



T
624.80285
065
0-2

ESCUELA SUPERIOR POLITECNICA DEL LITORAL

Facultad de Ingeniería en Mecánica

**"DISEÑO DE UN PUENTE-GRUA ELEVADO
ELECTRICO MEDIANTE COMPUTADORA
EMPLEANDO LAS RECOMENDACIONES
DEL MANUAL DEL A.I.S.C."**

TESIS DE GRADO

Previa a la obtención del Título de:

INGENIERO MECANICO

Presentado por:

LEONARDO FABIAN ORDOÑEZ JARA

GUAYAQUIL

ECUADOR

AÑO

1993

AGRADECIMIENTO

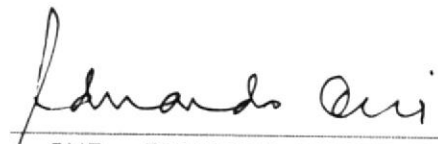
- Al Ing. Alfredo Torres por sus valiosos consejos y dirección en el desarrollo de este trabajo.
- Al Ing. Manuel Helguero por su desinteresada colaboración.
- A Rubén e Iván mis hermanos; por su apoyo moral y paciencia.

DEDICATORIA

- A Dios.
- A la felicidad de mi familia.
- A Juana, Gustavo, Jorge, Hans y Freddy, quienes me han honrado con su afecto y amistad.
- A Olga e Iván, a quienes siempre tengo en cuenta.


33-12506
DR. ALFREDO BARRIGA R.
DECANO DE LA FACULTAD
DE INGENIERIA MECANICA


ING. ALFREDO TORRES G.
DIRECTOR DE TESIS


ING. EDUARDO ORCES P.
MIEMBRO TRIBUNAL PRINCIPAL


ING. JORGE FELIX N.
MIEMBRO TRIBUNAL PRINCIPAL

DECLARACION EXPRESA

"La responsabilidad por los hechos, ideas y doctrinas expuestos en esta tesis, corresponden exclusivamente a su autor; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITECNICA DEL LITORAL".

(De acuerdo al Reglamento de Exámenes y Títulos Profesionales de la ESPOL).

LEONARDO FABIAN ORDOÑEZ JARA

RESUMEN

El proyecto que se presenta a continuación, tiene como objetivo el diseño de un puente-grúa elaborando programas de computadora para cada una de sus partes constitutivas.

Para cada elemento, se hace un análisis estructural, y luego mediante algunas recomendaciones se realiza el programa para el diseño, presentando su respectivo diagrama de flujo y algoritmo.

El lenguaje utilizado es el Turbo C, debido a que su empleo es ilimitado y significativo. Además tiene gran poder debido a la cuidadosa inclusión de las estructuras correctas de control y los tipos de datos.

Cada programa requiere de una serie de datos que deberán ser proporcionados por el usuario. Para el caso de los elementos estructurales, el programa nos proporciona el perfil más económico luego de verificar todas las especificaciones.

El proyecto presenta un costo final del diseño y su conveniencia o no para su construcción definitiva.

Las especificaciones y recomendaciones para el diseño han sido tomadas del manual de Construcción de Acero A.I.S.C., octava edición, 1980, así como del texto de Diseño Mecánico de Shigley, quinta edición, 1990. Y de estos se seleccionan los parámetros implícitos en el programa.

INDICE GENERAL

	Pág.
RESUMEN	VI
INDICE GENERAL	VII
INDICE DE FIGURAS	X
NOMENCLATURA	XI
INTRODUCCION	XIII
CAPITULO I	
INTRODUCCION TEORICA	
1.1 Las grúas: generalidades	15
1.2 Tipos de grúas	16
1.3 El puente-grúa elevado	18
1.3.1 Partes constitutivas	19
1.4 Características del puente a ser diseñado	21
1.5 Factibilidad	22
CAPITULO II	
ANALISIS Y DISEÑO DEL PROYECTO	
2.1 Factores de diseño	24
2.2 Factor de seguridad	25
2.3 Tipos de cargas	26
2.4 Características del puente-grúa	27
2.4.1 Recomendaciones para seleccionar dimensiones, cargas y velocidades	27
2.4.2 Diagrama de flujo y algoritmo	29
2.5 La viga principal	38
2.5.1 Análisis de deformaciones y determinación	

	de máximos	38
2.5.2	Recomendaciones del manual del A.I.S.C. para seleccionar el perfil más ligero	42
2.5.3	Diagrama de flujo y algoritmo	47
2.6	Los soportes laterales	69
2.6.1	Análisis de deformaciones y determinación de máximos	69
2.6.2	Recomendaciones del manual A.I.S.C. para seleccionar el perfil más ligero	72
2.6.3	Diagrama de flujo y algoritmo	75
2.7	La viga de asiento para el riel y sus placas de apoyo	91
2.7.1	Análisis de deformaciones y determinación de reacciones y momentos	91
2.7.2	Recomendaciones del manual A.I.S.C. para seleccionar el perfil más ligero y las dimensiones de las placas	97
2.7.3	Diagrama de flujo y algoritmo	101
2.8	El gancho del aparejo elevador	124
2.8.1	Análisis de esfuerzos	124
2.8.2	Recomendaciones para determinar el área óptima y dimensiones	127
2.8.3	Diagrama de flujo y algoritmo	130
2.9	Las uniones soldadas	137
2.9.1	Recomendaciones del manual A.I.S.C. para determinar las dimensiones	139
2.9.2	Diagrama de flujo y algoritmo	141

