial por el Radio del
engranaje menos el Radio del eje
I5=(%pi*((Radio_eje5^2)^4)/4)//inercia
Q5=(2/3)*(Radio_eje5^3)
J5=I5*2

//calculo de esfuerzos
sig5=(Momento_max5*Radio_eje5/I5) // esfuerzo flexor
maximo
taotormed5=(Torque5*Radio_eje5/J5) // esfuerzo cortante
torsor maximo
taotoralt5=(Torque5*Radio_eje5/J5)

//FACTOR DE TAMAÑO
if (Diametro_eje5 < 0.11)
f_tama5=1
end
if (Diametro_eje5 >= 0.11 & Diametro_eje5 <= 2)
f_tama5=0.879*Diametro_eje5^-0.107
end
if (Diametro_eje5 >= 3 & Diametro_eje5 <= 10)
f_tama5=0.91*(Diametro_eje5^(-0.157))
end
//factor de concentracion de esfuerzo
h5=Radio_eje5-Radio_eje4
relacion5=h5/Radio_muesca4

//0.1<=h/r<=2.0
C5_a1= 0.926+(1.157*sqrt(h5/Radio_muesca4))-
(0.099*(h5/Radio_muesca4))
C5_a2=0.012-
3.036*sqrt(h5/Radio_muesca4)+0.961*(h5/Radio_muesca4)
C5_a3=-0.302+3.977*sqrt(h5/Radio_muesca4)-
1.744*(h5/Radio_muesca4)
C5_a4=0.365-
2.098*sqrt(h5/Radio_muesca4)+0.878*(h5/Radio_muesca4)

//2.0<=h/r<=20
C5_a11=1.2+0.860*sqrt(h5/Radio_muesca4)-
0.022*(h5/Radio_muesca4)
C5_a22=-1.805-0.346*sqrt(h5/Radio_muesca4)-
0.038*(h5/Radio_muesca4)
C5_a33=2.198-
0.486*sqrt(h5/Radio_muesca4)+0.165*(h5/Radio_muesca4)
C5_a44=-0.593-0.028*sqrt(h5/Radio_muesca4)-
0.106*(h5/Radio_muesca4)

if (relacion5 >=0.1 & relacion5 <= 2)
C5_A1=C5_a1
C5_A2=C5_a2
C5_A3=C5_a3
C5_A4=C5_a4
Kt5_a=C5_A1+C5_A2*(2*h5/Diametro_eje5)+C5_A3*(2*h5/
Diametro_eje5)^2+C5_A4*(2*h5/Diametro_eje5)^3
end
if relacion5 >= 2 & relacion5 <= 20)
C5_F1=C5_f11
C5_F2=C5_f22
C5_F3=C5_f33
C5_F4=C5_f44
Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

if relacion5 < 0.1 then
Kt5_f=1
end
if relacion5 > 20 then
Kt5_f=1
end
KT5_F=Kt5_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C5_A1=C5_a11
C5_A2=C5_a22
C5_A3=C5_a33
C5_A4=C5_a44
Kt5_a=C5_A1+C5_A2*(2*h5/Diametro_eje5)+C5_A3*(2*h5/
Diametro_eje5)^2+C5_A4*(2*h5/Diametro_eje5)^3
end
if relacion5 < 0.1 then
Kt5_a=1
end
if relacion5 > 20 then
Kt5_a=1
end
KT5_A=Kt5_a
//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C5_f1=0.947+1.206*sqrt(h5/Radio_muesca4)-
0.131*(h5/Radio_muesca4)
C5_f2=0.022-
3.405*sqrt(h5/Radio_muesca4)+0.915*(h5/Radio_muesca4)
C5_f3=0.869+1.777*sqrt(h5/Radio_muesca4)-
0.555*(h5/Radio_muesca4)
C5_f4=-0.810+0.422*sqrt(h5/Radio_muesca4)-
0.260*(h5/Radio_muesca4)

//2.0<=h/r<=20
C5_f11=1.232+0.832*sqrt(h5/Radio_muesca4)-
0.008*(h5/Radio_muesca4)
C5_f22=-3.813+0.968*sqrt(h5/Radio_muesca4)-
0.260*(h5/Radio_muesca4)
C5_f33=7.423-
4.868*sqrt(h5/Radio_muesca4)+0.869*(h5/Radio_muesca4)
C5_f44=-3.839+3.070*sqrt(h5/Radio_muesca4)-
0.600*(h5/Radio_muesca4)

if (relacion5 >=0.1 & relacion5 <= 2)
C5_F1=C5_f1
C5_F2=C5_f2
C5_F3=C5_f3
C5_F4=C5_f4
Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

if (relacion5 > 2 & relacion5 <= 20)
C5_F1=C5_f11
C5_F2=C5_f22
C5_F3=C5_f33
C5_F4=C5_f44
Kt5_f=C5_F1+C5_F2*(2*h5/Diametro_eje5)+C5_F3*(2*h5/
Diametro_eje5)^2+C5_F4*(2*h5/Diametro_eje5)^3
end

if relacion5 < 0.1 then
Kt5_f=1
end
if relacion5 > 20 then
Kt5_f=1
end
KT5_F=Kt5_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0

```

```

C5_t1=0.905+0.783*sqrt(h5/Radio_muesca4)-
0.075*(h5/Radio_muesca4)
C5_t2=-0.437-
1.969*sqrt(h5/Radio_muesca4)+0.553*(h5/Radio_muesca4)
C5_t3=1.557+1.073*sqrt(h5/Radio_muesca4)-
0.578*(h5/Radio_muesca4)
C5_t4=-1.061+0.171*sqrt(h5/Radio_muesca4)-
0.086*(h5/Radio_muesca4)

```

```

if (relacion5 >=0.25 & relacion5 <= 4.0)
  C5_T1=C5_t1
  C5_T2=C5_t2
  C5_T3=C5_t3
  C5_T4=C5_t4
  Kt5_t=C5_T1+C5_T2*(2*h5/Diametro_eje5)+C5_T3*(2*h5/
  Diametro_eje5)^2+C5_T4*(2*h5/Diametro_eje5)^3
end
if relacion5 < 0.25 then
  Kt5_t=1
end
if relacion5 > 4.0 then
  Kt5_t=1
end
KT5_T=Kt5_t

```

```

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f flexión), (Kf_t torsión)
constante_reemplazo_a5=(2*(KT5_A-
1)/KT5_A)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_a=KT5_A/((1+constante_reemplazo_a5))

```

```

constante_reemplazo_f5=(2*(KT5_F-
1)/KT5_F)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_f=KT5_F/((1+constante_reemplazo_f5))

```

```

constante_reemplazo_t5=(2*(KT5_T-
1)/KT5_T)*(RAIZ_DE_A/sqrt(Radio_muesca4))
Kf5_t=KT5_T/((1+constante_reemplazo_t5))

```

```

//presion por interferencia
m2=(((2*Radio engrane2)^2-
Diametro_eje5^2)*(Diametro_eje5^2))
n2=(2*Radio engrane2)
PP5=((E*Interferen_maza_eje_max2)/(2*Diametro_eje5^3))*(
m2/n2)

```

```

Am5=[(PP5*f_carga_axial*Kf5_a/f_tama5)
(taotormed5*f_carga_torsion*Kf5_t/f_tama5)
0;(taotormed5*f_carga_torsion*Kf5_t/f_tama5) 0 0;0 0 0]

```

```

//tensor de esfuerzo medio//
[Em5]=spec(Am5)
sigvmm5=(((Em5(1)-Em5(2))^2)+((Em5(2)-
Em5(3))^2)+((Em5(3)-Em5(1))^2))/2^0.5

```

```

Aa5=[(siga5*f_carga_flexion*Kf5_f/f_tama5)
(taotort5*f_carga_torsion*Kf5_t/f_tama5) 0;0]
(taotort5*f_carga_torsion*Kf5_t/f_tama5) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea5]=spec(Aa5)
sigvma5=(((Ea5(1)-Ea5(2))^2)+((Ea5(2)-
Ea5(3))^2)+((Ea5(3)-Ea5(1))^2))/2^0.5

```

```

V_misses_med_LN5=[sigvmm5,
Coef_variacion_carga*sigvmm5]
V_misses_alt_LN5=[sigvma5,
Coef_variacion_carga*sigvma5]

```

```
[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)
```

```
SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]
```

```
if criterio_falla == 1 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
D_Va_SE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

```

```
[Div_Va_SE_5_Cuad, Desv_Div_Va_SE_5_Cuad]=
pot(D_Va_SE5,2)
Division_VaSE5 = [Div_Va_SE_5_Cuad,
Desv_Div_Va_SE_5_Cuad]

```

```
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

```

```
[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]

```

```
[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
```

```
ASME_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_ASME_5=ASME_5(2)/ASME_5(1)
CRITERIO_F=ASME_5
CF_CRITERIO_F=Cf_ASME_5
end

```

```
if criterio_falla == 2 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

```

```
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
D_Vm_Sut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

```

```
[Div_Vm_Sut_5_Cuad, Desv_Div_Vm_Sut_5_Cuad]=
pot(D_Vm_Sut5,2)
Division_VmSut5 = [Div_Vm_Sut_5_Cuad,
Desv_Div_Vm_Sut_5_Cuad]

```

```
[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
```

```
GERBER_5=[Suma5_VmSut,Desv_Suma5_VmSut]
Cf_GERBER_5=GERBER_5(2)/GERBER_5(1)
CRITERIO_F=GERBER_5
CF_CRITERIO_F=Cf_GERBER_5
end

```

```
if criterio_falla == 3 then
[Div_Va_SE_5,
Desv_Div_Va_SE_5]=div(V_misses_alt_LN5, SE)
Division_VaSE5 = [Div_Va_SE_5, Desv_Div_Va_SE_5]

```

```
[Div_Vm_Sut_5,
Desv_Div_Vm_Sut_5]=div(V_misses_med_LN5, Sut)
Division_VmSut5 = [Div_Vm_Sut_5, Desv_Div_Vm_Sut_5]

```

```
[Suma5_VmSut,Desv_Suma5_VmSut]
=suma(Division_VaSE5, Division_VmSut5)
GOODMAN_5=[Suma5_VmSut,Desv_Suma5_VmSut]

```

```

Cf_GOODMAN_5=GOODMAN_5(2)/GOODMAN_5(1)
CRITERIO_F=GOODMAN_5
CF_CRITERIO_F=Cf_GOODMAN_5
end

MEDCnormal_5=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_5=CF_CRITERIO_F

[Prob_5,Conf_5]=cdfnorr("PQ",0,abs(MEDCnormal_5),DESV
normal_5)

DIAMETRO_EJE5=0

if Prob_5<=1.5*10^-6 & Prob_5>=0.5*10^-6 then
    DIAMETRO_EJE5=Diametro_eje5
    break;
end
disp("El diametro del tramo 5 es")
disp(DIAMETRO_EJE5)

for Diametro_eje6=tol.tol:6

Radio_eje6=Diametro_eje6/2
Torque6=W_tangencial2*((Radio engrane2)-Radio_eje5)
I6=(%pi*((Radio_eje6^2)^4))/4//inercia
J6=I6*2
// calculo de esfuerzos
sigma6=(Momento_max6*Radio_eje6/I6) // esfuerzo flexor
maximo
taotormed6=(Torque6*Radio_eje6/J6) // esfuerzo cortante
torsor maximo
taotortal6=(Torque6*Radio_eje6/J6)

//FACTOR DE TAMAÑO
if (Diametro_eje6 < 0.11)
    f_tama6=1
end
if (Diametro_eje6 >= 0.11 & Diametro_eje6 <= 2)
    f_tama6=0.879*Diametro_eje6^-0.107
end
if (Diametro_eje6 >= 3 & Diametro_eje6 <= 10)
    f_tama6=0.91*(Diametro_eje6^(-0.157))
end

//factor de concentracion de esfuerzo
h6=Radio_eje6-Radio_eje5
relacion6=h6/Radio_muesca5

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C6_f1=0.947+1.206*sqrt(h6/Radio_muesca5)-
0.131*(h6/Radio_muesca5)
C6_f2=0.022-
3.405*sqrt(h6/Radio_muesca5)+0.915*(h6/Radio_muesca5)
C6_f3=0.869+1.777*sqrt(h6/Radio_muesca5)-
0.555*(h6/Radio_muesca5)
C6_f4=-0.810+0.422*sqrt(h6/Radio_muesca5)-
0.260*(h6/Radio_muesca5)

//2.0<=h/r<=20
C6_f11=1.232+0.832*sqrt(h6/Radio_muesca5)-
0.008*(h6/Radio_muesca5)
C6_f22=-3.813+0.968*sqrt(h6/Radio_muesca5)-
0.260*(h6/Radio_muesca5)

```

```

C6_f33=7.423-
4.868*sqrt(h6/Radio_muesca5)+0.869*(h6/Radio_muesca5)
C6_f44=-3.839+3.070*sqrt(h6/Radio_muesca5)-
0.600*(h6/Radio_muesca5)

if (relacion6 >=0.1 & relacion6 <= 2)
    C6_F1=C6_f1
    C6_F2=C6_f2
    C6_F3=C6_f3
    C6_F4=C6_f4
    Kt6_f=C6_F1+C6_F2*(2*h6/Diametro_eje6)+C6_F3*(2*h6/
    Diametro_eje6)^2+C6_F4*(2*h6/Diametro_eje6)^3
end

if (relacion6 > 2 & relacion6 <= 20)
    C6_F1=C6_f11
    C6_F2=C6_f22
    C6_F3=C6_f33
    C6_F4=C6_f44
    Kt6_f=C6_F1+C6_F2*(2*h6/Diametro_eje6)+C6_F3*(2*h6/
    Diametro_eje6)^2+C6_F4*(2*h6/Diametro_eje6)^3
end
if relacion6 < 0.1 then
    Kt6_f=1
end
if relacion6 > 20 then
    Kt6_f=1
end
KT6_F=Kt6_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C6_t1=0.905+0.783*sqrt(h6/Radio_muesca5)-
0.075*(h6/Radio_muesca5)
C6_t2=-0.437-
1.969*sqrt(h6/Radio_muesca5)+0.553*(h6/Radio_muesca5)
C6_t3=1.557+1.073*sqrt(h6/Radio_muesca5)-
0.578*(h6/Radio_muesca5)
C6_t4=-1.061+0.171*sqrt(h6/Radio_muesca5)-
0.086*(h6/Radio_muesca5)

if (relacion6 >=0.25 & relacion6 <= 4.0)
    C6_T1=C6_t1
    C6_T2=C6_t2
    C6_T3=C6_t3
    C6_T4=C6_t4
    Kt6_t=C6_T1+C6_T2*(2*h6/Diametro_eje6)+C6_T3*(2*h6/
    Diametro_eje6)^2+C6_T4*(2*h6/Diametro_eje6)^3
end
if relacion6 < 0.25 then
    Kt6_t=1
end
if relacion6 > 20 then
    Kt6_t=1
end
KT6_T=Kt6_t

constante_reemplazo_f6=(2*(KT6_F-
1)/KT6_F)*(RAIZ_DE_A/sqrt(Radio_muesca5))
Kf6_f=KT6_F/((1+constante_reemplazo_f6))

constante_reemplazo_t6=(2*(KT6_T-
1)/KT6_T)*(RAIZ_DE_A/sqrt(Radio_muesca5))
Kf6_t=KT6_T/((1+constante_reemplazo_t6))

```

```

Am6=[0 (taotormed6*f_carga_torsion*Kf6_t/f_tama6)
0;(taotormed6*f_carga_torsion*Kf6_t/f_tama6) 0 0;0 0 0]
//tensor de esfuerzo medio
[Em6]=spec(Am6)
sigvmm6=(((Em6(1)-Em6(2))^2)+((Em6(2)-
Em6(3))^2)+((Em6(3)-Em6(1))^2)/2)^0.5

Aa6=[(siga6*f_carga_flexion*Kf6_f/f_tama6)
(taotoralt6*f_carga_torsion*Kf6_t/f_tama6) 0;0
(taotoralt6*f_carga_torsion*Kf6_t/f_tama6) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea6]=spec(Aa6)
sigvma6=(((Ea6(1)-Ea6(2))^2)+((Ea6(2)-
Ea6(3))^2)+((Ea6(3)-Ea6(1))^2)/2)^0.5

V_misses_med_LN6=[sigvmm6,
Coef_variacion_carga*sigvmm6]
V_misses_alt_LN6=[sigvma6,
Coef_variacion_carga*sigvma6]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_6,
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
D_Va_SE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]

[Div_Va_SE_6_Cuad, Desv_Div_Va_SE_6_Cuad]=
pot(D_Va_SE6,2)
Division_VaSE6 = [Div_Va_SE_6_Cuad,
Desv_Div_Va_SE_6_Cuad]

Division_VmSut6 = [0,0]

Suma6_VmSut_VaSE=Division_VaSE6(1)+Division_VmSut6
(1)
Desv_Suma6_VmSut_VaSE=sqrt((Division_VaSE6(2)/Divisi
n_VaSE6(1))^2+(0)^2)

ASME_6=[Suma6_VmSut_VaSE,
Desv_Suma6_VmSut_VaSE]
Cf_ASME_6=ASME_6(2)/ASME_6(1)
CRITERIO_F=ASME_6
CF_CRITERIO_F=Cf_ASME_6
end

if criterio_falla == 2 then
[Div_Va_SE_6,
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
Division_VaSE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]

Division_VmSut6 = [0,0]

Suma6_VmSut_VaSE=Division_VaSE6(1)+Division_VmSut6
(1)
Desv_Suma6_VmSut_VaSE=sqrt((Division_VaSE6(2)/Divisi
n_VaSE6(1))^2+(0)^2)

GERBER_6=[Suma6_VmSut_VaSE,
Desv_Suma6_VmSut_VaSE]
Cf_GERBER_6=GERBER_6(2)/GERBER_6(1)
CRITERIO_F=GERBER_6
CF_CRITERIO_F=Cf_GERBER_6
end

```

```

if criterio_falla == 3 then
[Div_Va_SE_6,
Desv_Div_Va_SE_6]=div(V_misses_alt_LN6, SE)
Division_Va_SE6 = [Div_Va_SE_6, Desv_Div_Va_SE_6]

Division_VmSut6 = [0,0]

Suma6_VmSut_VaSE=Division_Va_SE6(1)+Division_VmSut
6(1)
Desv_Suma6_VmSut_VaSE=sqrt((Division_Va_SE6(2)/Divisi
on_Va_SE6(1))^2+(0)^2)
GOODMAN_6=[Suma6_VmSut_VaSE,
Desv_Suma6_VmSut_VaSE]
Cf_GOODMAN_6=GOODMAN_6(2)/GOODMAN_6(1)
CRITERIO_F=GOODMAN_6
CF_CRITERIO_F=Cf_GOODMAN_6
end

MEDCnormal_6=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_6=CF_CRITERIO_F

[Prob_6,Conf_6]=cdfnor("PQ",0,abs(MEDCnormal_6),DESV
normal_6)

DIAMETRO_EJE6=0

if Prob_6<=1.5*10^-6 & Prob_6>=0.5*10^-6 then
DIAMETRO_EJE6=Diametro_eje6
break;
end

end
disp("El diametro del tramo 6 es")
disp(DIAMETRO_EJE6)

for Diametro_eje7=tol:tol:6

Radio_eje7=Diametro_eje7/2

Torque7=W_tangencial3*((Radio engrane3)-Radio_eje7) // el
torque aplicado es la fuerza tangencial por el Radio del
engranaje menos el Radio del eje
I7=(%pi)*((Radio_eje7^2)^4)/4//inercia
Q7=(2/3)*(Radio_eje7^3)
J7=I7*2

//calculo de esfuerzos
siga7=(Momento_max7*Radio_eje7/I7) // esfuerzo flector
maximo
taotormed7=(Torque7*Radio_eje7/J7) // esfuerzo cortante
torsor maximo
taotoralt7=(Torque7*Radio_eje7/J7)