2.10	Los ejes de las ruedas y chavetas	150
2.10.1	Análisis de deformaciones y determinación de máximos	151
2.10.2	Recomendaciones para determinar el diámetro	153
2.10.3	Diagrama de flujo y algoritmo	157
2.11	Selección de los rodamientos para los ejes de las ruedas	170
2.12	Protección del puente-grúa	172
CAPITULO III		
RESULTADOS Y ANALISIS DE COSTOS		
3.1	Resultados obtenidos	175
3.2	Costos de cada elemento del puente-grúa	194
3.3	Costo total del puente-grúa	195
CONCLUSIONES Y RECOMENDACIONES		196
APENDICE A: MANUAL PARA EL USUARIO		200
PROGRAMA FINAL		201
APENDICE B: PLANOS		204
BIBLIOGRAFIA		210

INDICE DE FIGURAS

No.		Pág.
1	El puente-grúa elevado	23
2	La viga principal	24
3	El soporte lateral	24
4	La viga porta-grúa	24
5	Conjunto del puente-grúa y viga porta-grúa	25
6	Viga para el carro del trole sometida a flexión .	45
7	Viga principal sometida a flexión	46
8	Soporte lateral sometido a flexión	74
9	Viga porta-grúa sometida a flexión	100
10	Gancho del aparejo elevador	129
11	Eje de las ruedas sometido a flexión	156

NOMENCLATURA

AISC	American Institute of Steel Construction
Ag	Area de la sección transversal
Cb	Coefficiente de flexión
E	Módulo de elasticidad del acero (29000 ksi)
F	Fuerza equivalente de impacto
Fb	Esfuerzo permisible en flexión
Fc	Esfuerzo de aplastamiento
Ft	Esfuerzo permisible en tensión
Fu	Resistencia última de acuerdo al tipo de acero
Fv	Resistencia permisible en cortante
Fy	Resistencia a la fluencia de acuerdo al tipo de acero
I	Momento de inercia de la sección
K	Factor de longitud efectiva
L	Longitud de la viga
Lc	Longitud mínima sin arriostrar para el patín de compresión
Lna	Longitud no arriostrada
Lu	Longitud mínima sin arriostrar después de la cual $F_b \leq 0.6 F_y$
M	Momento
N	Longitud mínima de apoyo para que el alma de la viga no se aplaste
S	Módulo elástico de la sección transversal
T	Carga concentrada
Tm	Torque producido por el motor sobre el eje

V	Cortante
f_b	Esfuerzo actual de flexión
f_c	Esfuerzo actual de aplastamiento
f_v	Esfuerzo actual en cortante
m	Masa de un sólido
r_t	Radio del giro
W	Carga uniformemente distribuida
Kip	1000 lb
Δ	Desplazamiento vertical de la viga
δ	Deformación elástica dinámica o de impacto
δ_{st}	Deformación elástica estática
σ	Esfuerzo de flexión
τ	Esfuerzo cortante

INTRODUCCION

El presente trabajo tiene como propósito, el diseño de un puente-grúa elevado de corredera mediante programas de computadora y presentarlos en forma de menú para facilitar la manipulación por parte del usuario.

Además se hace un estudio de costos, para ver si es factible su construcción y montaje; de esta manera se tendrá una idea hasta que punto es rentable este tipo de diseño.

El diseño estructural se hace en base a las recomendaciones que nos presenta el manual de la American Institute of Steel Construction, Inc., octava edición de 1980; además se presentan unas tablas que se han utilizado para seleccionar algunas características del puente-grúa.

En la primera parte de este trabajo, se hace una introducción técnica general acerca de la grúas, sus características, tipos, partes que lo constituyen, un estudio de factibilidad y las características de puente-grúa a ser diseñado.

En una segunda parte se hace el análisis de cada elemento del puente, previo a la realización de los programas.

Cada programa contiene recomendaciones y especificaciones

para su correcto diseño, un diagrama de flujo y su respectivo algoritmo que está codificado en Turbo C.

Una tercera parte nos presenta los resultados que nos proporcionan los programas para ciertas condiciones específicas; además de un estudio económico del proyecto, válido hasta la fecha, el mismo que varía por efectos de la inflación.

Por último se dan una serie de alternativas para proteger al puente-grúa del efecto de la corrosión y se presenta un manual para manejo por parte del usuario.

CAPITULO I

INTRODUCCION TEORICA

1.1 LAS GRUAS: GENERALIDADES.

En la actualidad los sistemas utilizados para manejo de materiales, incluyen con frecuencia muchos mecanismos individuales integrados en una red que se vuelve un factor dominante en el proyecto de la planta, así como en los diferentes procesos de fabricación.

Pueden requerirse movimientos individuales como elevación, traslación o una complicada combinación de los dos. La secuencia de los movimientos pueden ser de ida y vuelta sobre la misma trayectoria, o bien, puede ser unidireccional sobre un sistema transportador continuo o circulante.

El movimiento necesario, la clase y consistencia del artículo que se va a mover y el proceso con el cual se van a combinar, constituyen una compleja serie de requisitos que se deben estudiar con cuidado, a fin de asegurar una justificación económica y la suficiencia técnica.

El puente-grúa se define como un vehículo que sirve

para elevar, transportar y descender cargas.

A manera muy general consiste de un puente soportado por ruedas en sus extremos; que lleva un aparejo elevador y se mueve sobre una vía elevada, y en el caso de las grúas de pórtico sobre rieles colocados en el piso de la planta.

Los movimientos de la grúa, o sea elevación, recorrido del carro y del puente, pueden ser manuales, eléctricos, neumáticos, hidráulicos o una combinación de ellos.

Las grúas neumáticas se utilizan en donde se vuelve peligroso utilizar electricidad o cuando existe algún sistema que suministre aire. Las grúas más comunes y las más utilizadas se pueden construir para levantar hasta sobre las 450 toneladas y para claros que van más allá de los 40 metros.

1.2 TIPOS DE PUENTE-GRUAS.

De la inmensa variedad existente de grúas, mencionaremos solamente las más utilizadas, y que han sido instaladas dentro de áreas industriales con la finalidad de que sirvan para el traslado de maquinaria o manejo de materiales.

Entre ellas tenemos el puente-grúa de corredera, que en su forma más sencilla consiste de una viga I, soportada

por ruedas en sus extremos, el carro o también llamado trole se mueve sobre las alas o patines inferiores y lleva el aparejo de cadena o cable, que forma la unidad elevadora.

El puente-grúa viajero o eléctrico, consta de dos vigas que son las que forman un puente y en su parte superior tiene los rieles sobre los cuales corre el aparejo elevador. Las vigas armadas están soportadas en sus extremos por carros de plataforma.

Las vigas del puente para las grúas pequeñas son del tipo de viguetas I; pero cuando se trata de claros grandes, se usan vigas de caja para darles rigidez torsional y lateral.

Se puede construir una amplia variedad con el fin de satisfacer condiciones especiales o necesidades de manejo; ejemplos comunes son las grúas apiladoras para introducir y sacar materiales de enrejados, grúas de pared que corren sobre vías colocadas en uno de los lados de un edificio, grúas de movimiento circular o de pivote y semigrúas de caballete (1).

1.3 EL PUENTE-GRUA ELEVADO.

Estas grúas consisten de un aparejo elevador montado en un puente de guía soportado por dos carros con ruedas

(1) Manual del Ingeniero Mecánico. McGraw-Hill, 1984, Pág 10-31.

en cada extremo y se mueve sobre vías soportadas por elementos estructurales (Figura No.1).

Los puentes con vía superior para los carros pueden soportar un peso combinado de la grúa y de la carga de centenares de toneladas, pero los puentes con suspensión o vías inferiores en que los carros van suspendidos de las cajas inferiores de la vía, se utilizan para cargas de menos de 20 toneladas (2).

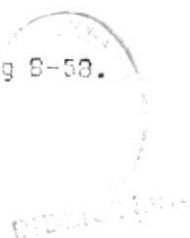
1.3.1 PARTES CONSTITUTIVAS

Constan generalmente de dos vigas armadas que forman el puente, estas vigas son del tipo I (si son muy grandes se utilizan vigas de caja), sobre las cuales va montado el riel donde correrá el carro del aparejo elevador (Figura No.2).

Las vigas I, también se las puede construir, uniendo mediante soldadura dos placas, una inferior y otra superior (alas), una pared vertical (alba), los nervios de refuerzo (rigidizadores) y los nervios laterales (topes) (3).

El puente está soportado en sus extremos, por carros de plataforma con número de ruedas según el tamaño del puente (Figura No.3).

(2) Manual de Mantenimiento Industrial. McGraw-Hill, 1990, Pág 9-58.
(3) Producción de estructuras metálicas. Mir, 1982, Pág 13.



Estos carros se construyen con perfiles estructurales de acero, y las ruedas van en la parte inferior, que se mueven mediante un motor que puede ser colocado en la parte superior del carro (4).

El puente-grúa se mueve a lo largo de vías (vía por lagrías) sobre las cuales van colocadas los rieles por donde se desplaza el carro de plataforma; estas vías son generalmente vigas de tipo I (Figura No.4).

El puente-grúa se desplaza por medio de un motor, que puede impulsar los dos carros, mediante un eje y con la ayuda de engranajes colocados en las ruedas, o puede utilizar dos motores, uno para cada carro.

El aparato eléctrico se utiliza para elevarlo a gran velocidad.

Se pueden utilizar de dos clases; de cadena con capacidad hasta 5 toneladas; de cable de acero, con capacidad hasta 30 toneladas.

El aparato tiene un tambor centrado en el bastidor, el motor y los engranes en extremos opuestos, y el eje del motor pasa a través o a un lado del tambor.

(4) Manual de Mantenimiento Industrial. McGraw-Hill, 1990, Pág 8-73.

El aparejo puede suspenderse de un gancho de ojal, que varía sus dimensiones según la capacidad del puente grúa (5).

Existen además otros elementos, como frenos, interruptores, botoneras, cables, y los carros suelen estar equipados con un segundo aparejo auxiliar de menor capacidad (esto no se cumple siempre).

1.4 CARACTERÍSTICAS DEL PUENTE A SER DISEÑADO.

Para el diseño en lo que a la parte estructural se refiere, es decir diseño de la viga, apoyos laterales, uniones soldadas se utilizará las recomendaciones del manual de la A.I.S.C. (American Institute of Steel Construction) 8ava edición 1980.

La Dresser Industries Inc. - División de Grúas y Elevadores, recomienda el número de ruedas, velocidades y dimensiones en función de la capacidad de la grúa. Ver la tabla No.1 del Apéndice B.

Para el diseño de los ejes de las ruedas y chavetas, se utilizarán los criterios y especificaciones que están detallados en el libro Diseño Mecánico de Shigley.