//FACTOR DE TAMAÑO

if (Diametro_eje7 < 0.11)
f_tama7=1
end
if (Diametro_eje7 >= 0.11 & Diametro_eje7 <= 2)
f_tama7=0.879*Diametro_eje7^-0.107
end
if (Diametro_eje7 >= 3 & Diametro_eje7 <= 10)
f_tama7=0.91*(Diametro_eje7^(-0.177))
end

```

//factor de concentracion de esfuerzo

h7=Radio_eje7-Radio_eje6
relacion7=h7/Radio_muesca6

//0.1 <= h/r <= 2.0

C7_a1= 0.926+(1.157*sqrt(h7/Radio_muesca6))-
(0.099*(h7/Radio_muesca6))
C7_a2=0.012-
3.036*sqrt(h7/Radio_muesca6)+0.961*(h7/Radio_muesca6)
C7_a3=-0.302+3.977*sqrt(h7/Radio_muesca6)-
1.744*(h7/Radio_muesca6)

C7_a4=0.365-
2.098*sqrt(h7/Radio_muesca6)+0.878*(h7/Radio_muesca6)

//2.0 <= h/r <= 20

C7_a11=1.2+0.860*sqrt(h7/Radio_muesca6)-
0.022*(h7/Radio_muesca6)
C7_a22=-1.805-0.346*sqrt(h7/Radio_muesca6)-
0.038*(h7/Radio_muesca6)
C7_a33=2.198-
0.486*sqrt(h7/Radio_muesca6)+0.165*(h7/Radio_muesca6)
C7_a44=-0.593-0.028*sqrt(h7/Radio_muesca6)-
0.106*(h7/Radio_muesca6)

if (relacion7 >=0.1 & relacion7 <= 2)

C7_A1=C7_a1
C7_A2=C7_a2
C7_A3=C7_a3
C7_A4=C7_a4

Kt10_a=C7_A1+C7_A2*(2*h7/Diametro_eje7)+C7_A3*(2*h
7/Diametro_eje7)^2+C7_A4*(2*h7/Diametro_eje7)^3
end

if (relacion7 > 2 & relacion7 <= 20)

C7_A1=C7_a11
C7_A2=C7_a22
C7_A3=C7_a33
C7_A4=C7_a44

Kt10_a=C7_A1+C7_A2*(2*h7/Diametro_eje7)+C7_A3*(2*h
7/Diametro_eje7)^2+C7_A4*(2*h7/Diametro_eje7)^3
end

if relacion7 < 0.1 then

Kt10_a=1

end

if relacion7 > 20 then

Kt10_a=1

end

KT7_A=Kt10_a

//Kt para esfuerzos de flexión (Kf)

//0.1 <= h/r <= 2.0

C7_f1=0.947+1.206*sqrt(h7/Radio_muesca6)-
0.131*(h7/Radio_muesca6)
C7_f2=0.022-
3.405*sqrt(h7/Radio_muesca6)+0.915*(h7/Radio_muesca6)
C7_f3=0.869+1.777*sqrt(h7/Radio_muesca6)-
0.555*(h7/Radio_muesca6)
C7_f4=-0.810+0.422*sqrt(h7/Radio_muesca6)-
0.260*(h7/Radio_muesca6)

//2.0 <= h/r <= 20

C7_f11=1.232+0.832*sqrt(h7/Radio_muesca6)-
0.008*(h7/Radio_muesca6)
C7_f22=-3.813+0.968*sqrt(h7/Radio_muesca6)-
0.260*(h7/Radio_muesca6)

C7_f33=7.423-

4.868*sqrt(h7/Radio_muesca6)+0.869*(h7/Radio_muesca6)

C7_f44=-3.839+3.070*sqrt(h7/Radio_muesca6)-

0.600*(h7/Radio_muesca6)

if (relacion7 >=0.1 & relacion7 <= 2)

C7_F1=C7_f1

C7_F2=C7_f2

C7_F3=C7_f3

C7_F4=C7_f4

Kt10_f=C7_F1+C7_F2*(2*h7/Diametro_eje7)+C7_F3*(2*h7/
Diametro_eje7)^2+C7_F4*(2*h7/Diametro_eje7)^3

end

if (relacion7 > 2 & relacion7 <= 20)

C7_F1=C7_f11

C7_F2=C7_f22

C7_F3=C7_f33

C7_F4=C7_f44

Kt10_f=C7_F1+C7_F2*(2*h7/Diametro_eje7)+C7_F3*(2*h7/
Diametro_eje7)^2+C7_F4*(2*h7/Diametro_eje7)^3

end

if relacion7 < 0.1 then

Kt10_f=1

end

if relacion7 > 20 then

Kt10_f=1

end

KT7_F=Kt10_f

//Kt para esfuerzos de torsión (Ktf)

//0.25 <= h/r <= 4.0

C7_t1=0.905+0.783*sqrt(h7/Radio_muesca6)-

0.075*(h7/Radio_muesca6)

C7_t2=-0.437-

1.969*sqrt(h7/Radio_muesca6)+0.553*(h7/Radio_muesca6)

C7_t3=1.557+1.073*sqrt(h7/Radio_muesca6)-

0.578*(h7/Radio_muesca6)

C7_t4=-1.061+0.171*sqrt(h7/Radio_muesca6)-

0.086*(h7/Radio_muesca6)

if (relacion7 >= 0.25 & relacion7 <= 4.0)

C7_T1=C7_t1

C7_T2=C7_t2

C7_T3=C7_t3

C7_T4=C7_t4

Kt7_t=C7_T1+C7_T2*(2*h7/Diametro_eje7)+C7_T3*(2*h7/
Diametro_eje7)^2+C7_T4*(2*h7/Diametro_eje7)^3

end

if relacion7 < 0.25 then

Kt7_t=1

end

if relacion7 > 0.25 then

Kt7_t=1

end

KT7_T=Kt7_t

//Factor de concentración de esfuerzo de fatiga (Kf_a - axial),

(Kf_f - flexión), (Kf_t - torsión)

constante_reemplazo_a7=(2*(KT7_A-

1)/KT7_A)*(RAIZ_DE_A/sqrt(Radio_muesca6))

Kf7_a=KT7_A/((1+constante_reemplazo_a7))


```

constante_reemplazo_f7=(2*(KT7_F-
1)/KT7_F)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_f=KT7_F/((1+constante_reemplazo_f7))

constante_reemplazo_t7=(2*(KT7_T-
1)/KT7_T)*(RAIZ_DE_A/sqrt(Radio_muesca6))
Kf7_t=KT7_T/((1+constante_reemplazo_t7))
//presion por interferencia
m3=(((2*Radio engrane3)^2-
Diametro_eje7^2)*(Diametro_eje7^2))
n3=(2*Radio engrane3)
PP7=((E*Interferen_maza_eje_max3)/(2*Diametro_eje7^3))*(
m3/n3)

Am7=[(PP7*f_carga_axial*Kf7_a/f_tama7)
(taotormed7*f_carga_torsion*Kf7_t/f_tama7)
0;(taotormed7*f_carga_torsion*Kf7_t/f_tama7) 0 0;0 0 0]
//tensor de esfuerzo medio//
[Em7]=spec(Am7)
sigvmm7=(((Em7(1)-Em7(2))^2)+((Em7(2)-
Em7(3))^2)+((Em7(3)-Em7(1))^2))/2^0.5

Aa7=[(sigaf7*f_carga_flexion*Kf7_f/f_tama7)
(taotormed7*f_carga_torsion*Kf7_t/f_tama7) 0;0
(taotormed7*f_carga_torsion*Kf7_t/f_tama7) 0;0 0 0] //tensor
de esfuerzo alternante
[Ea7]=spec(Aa7)
sigvma7=(((Ea7(1)-Ea7(2))^2)+((Ea7(2)-
Ea7(3))^2)+((Ea7(3)-Ea7(1))^2))/2^0.5

V_misses_med_LN7=[sigvmm7,
Coef_variacion_carga*sigvmm7]
V_misses_alt_LN7=[sigvma7,
Coef_variacion_carga*sigvma7]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
D_Va_SE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

[Div_Va_SE_7_Cuad,          Desv_Div_Va_SE_7_Cuad]=
pot(D_Va_SE7,2)
Division_VaSE7 = [Div_Va_SE_7_Cuad,
Desv_Div_Va_SE_7_Cuad]

[Div_Vm_Sut_7,
Desv_Div_Vm_Sut_7]=div(V_misses_med_LN7, Sut)
D_Vm_Sut7 = [Div_Vm_Sut_7, Desv_Div_Vm_Sut_7]

[Div_Vm_Sut_7_Cuad,          Desv_Div_Vm_Sut_7_Cuad]=
pot(D_Vm_Sut7,2)
Division_VmSut7 = [Div_Vm_Sut_7_Cuad,
Desv_Div_Vm_Sut_7_Cuad]

[Suma7_VmSut,Desv_Suma7_VmSut]
=suma(Division_VaSE7, Division_VmSut7)

ASME_7=[Suma7_VmSut,Desv_Suma7_VmSut]
Cf_ASME_7=ASME_7(2)/ASME_7(1)
CRITERIO_F=ASME_7
CF_CRITERIO_F=Cf_ASME_7
end

```

```

if criterio_falla == 2 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
Division_VaSE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]

[Div_Vm_Sut_7,
Desv_Div_Vm_Sut_7]=div(V_misses_med_LN7, Sut)
D_Vm_Sut7 = [Div_Vm_Sut_7, Desv_Div_Vm_Sut_7]

[Div_Vm_Sut_7_Cuad,          Desv_Div_Vm_Sut_7_Cuad]=
pot(D_Vm_Sut7,2)
Division_VmSut7 = [Div_Vm_Sut_7_Cuad,
Desv_Div_Vm_Sut_7_Cuad]

[Suma7_VmSut,Desv_Suma7_VmSut]
=suma(Division_VaSE7, Division_VmSut7)

GERBER_7=[Suma7_VmSut,Desv_Suma7_VmSut]
Cf_GERBER_7=GERBER_7(2)/GERBER_7(1)
CRITERIO_F=GERBER_7
CF_CRITERIO_F=Cf_GERBER_7
end

if criterio_falla == 3 then
[Div_Va_SE_7,
Desv_Div_Va_SE_7]=div(V_misses_alt_LN7, SE)
Division_VaSE7 = [Div_Va_SE_7, Desv_Div_Va_SE_7]
[Div_Vm_Sut_7,
Desv_Div_Vm_Sut_7]=div(V_misses_med_LN7, Sut)
Division_VmSut7 = [Div_Vm_Sut_7, Desv_Div_Vm_Sut_7]

[Suma7_VmSut,Desv_Suma7_VmSut]
=suma(Division_VaSE7, Division_VmSut7)
GOODMAN_7=[Suma7_VmSut,Desv_Suma7_VmSut]
Cf_GOODMAN_7=GOODMAN_7(2)/GOODMAN_7(1)
CRITERIO_F=GOODMAN_7
CF_CRITERIO_F=Cf_GOODMAN_7
end
MEDCnormal_7=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESNnormal_7=CF_CRITERIO_F

[Prob_7,Conf_7]=cdfnorr("PQ",0,abs(MEDCnormal_7),DESN
normal_7)

DIAMETRO_EJE7=0

if Prob_7<=1.5*10^-6 & Prob_7>=0.5*10^-6 then
DIAMETRO_EJE7=Diametro_eje7
break;
end

end
disp("El diametro del tramo 7 es")
disp(DIAMETRO_EJE7)

for Diametro_eje8=tol:tol:6

Radio_eje8=Diametro_eje8/2
I8=(%pi*(((Radio_eje8*2)^4))/4)/inercia
J8=I8*2

// calculo de esfuerzos
sigaf8=(Momento_max8*Radio_eje8/I8) // esfuerzo flector
maximo

```

```

//FACTOR DE TAMAÑO
if (Diametro_eje8 < 0.11)
  f_tama8=1
end
if (Diametro_eje8 >= 0.11 & Diametro_eje8 <= 2)
  f_tama8=0.879*Diametro_eje8^-0.107
end
if (Diametro_eje8 >= 3 & Diametro_eje8 <= 10)
  f_tama8=0.91*(Diametro_eje8^(-0.157))
end

//factor de concentracion de esfuerzo
h8=Radio_eje8-Radio_eje7
relacion8=h8/Radio_muesca7

//Kt para esfuerzos de flexión (Ktf)

//0.1<=h/r<=2.0
C8_f1=0.947+1.206*sqrt(h8/Radio_muesca7)-
0.131*(h8/Radio_muesca7)
C8_f2=0.022-
3.405*sqrt(h8/Radio_muesca7)+0.915*(h8/Radio_muesca7)
C8_f3=0.869+1.777*sqrt(h8/Radio_muesca7)-
0.555*(h8/Radio_muesca7)
C8_f4=-0.810+0.422*sqrt(h8/Radio_muesca7)-
0.260*(h8/Radio_muesca7)

//2.0<=h/r<=20
C8_f11=1.232+0.832*sqrt(h8/Radio_muesca7)-
0.008*(h8/Radio_muesca7)
C8_f22=-3.813+0.968*sqrt(h8/Radio_muesca7)-
0.260*(h8/Radio_muesca7)
C8_f33=7.423-
4.868*sqrt(h8/Radio_muesca7)+0.869*(h8/Radio_muesca7)
C8_f44=-3.839+3.070*sqrt(h8/Radio_muesca7)-
0.600*(h8/Radio_muesca7)

if (relacion8 >= 0.1 & relacion8 <= 2)
  C8_F1=C8_f1
  C8_F2=C8_f2
  C8_F3=C8_f3
  C8_F4=C8_f4
  Kt8_f=C8_F1+C8_F2*(2*h8/Diametro_eje8)+C8_F3*(2*h8/
Diametro_eje8)^2+C8_F4*(2*h8/Diametro_eje8)^3
end

if (relacion8 > 2 & relacion8 <= 20)
  C8_F1=C8_f11
  C8_F2=C8_f22
  C8_F3=C8_f33
  C8_F4=C8_f44
  Kt8_f=C8_F1+C8_F2*(2*h8/Diametro_eje8)+C8_F3*(2*h8/
Diametro_eje8)^2+C8_F4*(2*h8/Diametro_eje8)^3
end

if relacion8 < 0.1 then
  Kt8_f=1
end
if relacion8 > 20 then
  Kt8_f=1
end
end
KT8_F=Kt8_f
//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C8_t1=0.905+0.783*sqrt(h8/Radio_muesca7)-
0.075*(h8/Radio_muesca7)

```

```

C8_t2=-0.437-
1.969*sqrt(h8/Radio_muesca7)+0.553*(h8/Radio_muesca7)
C8_t3=1.557+1.073*sqrt(h8/Radio_muesca7)-
0.578*(h8/Radio_muesca7)
C8_t4=-1.061+0.171*sqrt(h8/Radio_muesca7)-
0.086*(h8/Radio_muesca7)

if (relacion8 >= 0.25 & relacion8 <= 4.0)
  C8_T1=C8_t1
  C8_T2=C8_t2
  C8_T3=C8_t3
  C8_T4=C8_t4
  Kt8_t=C8_T1+C8_T2*(2*h8/Diametro_eje8)+C8_T3*(2*h8/
Diametro_eje8)^2+C8_T4*(2*h8/Diametro_eje8)^3
end

if relacion8 < 0.25 then
  Kt8_t=1
end
if relacion8 > 4.0 then
  Kt8_t=1
end
KT8_T=Kt8_t

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f8=(2*(KT8_F-
1)/KT8_F)*(RAIZ_DE_A/sqrt(Radio_muesca7))
Kf8_f=KT8_F/((1+constante_reemplazo_f8))
constante_reemplazo_t8=(2*(KT8_T-
1)/KT8_T)*(RAIZ_DE_A/sqrt(Radio_muesca7))
Kf8_t=KT8_T/((1+constante_reemplazo_t8))
Torque8=Torque7

//Esfuerzo cortante torsor alternante:
taotormed8=(Torque8*Radio_eje8/J8)
//Esfuerzo cortante torsor medio:
taotoralt8=(Torque8*Radio_eje8/J8)

Am8=[0 (taotormed8*f_carga_torsion*Kf8_t/f_tama8)
0;(taotormed8*f_carga_torsion*Kf8_t/f_tama8) 0 0;0 0 0]
//tensor de esfuerzo medio//
[Em8]=spec(Am8)
sigvmm8=(((Em8(1)-Em8(2))^2)+((Em8(2)-
Em8(3))^2)+((Em8(3)-Em8(1))^2)/2)^0.5

Aa8=[(siga8*f_carga_flexion*Kf8_f/f_tama8)
(taotoralt8*f_carga_torsion*Kf8_t/f_tama8) 0;0 0 0]
[Ea8]=spec(Aa8)
sigvma8=(((Ea8(1)-Ea8(2))^2)+((Ea8(2)-
Ea8(3))^2)+((Ea8(3)-Ea8(1))^2)/2)^0.5

V_misses_med_LN8=[sigvmm8,
Coef_variacion_carga*sigvmm8]
V_misses_alt_LN8=[sigvma8,
Coef_variacion_carga*sigvma8]

[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)

SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]

if criterio_falla == 1 then
[Div_Va_SE_8,
Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)

```



```
D_Va_SE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]
```

```
[Div_Va_SE_8_Cuad, Desv_Div_Va_SE_8_Cuad]=  
pot(D_Va_SE8,2)  
Division_VaSE8 = [Div_Va_SE_8_Cuad,  
Desv_Div_Va_SE_8_Cuad]
```

```
[Div_Vm_Sut_8,  
Desv_Div_Vm_Sut_8]=div(V_misses_med_LN8, Sut)  
D_Vm_Sut8 = [Div_Vm_Sut_8, Desv_Div_Vm_Sut_8]
```

```
[Div_Vm_Sut_8_Cuad, Desv_Div_Vm_Sut_8_Cuad]=  
pot(D_Vm_Sut8,2)  
Division_VmSut8 = [Div_Vm_Sut_8_Cuad,  
Desv_Div_Vm_Sut_8_Cuad]
```

```
[Suma8_VmSut,Desv_Suma8_VmSut]  
=suma(Division_VaSE8, Division_VmSut8)
```

```
ASME_8=[Suma8_VmSut,Desv_Suma8_VmSut]  
Cf_ASME_8=ASME_8(2)/ASME_8(1)  
CRITERIO_F=ASME_8  
CF_CRITERIO_F=Cf_ASME_8  
end
```

```
if criterio_falla == 2 then  
[Div_Va_SE_8,  
Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)  
Division_VaSE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]
```

```
[Div_Vm_Sut_8,  
Desv_Div_Vm_Sut_8]=div(V_misses_med_LN8, Sut)  
D_Vm_Sut8 = [Div_Vm_Sut_8, Desv_Div_Vm_Sut_8]
```

```
[Div_Vm_Sut_8_Cuad, Desv_Div_Vm_Sut_8_Cuad]=  
pot(D_Vm_Sut8,2)  
Division_VmSut8 = [Div_Vm_Sut_8_Cuad,  
Desv_Div_Vm_Sut_8_Cuad]
```

```
[Suma8_VmSut,Desv_Suma8_VmSut]  
=suma(Division_VaSE8, Division_VmSut8)
```

```
GERBER_8=[Suma8_VmSut,Desv_Suma8_VmSut]  
Cf_GERBER_8=GERBER_8(2)/GERBER_8(1)  
CRITERIO_F=GERBER_8  
CF_CRITERIO_F=Cf_GERBER_8  
end
```

```
if criterio_falla == 3 then  
[Div_Va_SE_8,  
Desv_Div_Va_SE_8]=div(V_misses_alt_LN8, SE)  
Division_VaSE8 = [Div_Va_SE_8, Desv_Div_Va_SE_8]  
[Div_Vm_Sut_8,  
Desv_Div_Vm_Sut_8]=div(V_misses_med_LN8, Sut)  
Division_VmSut8 = [Div_Vm_Sut_8, Desv_Div_Vm_Sut_8]
```

```
[Suma8_VmSut,Desv_Suma8_VmSut]  
=suma(Division_VaSE8, Division_VmSut8)  
GOODMAN_8=[Suma8_VmSut,Desv_Suma8_VmSut]  
Cf_GOODMAN_8=GOODMAN_8(2)/GOODMAN_8(1)  
CRITERIO_F=GOODMAN_8  
CF_CRITERIO_F=Cf_GOODMAN_8  
end
```

```
MEDCnormal_8=log(CRITERIO_F(1))-  
((0.5)*(CF_CRITERIO_F)^2)  
DESVnormal_8=CF_CRITERIO_F
```

```
[Prob_8,Conf_8]=cdfnor("PQ",0,abs(MEDCnormal_8),DESV  
normal_8)
```

```
DIAMETRO_EJE8=0
```

```
if Prob_8<=1.5*10^-6 & Prob_8>=0.5*10^-6 then  
DIAMETRO_EJE8=Diametro_eje8  
break;
```

```
end  
end  
disp("El diametro del tramo 8 es")  
disp(DIAMETRO_EJE8)
```

```
for Diametro_eje9=tol:tol:6
```

```
Radio_eje9=Diametro_eje9/2  
I9=(%pi*(((Radio_eje9*2)^4))/4)/inercia  
Q9=(2/3)*(Radio_eje9^3)  
J9=I9*2  
// calculo de esfuerzos  
sigma9=(Momento_max9*Radio_eje9/I9) // esfuerzo flexor  
maximo  
Torque9=W_tangencial4*((Radio engrane4)-Radio_eje9)  
taotormed9=(Torque9*Radio_eje9/J9)  
//Esfuerzo cortante torsor medio:  
taotoralt9=(Torque3*Radio_eje9/J9)
```

```
//FACTOR DE TAMAÑO
```

```
if (Diametro_eje9 < 0.11)  
f_tama9=1  
end  
if (Diametro_eje9 >= 0.11 & Diametro_eje9 <= 2)  
f_tama9=0.879*Diametro_eje9^-0.107  
end  
if (Diametro_eje9 >= 3 & Diametro_eje9 <= 10)  
f_tama9=0.91*(Diametro_eje9^(-0.157))  
end
```

```
//factor de concentracion de esfuerzo
```

```
h9=Radio_eje9-Radio_eje8  
relacion9=h9/Radio_muesca8
```

```
//Kt para esfuerzos de flexión (Ktf)
```

```
//0.1<=h/r<=2.0  
C9_f1=0.947+1.206*sqrt(h9/Radio_muesca8)-  
0.131*(h9/Radio_muesca8)  
C9_f2=0.022-  
3.405*sqrt(h9/Radio_muesca8)+0.915*(h9/Radio_muesca8)  
C9_f3=0.869+1.777*sqrt(h9/Radio_muesca8)-  
0.555*(h9/Radio_muesca8)  
C9_f4=-0.810+0.422*sqrt(h9/Radio_muesca8)-  
0.260*(h9/Radio_muesca8)
```

```
//2.0<=h/r<=20  
C9_f11=1.232+0.832*sqrt(h9/Radio_muesca8)-  
0.008*(h9/Radio_muesca8)  
C9_f22=-3.813+0.968*sqrt(h9/Radio_muesca8)-  
0.260*(h9/Radio_muesca8)  
C9_f33=7.423-  
4.868*sqrt(h9/Radio_muesca8)+0.869*(h9/Radio_muesca8)  
C9_f44=-3.839+3.070*sqrt(h9/Radio_muesca8)-  
0.600*(h9/Radio_muesca8)
```

```
if (relacion9 >=0.1 & relacion9 <= 2)
```

```

C9_F1=C9_f1
C9_F2=C9_f2
C9_F3=C9_f3
C9_F4=C9_f4
Kt9_f=C9_F1+C9_F2*(2*h9/Diametro_eje9)+C9_F3*(2*h9/
Diametro_eje9)^2+C9_F4*(2*h9/Diametro_eje9)^3
end

if (relacion9 > 2 & relacion9 <= 20)
  C9_F1=C9_f11
  C9_F2=C9_f22
  C9_F3=C9_f33
  C9_F4=C9_f44
  Kt9_f=C9_F1+C9_F2*(2*h9/Diametro_eje9)+C9_F3*(2*h9/
  Diametro_eje9)^2+C9_F4*(2*h9/Diametro_eje9)^3
end
if relacion9 < 0.1 then
  Kt9_f=1
end
if relacion9 > 0.1 then
  Kt9_f=1
end
Kt9_F=Kt9_f

//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C9_t1=0.905+0.783*sqrt(h9/Radio_muesca8)-
0.075*(h9/Radio_muesca8)
C9_t2=-0.437-
1.969*sqrt(h9/Radio_muesca8)+0.553*(h9/Radio_muesca8)
C9_t3=1.557+1.073*sqrt(h9/Radio_muesca8)-
0.578*(h9/Radio_muesca8)
C9_t4=-1.061+0.171*sqrt(h9/Radio_muesca8)-
0.086*(h9/Radio_muesca8)

if (relacion9 >=0.25 & relacion9 <= 4.0)
  C9_T1=C9_t1
  C9_T2=C9_t2
  C9_T3=C9_t3
  C9_T4=C9_t4
  Kt9_t=C9_T1+C9_T2*(2*h9/Diametro_eje9)+C9_T3*(2*h9/
  Diametro_eje9)^2+C9_T4*(2*h9/Diametro_eje9)^3
end

if relacion9 < 0.25 then
  Kt9_t=1
end
if relacion9 > 4.0 then
  Kt9_t=1
end
Kt9_T=Kt9_t

//constantes para carga axial
//0.1<=h/r<=2.0
C9_a1= 0.926+(1.157*sqrt(h9/Radio_muesca8))-
(0.099*(h9/Radio_muesca8))
C9_a2=0.012-
3.036*sqrt(h9/Radio_muesca8)+0.961*(h9/Radio_muesca8)
C9_a3=-0.302+3.977*sqrt(h9/Radio_muesca8)-
1.744*(h9/Radio_muesca8)
C9_a4=0.365-
2.098*sqrt(h9/Radio_muesca8)+0.878*(h9/Radio_muesca8)

//2.0<=h/r<=20
C9_a11=1.2+0.860*sqrt(h9/Radio_muesca8)-
0.022*(h9/Radio_muesca8)

```

```

C9_a22=-1.805-0.346*sqrt(h9/Radio_muesca8)-
0.038*(h9/Radio_muesca8)
C9_a33=2.198-
0.486*sqrt(h9/Radio_muesca8)+0.165*(h9/Radio_muesca8)
C9_a44=-0.593-0.028*sqrt(h9/Radio_muesca8)-
0.106*(h9/Radio_muesca8)

if (relacion9 >=0.1 & relacion9 <= 2)
  C9_A1=C9_a1
  C9_A2=C9_a2
  C9_A3=C9_a3
  C9_A4=C9_a4
  Kt9_a=C9_A1+C9_A2*(2*h9/Diametro_eje9)+C9_A3*(2*h9/
  Diametro_eje9)^2+C9_A4*(2*h9/Diametro_eje9)^3
end

if (relacion9 > 2 & relacion9 <= 20)
  C9_A1=C9_a11
  C9_A2=C9_a22
  C9_A3=C9_a33
  C9_A4=C9_a44
  Kt9_a=C9_A1+C9_A2*(2*h9/Diametro_eje9)+C9_A3*(2*h9/
  Diametro_eje9)^2+C9_A4*(2*h9/Diametro_eje9)^3
end
if relacion9 < 0.1 then
  Kt9_a=1
end

if relacion9 > 20 then
  Kt9_a=1
end
Kt9_A=Kt9_a

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f9=(2*(Kt9_F-
1)/Kt9_F)*(RAIZ_DE_A/sqrt(Radio_muesca8))
Kf9_f=Kt9_F/((1+constante_reemplazo_f9))
constante_reemplazo_t9=(2*(Kt9_T-
1)/Kt9_T)*(RAIZ_DE_A/sqrt(Radio_muesca8))
Kf9_t=Kt9_T/((1+constante_reemplazo_t9))
constante_reemplazo_a9=(2*(Kt9_A-
1)/Kt9_A)*(RAIZ_DE_A/sqrt(Radio_muesca8))
Kf9_a=Kt9_A/((1+constante_reemplazo_a9))

//CÁLCULO DE LA PRESIÓN POR INTERFERENCIA
m4=(((2*Radio engrane4)^2-
Diametro_eje9^2))*(Diametro_eje9^2))
n4=(2*Radio engrane4)
PP9=((E*Interferen_maza_eje_max4)/(2*Diametro_eje9^3))*
(m4/n4)

Am9=[(PP9*f_carga_axial*Kf9_a/f_tama9)
(taotormed9*f_carga_torsion*Kf9_t/f_tama9)
0;(taotormed9*f_carga_torsion*Kf9_t/f_tama9) 0 0;0 0 0]
//tensor de esfuerzo medio//
[Em9]=spec(Am9)
sigvmm9=(((Em9(1)-Em9(2))^2)+((Em9(2)-
Em9(3))^2)+((Em9(3)-Em9(1))^2))/2^0.5

Aa9=[(siga9*f_carga_flexion*Kf9_f/f_tama9)
(taotort9*f_carga_torsion*Kf9_t/f_tama9) 0;0
(taotort9*f_carga_torsion*Kf9_t/f_tama9) 0;0 0 0] //Tensor
de esfuerzos alternantes
[Ea9]=spec(Aa9)

```

```
sigvma9=(((Ea9(1)-Ea9(2))^2)+((Ea9(2)-Ea9(3))^2)+((Ea9(3)-Ea9(1))^2))/2)^0.5
```

```
V_misses_med_LN9=[sigvmm9,
Coef_variacion_carga*sigvmm9]
V_misses_alt_LN9=[sigvma9,
Coef_variacion_carga*sigvma9]
```

```
[Media_Se,Desv_Se]=mult(Se_Prima,Ka_Kd)
```

```
SE=[Media_Se, Desv_Se]
Sut=[Valor_medio_sut,Desv_sut]
```

```
if criterio_falla == 1 then
[Div_Va_SE_9,
Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
D_Va_SE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]
```

```
[Div_Va_SE_9_Cuad, Desv_Div_Va_SE_9_Cuad]=
pot(D_Va_SE9,2)
Division_VaSE9 = [Div_Va_SE_9_Cuad,
Desv_Div_Va_SE_9_Cuad]
```

```
[Div_Vm_Sut_9,
Desv_Div_Vm_Sut_9]=div(V_misses_med_LN9, Sut)
D_Vm_Sut9 = [Div_Vm_Sut_9, Desv_Div_Vm_Sut_9]
```

```
[Div_Vm_Sut_9_Cuad, Desv_Div_Vm_Sut_9_Cuad]=
pot(D_Vm_Sut9,2)
Division_VmSut9 = [Div_Vm_Sut_9_Cuad,
Desv_Div_Vm_Sut_9_Cuad]
```

```
[Suma9_VmSut,Desv_Suma9_VmSut]
=suma(Division_VaSE9, Division_VmSut9)
```

```
ASME_9=[Suma9_VmSut,Desv_Suma9_VmSut]
Cf_ASME_9=ASME_9(2)/ASME_9(1)
CRITERIO_F=ASME_9
CF_CRITERIO_F=Cf_ASME_9
end
```

```
if criterio_falla == 2 then
[Div_Va_SE_9,
Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
Division_VaSE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]
```

```
[Div_Vm_Sut_9,
Desv_Div_Vm_Sut_9]=div(V_misses_med_LN9, Sut)
D_Vm_Sut9 = [Div_Vm_Sut_9, Desv_Div_Vm_Sut_9]
```

```
[Div_Vm_Sut_9_Cuad, Desv_Div_Vm_Sut_9_Cuad]=
pot(D_Vm_Sut9,2)
Division_VmSut9 = [Div_Vm_Sut_9_Cuad,
Desv_Div_Vm_Sut_9_Cuad]
```

```
[Suma9_VmSut,Desv_Suma9_VmSut]
=suma(Division_VaSE9, Division_VmSut9)
```

```
GERBER_9=[Suma9_VmSut,Desv_Suma9_VmSut]
Cf_GERBER_9=GERBER_9(2)/GERBER_9(1)
CRITERIO_F=GERBER_9
CF_CRITERIO_F=Cf_GERBER_9
end
```

```
if criterio_falla == 3 then
[Div_Va_SE_9,
Desv_Div_Va_SE_9]=div(V_misses_alt_LN9, SE)
```

```
Division_VaSE9 = [Div_Va_SE_9, Desv_Div_Va_SE_9]
[Div_Vm_Sut_9,
Desv_Div_Vm_Sut_9]=div(V_misses_med_LN9, Sut)
Division_VmSut9 = [Div_Vm_Sut_9, Desv_Div_Vm_Sut_9]
```

```
[Suma9_VmSut,Desv_Suma9_VmSut]
=suma(Division_VaSE9, Division_VmSut9)
GOODMAN_9=[Suma9_VmSut,Desv_Suma9_VmSut]
Cf_GOODMAN_9=GOODMAN_9(2)/GOODMAN_9(1)
CRITERIO_F=GOODMAN_9
CF_CRITERIO_F=Cf_GOODMAN_9
end
```

```
MEDCnormal_9=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_9=CF_CRITERIO_F
```

```
[Prob_9,Conf_9]=cdfnor("PQ",0,abs(MEDCnormal_9),DESV
normal_9)
```

```
DIAMETRO_EJE9=0
```

```
if Prob_9<=1.5*10^-6 & Prob_9>=0.5*10^-6 then
DIAMETRO_EJE9=Diametro_eje9
break;
end
end
disp("El diametro del tramo 9 es")
disp(DIAMETRO_EJE9)
```

```
//ITERACIÓN DEL DÉCIMO DIÁMETRO:
for Diametro_eje10=tol:tol:6
```

```
Radio_eje10=Diametro_eje10/2
//MOMENTO DE INERCIA:
I10=(%pi*((Radio_eje10^2)^4))/4)
//MOMENTO DE ÁREA:
Q10=(2/3)*(Radio_eje10^3)
//MOMENTO POLAR DE INERCIA:
J10=I10*2
//CÁLCULO DE ESFUERZOS:
//Esfuerzo flector máximo:
sig10=(Momento_max10*Radio_eje10/I10)
```

```
//FACTOR DE TAMAÑO
if (Diametro_eje10 < 0.11)
f_tama10=1
end
if (Diametro_eje10 >= 0.11 & Diametro_eje10 <= 2)
f_tama10=0.879*Diametro_eje10^-0.107
end
if (Diametro_eje10 >= 3 & Diametro_eje10 <= 10)
f_tama10=0.91*(Diametro_eje10^(-0.157))
end
```

```
//Factor de concentración de esfuerzo
h10=Radio_eje10-Radio_eje9
relacion10=h10/Radio_muesca9
```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON:
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON PARA FLEXIÓN:
//0.1<=h/r<=2.0
C7_f1=0.947+1.206*sqrt(h10/Radio_muesca9)-
0.131*(h10/Radio_muesca9)
```

```

C7_f2=0.022-
3.405*sqrt(h10/Radio_muesca9)+0.915*(h10/Radio_muesca9)
C7_f3=0.869+1.777*sqrt(h10/Radio_muesca9)-
0.555*(h10/Radio_muesca9)
C7_f4=-0.810+0.422*sqrt(h10/Radio_muesca9)-
0.260*(h10/Radio_muesca9)

```

```

//2.0<=h/r<=20
C7_f11=1.232+0.832*sqrt(h10/Radio_muesca9)-
0.008*(h10/Radio_muesca9)
C7_f22=-3.813+0.968*sqrt(h10/Radio_muesca9)-
0.260*(h10/Radio_muesca9)
C7_f33=7.423-
4.868*sqrt(h10/Radio_muesca9)+0.869*(h10/Radio_muesca9)
C7_f44=-3.839+3.070*sqrt(h10/Radio_muesca9)-
0.600*(h10/Radio_muesca9)

```

```

if (relacion10 >=0.1 & relacion10 <= 2)
  C7_F1=C7_f1
  C7_F2=C7_f2
  C7_F3=C7_f3
  C7_F4=C7_f4

```

```

Kt10_f=C7_F1+C7_F2*(2*h10/Diametro_eje10)+C7_F3*(2*
h10/Diametro_eje10)^2+C7_F4*(2*h10/Diametro_eje10)^3
end

```

```

if (relacion10 > 2 & relacion10 <= 20)
  C7_F1=C7_f11
  C7_F2=C7_f22
  C7_F3=C7_f33
  C7_F4=C7_f44

```

```

Kt10_f=C7_F1+C7_F2*(2*h10/Diametro_eje10)+C7_F3*(2*
h10/Diametro_eje10)^2+C7_F4*(2*h10/Diametro_eje10)^3
end

```

```

if relacion10 < 0.1 then
  Kt10_f=1
end

```

```

if relacion10 > 20 then
  Kt10_f=1
end

```

```

Kt10_F=Kt10_f

```

```

//Kt para esfuerzos de torsión (Ktf)

```

```

//0.25<=h/r<=4.0
C10_t1=0.905+0.783*sqrt(h10/Radio_muesca9)-
0.075*(h10/Radio_muesca9)
C10_t2=-0.437-
1.969*sqrt(h10/Radio_muesca9)+0.553*(h10/Radio_muesca9)
C10_t3=1.557+1.073*sqrt(h10/Radio_muesca9)-
0.578*(h10/Radio_muesca9)
C10_t4=-1.061+0.171*sqrt(h10/Radio_muesca9)-
0.086*(h10/Radio_muesca9)

```

```

if (relacion10 >=0.25 & relacion10 <= 4.0)
  C10_T1=C10_t1
  C10_T2=C10_t2
  C10_T3=C10_t3
  C10_T4=C10_t4
  Kt10_t=C10_T1+C10_T2*(2*h10/Diametro_eje10)+C10_T3*
(2*h10/Diametro_eje10)^2+C10_T4*(2*h10/Diametro_eje10)
^3
end

```

```

if relacion10 < 0.25 then
  Kt10_t=1
end
if relacion10 > 4.0 then
  Kt10_t=1
end
Kt10_T=Kt10_t

```

```

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

```

```

constante_reemplazo_f10=(2*(Kt10_F-
1)/Kt10_F)*(RAIZ_DE_A/sqrt(Radio_muesca9))
Kf10_f=Kt10_F/((1+constante_reemplazo_f10))
constante_reemplazo_t10=(2*(Kt10_T-
1)/Kt10_T)*(RAIZ_DE_A/sqrt(Radio_muesca9))
Kf10_t=Kt10_T/((1+constante_reemplazo_t10))

```

```

Torque10=Torque9

```

```

//Esfuerzo cortante torsor alternante:

```

```

taotormed10=(Torque10*Radio_eje10/J10)

```

```

//Esfuerzo cortante torsor medio:

```

```

taotoralt10=(Torque10*Radio_eje10/J10)

```

```

//TENSOR DE ESFUERZOS MEDIOS:

```

```

Am10=[0 (taotormed10*f_carga_torsion*Kf10_t/f_tama10)
0;(taotormed10*f_carga_torsion*Kf10_t/f_tama10) 0 0 0 0]

```

```

//CÁLCULO DE AUTOVALORES:

```

```

[Em10]=spec(Am10)

```

```

//CRITERIO DE VON-MISSES:

```

```

sigvmm10=(((Em10(1)-Em10(2))^2)+((Em10(2)-
Em10(3))^2)+((Em10(3)-Em10(1))^2)/2)^0.5

```

```

//TENSOR DE ESFUERZOS ALTERNANTES:

```

```

Aa10=[(sigma10*f_carga_flexion*Kf10_f/f_tama10)
(taotoralt10*f_carga_torsion*Kf10_t/f_tama10) 0;0
(taotoralt10*f_carga_torsion*Kf10_t/f_tama10) 0;0 0 0]

```

```

//CÁLCULO DE AUTOVALORES:

```

```

[Ea10]=spec(Aa10)

```

```

//CRITERIO DE VON-MISSES:

```

```

sigvma10=(((Ea10(1)-Ea10(2))^2)+((Ea10(2)-
Ea10(3))^2)+((Ea10(3)-Ea10(1))^2)/2)^0.5

```

```

V_misses_med_LN10=[sigvmm10,
Coef_variacion_carga*sigvmm10]

```

```

V_misses_alt_LN10=[sigvma10,
Coef_variacion_carga*sigvma10]