Para diseñar el gancho se recurrirá a la teoría de vigas curvas. El puente-grúa requerido será diseñado

(5) Fruntuario de Máquinas. Paramito N. Larburu, 1990, Pág 308.

para ser construido posteriormente, y estará ubicado en el C.I.T.D. (Centro de Innovación Tecnológico y Desarrollo) de la Facultad de Ingeniería en Mecánica perteneciente a la ESFOL.

El puente-grúa tendrá los siguientes requerimientos:

- Capacidad de 3 toneladas
- Una luz de 14,40 metros
- Longitud total requerida de la viga portagrúa y de los rieles que cubre toda la longitud del laboratorio, de 34 metros.
- Salvar obstáculos hasta una altura máxima de 3 m, esto para que no haya problemas en el momento de transportar cualquier máquina o material en el laboratorio (Figura No.3).

1.5 FACTIBILIDAD.

La necesidad de transportar maquinaria o materiales a cualquier punto del laboratorio, obligó a realizar un proyecto para el diseño de un puente-grúa.

Debido a que es posible colocar el puente-grúa sobre columnas de concreto ya existentes, se decidió que sería del tipo elevado y eléctrico.

El diseño del proyecto considera ciertos factores importantes:

- El puente-grúa va a ser manipulado dentro de un laboratorio en el cual existe circulación de personas, así que es importante que exista seguridad y calidad que debería brindar.
- Debería ser fácil de usar, es decir, que tenga funcionalidad.
- tener una estética agradable.
- se debe utilizar aceros estructurales para su construcción, y utilizar lo más posible materias existentes en el país.
- En el país se dispone de algunos tipos de perfiles, como por ejemplo: vigas C, vigas L; las vigas I que no se tienen, se pueden construir con la unión de otros perfiles, o de placas; de esta manera evitamos tener que importarlas y bajar de esta manera su costo.
- Las uniones soldadas serán del tipo de filete, y el proceso utilizado será, el SMAW (Shielded metal arc welding).
- Las especificaciones para el acero estructural A.36 y para la soldadura SMAW E60 han sido tomados del Manual de construcción en acero del A.I.S.C.
- Los ejes son de acero de transmisión SAE 1035-CD, y los rodamientos son rígidos de bolas con tapas de protección de la NTN Co.

FIGURA N.º 1 : EL PUENTE-GRUA ELEVADO

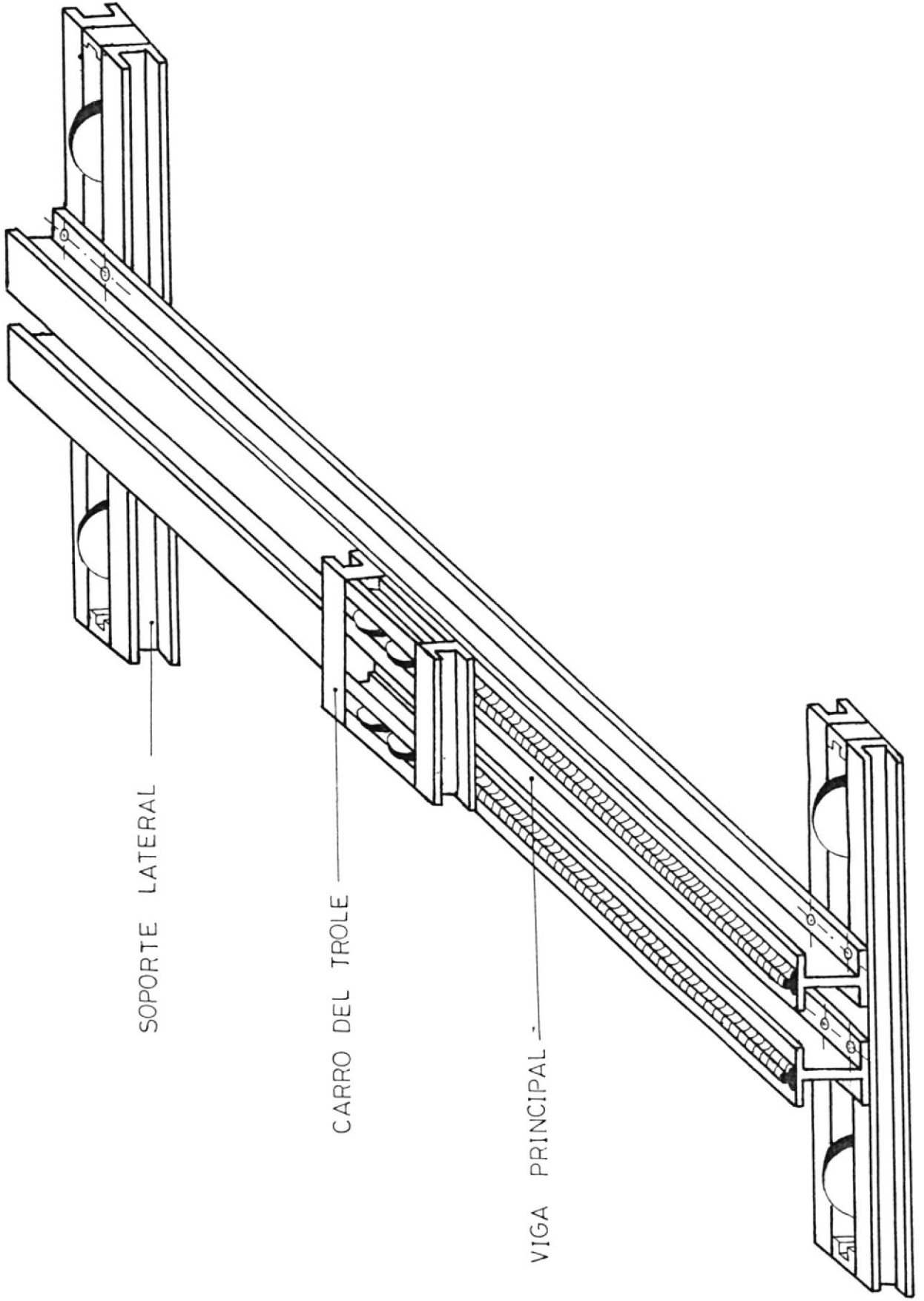


FIGURA Nº 3 : EL SOPORTE LATERAL

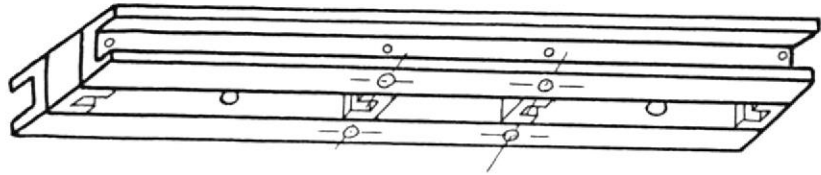
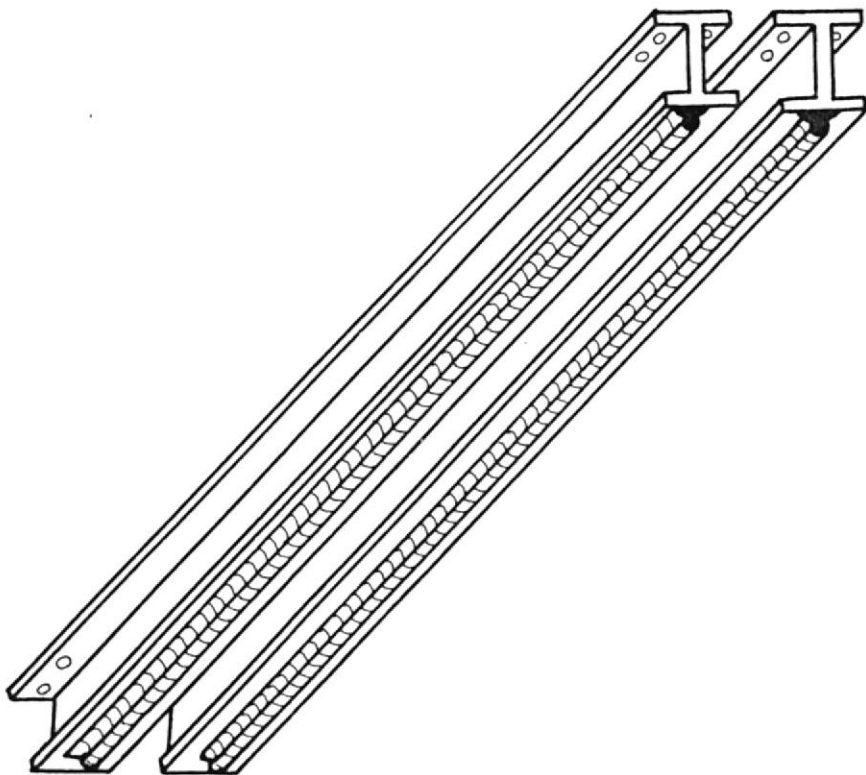