```

```

//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:

```

```

if criterio_falla == 1 then

```

```

  [Div_Va_SE_10,

```

```

  Desv_Div_Va_SE_10]=div(V_misses_alt_LN10, SE)

```

```

  D_Va_SE10 = [Div_Va_SE_10, Desv_Div_Va_SE_10]

```

```

  [Div_Va_SE_10_Cuad, Desv_Div_Va_SE_10_Cuad]=
  pot(D_Va_SE10,2)

```

```

  Division_VaSE10 = [Div_Va_SE_10_Cuad,
  Desv_Div_Va_SE_10_Cuad]

```

```

  Division_VmSut10 = [0,0]

```

```

  Suma10_VmSut_VaSE=Division_VaSE10(1)+Division_VmS
  ut10(1)

```

```

  Desv_Suma10_VmSut_VaSE=sqrt((Division_VaSE10(2)/Divi
  sion_VaSE10(1))^2+(0)^2)

```

```

  ASME_10=[Suma10_VmSut_VaSE,
  Desv_Suma10_VmSut_VaSE]

```

```

Cf_ASME_10=ASME_10(2)/ASME_10(1)
CRITERIO_F=ASME_10
CF_CRITERIO_F=Cf_ASME_10
end

if criterio_falla == 2 then
|Div_Va_SE_10,
Desv_Div_Va_SE_10]=div(V_misses_alt_LN10, SE)
Division_VaSE10 = [Div_Va_SE_10, Desv_Div_Va_SE_10]

Division_VmSut10 = [0,0]

Suma10_VmSut_VaSE=Division_VaSE10(1)+Division_VmSut10(1)
Desv_Suma10_VmSut_VaSE=sqrt((Division_VaSE10(2)/Division_VaSE10(1))^2+(0)^2)

GERBER_10=[Suma10_VmSut_VaSE,
Desv_Suma10_VmSut_VaSE]
Cf_GERBER_10=GERBER_10(2)/GERBER_10(1)
CRITERIO_F=GERBER_10
CF_CRITERIO_F=Cf_GERBER_10
end

if criterio_falla == 3 then
|Div_Va_SE_10,
Desv_Div_Va_SE_10]=div(V_misses_alt_LN10, SE)
Division_Va_SE10 = [Div_Va_SE_10, Desv_Div_Va_SE_10]

Division_VmSut10 = [0,0]

Suma10_VmSut_VaSE=Division_Va_SE10(1)+Division_VmSut10(1)
Desv_Suma10_VmSut_VaSE=sqrt((Division_Va_SE10(2)/Division_Va_SE10(1))^2+(0)^2)
GOODMAN_10=[Suma10_VmSut_VaSE,
Desv_Suma10_VmSut_VaSE]
Cf_GOODMAN_10=GOODMAN_10(2)/GOODMAN_10(1)
CRITERIO_F=GOODMAN_10
CF_CRITERIO_F=Cf_GOODMAN_10
end

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A LA DISTRIBUCIÓN NORMAL
MEDCnormal_10=log(CRITERIO_F(1))-((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_10=CF_CRITERIO_F

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_10,Conf_10]=cdfnor("PQ",0,abs(MEDCnormal_10),DESVnormal_10)

DIAMETRO_EJE10=0

if Prob_10<=1.5*10^-6 & Prob_10>=0.5*10^-6 then
DIAMETRO_EJE10=Diametro_eje10
break;
end
end
disp("El diametro del tramo 10 es")
disp(DIAMETRO_EJE10)

//ITERACIÓN DEL UNDÉCIMO DIÁMETRO:
for Diametro_eje11=tol:tol:6

Radio_eje11=Diametro_eje11/2
//MOMENTO DE INERCIA:

```

```

I11=(%pi*((Radio_eje11*2)^4))/4
//MOMENTO DE ÁREA:
Q11=(2/3)*(Radio_eje11^3)
//MOMENTO POLAR DE INERCIA:
J11=I11*2
//CÁLCULO DE ESFUERZOS:
//Esfuerzo flector máximo:
sigal1=(Momento_max11*Radio_eje11/I11)
Torque11=W_tangencial5*((Radio_engrane5)-Radio_eje11)
taotormed11=(Torque11*Radio_eje11/J11)
//Esfuerzo cortante torsor medio:
taotoralt11=(Torque11*Radio_eje11/J11)

//FACTOR DE TAMAÑO
if (Diametro_eje11 < 0.11)
f_tamall=1
end
if (Diametro_eje11 >= 0.11 & Diametro_eje11 <= 2)
f_tamall=0.879*Diametro_eje11^-0.107
end
if (Diametro_eje11 >= 3 & Diametro_eje11 <= 10)
f_tamall=0.91*(Diametro_eje11^(-0.157))
end

//Factor de concentración de esfuerzo
h11=Radio_eje11-Radio_eje10
relacion11=h11/Radio_muesca10

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE PETERSON:
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE PETERSON PARA FLEXIÓN:
//0.1<=h/r<=2.0
C7_f1=0.947+1.206*sqrt(h11/Radio_muesca10)-0.131*(h11/Radio_muesca10)
C7_f2=0.022-3.405*sqrt(h11/Radio_muesca10)+0.915*(h11/Radio_muesca10)
C7_f3=0.869+1.777*sqrt(h11/Radio_muesca10)-0.555*(h11/Radio_muesca10)
C7_f4=-0.810+0.422*sqrt(h11/Radio_muesca10)-0.260*(h11/Radio_muesca10)

//2.0<=h/r<=20
C7_f11=1.232+0.832*sqrt(h11/Radio_muesca10)-0.008*(h11/Radio_muesca10)
C7_f22=-3.813+0.968*sqrt(h11/Radio_muesca10)-0.260*(h11/Radio_muesca10)
C7_f33=7.423-4.868*sqrt(h11/Radio_muesca10)+0.869*(h11/Radio_muesca10)
C7_f44=-3.839+3.070*sqrt(h11/Radio_muesca10)-0.600*(h11/Radio_muesca10)

if (relacion11 >=0.1 & relacion11 <= 2)
C7_F1=C7_f1
C7_F2=C7_f2
C7_F3=C7_f3
C7_F4=C7_f4
Kt11_f=C7_F1+C7_F2*(2*h11/Diametro_eje11)+C7_F3*(2*h11/Diametro_eje11)^2+C7_F4*(2*h11/Diametro_eje11)^3
end

if (relacion11 > 2 & relacion11 <= 20)
C7_F1=C7_f11
C7_F2=C7_f22
C7_F3=C7_f33

```

```

C7_F4=C7_f44
Kt11_f=C7_F1+C7_F2*(2*h11/Diametro_eje11)+C7_F3*(2*
h11/Diametro_eje11)^2+C7_F4*(2*h11/Diametro_eje11)^3
end

if relacion11 < 0.1 then
  Kt11_f=1
end

if relacion11 > 20 then
  Kt11_f=1
end

KT11_F=Kt11_f
//Kt para esfuerzos de torsión (Ktf)

//0.25<=h/r<=4.0
C11_t1=0.905+0.783*sqrt(h9/Radio_muesca10)-
0.075*(h9/Radio_muesca10)
C11_t2=-0.437-
1.969*sqrt(h9/Radio_muesca10)+0.553*(h9/Radio_muesca10)
C11_t3=1.557+1.073*sqrt(h9/Radio_muesca10)-
0.578*(h9/Radio_muesca10)
C11_t4=-1.061+0.171*sqrt(h9/Radio_muesca10)-
0.086*(h9/Radio_muesca10)

if (relacion11 >=0.25 & relacion11 <= 4.0)
  C11_T1=C11_t1
  C11_T2=C11_t2
  C11_T3=C11_t3
  C11_T4=C11_t4
Kt11_t=C11_T1+C11_T2*(2*h11/Diametro_eje11)+C11_T3*
(2*h11/Diametro_eje11)^2+C11_T4*(2*h11/Diametro_eje11)
^3
end

if relacion11 < 0.25 then
  Kt11_t=1
end
if relacion11 > 4.0 then
  Kt11_t=1
end
KT11_T=Kt11_t

//constantes para carga axial
//0.1<=h/r<=2.0
C11_a1=0.926+(1.157*sqrt(h11/Radio_muesca10))-
(0.099*(h11/Radio_muesca10))
C11_a2=0.012-
3.036*sqrt(h11/Radio_muesca10)+0.961*(h11/Radio_muesca1
0)
C11_a3=-0.302+3.977*sqrt(h11/Radio_muesca10)-
1.744*(h11/Radio_muesca10)
C11_a4=0.365-
2.098*sqrt(h11/Radio_muesca10)+0.878*(h11/Radio_muesca1
0)

//2.0<=h/r<=20
C11_a11=1.2+0.860*sqrt(h11/Radio_muesca10)-
0.022*(h11/Radio_muesca10)
C11_a22=-1.805-0.346*sqrt(h11/Radio_muesca10)-
0.038*(h11/Radio_muesca10)
C11_a33=2.198-
0.486*sqrt(h11/Radio_muesca10)+0.165*(h11/Radio_muesca1
0)
C11_a44=-0.593-0.028*sqrt(h11/Radio_muesca10)-
0.106*(h11/Radio_muesca10)

if (relacion11 >=0.1 & relacion11 <= 2)
  C11_A1=C11_a1
  C11_A2=C11_a2
  C11_A3=C11_a3
  C11_A4=C11_a4
Kt11_a=C11_A1+C11_A2*(2*h11/Diametro_eje11)+C11_A3
*(2*h11/Diametro_eje11)^2+C11_A4*(2*h11/Diametro_eje11)
^3
end

if (relacion11 > 2 & relacion11 <= 20)
  C11_A1=C11_a11
  C11_A2=C11_a22
  C11_A3=C11_a33
  C11_A4=C11_a44
Kt11_a=C11_A1+C11_A2*(2*h11/Diametro_eje11)+C11_A3
*(2*h11/Diametro_eje11)^2+C11_A4*(2*h11/Diametro_eje11)
^3
end
if relacion11 < 0.1 then
  Kt11_a=1
end
if relacion11 > 20 then
  Kt11_a=1
end
KT11_A=Kt11_a

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f- flexión), (Kf_t-torsión)

constante_reemplazo_f11=(2*(KT11_F-
1)/KT11_F)*(RAIZ_DE_A/sqrt(Radio_muesca10))
Kf11_f=KT11_F/((1+constante_reemplazo_f11))
constante_reemplazo_t11=(2*(KT11_T-
1)/KT11_T)*(RAIZ_DE_A/sqrt(Radio_muesca10))
Kf11_t=KT11_T/((1+constante_reemplazo_t11))
constante_reemplazo_a11=(2*(KT11_A-
1)/KT11_A)*(RAIZ_DE_A/sqrt(Radio_muesca10))
Kf11_a=KT11_A/((1+constante_reemplazo_a11))

//CÁLCULO DE LA PRESIÓN POR INTERFERENCIA
m5=(((2*Radio engrane5)^2-
Diametro_eje11^2)*(Diametro_eje11^2))
n5=(2*Radio engrane5)
PP11=((E*Interferen_maza_eje_max5)/(2*Diametro_eje11^3)
)*(m5/n5)

//TENSOR DE ESFUERZOS MEDIOS:
Am11=[(PP11*f_carga_axial*Kf11_a/f_tama11)
(taotormed11*f_carga_torsion*Kf11_t/f_tama11)
0;(taotormed11*f_carga_torsion*Kf11_t/f_tama11) 0 0 0 0]
//tensor de esfuerzo medio//
//CÁLCULO DE AUTOVALORES:
[Em11]=spec(Am11)
//CRITERIO DE VON-MISSES:
sigvmm11=(((Em11(1)-Em11(2))^2)+((Em11(2)-
Em11(3))^2)+((Em11(3)-Em11(1))^2))/2^0.5

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa11=[(sigma11*f_carga_flexion*Kf11_f/f_tama11)
(taotort11*f_carga_torsion*Kf11_t/f_tama11) 0;0 0 0]
(taotort11*f_carga_torsion*Kf11_t/f_tama11) 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Ea11]=spec(Aa11)
//CRITERIO DE VON-MISSES:

```



```
sigvma11=(((Ea11(1)-Ea11(2))^2)+((Ea11(2)-Ea11(3))^2)+((Ea11(3)-Ea11(1))^2))/2)^0.5
```

```
V_misses_med_LN11=[sigvmm11,
Coef_variacion_carga*sigvmm11]
V_misses_alt_LN11=[sigvma11,
Coef_variacion_carga*sigvma11]
```

```
//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:
```

```
if criterio_falla == 1 then
[Div_Va_SE_11,
Desv_Div_Va_SE_11]=div(V_misses_alt_LN11, SE)
D_Va_SE11 = [Div_Va_SE_11, Desv_Div_Va_SE_11]
```

```
[Div_Va_SE_11_Cuad, Desv_Div_Va_SE_11_Cuad]=
pot(D_Va_SE11,2)
Division_VaSE11 = [Div_Va_SE_11_Cuad,
Desv_Div_Va_SE_11_Cuad]
```

```
Division_VmSut11 = [0,0]
```

```
Suma11_VmSut_VaSE=Division_VaSE11(1)+Division_VmS
ut11(1)
Desv_Suma11_VmSut_VaSE=sqrt((Division_VaSE11(2)/Divi
sion_VaSE11(1))^2+(0)^2)
```

```
ASME_11=[Suma11_VmSut_VaSE,
Desv_Suma11_VmSut_VaSE]
Cf_ASME_11=ASME_11(2)/ASME_11(1)
CRITERIO_F=ASME_11
CF_CRITERIO_F=Cf_ASME_11
end
```

```
if criterio_falla == 2 then
[Div_Va_SE_11,
Desv_Div_Va_SE_11]=div(V_misses_alt_LN11, SE)
Division_VaSE11 = [Div_Va_SE_11, Desv_Div_Va_SE_11]
```

```
Division_VmSut11 = [0,0]
```

```
Suma11_VmSut_VaSE=Division_VaSE11(1)+Division_VmS
ut11(1)
Desv_Suma11_VmSut_VaSE=sqrt((Division_VaSE11(2)/Divi
sion_VaSE11(1))^2+(0)^2)
```

```
GERBER_11=[Suma11_VmSut_VaSE,
Desv_Suma11_VmSut_VaSE]
Cf_GERBER_11=GERBER_11(2)/GERBER_11(1)
CRITERIO_F=GERBER_11
CF_CRITERIO_F=Cf_GERBER_11
end
```

```
if criterio_falla == 3 then
[Div_Va_SE_11,
Desv_Div_Va_SE_11]=div(V_misses_alt_LN11, SE)
Division_Va_SE11 = [Div_Va_SE_11, Desv_Div_Va_SE_11]
```

```
Division_VmSut11 = [0,0]
```

```
Suma11_VmSut_VaSE=Division_Va_SE11(1)+Division_Vm
Sut11(1)
Desv_Suma11_VmSut_VaSE=sqrt((Division_Va_SE11(2)/Di
vision_Va_SE11(1))^2+(0)^2)
GOODMAN_11=[Suma11_VmSut_VaSE,
Desv_Suma11_VmSut_VaSE]
Cf_GOODMAN_11=GOODMAN_11(2)/GOODMAN_11(1)
CRITERIO_F=GOODMAN_11
```

```
CF_CRITERIO_F=Cf_GOODMAN_11
end
```

```
//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL
```

```
MEDCnormal_11=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_11=CF_CRITERIO_F
```

```
//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
```

```
[Prob_11,Conf_11]=cdfnor("PQ",0,abs(MEDCnormal_11),DE
SVnormal_11)
```

```
DIAMETRO_EJE11=0
```

```
if Prob_11<=1.5*10^-6 & Prob_11>=0.5*10^-6 then
DIAMETRO_EJE11=Diametro_eje11
break;
```

```
end
end
disp("El diametro del tramo 11 es")
disp(DIAMETRO_EJE11)
```

```
//ITERACIÓN DEL 12 DIÁMETRO:
```

```
for Diametro_eje12=tol:tol:6
```

```
Radio_eje12=Diametro_eje12/2
```

```
//MOMENTO DE INERCIA:
```

```
I12=(%pi*((Radio_eje12^2)^4))/4)
```

```
//MOMENTO DE ÁREA:
```

```
Q12=(2/3)*(Radio_eje12^3)
```

```
//MOMENTO POLAR DE INERCIA:
```

```
J12=I12*2
```

```
//CÁLCULO DE ESFUERZOS:
```

```
//Esfuerzo flector máximo:
```

```
sigla12=(Momento_max12*Radio_eje12/I12)
```

```
//FACTOR DE TAMAÑO
```

```
if (Diametro_eje12 < 0.11)
```

```
f_tama12=1
```

```
end
```

```
if (Diametro_eje12 >= 0.11 & Diametro_eje12 <= 2)
```

```
f_tama12=0.879*Diametro_eje12^-0.107
```

```
end
```

```
if (Diametro_eje12 >= 3 & Diametro_eje12 <= 10)
```

```
f_tama12=0.91*(Diametro_eje12^(-0.157))
```

```
end
```

```
//Factor de concentración de esfuerzo
```

```
h12=Radio_eje12-Radio_eje11
```

```
relacion12=h12/Radio_muesca11
```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON:
```

```
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETERSON PARA FLEXIÓN:
```

```
//0.1<=h/r<=2.0
```

```
C7_f1=0.947+1.206*sqrt(h12/Radio_muesca11)-
```

```
0.131*(h12/Radio_muesca11)
```

```
C7_f2=0.022-
```

```
3.405*sqrt(h12/Radio_muesca11)+0.915*(h12/Radio_muesca11)
```

```
1)
```

```
C7_f3=0.869+1.777*sqrt(h12/Radio_muesca11)-
```

```
0.555*(h12/Radio_muesca11)
```

```
C7_f4=-0.810+0.422*sqrt(h12/Radio_muesca11)-
```

```
0.260*(h12/Radio_muesca11)
```

```

//2.0<=h/r<=20
C7_f11=1.232+0.832*sqrt(h12/Radio_muesca11)-
0.008*(h12/Radio_muesca11)
C7_f22=-3.813+0.968*sqrt(h12/Radio_muesca11)-
0.260*(h12/Radio_muesca11)
C7_f33=7.423-
4.868*sqrt(h12/Radio_muesca11)+0.869*(h12/Radio_muesca11)
C7_f44=-3.839+3.070*sqrt(h12/Radio_muesca11)-
0.600*(h12/Radio_muesca11)

if (relacion12 >=0.1 & relacion12 <= 2)
  C7_F1=C7_f1
  C7_F2=C7_f2
  C7_F3=C7_f3
  C7_F4=C7_f4
Kt12_f=C7_F1+C7_F2*(2*h12/Diametro_eje12)+C7_F3*(2*
h12/Diametro_eje12)^2+C7_F4*(2*h12/Diametro_eje12)^3
end

if (relacion12 > 2 & relacion12 <= 20)
  C7_F1=C7_f11
  C7_F2=C7_f22
  C7_F3=C7_f33
  C7_F4=C7_f44
Kt12_f=C7_F1+C7_F2*(2*h12/Diametro_eje12)+C7_F3*(2*
h12/Diametro_eje12)^2+C7_F4*(2*h12/Diametro_eje12)^3
end

if relacion12 < 0.1 then
  Kt12_f=1
end

if relacion12 > 20 then
  Kt12_f=1
end

KT12_F=Kt12_f

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f12=(2*(KT12_F-
1)/KT12_F)*(RAIZ_DE_A/sqrt(Radio_muesca11))
Kf12_f=KT12_F/((1+constante_reemplazo_f12))

//TENSOR DE ESFUERZOS MEDIOS:
Am12=[0 0 0;0 0 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Em12]=spec(Am12)
//CRITERIO DE VON-MISES:
sigvmm12=(((Em12(1)-Em12(2))^2)+((Em12(2)-
Em12(3))^2)+((Em12(3)-Em12(1))^2))/2^0.5

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa12=[(sigal2*f_carga_flexion*Kf12_f/f_tama12) 0 0;0 0 0;0
0 0]
//CÁLCULO DE AUTOVALORES:
[Ea12]=spec(Aa12)
//CRITERIO DE VON-MISES:
sigvma12=(((Ea12(1)-Ea12(2))^2)+((Ea12(2)-
Ea12(3))^2)+((Ea12(3)-Ea12(1))^2))/2^0.5

V_misses_med_LN12=[sigvmm12,
Coef_variacion_carga*sigvmm12]
V_misses_alt_LN12=[sigvma12,
Coef_variacion_carga*sigvma12]

```

//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:

```

if criterio_falla == 1 then
  [Div_Va_SE_12,
Desv_Div_Va_SE_12]=div(V_misses_alt_LN12, SE)
D_Va_SE12 = [Div_Va_SE_12, Desv_Div_Va_SE_12]

[Div_Va_SE_12_Cuad, Desv_Div_Va_SE_12_Cuad]=
pot(D_Va_SE12,2)
Division_VaSE12 = [Div_Va_SE_12_Cuad,
Desv_Div_Va_SE_12_Cuad]

Division_VmSut12 = [0,0]

Suma12_VmSut_VaSE=Division_VaSE12(1)+Division_VmS
ut12(1)
Desv_Suma12_VmSut_VaSE=sqrt((Division_VaSE12(2)/Divi
sion_VaSE12(1))^2+(0)^2)

ASME_12=[Suma12_VmSut_VaSE,
Desv_Suma12_VmSut_VaSE]
Cf_ASME_12=ASME_12(2)/ASME_12(1)
CRITERIO_F=ASME_12
CF_CRITERIO_F=Cf_ASME_12
end

if criterio_falla == 2 then
  [Div_Va_SE_12,
Desv_Div_Va_SE_12]=div(V_misses_alt_LN12, SE)
Division_VaSE12 = [Div_Va_SE_12, Desv_Div_Va_SE_12]

Division_VmSut12 = [0,0]

Suma12_VmSut_VaSE=Division_VaSE12(1)+Division_VmS
ut12(1)
Desv_Suma12_VmSut_VaSE=sqrt((Division_VaSE12(2)/Divi
sion_VaSE12(1))^2+(0)^2)

GERBER_12=[Suma12_VmSut_VaSE,
Desv_Suma12_VmSut_VaSE]
Cf_GERBER_12=GERBER_12(2)/GERBER_12(1)
CRITERIO_F=GERBER_12
CF_CRITERIO_F=Cf_GERBER_12
end

if criterio_falla == 3 then
  [Div_Va_SE_12,
Desv_Div_Va_SE_12]=div(V_misses_alt_LN12, SE)
Division_Va_SE12 = [Div_Va_SE_12, Desv_Div_Va_SE_12]

Division_VmSut12 = [0,0]

Suma12_VmSut_VaSE=Division_Va_SE12(1)+Division_Vm
Sut12(1)
Desv_Suma12_VmSut_VaSE=sqrt((Division_Va_SE12(2)/Di
vision_Va_SE12(1))^2+(0)^2)
GOODMAN_12=[Suma12_VmSut_VaSE,
Desv_Suma12_VmSut_VaSE]
Cf_GOODMAN_12=GOODMAN_12(2)/GOODMAN_12(1)
CRITERIO_F=GOODMAN_12
CF_CRITERIO_F=Cf_GOODMAN_12
end

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL
MEDCnormal_12=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)

```



```

DESVnormal_12=CF_CRITERIO_F

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_12,Conf_12]=cdfnorf("PQ",0,abs(MEDCnormal_12),DE
SVnormal_12)

DIAMETRO_EJE12=0

if Prob_12<=1.5*10^-6 & Prob_12>=0.5*10^-6 then
    DIAMETRO_EJE12=Diametro_eje12
    break;
end
end
disp("El diametro del tramo 12 es")
disp(DIAMETRO_EJE12)

//ITERACIÓN DEL 13 DIÁMETRO:
for Diametro_eje13=tol:tol:6

Radio_eje13=Diametro_eje13/2
//MOMENTO DE INERCIA:
I13=(%pi*(((Radio_eje13^2)^4))/4)
//MOMENTO DE ÁREA:
Q13=(2/3)*(Radio_eje13^3)
//MOMENTO POLAR DE INERCIA:
J13=I13*2
//CÁLCULO DE ESFUERZOS:
//Esfuerzo flector máximo:
sigal3=(Momento_max13*Radio_eje13/I13)

//FACTOR DE TAMAÑO
if (Diametro_eje13 < 0.11)
    f_tama13=1
end
if (Diametro_eje13 >= 0.11 & Diametro_eje13 <= 2)
    f_tama13=0.879*Diametro_eje13^-0.107
end
if (Diametro_eje13 >= 3 & Diametro_eje13 <= 10)
    f_tama13=0.91*(Diametro_eje13^(-0.157))
end

//Factor de concentración de esfuerzo
h13=Radio_eje13-Radio_eje12
relacion13=h13/Radio_muesca12

//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON:
//CONSTANTES DE CONCENTRADOR DE ESFUERZO DE
PETTERSON PARA FLEXIÓN:
//0.1<=h/r<=2.0
C7_f1=0.947+1.206*sqrt(h13/Radio_muesca12)-
0.131*(h13/Radio_muesca12)
C7_f2=0.022-
3.405*sqrt(h13/Radio_muesca12)+0.915*(h13/Radio_muesca1
2)
C7_f3=0.869+1.777*sqrt(h13/Radio_muesca12)-
0.555*(h13/Radio_muesca12)
C7_f4=-0.810+0.422*sqrt(h13/Radio_muesca12)-
0.260*(h13/Radio_muesca12)

//2.0<=h/r<=20
C7_f11=1.232+0.832*sqrt(h13/Radio_muesca12)-
0.008*(h13/Radio_muesca12)
C7_f22=-3.813+0.968*sqrt(h13/Radio_muesca12)-
0.260*(h13/Radio_muesca12)

C7_f33=7.423-
4.868*sqrt(h13/Radio_muesca12)+0.869*(h13/Radio_muesca1
2)
C7_f44=-3.839+3.070*sqrt(h13/Radio_muesca12)-
0.600*(h13/Radio_muesca12)

if (relacion13 >=0.1 & relacion13 <= 2)
    C7_F1=C7_f1
    C7_F2=C7_f2
    C7_F3=C7_f3
    C7_F4=C7_f4
    Kt13_f=C7_F1+C7_F2*(2*h13/Diametro_eje13)+C7_F3*(2*
h13/Diametro_eje13)^2+C7_F4*(2*h13/Diametro_eje13)^3
end

if (relacion13 > 2 & relacion13 <= 20)
    C7_F1=C7_f11
    C7_F2=C7_f22
    C7_F3=C7_f33
    C7_F4=C7_f44
    Kt13_f=C7_F1+C7_F2*(2*h13/Diametro_eje13)+C7_F3*(2*
h13/Diametro_eje13)^2+C7_F4*(2*h13/Diametro_eje13)^3
end

if relacion13 < 0.1 then
    Kt13_f=1
end

if relacion13 > 20 then
    Kt13_f=1
end

KT13_F=Kt13_f

//Factor de concentración de esfuerzo de fatiga(Kf_a - axial),
(Kf_f-flexión), (Kf_t-torsión)

constante_reemplazo_f13=(2*(KT13_F-
1)/KT13_F)*(RAIZ_DE_A/sqrt(Radio_muesca12))
Kf13_f=KT13_F/((1+constante_reemplazo_f13))

//TENSOR DE ESFUERZOS MEDIOS:
Am13=[0 0 0;0 0 0;0 0 0]
//CÁLCULO DE AUTOVALORES:
[Em13]=spec(Am13)
//CRITERIO DE VON-MISSES:
sigvmm13=(((Em13(1)-Em13(2))^2)+((Em13(2)-
Em13(3))^2)+((Em13(3)-Em13(1))^2))/2^0.5

//TENSOR DE ESFUERZOS ALTERNANTES:
Aa13=[(sigal3*f_carga_flexion*Kf13_f/f_tama13) 0 0;0 0 0;0
0 0]
//CÁLCULO DE AUTOVALORES:
[Ea13]=spec(Aa13)
//CRITERIO DE VON-MISSES:
sigvma13=(((Ea13(1)-Ea13(2))^2)+((Ea13(2)-
Ea13(3))^2)+((Ea13(3)-Ea13(1))^2))/2^0.5

V_misses_med_LN13=[sigvmm13,
Coef_variacion_carga*sigvmm13]
V_misses_alt_LN13=[sigvma13,
Coef_variacion_carga*sigvma13]

//EVALUACIÓN DEL CRITERIO DE FALLA DESEADO:
if criterio_falla == 1 then
    [Div_Va_SE_13,
Desv_Div_Va_SE_13]=div(V_misses_alt_LN13, SE)

```

```

D_Va_SE13 = [Div_Va_SE_13, Desv_Div_Va_SE_13]

[Div_Va_SE_13_Cuad,      Desv_Div_Va_SE_13_Cuad]=
pot(D_Va_SE13,2)
Division_VaSE13      =      [Div_Va_SE_13_Cuad,
Desv_Div_Va_SE_13_Cuad]

Division_VmSut13 = [0,0]

Suma13_VmSut_VaSE=Division_VaSE13(1)+Division_VmS
ut13(1)
Desv_Suma13_VmSut_VaSE=sqrt((Division_VaSE13(2)/Divi
sion_VaSE13(1))^2+(0)^2)

ASME_13=[Suma13_VmSut_VaSE,
Desv_Suma13_VmSut_VaSE]
Cf_ASME_13=ASME_13(2)/ASME_13(1)
CRITERIO_F=ASME_13
CF_CRITERIO_F=Cf_ASME_13
end

if criterio_falla == 2 then
[Div_Va_SE_13,
Desv_Div_Va_SE_13]=div(V_misses_alt_LN13, SE)
Division_VaSE13 = [Div_Va_SE_13, Desv_Div_Va_SE_13]

Division_VmSut13 = [0,0]

Suma13_VmSut_VaSE=Division_VaSE13(1)+Division_VmS
ut13(1)
Desv_Suma13_VmSut_VaSE=sqrt((Division_VaSE13(2)/Divi
sion_VaSE13(1))^2+(0)^2)

GERBER_13=[Suma13_VmSut_VaSE,
Desv_Suma13_VmSut_VaSE]
Cf_GERBER_13=GERBER_13(2)/GERBER_13(1)
CRITERIO_F=GERBER_13
CF_CRITERIO_F=Cf_GERBER_13
end

if criterio_falla == 3 then
[Div_Va_SE_13,
Desv_Div_Va_SE_13]=div(V_misses_alt_LN13, SE)
Division_Va_SE13 = [Div_Va_SE_13, Desv_Div_Va_SE_13]

Division_VmSut13 = [0,0]

Suma13_VmSut_VaSE=Division_Va_SE13(1)+Division_Vm
Sut13(1)
Desv_Suma13_VmSut_VaSE=sqrt((Division_Va_SE13(2)/Di
vision_Va_SE13(1))^2+(0)^2)
GOODMAN_13=[Suma13_VmSut_VaSE,
Desv_Suma13_VmSut_VaSE]
Cf_GOODMAN_13=GOODMAN_13(2)/GOODMAN_13(1)
CRITERIO_F=GOODMAN_13
CF_CRITERIO_F=Cf_GOODMAN_13
end

//CONVERSIÓN DE LA DISTRIBUCIÓN LOG-NORMAL A
LA DISTRIBUCIÓN NORMAL
MEDCnormal_13=log(CRITERIO_F(1))-
((0.5)*(CF_CRITERIO_F)^2)
DESVnormal_13=CF_CRITERIO_F

//EVALUACIÓN DE LA PROBABILIDAD DE FALLA:
[Prob_13,Conf_13]=cdfnor("PQ",0,abs(MEDCnormal_13),DE
SVnormal_13)

```

```
DIAMETRO_EJE13=0
```

```

if Prob_13<=1.5*10^-6 & Prob_13>=0.5*10^-6 then
    DIAMETRO_EJE13=Diametro_eje13
    break;
end
end
disp("El diametro del tramo 13 es")
disp(DIAMETRO_EJE13)

Sut9=0.9*Sut (1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=((sigvma13)-Sut9+(pend*3))/pend //valor de x para el
esfuerzo alterante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida del eje es :")
disp(ViRe)

```

14. 8_VidaRemanente.sci

```

clc
clear all
if sigvma13 > sigvma12 & sigvma13 > sigvma11 & sigvma13
> sigvma10 & sigvma13 > sigvma9 & sigvma13 > sigvma8 &
sigvma13 > sigvma7 & sigvma13 > sigvma6 & sigvma13 >
sigvma5 & sigvma13 > sigvma4 & sigvma13 > sigvma3 &
sigvma13 > sigvma2 & sigvma13 > sigvma1 then
sigvma=sigvma13
end
if sigvma12 > sigvma13 & sigvma12 > sigvma11 & sigvma12
> sigvma10 & sigvma12 > sigvma9 & sigvma12 > sigvma8 &
sigvma12 > sigvma7 & sigvma12 > sigvma6 & sigvma12 >
sigvma5 & sigvma12 > sigvma4 & sigvma12 > sigvma3 &
sigvma12 > sigvma2 & sigvma12 > sigvma1 then
sigvma=sigvma12
end
if sigvma11 > sigvma13 & sigvma11 > sigvma12 & sigvma11
> sigvma10 & sigvma11 > sigvma9 & sigvma11 > sigvma8 &
sigvma11 > sigvma7 & sigvma11 > sigvma6 & sigvma11 >
sigvma5 & sigvma11 > sigvma4 & sigvma11 > sigvma3 &
sigvma11 > sigvma2 & sigvma11 > sigvma1 then
sigvma=sigvma11
end
if sigvma10 > sigvma13 & sigvma10 > sigvma12 & sigvma10
> sigvma11 & sigvma10 > sigvma9 & sigvma10 > sigvma8 &
sigvma10 > sigvma7 & sigvma10 > sigvma6 & sigvma10 >
sigvma5 & sigvma10 > sigvma4 & sigvma10 > sigvma3 &
sigvma10 > sigvma2 & sigvma10 > sigvma1 then
sigvma=sigvma10
end
if sigvma9 > sigvma13 & sigvma9 > sigvma12 & sigvma9 >
sigvma11 & sigvma9 > sigvma10 & sigvma9 > sigvma8 &
sigvma9 > sigvma7 & sigvma9 > sigvma6 & sigvma9 >
sigvma5 & sigvma9 > sigvma4 & sigvma9 > sigvma3 &
sigvma9 > sigvma2 & sigvma9 > sigvma1 then
sigvma=sigvma9
end
if sigvma8 > sigvma13 & sigvma8 > sigvma12 & sigvma8 >
sigvma11 & sigvma8 > sigvma10 & sigvma8 > sigvma9 &
sigvma8 > sigvma7 & sigvma8 > sigvma6 & sigvma8 >

```

```

sigvma5 & sigvma8 > sigvma4 & sigvma8 > sigvma3 &
sigvma8 > sigvma2 & sigvma8 > sigvma1 then
sigvma=sigvma8
end
if sigvma7 > sigvma13 & sigvma7 > sigvma12 & sigvma7 >
sigvma11 & sigvma7 > sigvma10 & sigvma7 > sigvma9 &
sigvma7 > sigvma8 & sigvma7 > sigvma6 & sigvma7 >
sigvma5 & sigvma7 > sigvma4 & sigvma7 > sigvma3 &
sigvma7 > sigvma2 & sigvma7 > sigvma1 then
sigvma=sigvma7
end
if sigvma6 > sigvma13 & sigvma6 > sigvma12 & sigvma6 >
sigvma11 & sigvma6 > sigvma10 & sigvma6 > sigvma9 &
sigvma6 > sigvma8 & sigvma6 > sigvma7 & sigvma6 >
sigvma5 & sigvma6 > sigvma4 & sigvma6 > sigvma3 &
sigvma6 > sigvma2 & sigvma6 > sigvma1 then
sigvma=sigvma6
end
if sigvma5 > sigvma13 & sigvma5 > sigvma12 & sigvma5 >
sigvma11 & sigvma5 > sigvma10 & sigvma5 > sigvma9 &
sigvma5 > sigvma8 & sigvma5 > sigvma7 & sigvma5 >
sigvma6 & sigvma5 > sigvma4 & sigvma5 > sigvma3 &
sigvma5 > sigvma2 & sigvma5 > sigvma1 then
sigvma=sigvma5
end
if sigvma4 > sigvma13 & sigvma4 > sigvma12 & sigvma4 >
sigvma11 & sigvma4 > sigvma10 & sigvma4 > sigvma9 &
sigvma4 > sigvma8 & sigvma4 > sigvma7 & sigvma4 >
sigvma6 & sigvma4 > sigvma5 & sigvma4 > sigvma3 &
sigvma4 > sigvma2 & sigvma4 > sigvma1 then
sigvma=sigvma4
end
if sigvma3 > sigvma13 & sigvma3 > sigvma12 & sigvma3 >
sigvma11 & sigvma3 > sigvma10 & sigvma3 > sigvma9 &
sigvma3 > sigvma8 & sigvma3 > sigvma7 & sigvma3 >
sigvma6 & sigvma3 > sigvma5 & sigvma3 > sigvma4 &
sigvma3 > sigvma2 & sigvma3 > sigvma1 then
sigvma=sigvma3
end
if sigvma2 > sigvma13 & sigvma2 > sigvma12 & sigvma2 >
sigvma11 & sigvma2 > sigvma10 & sigvma2 > sigvma9 &
sigvma2 > sigvma8 & sigvma2 > sigvma7 & sigvma2 >
sigvma6 & sigvma2 > sigvma5 & sigvma2 > sigvma4 &
sigvma2 > sigvma3 & sigvma2 > sigvma1 then
sigvma=sigvma2
end
if sigvma1 > sigvma13 & sigvma1 > sigvma12 & sigvma1 >
sigvma11 & sigvma1 > sigvma10 & sigvma1 > sigvma9 &
sigvma1 > sigvma8 & sigvma1 > sigvma7 & sigvma1 >
sigvma6 & sigvma1 > sigvma5 & sigvma1 > sigvma4 &
sigvma1 > sigvma3 & sigvma1 > sigvma2 then
sigvma=sigvma1
end
SUT=0.9*Sut (1)
m1=(SUT-SE(1))
m2=(3-6)
pend=m1/m2 // calculo de la pendiente
x=((sigvma)-SUT+(pend*3))/pend //valor de x para el esfuerzo
alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)

```

14_CincoEngranes.sci

```

clc
clear all

//Entrada de datos de las sesiones, muescas, ángulos, ubicación
y fuerzas:
exec('14.1_DistanciaSesionesMuecasYUbicacionEngrane.sci',
-1)

//Entrada del ajuste por interferencia para el primer engrane:
exec('11.2_OpcionesInterferenciaUno.sci',-1)

if interferencia == 1 then
exec('10.3_InterferenciaUno.sci',-1) //Para la opción 1
end

if interferencia == 2 then
exec('10.4_InterferenciaDos.sci',-1) // Para la opción 2
end

if interferencia == 3 then
exec('10.5_InterferenciaTres.sci',-1) //Para la opción 3
end

Interferen_maza_eje_min=Interf_maza_eje_min
Interferen_maza_eje_max=Interf_maza_eje_max

//Entrada del ajuste por interferencia para el segundo
engrane:
exec('11.3_OpcionesInterferenciaDos.sci',-1)

if interferencia2 == 1 then
exec('11.4_InterferenciaUno.sci',-1) //Para la opción 1
end

if interferencia2 == 2 then
exec('11.5_InterferenciaDos.sci',-1) // Para la opción 2
end

if interferencia2 == 3 then
exec('11.6_InterferenciaTres.sci',-1) //Para la opción 3
end

Interferen_maza_eje_min2=Interf_maza_eje_min2
Interferen_maza_eje_max2=Interf_maza_eje_max2

//Entrada del ajuste por interferencia para el tercer engrane:
exec('12.2_OpcionesInterferenciaTres.sci',-1)

if interferencia3 == 1 then
exec('12.3_InterferenciaUno.sci',-1) //Para la opción 1
end

if interferencia3 == 2 then
exec('12.4_InterferenciaDos.sci',-1) // Para la opción 2
end

if interferencia3 == 3 then
exec('12.5_InterferenciaTres.sci',-1) //Para la opción 3
end