FIGURA Nº 2 : LA VIGA PRINCIPAL



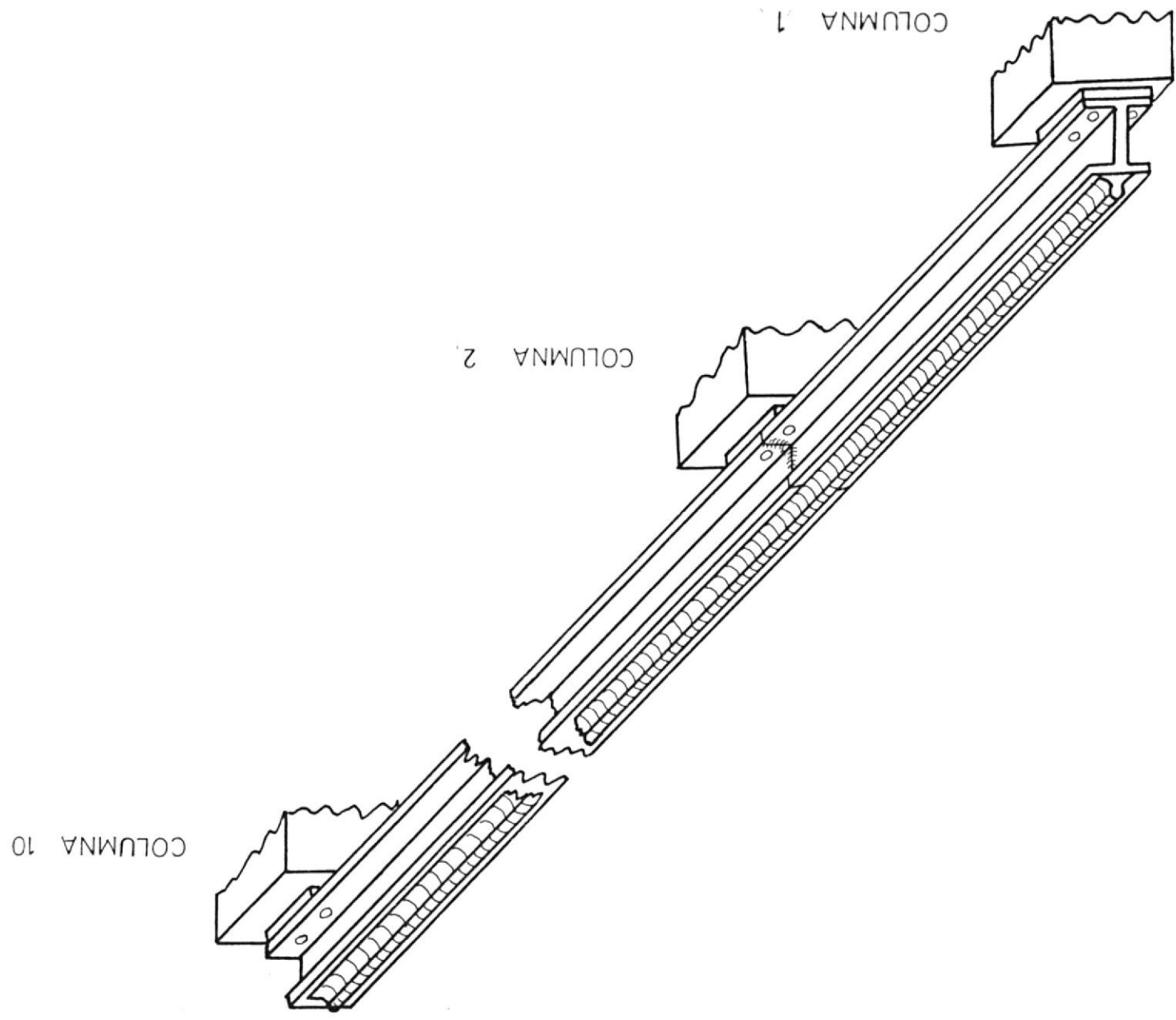
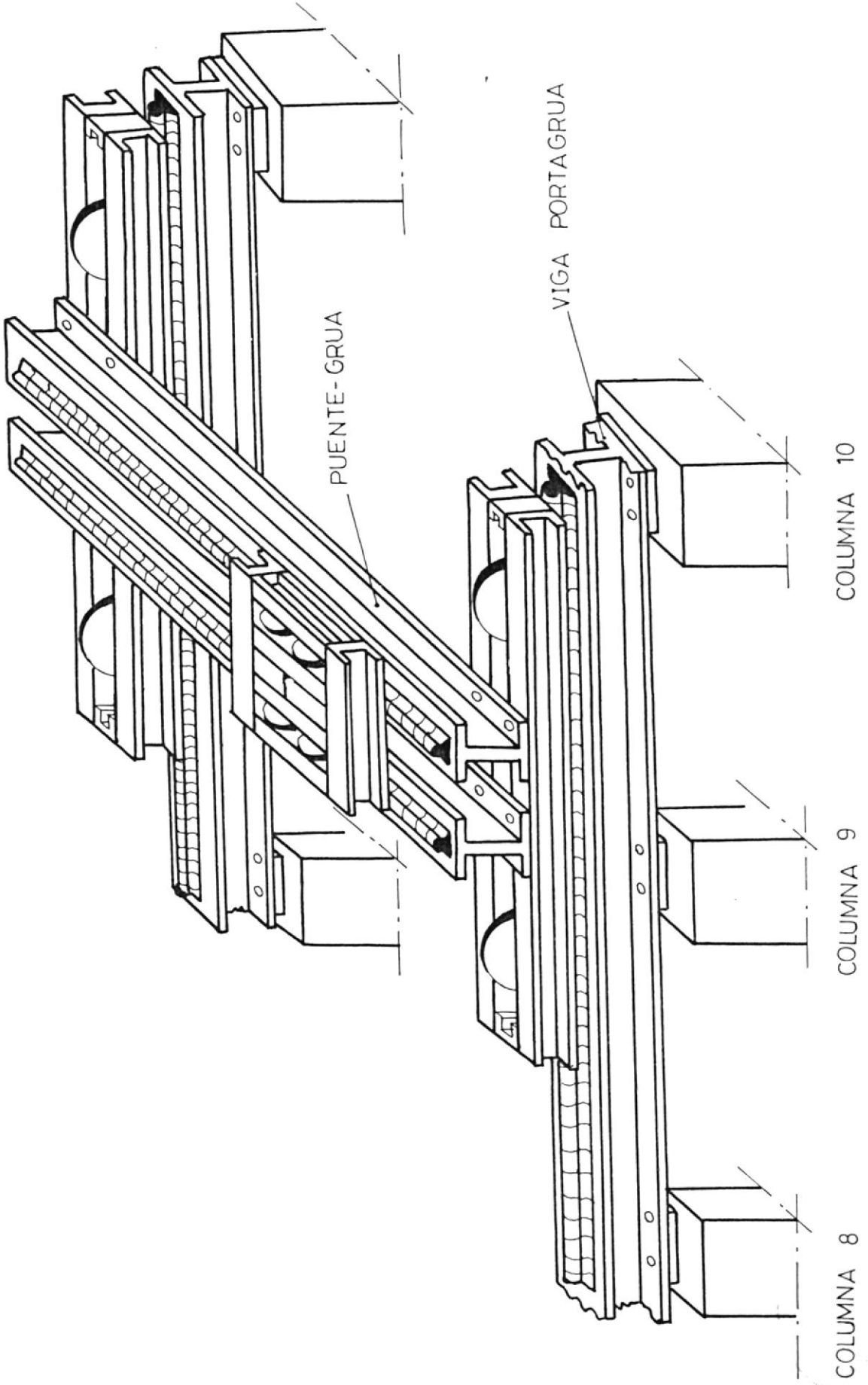


FIGURA N°4 : LA VIGA PORTAGRUA

FIGURA Nº 5 : CONJUNTO PUENTE — VIGA PORTAGRUA



CAPITULO II

ANALISIS Y DISEÑO DEL PROYECTO

2.1 FACTORES DE DISEÑO.

El factor de diseño de los costos es tan importante en el proceso de las decisiones para diseño, y por tanto hay que darle la debida importancia.