Interferen_maza_eje_min3=Interf_maza_eje_min3
Interferen_maza_eje_max3=Interf_maza_eje_max3

//Entrada del ajuste por interferencia para el cuarto engrane:
exec('13.2_OpcionesInterferenciaCuatro.sci',-1)

if interferencia4 == 1 then

```

```
exec('13.3_InterferenciaUno.sci',-1) //Para la opción 1
end
```

```
if interferencia4 == 2 then
exec('13.4_InterferenciaDos.sci',-1) //Para la opción 2
end
```

```
if interferencia4 == 3 then
exec('13.5_InterferenciaTres.sci',-1) //Para la opción 3
end
```

```
Interferen_maza_eje_min4=Interf_maza_eje_min4
Interferen_maza_eje_max4=Interf_maza_eje_max4
```

```
//Entrada del ajuste por interferencia para el quinto engrane:
exec('14.2_OpcionesInterferenciaCinco.sci',-1)
```

```
if interferencia5 == 1 then
exec('14.3_InterferenciaUno.sci',-1) //Para la opción 1
end
```

```
if interferencia5 == 2 then
exec('14.4_InterferenciaDos.sci',-1) //Para la opción 2
end
```

```
if interferencia5 == 3 then
exec('14.5_InterferenciaTres.sci',-1) //Para la opción 3
end
```

```
Interferen_maza_eje_min5=Interf_maza_eje_min5
Interferen_maza_eje_max5=Interf_maza_eje_max5
```

```
//Cálculo de diagramas de momentos con sus diagramas:
exec('14.6_TorquesMomentosYGraficas.sci',-1)
//Multiplicación de los factores de Marin Ka por Kd:
exec('10.8_MultiplicacionKaKd.sci',-1)
//Iteraciones de los diámetros:
exec('14.7_IteracionesDeLosDiametros.sci',-1)
```

16_VidaRemanente_UnEngrane.sci

```
clc
clear all

//CALCULO DE VIDA REMANENTE
Sut9=0.9*Sut (1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=((sigvma5)-Sut9+(pend*3))/pend //valor de x para el
esfuerzo alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)
```

17_VidaRemanente_DosEngrane.sci

```
clc
clear all
//CALCULO DE VIDA REMANENTE
if sigvma3 > sigvma5 & sigvma3 > sigvma7 then
sigvma3=sigvma
end
```

```
if sigvma5 > sigvma3 & sigvma5 > sigvma7 then
sigvma5=sigvma
end
if sigvma7 > sigvma5 & sigvma7 > sigvma3 & sigvma7 then
sigvma7=sigvma
end
```

```
Sut9=0.9*Sut (1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=((sigvma)-sut9+(pend*3))/pend //valor de x para el esfuerzo
alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)
```

18_VidaRemanente_TresEngranes.sci

```
clc
clear all
//CALCULO DE VIDA REMANENTE
if sigvma3 > sigvma5 & sigvma3 > sigvma7 & sigvma3 >
sigvma9 then
sigvma3=sigvma
end
if sigvma5 > sigvma3 & sigvma5 > sigvma7 & sigvma5 >
sigvma9 then
sigvma5=sigvma
end
if sigvma7 > sigvma5 & sigvma7 > sigvma3 & sigvma7 >
sigvma9 then
sigvma7=sigvma
end
if sigvma9 > sigvma5 & sigvma9 > sigvma7 & sigvma9 >
sigvma3 then
sigvma9=sigvma
end
```

```
Sut9=0.9*Sut (1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=((sigvma)-sut9+(pend*3))/pend //valor de x para el esfuerzo
alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)
```

19_VidaRemanente_CuatroEngranes.sci

```
clc
clear all
//CALCULO VIDA REMANENTE
if sigvma3 > sigvma5 & sigvma3 > sigvma7 & sigvma3 >
sigvma9 & sigvma9 > sigvma11 then
sigvma3=sigvma
end
if sigvma5 > sigvma3 & sigvma5 > sigvma7 & sigvma5 >
sigvma9 & sigvma9 > sigvma11 then
sigvma5=sigvma
```

```

end
if sigvma7 > sigvma5 & sigvma7 > sigvma3 & sigvma7 >
sigvma9 & sigvma9 > sigvma11 then
sigvma7=sigvma
end
if sigvma9 > sigvma5 & sigvma9 > sigvma7 & sigvma9 >
sigvma3 & sigvma9 > sigvma11 then
sigvma9=sigvma
end
if sigvma11 > sigvma5 & sigvma11 > sigvma7 & sigvma11 >
sigvma9 & sigvma9 > sigvma11 then
sigvma11=sigvma
end

```

```

Sut9=0.9*Sut (1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=((sigvma)-sut9+(pend*3))/pend //valor de x para el
esfuerzo alternante
CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)

```

20_VidaRemanente_CincoEngranes.sci

```

clc
clear all
//CALCULO VIDA REMANENTE
if sigvma3 > sigvma5 & sigvma3 > sigvma7 & sigvma3 >
sigvma9 & sigvma9 > sigvma11 then
sigvma3=sigvma
end
if sigvma5 > sigvma3 & sigvma5 > sigvma7 & sigvma5 >
sigvma9 & sigvma9 > sigvma11 then
sigvma5=sigvma
end
if sigvma7 > sigvma5 & sigvma7 > sigvma3 & sigvma7 >
sigvma9 & sigvma9 > sigvma11 then
sigvma7=sigvma
end
if sigvma9 > sigvma5 & sigvma9 > sigvma7 & sigvma9 >
sigvma3 & sigvma9 > sigvma11 then
sigvma9=sigvma
end
if sigvma11 > sigvma5 & sigvma11 > sigvma7 & sigvma11 >
sigvma9 & sigvma9 > sigvma11 then
sigvma11=sigvma
end
if sigvma11 > sigvma5 & sigvma11 > sigvma7 & sigvma11 >
sigvma9 & sigvma9 > sigvma11 then
sigvma11=sigvma
end
if sigvma13 > sigvma5 & sigvma13 > sigvma7 & sigvma13 >
sigvma9 & sigvma9 > sigvma13 then
sigvma13=sigvma
end
end

```

```

Sut9=0.9*Sut (1)
m1=(Sut9-SE(1))
m22=(3-6)
pend=m1/m22 // calculo de la pendiente
x=((sigvma)-sut9+(pend*3))/pend //valor de x para el
esfuerzo alternante

```

```

CiTo=10^x //REVISAR VARIABLE "x".
ViRe=CiTo
disp("la vida remanente del eje es :")
disp(ViRe)

```

Tesis-Prototipo.sce

```

clc
clear

```

```

K=2490.3/1000 //constante del resorte en N/mm
sut=786.11// la resistencia ultima del material acero AISI 1137
en Megapascuales
desv=58.351 // desviacion del esfuerzo ultimo
Cf_sut=desv/sut // coeficiente de desviacion del sut

ri=0// radio interno en milimetros
re=2.1 // radio externo en milimetros
D=re*2 // diametro del radio externo en milimetro
// calculo de inercia
I=(%pi*(((re)^4)-((ri)^4))/4)//inercia en mm4
A=(%pi*(((re)^2)-((ri)^2)))/area en mm2
J=I*2// momento polar de inercia en mm4

```

```

//calculo de esfuerzo torsion
Pot=500 // Watts
f=11000

```

```

//rpm
W=f*2*%pi/60 // velocidad angular rad/s
T=(Pot/W)*1000 // en Nmm
tao=(T*re)/J // en megapascuales

```

```

//Calculos de potencia absorbida por la polea

```

```

Rpol=70 //radio de la polea en mm
F1= T/Rpol // fuerza absorbida por el resorte
Est=F1/K//estiramiento del resorte

```

```

//calculo de esfuerzo flector

```

```

Mflex=110*F1 // momento flector maximo sobre el eje en N
sigflex=(Mflex*re)/I // esuferzo flector en Megapascuales

```

```

//CONCENTRADORES DE ESFUERZO QUE DEPENDEN
DEL DIÁMETRO

```

```

//constantes para hallar la concentración de esfuerzos de
flexión (kf)
r1=(1) // radio de la muesca en mm
h1=(2) // dimetro interno
b1=(2*h1/D)

```

```

c11=0.965+(1.926*sqrt(h1/r1))
c21=-2.773-(4.414*sqrt(h1/r1))-(0.017*h1/r1)
c31=4.785+(4.681*sqrt(h1/r1))+(0.096*h1/r1)
c41=-1.995-(2.241*sqrt(h1/r1))-(0.074*h1/r1)
Kf=c11+(c21*b1)+(c31*(b1^2))+(c41*(b1^3)) // es requerido
para calcular el factor de concentracion para flexion

```

```

//concentrador de esfuerzo para torsión

```

```

C11=1.089+(0.924*sqrt(h1/r1))+(0.018*h1/r1)
C21=-1.504-(2.141*sqrt(h1/r1))-(0.047*h1/r1)
C31=2.486+(2.289*sqrt(h1/r1))+(0.091*h1/r1)
C41=-1.056-(1.104*sqrt(h1/r1))-(0.059*h1/r1)

```

$K_s = C_{11} + (C_{21} * b_1) + (C_{31} * (b_1^2)) + (C_{41} * (b_1^3))$ // es requerido para calcular el factor de concentracion para flexion

```
//FACTORES DE MARÍN PAGINAS 317-320
//FACTOR DE CARGA DE FLEXIÓN
Kcf=[sut^0,0] // Kc en estocastico
mediaKcf=Kcf(1) // valor medio de
CfKcf=0 // coeficiente de desviacion
//FACTOR DE CARGA DE TORSIÓN
Kct=[0.258*sut^-0.125,0.125]
mediaKct=Kct(1)
CfKcs=Kct(2)
//FACTOR DE CARGA DE CARGA AXIAL
Kca=[1.43*sut^-0.0778,0.125] // de torsion que reduce
mayormente la resistencia a la fatiga
mediaKca=Kca(1)
CfKca=Kca(2)
//FACTOR DE TAMAÑO PARA CARGA AXIAL,TORSION Y
FLEXION
// el de flexion es el mismo que de torsion
Kba=1
Kbf=0.6
//TENSORES, antes de ingresar los esfuerzos en el tensor, se
operan los Kf, Ka y Kba
//los Kf se multiplican y los otros se dividen
```

$Am_2 = (\tau_{ao}) / (K_{ct}(1) * K_{bf})$ // esfuerzo correspondiente por la torsion
 $Aa_1 = (\sigma_{flex}) / (K_{cf}(1) * K_{bf})$ // esfuerzo correspondiente por momento

```
Am=[ 0 0 -Am2 ;0 0 0;-Am2 0 0] //tensor de esfuerzo
medio//
Aa=[(-Aa1) 0 0;0 0 0;0 0 0] //tensor de esfuerzo alternante,
suma del esfuerzo flector con el longitudinal
auto=spec(Am) //funcion de autovalores//
[rr,dd]=spec(Am)
sigm1=dd(1,1)
sigm2=dd(2,2)
sigm3=dd(3,3) //nombramos los autovalores //
```

```
auto=spec(Aa) //funcion de autovalores//
[rr,dd]=spec(Aa)
siga1=dd(1,1)
siga2=dd(2,2)
siga3=dd(3,3) //nombramos los autovalores //
```

$\sigma_{vmm} = (((\sigma_1 - \sigma_2)^2) + ((\sigma_2 - \sigma_3)^2) + ((\sigma_3 - \sigma_1)^2)) / 2)^{0.5}$ //calculo esfuerzo Von Mises medio
 $\sigma_{vma} = (((\sigma_1 - \sigma_2)^2) + ((\sigma_2 - \sigma_3)^2) + ((\sigma_3 - \sigma_1)^2)) / 2)^{0.5}$ //calculo esfuerzo Von Mises alternante

```
//MULTIPLICACIÓN DE LOS FACTORES DE MARÍN
//OPERACIONES DADAS POR LA TABLA 20-6
//FACTOR DE SUPERFICIE
Ka=[271*sut^-0.995,0.145]
CfKa=Ka(2)
//FACTOR DE TEMPERATURA
Kd=[0.672,0.11]
Cfkd=(Kd(2))
//multiplicacion de ka y kd
mediaKaKd=Ka(1)*Kd(1)//Valor medio de la multiplicación
de los factores de marín ka y kd
```

$desv_KaKd = mediaKaKd * \sqrt{CfKa^2 + Cfkd^2 + (CfKa^2) * Cfkd^2}$ // Desviación estándar de la multiplicación de los factores de marín ka y kd
 $C_{KaKd} = (desv_KaKd / mediaKaKd)$
//SE PRIMA
 $Se_prima = [1230, 0.138 * 0.506 * sut]$
//[0.506*sut,0.138*0.506*sut]
 $CfSe_prima = Se_prima(2) / Se_prima(1)$

```
//Multiplicación de Ka, Kd, Se prima
SE_MEDIO=mediaKaKd*Se_prima(1)
SE_DESVIA=SE_MEDIO*sqrt(CfSe_prima^2+CfKaKd^2+(CfSe_prima^2)*(CfKaKd^2))
Cf_SE=(SE_DESVIA/SE_MEDIO)
```

```
//ASME-ELIPTICA
//VM/SUT
Zmedio=sigvmm/sut
Zdesvia=Zmedio*sqrt(((0.2^2)+(Cf_sut)^2)*(1+Cf_sut^2))
CZ=Zdesvia/Zmedio
//VM2/SUT^2
Z2medio=Zmedio^2*(1+CZ^2)
Z2desvia=2*Zmedio^2*CZ*(1+0.25*CZ^2)
```

```
//VA/SE
Zamedio=sigvma/SE_MEDIO
ZAdesvia=Zamedio*sqrt(((0.2^2)+(Cf_SE)^2)*(1+Cf_SE^2))
CZA=ZAdesvia/Zamedio
```

```
//VA^2/SE^2
Z2amedio=Zamedio^2*(1+CZA^2)
Z2Adesvia=2*Zamedio^2*CZA*(1+0.25*CZA^2)
```

```
//SUMA DE VA2/SE2+VM2/SUT2
ASMEM=Z2amedio+Z2medio
ASMED=sqrt(Z2desvia^2 + Z2Adesvia^2)
Cf_ASME=ASMED/ASMEM
```

```
//PASAMOS DE TERMOINOS LOG NORMAL A NORMAL,
FORMULAS DE LA PAGINA 240
MEDCnormal=log(ASMEM)-((0.5)*(Cf_ASME)^2)
DESVnormal=Cf_ASME
[Prob,Qconf]=cdfnor("PQ",0,abs
(MEDCnormal),DESVnormal)
```

$m_1 = (sut(1) - Se_prima(1))$
 $m_2 = (3 - 6)$
 $pend = m_1 / m_2$ // calculo de la pendiente
 $x = ((\sigma_{vmm}) - sut + (pend * 3)) / pend$ //valor de x para el esfuerzo alternante
 $CiTo = 10^x$ //REVISAR VARIABLE "x".
 $ViRe = CiTo$

```
disp ("Probabilidad de falla")
disp(Prob)
disp ("el diametro del eje en pulgadas es ")
disp(re*2/25.4)
disp ("el valor de Se es ")
disp(Se_prima)
disp ("el esfuerzo von missses alternante es ")
disp(sigvma)
disp ("el esfuerzo von missses medio es ")
disp(sigvmm)
disp ("el numero de ciclos es ")
disp(ViRe)
```

Anexo B. Manual para el usuario

El programa se ejecuta desde el archivo ScilabPrincipal.sce, pero antes es necesario colocar la ubicación de la carpeta en la que se encuentra los archivos. Esta se coloca en la línea indicada a continuación:

```
ScilabPrincipal.sce [X]
1  //-----Borrar variables y funciones previamente definidas
2  clear
3  clear all
4  //-----No mensajes de alerta por redefinir funciones
5  funcprot(0);
6  //-----Memoria para variables y funciones
7  //... (medido en unidades de numeros de doble precision)
8  stacksize(25000000);
9  cd "C:\Users\Usuario\Desktop\codigo";
10
11
12 exec('00_FuncionesEstocasticas.sci',-1); //Funciones para transformaciones estocasticas.
13 exec('01_TablasPresionInterferencia.sci',-1); //tablas de presion de interferencia
14 exec('02_TablaMateriales.sci',-1); //Menu y tabla de los materiales a escoger
15
```

En este manual del usuario se explica, cada una de las pantallas sobre las cuales el usuario ingresara los datos requeridos para el diseño.

1. Opciones para escoger el material

En esta sección, el usuario debe escribir el número, al cual corresponda el material usado en el eje.

2. Ingrese el porcentaje del coeficiente variación de las cargas

Como se realizan cálculos estocásticos, hay la posibilidad de colocar porcentaje del coeficiente de variación de la carga que se somete el eje.

3. Opciones para escoger el acabado superficial

En esta sección, el usuario debe escribir el número, al cual corresponda el acabado superficial usado en el eje.

4. Opciones para escoger la temperatura de trabajo

En esta sección, el usuario debe escribir el número, al cual corresponda temperatura (C°) a la cual trabajará el eje.

5. Opciones para el tipo de muesca deseado

En esta sección, el usuario debe escribir el número, al cual corresponda el tipo de muesca que tendrá el eje.

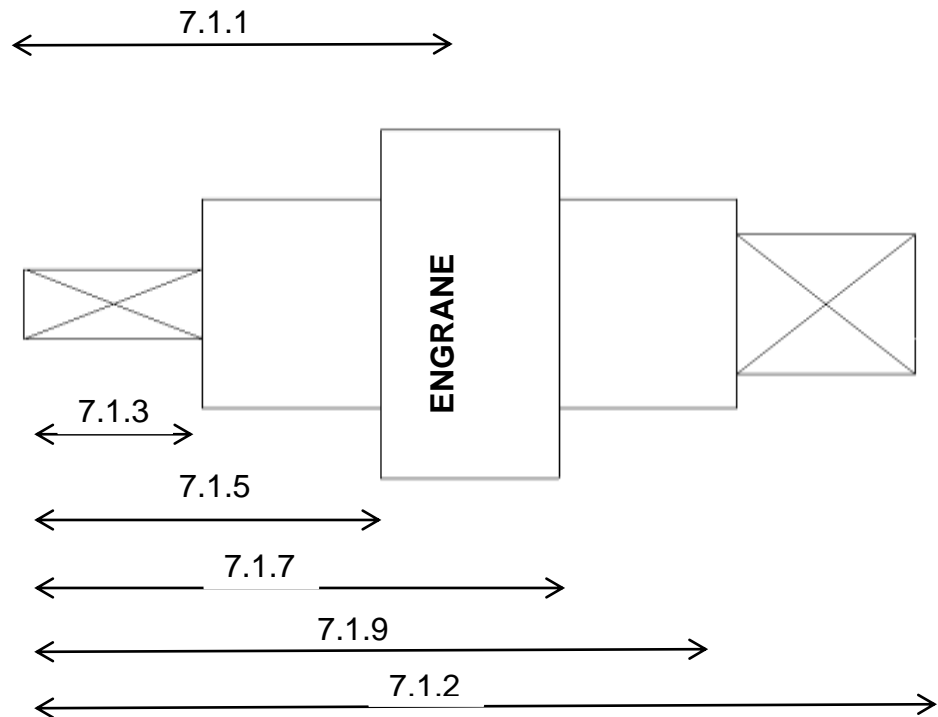
6. Opciones para criterio de falla

En esta sección, el usuario debe escribir el número, al cual corresponda la teoría deseada para realizar el diseño del eje.

7. Ingrese el número de engranes

Ingrese el número de engranes que tendrá el eje. Todas las distancias se miden desde un mismo extremo del eje, hasta el otro. De izquierda a derecha, como se muestran en los gráficos más adelante.

7.1. Un engrane



7.1.1. Ingrese la velocidad angular del eje

Ingrese la velocidad angular que tendrá el eje en rpm.

7.1.2. Ingrese la longitud del eje

Ingrese la longitud total del eje en pulgadas.

7.1.3. Ingrese la distancia de la sección 1

Ingrese la distancia de la sección en pulgadas, en la cual está el cojinete (sección 1). Como se muestra en el gráfico.

7.1.4. Ingrese el radio de la muesca de la sección 1

Ingrese el radio de la muesca del primer cambio diámetro en pulgadas.

7.1.5. Ingrese la distancia de la sección 2

Ingrese la distancia de la sección 2 en pulgadas. Como se muestra en el gráfico.

7.1.6. Ingrese el radio de la muesca de la sección 2

Ingrese el radio de la muesca del segundo cambio diámetro en pulgadas.

7.1.7. Ingrese la distancia de la sección 3

Ingrese la distancia de la sección 3 en pulgadas. Como se muestra en el gráfico.

7.1.8. Ingrese el radio de la muesca de la sección 3

Ingrese el radio de la muesca del tercer diámetro en pulgadas.

7.1.9. Ingrese la distancia de la sección 4

Ingrese la distancia de la sección 4 en pulgadas. Como se muestra en el gráfico.

7.1.10. Ingrese el radio de la muesca de la sección 4

Ingrese el radio de la muesca del cuarto cambio diámetro en pulgadas.

7.1.11. Ingrese la distancia a la que se encuentra el engrane

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 2 y 3 en pulgadas, como se muestra en el gráfico.

7.1.12. Ingrese la potencia transmitida en el engrane en hp.

7.1.13. Ingrese el radio del engrane en pulgadas.

7.1.14. Opción de los planos

Seleccione el plano sobre el cual ejerce el engrane.

7.1.15. Ingrese el ángulo de presión en grados del engrane.

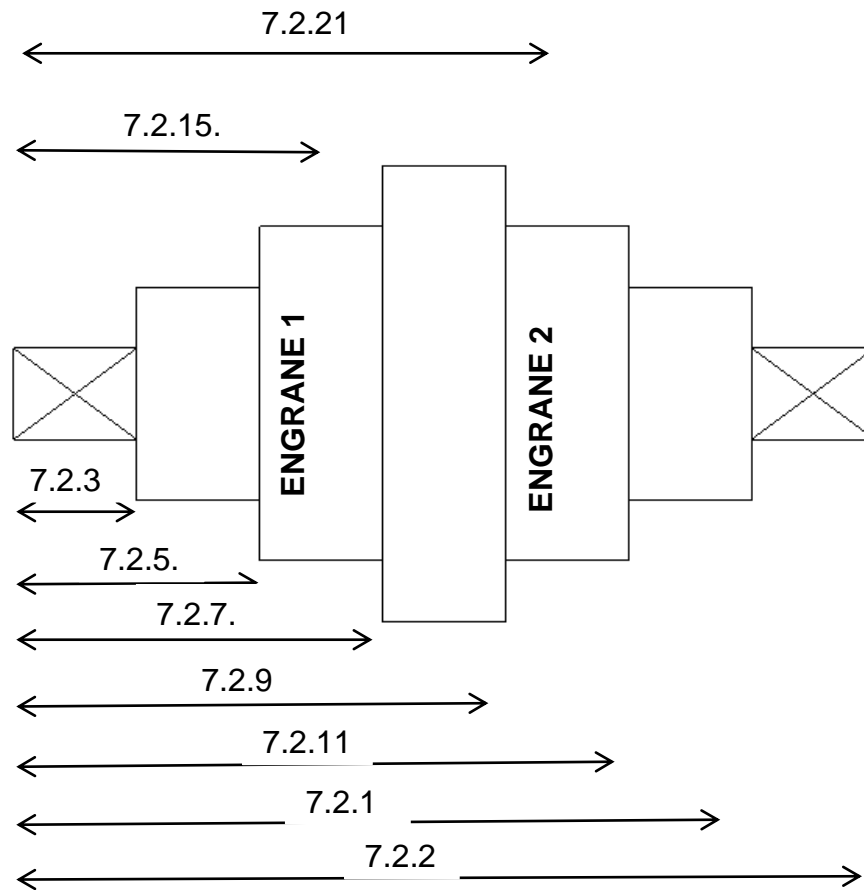
7.1.16. Ingrese el ángulo en grados de hélice del engrane

7.1.17. Opción de ajuste de engrane

7.1.18. Ingrese el tamaño básico

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.2. Dos engranes



7.2.1. Ingrese la velocidad angular del eje

Ingrese la velocidad angular que tendrá el eje en rpm.

7.2.2. Ingrese la longitud del eje

Ingrese la longitud total del eje en pulgadas.

7.2.3. Ingrese la distancia de la sección 1

Ingrese la distancia de la sección en pulgadas, en la cual está el cojinete (sección 1). Como se muestra en el gráfico.

7.2.4. Ingrese el radio de la muesca de la sección 1

Ingrese el radio de la muesca del primer cambio diámetro en pulgadas.

7.2.5. Ingrese la distancia de la sección 2

Ingrese la distancia de la sección 2 en pulgadas. Como se muestra en el gráfico.

7.2.6. Ingrese el radio de la muesca de la sección 2

Ingrese el radio de la muesca del segundo cambio diámetro en pulgadas.

7.2.7. Ingrese la distancia de la sección 3

Ingrese la distancia de la sección 3 en pulgadas. Como se muestra en el gráfico.

7.2.8. Ingrese el radio de la muesca de la sección 3

Ingrese el radio de la muesca del tercer cambio diámetro en pulgadas.

7.2.9. Ingrese la distancia de la sección 4

Ingrese la distancia de la sección 4 en pulgadas. Como se muestra en el gráfico.

7.2.10. Ingrese el radio de la muesca de la sección 4

Ingrese el radio de la muesca del cuarto cambio diámetro en pulgadas.

7.2.11. Ingrese la distancia de la sección 5

Ingrese la distancia de la sección 5 en pulgadas. Como se muestra en el gráfico.

7.2.12. Ingrese el radio de la muesca de la sección 5

Ingrese el radio de la muesca del quinto cambio diámetro en pulgadas.

7.2.13. Ingrese la distancia de la sección 6

Ingrese la distancia de la sección 6 en pulgadas. Como se muestra en el gráfico.

7.2.14. Ingrese el radio de la muesca de la sección 6

Ingrese el radio de la muesca del sexto cambio diámetro en pulgadas.

7.2.15. Ingrese la distancia a la que se encuentra el engrane 1

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 2 y 3 en pulgadas, como se muestra en el gráfico.

7.2.16. Ingrese la potencia transmitida en el engrane 1 en hp.

7.2.17. Ingrese el radio del engrane 1.

7.2.18. Opción de los planos 1

Seleccione el plano sobre el cual ejerce el engrane 1

7.2.19. Ingrese el ángulo de presión del engrane 1 en grados.

7.2.20. Ingrese el ángulo de hélice del engrane 1 en grados.

7.2.21. Ingrese la distancia a la que se encuentra el engrane 2

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 4 y 5 en pulgadas, como se muestra en el gráfico.

7.2.22. Ingrese la potencia transmitida en el engrane 2 en hp.

7.2.23. Ingrese el radio del engrane 2 en pulgadas

7.2.24. Opción de los planos 2

Seleccione el plano sobre el cual ejerce el engrane 2

7.2.25. Ingrese el ángulo de presión del engrane 2 en grados.

7.2.26. Ingrese el ángulo de hélice del engrane 2 en grados.

7.2.27. Opción de ajuste de engrane 1

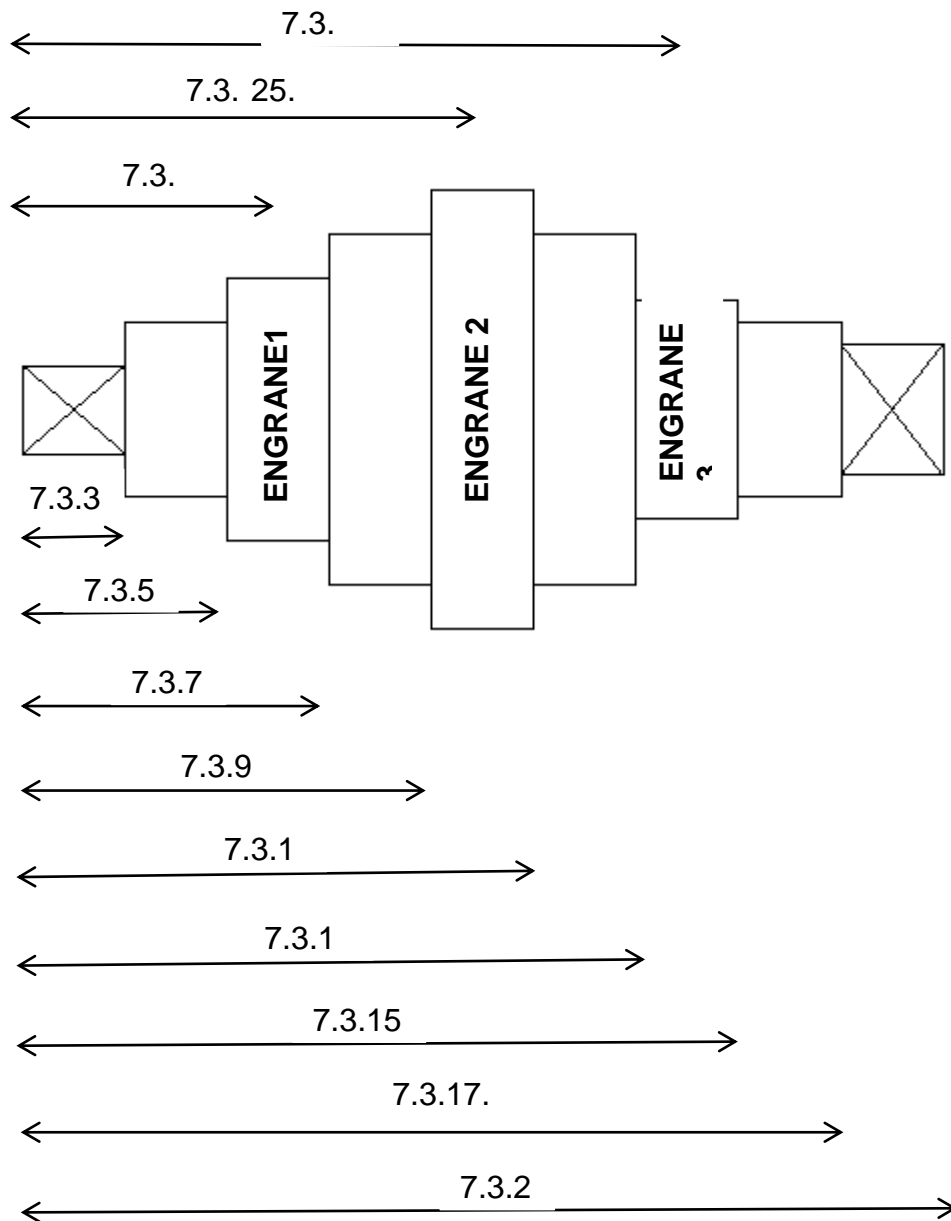
7.2.28. Ingrese el tamaño básico 1 en pulgadas.

7.2.29. Opción de ajuste de engrane 2

7.2.30. Ingrese el tamaño básico 2

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.3. Tres engranes



- 7.3.1. Ingrese la velocidad angular del eje**
Ingrese la velocidad angular que tendrá el eje en rpm.
- 7.3.2. Ingrese la longitud del eje.**
Ingrese la longitud total del eje en pulgadas.
- 7.3.3. Ingrese la distancia de la sección 1**
Ingrese la distancia de la sección, en la cual está el cojinete (sección 1) en pulgadas. Como se muestra en el gráfico.
- 7.3.4. Ingrese el radio de la muesca de la sección 1**
Ingrese el radio de la muesca del primer cambio diámetro en pulgadas.
- 7.3.5. Ingrese la distancia de la sección 2**
Ingrese la distancia de la sección 2 en pulgadas. Como se muestra en el gráfico.
- 7.3.6. Ingrese el radio de la muesca de la sección 2**
Ingrese el radio de la muesca del segundo cambio diámetro en pulgadas.
- 7.3.7. Ingrese la distancia de la sección 3**
Ingrese la distancia de la sección 3 en pulgadas. Como se muestra en el gráfico.
- 7.3.8. Ingrese el radio de la muesca de la sección 3**
Ingrese el radio de la muesca del tercer cambio diámetro en pulgadas.
- 7.3.9. Ingrese la distancia de la sección 4**
Ingrese la distancia de la sección 4 en pulgadas. Como se muestra en el gráfico.
- 7.3.10. Ingrese el radio de la muesca de la sección 4**
Ingrese el radio de la muesca del cuarto cambio diámetro en pulgadas.
- 7.3.11. Ingrese la distancia de la sección 5**
Ingrese la distancia de la sección 5 en pulgadas. Como se muestra en el gráfico.
- 7.3.12. Ingrese el radio de la muesca de la sección 5**
Ingrese el radio de la muesca del quinto cambio diámetro en pulgadas.
- 7.3.13. Ingrese la distancia de la sección 6**
Ingrese la distancia de la sección 6 en pulgadas. Como se muestra en el gráfico.
- 7.3.14. Ingrese el radio de la muesca de la sección 6**
Ingrese el radio de la muesca del sexto cambio diámetro en pulgadas.
- 7.3.15. Ingrese la distancia de la sección 7**
Ingrese la distancia de la sección 7 en pulgadas. Como se muestra en el gráfico.
- 7.3.16. Ingrese el radio de la muesca de la sección 7**

Ingrese el radio de la muesca del séptimo cambio diámetro en pulgadas.

7.3.17. Ingrese la distancia de la sección 8

Ingrese la distancia de la sección 8 en pulgadas. Como se muestra en el gráfico.

7.3.18. Ingrese el radio de la muesca de la sección 8

Ingrese el radio de la muesca del octavo cambio diámetro en pulgadas.

7.3.19. Ingrese la distancia a la que se encuentra el engrane 1

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 2 y 3 en pulgadas, como se muestra en el gráfico.

7.3.20. Ingrese la potencia transmitida en el engrane 1 en hp.

7.3.21. Ingrese el radio del engrane 1 en pulgadas.

7.3.22. Opción de los planos 1

Seleccione el plano sobre el cual ejerce el engrane 1

7.3.23. Ingrese el ángulo de presión del engrane 1 en grados.

7.3.24. Ingrese el ángulo de hélice del engrane 1 en grados.

7.3.25. Ingrese la distancia a la que se encuentra el engrane 2

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 4 y 5 en pulgadas, como se muestra en el gráfico.

7.3.26. Ingrese la potencia transmitida en el engrane 2 en hp.

7.3.27. Ingrese el radio del engrane 2 en pulgadas.

7.3.28. Opción de los planos 2

Seleccione el plano sobre el cual ejerce el engrane 3

7.3.29. Ingrese el ángulo de presión del engrane 2 en grados.

7.3.30. Ingrese el ángulo de hélice del engrane 2 en grados.

7.3.31. Ingrese la distancia a la que se encuentra el engrane 3 en pulgadas.

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 6 y 7 en pulgadas, como se muestra en el gráfico.

7.3.32. Ingrese la potencia transmitida en el engrane 3 en hp.

7.3.33. Ingrese el radio del engrane 3 en pulgadas.

7.3.34. Opción de los planos 3

Seleccione el plano sobre el cual ejerce el engrane 3

7.3.35. Ingrese el ángulo de presión del engrane 3 en grados.

7.3.36. Ingrese el ángulo de hélice del engrane 3 en grados.

7.3.37. Opción de ajuste de engrane 1

7.3.38. Ingrese el tamaño básico 1 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.3.39. Opción de ajuste de engrane 2

7.3.40. Ingrese el tamaño básico 2 en pulgadas

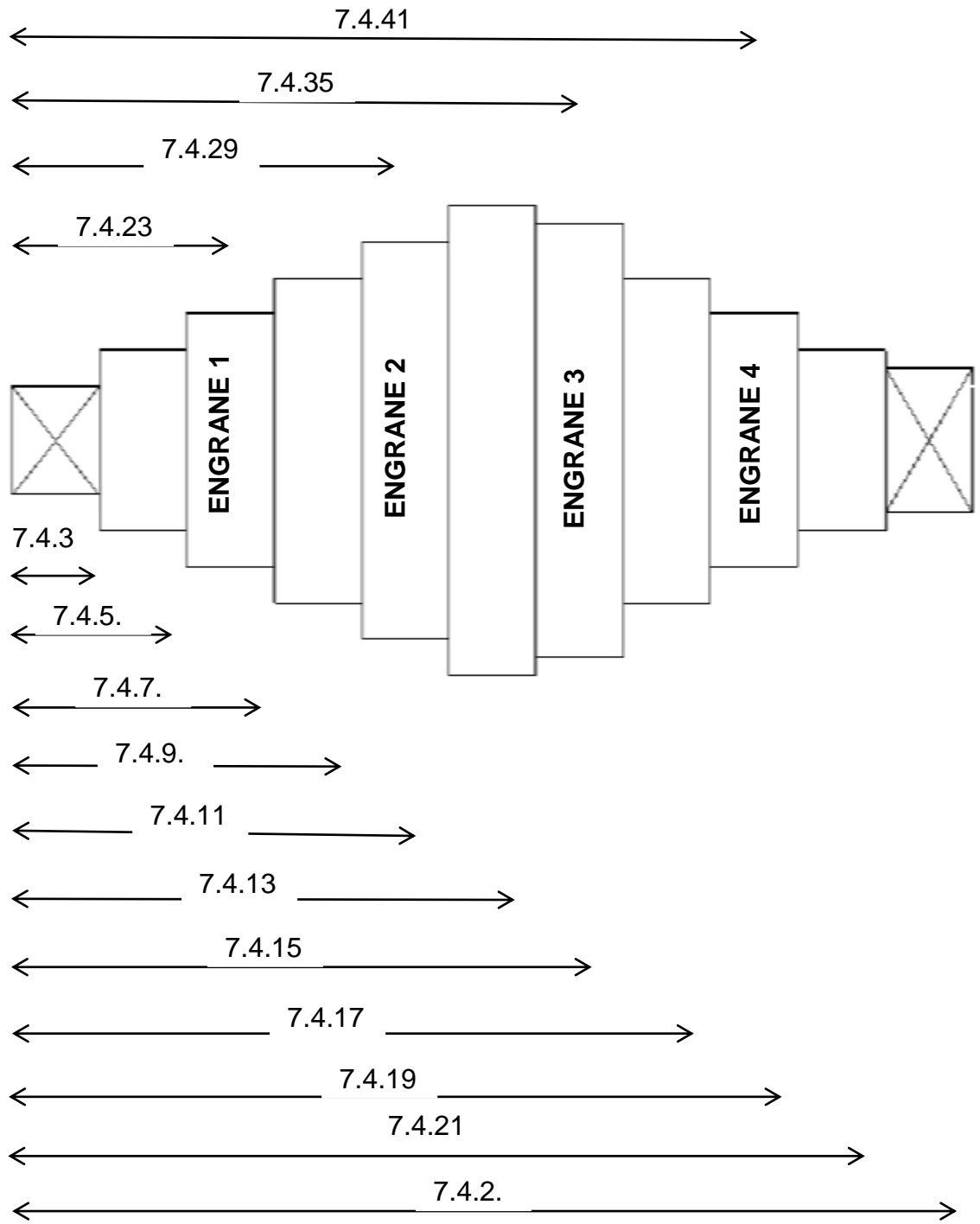
Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.3.41. Opción de ajuste de engrane 3

7.3.42. Ingrese el tamaño básico 3 en pulgadas

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.4. Cuatro engranes



7.4.1. Ingrese la velocidad angular del eje

Ingrese la velocidad angular que tendrá el eje en rpm.

7.4.2. Ingrese la longitud del eje

Ingrese la longitud total del eje en pulgadas.

7.4.3. Ingrese la distancia de la sección 1

Ingrese la distancia de la sección, en la cual está el cojinete (sección 1) en pulgadas. Como se muestra en el gráfico.

7.4.4. Ingrese el radio de la muesca de la sección 1

Ingrese el radio de la muesca del primer cambio diámetro en pulgadas.

7.4.5. Ingrese la distancia de la sección 2

Ingrese la distancia de la sección 2 en pulgadas. Como se muestra en el gráfico.

7.4.6. Ingrese el radio de la muesca de la sección 2

Ingrese el radio de la muesca del segundo cambio diámetro en pulgadas.

7.4.7. Ingrese la distancia de la sección 3

Ingrese la distancia de la sección 3 en pulgadas. Como se muestra en el gráfico.

7.4.8. Ingrese el radio de la muesca de la sección 3