Existen algunos principios básicos para reducir los costos. Por ejemplo el uso de tamaños estándares, uso de amplias tolerancias (dimensiones y variación de las propiedades mecánicas), punto de equilibrio (comparación de dos o más diseños), estimación de costos (algunos estimadores como el área, volumen, potencia, capacidad, velocidad, etc.).

Los perfiles utilizados en el puente-grúa son aceros estructurales A36, los ejes de las ruedas de acero de transmisión, el gancho de acero dúctil, pernos de acero, y el cable para elevación de acero lo más flexible que se pueda.

El puente-grúa estará sometido a los siguientes tipos de cargas.

* Carga Muerta, que es el peso propio de la viga o

viguetas y el de los apoyos laterales. El peso de estos dos elementos deberán ser asumidos en un principio.

- * Carga Viva. Que en este caso será la carga que va a levantar la grúa, más el peso del motor y bobina (incluido el gancho). Estos valores no son asumidos.
- $$W_V = W_{LEVANT.} + W_{MOTOR} + W_{BOBINA} + W_{GANCHO}$$

Para saber el tipo de estructura a diseñar, se considerará los siguientes factores:

Situación Geográfica. El puente-grúa será construido dentro de un galpón y en un medio corrosivo, por lo tanto al seleccionar los materiales habrá que tomar en cuenta esta consideración.

Durabilidad: El puente-grúa, tendrá un tiempo de vida útil de unos 20 años.

tipo de perfil de la viga principal: El perfil escogido será un I de la ancha, por ser las más económicas y además resistir los esfuerzos a los que está sometido.

tipo de perfil de los apoyos laterales: Se ha escogido

dos perfiles en L para cada apoyo.

2.2 FACTOR DE SEGURIDAD.

La expresión factor de seguridad significa alguna característica o consideración que influye en el diseño de un elemento o, quizás, en todo el sistema.

Por lo general se tiene que tomar en cuenta varios de esos factores en un caso de diseño determinado. En ocasiones, alguno de esos factores será crítico y, si se satisfacen sus condiciones, ya no será necesario considerar los demás.

El criterio utilizado en este diseño es el factor de seguridad por resistencia, que es un número que evalúa el comportamiento seguro de un elemento.

Empleando el esfuerzo permisible que es el máximo que se puede aplicar con seguridad a un elemento y la resistencia que es una propiedad intrínseca, resulta:

$$n = \frac{S}{\sigma_{\text{PERMISIBLE}}} \left(\frac{\text{Resistencia}}{\text{Esfuerzo normal}} \right)$$

$$n = \frac{S}{\sigma_{\text{PERMISIBLE}}} \left(\frac{\text{Resistencia}}{\text{Esfuerzo cortante}} \right)$$

Donde:

$\sigma_{\text{PERMISIBLE}}$ FLEXION $\leq 0.6 S_y$

$\sigma_{\text{PERMISIBLE}}$ CORTANTE $\leq 0.4 S_y$

EMPLASTAMIENTO ≤ 0.75 Sy

2.3 TIPOS DE CARGAS.

Las cargas que actúan generalmente sobre una estructura y que hay que considerar son la carga permanente, la sobrecarga y los efectos dinámicos de la sobrecarga.

Cuando la sobrecarga actúa sobre la estructura de un modo instantáneo es una sobrecarga móvil, hay que incrementar el valor de la sobrecarga para tener en cuenta los efectos dinámicos. Este incremento se llama impacto.

Las cargas horizontales tales como fuerzas del viento y fuerzas longitudinales han sido despreciadas en vista de que sus efectos no serán relevantes en el diseño del proyecto, debido a que el puente-grúa se encuentra dentro de un galpón con lo cual no se verá afectado por el viento, y debido a lo bajo de la velocidad del trole y los esfuerzos de rozamiento.

2.4 CARACTERISTICAS DEL PUENTE-GRUA.

2.4.1 RECOMENDACIONES PARA SELECCIONAR DIMENSIONES, CARGAS Y VELOCIDADES.

Las grúas eléctricas se construyen para corriente alterna o directa y predomina la primera. Los motores para ambos tipos de corrientes son

especiales para grúas. Los motores de corriente directa suelen tener arrollamientos en serie; los motores de corriente alterna son del tipo de rotor devanado o de jaula de ardilla de dos velocidades los voltajes usuales de C.A. son de 220 y 440 y el más común es 440 (1).

Las características han sido tomadas de la tabla de dimensiones, cargas y velocidades de grúas de tipo industrial que recomienda la división de grúas y elevadores de la Dresser Industries, Inc.