Ingrese el radio de la muesca del tercer cambio diámetro en pulgadas.

7.4.9. Ingrese la distancia de la sección 4

Ingrese la distancia de la sección 4 en pulgadas. Como se muestra en el gráfico.

7.4.10. Ingrese el radio de la muesca de la sección 4

Ingrese el radio de la muesca del cuarto cambio diámetro pulgadas.

7.4.11. Ingrese la distancia de la sección 5

Ingrese la distancia de la sección 5 en pulgadas. Como se muestra en el gráfico.

7.4.12. Ingrese el radio de la muesca de la sección 5

Ingrese el radio de la muesca del quinto cambio diámetro en pulgadas.

7.4.13. Ingrese la distancia de la sección 6

Ingrese la distancia de la sección 6 en pulgadas. Como se muestra en el gráfico.

7.4.14. Ingrese el radio de la muesca de la sección 6

Ingrese el radio de la muesca del sexto cambio diámetro en pulgadas.

7.4.15. Ingrese la distancia de la sección 7

Ingrese la distancia de la sección 7 en pulgadas. Como se muestra en el gráfico.

7.4.16. Ingrese el radio de la muesca de la sección 7

Ingrese el radio de la muesca del séptimo cambio diámetro en pulgadas.

7.4.17. Ingrese la distancia de la sección 8

Ingrese la distancia de la sección 8 en pulgadas. Como se muestra en el gráfico.

7.4.18. Ingrese el radio de la muesca de la sección 8

Ingrese el radio de la muesca del octavo cambio diámetro en pulgadas.

7.4.19. Ingrese la distancia de la sección 9

Ingrese la distancia de la sección 9 en pulgadas. Como se muestra en el gráfico.

7.4.20. Ingrese el radio de la muesca de la sección 9

Ingrese el radio de la muesca del noveno cambio diámetro en pulgadas.

7.4.21. Ingrese la distancia de la sección 10

Ingrese la distancia de la sección 10 en pulgadas. Como se muestra en el gráfico.

7.4.22. Ingrese el radio de la muesca de la sección 10

Ingrese el radio de la muesca del décimo cambio diámetro en pulgadas.

7.4.23. Ingrese la distancia a la que se encuentra el engrane 1

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 2 y 3 en pulgadas, como se muestra en el gráfico.

7.4.24. Ingrese la potencia transmitida en el engrane 1 en hp.

7.4.25. Ingrese el radio del engrane 1 en pulgadas.

7.4.26. Opción de los planos 1

Seleccione el plano sobre el cual ejerce el engrane 1

7.4.27. Ingrese el ángulo de presión del engrane 1 en grados.

7.4.28. Ingrese el ángulo de hélice del engrane 1 en grados.

7.4.29. Ingrese la distancia a la que se encuentra el engrane 2

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 4 y 5 en pulgadas, como se muestra en el gráfico.

7.4.30. Ingrese la potencia transmitida en el engrane 2 en hp.

7.4.31. Ingrese el radio del engrane 2 en pulgadas

7.4.32. Opción de los planos 2

Seleccione el plano sobre el cual ejerce el engrane 2

7.4.33. Ingrese el ángulo de presión del engrane 2 en grados.

7.4.34. Ingrese el ángulo de hélice del engrane 2 en grados.

7.4.35. Ingrese la distancia a la que se encuentra el engrane 3 en pulgadas.

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 6 y 7 en pulgadas, como se muestra en el gráfico.

7.4.36. Ingrese la potencia transmitida en el engrane 3 en hp.

7.4.37. Ingrese el radio del engrane 3 en pulgadas.

7.4.38. Opción de los planos 3

Seleccione el plano sobre el cual ejerce el engrane 3

7.4.39. Ingrese el ángulo de presión del engrane 3 en grados.

7.4.40. Ingrese el ángulo de hélice del engrane 3 en grados.

7.4.41. Ingrese la distancia a la que se encuentra el engrane 4 en pulgadas.

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 8 y 9 en pulgadas, como se muestra en el gráfico.

7.4.42. Ingrese la potencia transmitida en el engrane 4 en hp.

7.4.43. Ingrese el radio del engrane 4 en pulgadas.

7.4.44. Opción de los planos 4

Seleccione el plano sobre el cual ejerce el engrane 4

7.4.45. Ingrese el ángulo de presión del engrane 4 en grados.

7.4.46. Ingrese el ángulo de hélice del engrane 4 en grados.

7.4.47. Opción de ajuste de engrane 1

7.4.48. Ingrese el tamaño básico 1 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.4.49. Opción de ajuste de engrane 2

7.4.50. Ingrese el tamaño básico 2 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.4.51. Opción de ajuste de engrane 3

7.4.52. Ingrese el tamaño básico 3 en pulgadas.

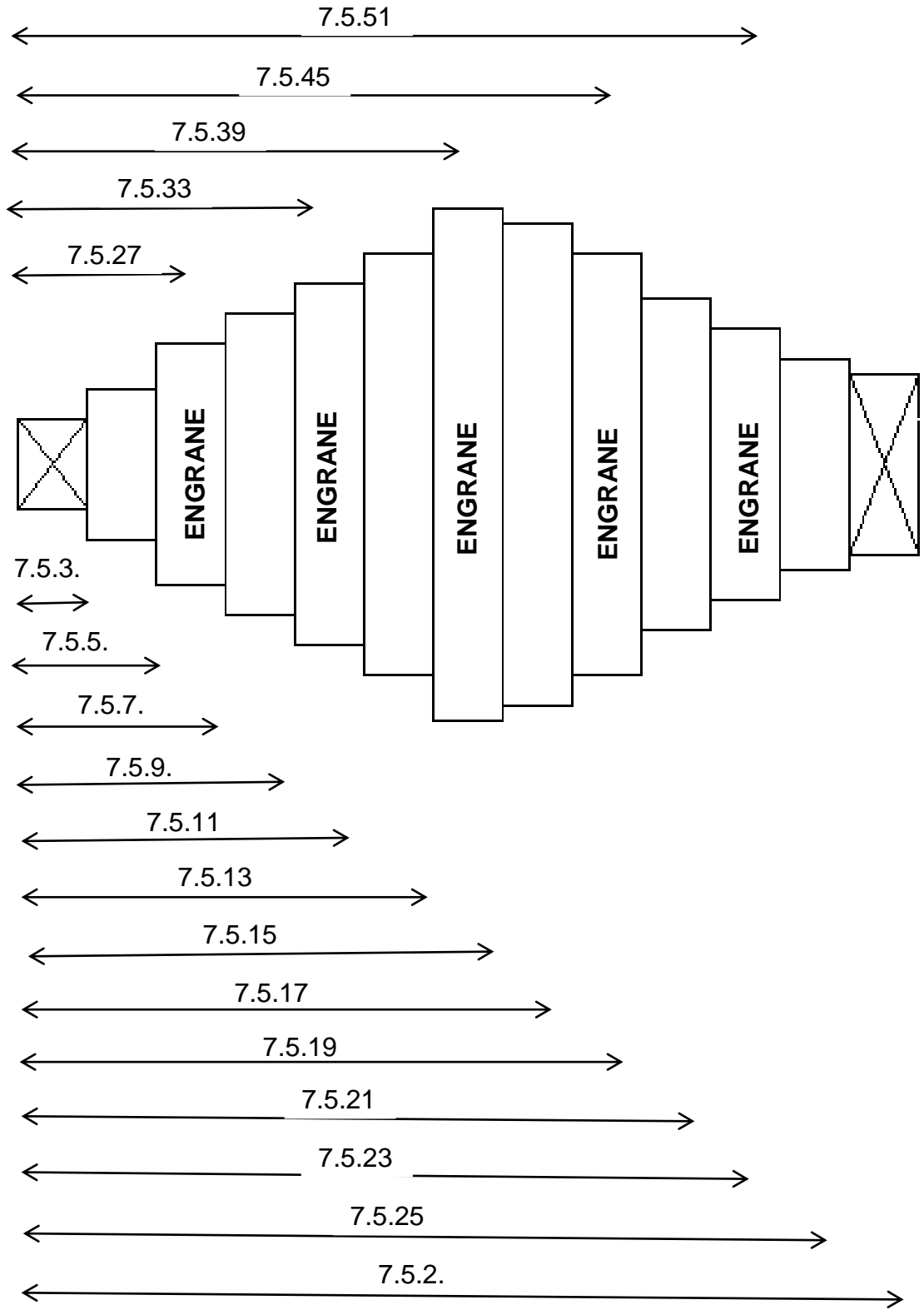
Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.4.53. Opción de ajuste de engrane 4

7.4.54. Ingrese el tamaño básico 4 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.5. Cinco engranes



- 7.5.1. Ingrese la velocidad angular del eje**
Ingrese la velocidad angular que tendrá el eje en rpm.
- 7.5.2. Ingrese la longitud del eje**
Ingrese la longitud total del eje en pulgadas.
- 7.5.3. Ingrese la distancia de la sección 1**
Ingrese la distancia de la sección, en la cual está el cojinete (sección 1) en pulgadas. Como se muestra en el gráfico.
- 7.5.4. Ingrese el radio de la muesca de la sección 1**
Ingrese el radio de la muesca del primer cambio diámetro en pulgadas.
- 7.5.5. Ingrese la distancia de la sección 2**
Ingrese la distancia de la sección 2 en pulgadas. Como se muestra en el gráfico.
- 7.5.6. Ingrese el radio de la muesca de la sección 2**
Ingrese el radio de la muesca del segundo cambio diámetro en pulgadas.
- 7.5.7. Ingrese la distancia de la sección 3**
Ingrese la distancia de la sección 3 en pulgadas. Como se muestra en el gráfico.
- 7.5.8. Ingrese el radio de la muesca de la sección 3**
Ingrese el radio de la muesca del tercer cambio diámetro en pulgadas.
- 7.5.9. Ingrese la distancia de la sección 4**
Ingrese la distancia de la sección 4 en pulgadas. Como se muestra en el gráfico.
- 7.5.10. Ingrese el radio de la muesca de la sección 4**
Ingrese el radio de la muesca del cuarto cambio diámetro en pulgadas.
- 7.5.11. Ingrese la distancia de la sección 5**
Ingrese la distancia de la sección 5 en pulgadas. Como se muestra en el gráfico.
- 7.5.12. Ingrese el radio de la muesca de la sección 5**
Ingrese el radio de la muesca del quinto cambio diámetro en pulgadas.
- 7.5.13. Ingrese la distancia de la sección 6**
Ingrese la distancia de la sección 6 en pulgadas. Como se muestra en el gráfico.
- 7.5.14. Ingrese el radio de la muesca de la sección 6**
Ingrese el radio de la muesca del sexto cambio diámetro en pulgadas.
- 7.5.15. Ingrese la distancia de la sección 7**
Ingrese la distancia de la sección 7 en pulgadas. Como se muestra en el gráfico.
- 7.5.16. Ingrese el radio de la muesca de la sección 7**

Ingrese el radio de la muesca del séptimo cambio diámetro en pulgadas.

7.5.17. Ingrese la distancia de la sección 8

Ingrese la distancia de la sección 8 en pulgadas. Como se muestra en el gráfico.

7.5.18. Ingrese el radio de la muesca de la sección 8

Ingrese el radio de la muesca del octavo cambio diámetro en pulgadas.

7.5.19. Ingrese la distancia de la sección 9

Ingrese la distancia de la sección 9 en pulgadas. Como se muestra en el gráfico.

7.5.20. Ingrese el radio de la muesca de la sección 9

Ingrese el radio de la muesca del noveno cambio diámetro en pulgadas.

7.5.21. Ingrese la distancia de la sección 10

Ingrese la distancia de la sección 10 en pulgadas. Como se muestra en el gráfico.

7.5.22. Ingrese el radio de la muesca de la sección 10

Ingrese el radio de la muesca del décimo cambio diámetro en pulgadas.

7.5.23. Ingrese la distancia de la sección 11

Ingrese la distancia de la sección 11 en pulgadas. Como se muestra en el gráfico.

7.5.24. Ingrese el radio de la muesca de la sección 11

Ingrese el radio de la muesca del onceavo cambio diámetro en pulgadas.

7.5.25. Ingrese la distancia de la sección 12

Ingrese la distancia de la sección 12 en pulgadas. Como se muestra en el gráfico.

7.5.26. Ingrese el radio de la muesca de la sección 12

Ingrese el radio de la muesca del doceavo cambio diámetro en pulgadas.

7.5.27. Ingrese la distancia a la que se encuentra el engrane 1

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 2 y 3 en pulgadas, como se muestra en el gráfico.

7.5.28. Ingrese la potencia transmitida en el engrane 1 en hp.

7.5.29. Ingrese el radio del engrane 1 en pulgadas.

7.5.30. Opción de los planos 1

Seleccione el plano sobre el cual ejerce el engrane 1

7.5.31. Ingrese el ángulo de presión del engrane 1 en grados.

7.5.32. Ingrese el ángulo de hélice del engrane 1 en grados.

7.5.33. Ingrese la distancia a la que se encuentra el engrane 2 en pulgadas

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 4 y 5 en pulgadas, como se muestra en el gráfico.

7.5.34. Ingrese la potencia transmitida en el engrane 2 en hp.

7.5.35. Ingrese el radio del engrane 2 en pulgadas

7.5.36. Opción de los planos 2

Seleccione el plano sobre el cual ejerce el engrane 2

7.5.37. Ingrese el ángulo de presión del engrane 2 en grados.

7.5.38. Ingrese el ángulo de hélice del engrane 2 en grados.

7.5.39. Ingrese la distancia a la que se encuentra el engrane 3

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 6 y 7 en pulgadas, como se muestra en el gráfico.

7.5.40. Ingrese la potencia transmitida en el engrane 3 en hp.

7.5.41. Ingrese el radio del engrane 3 en pulgadas.

7.5.42. Opción de los planos 3

Seleccione el plano sobre el cual ejerce el engrane 3

7.5.43. Ingrese el ángulo de presión del engrane 3 en grados.

7.5.44. Ingrese el ángulo de hélice del engrane 3 en grados.

7.5.45. Ingrese la distancia a la que se encuentra el engrane 4

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 8 y 9 en pulgadas, como se muestra en el gráfico.

7.5.46. Ingrese la potencia transmitida en el engrane 4 en hp.

7.5.47. Ingrese el radio del engrane 4 en pulgadas.

7.5.48. Opción de los planos 4

Seleccione el plano sobre el cual ejerce el engrane 4

7.5.49. Ingrese el ángulo de presión del engrane 4 en grados.

7.5.50. Ingrese el ángulo de hélice del engrane 4 en grados.

7.5.51. Ingrese la distancia a la que se encuentra el engrane 5

Ingrese la distancia a la que se encuentra el engrane, tiene que estar entre la distancia de la sección 10 y 11 en pulgadas, como se muestra en el gráfico.

7.5.52. Ingrese la potencia transmitida en el engrane 5 en hp.

7.5.53. Ingrese el radio del engrane 5 en pulgadas.

7.5.54. Opción de los planos 5

Seleccione el plano sobre el cual ejerce el engrane 5

7.5.55. Ingrese el ángulo de presión del engrane 5 en grados.

7.5.56. Ingrese el ángulo de hélice del engrane 5 en grados.

7.5.57. Opción de ajuste de engrane 1

7.5.58. Ingrese el tamaño básico 1 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.5.59. Opción de ajuste de engrane 2

7.5.60. Ingrese el tamaño básico 2 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.5.61. Opción de ajuste de engrane 3

7.5.62. Ingrese el tamaño básico 3 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.5.63. Opción de ajuste de engrane 4

7.5.64. Ingrese el tamaño básico 4 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

7.5.65. Opción de ajuste de engrane 5

7.5.66. Ingrese el tamaño básico 5 en pulgadas.

Ingrese el tamaño básico en pulgadas correspondiente al tipo de ajuste por interferencia.

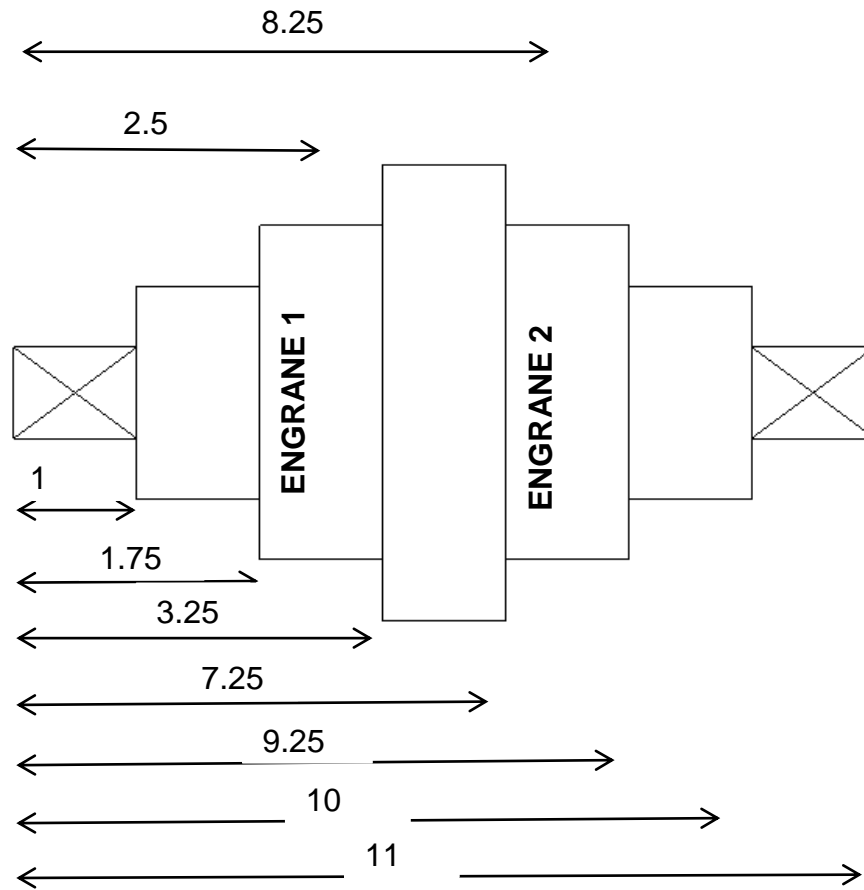
Anexo C. Ejercicio resuelto con aplicación

Además se realizó ejercicio de ejemplo 7-2 del libro Shigley ([2]. Shigley) por medio de la aplicación, en el cual se obtuvieron resultados acordes al ejemplo.

Algunos de los datos ingresados en en la aplicación fueron siguientes:

Variables	Datos
Sut	68Kpsi
Numero de engranes	2
Carga tangencial engrane #1	540lbf
Carga tangencial engrane #2	2431lbf
Carga radial engrane #1	197lbf plano xz
Carga radia engrane #2	885lbf plano xy
Ajuste	Hombro
Radio concentrador	0.16
Criterio	Goodman
Kc=Kd	1

Ademas, la longitud del eje en pulgadas con cada una de las secciones



Los resultados obtenidos se encuentran en la siguiente tabla

Diámetro	Resultados libro	Resultados aplicación
D1	1 pulg.	0.885 pulg.
D2	1.4 pulg.	1.183 pulg.
D3	1.625 pulg.	2.155 pulg.
D4	2 pulg.	2.129 pulg.
D5	1.625 pulg.	2.174 pulg.
D6	1.4 pulg.	1.469 pulg.
D7	1 pulg.	1.211 pulg.
Seg./Prob.	1.56	$1 \cdot 10^{-6}$

Cabe resaltar que la metodología usada en el libro es determinística y se uso un factor de seguridad de 1.56 y el usado por la aplicación evalua el criterio con una probabilidad de falla $1 \cdot 10^{-6}$.