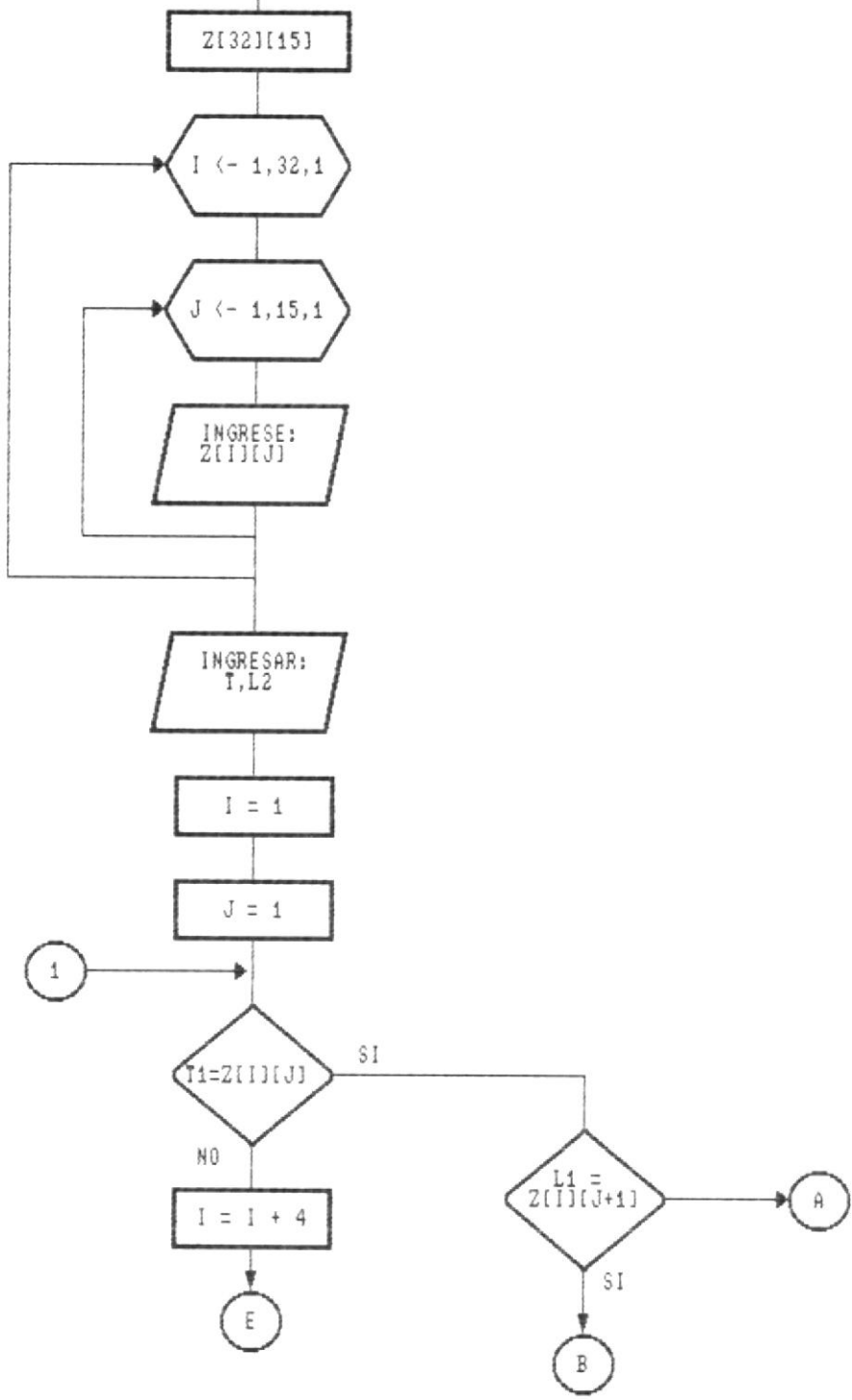
Para grúas de 30 Ton o menos, se recomienda una velocidad normal del carro del trole de 100 a 150 pie/min, y la del puente de 175 a 300 pie/min.

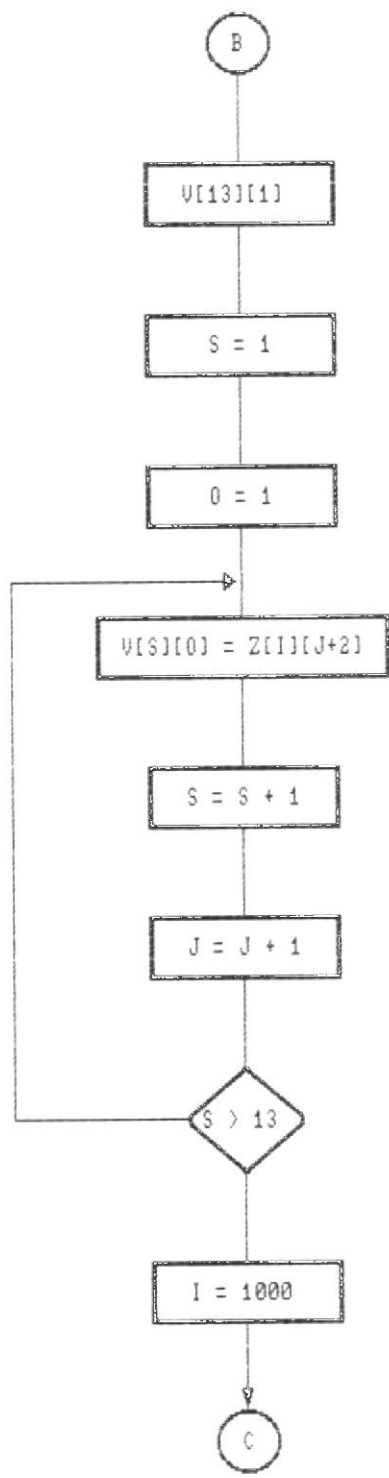
Las dimensiones, distancia entre ruedas del trole, y número de ruedas del puente-grúa están en la Tabla No.1 del Apéndice B.

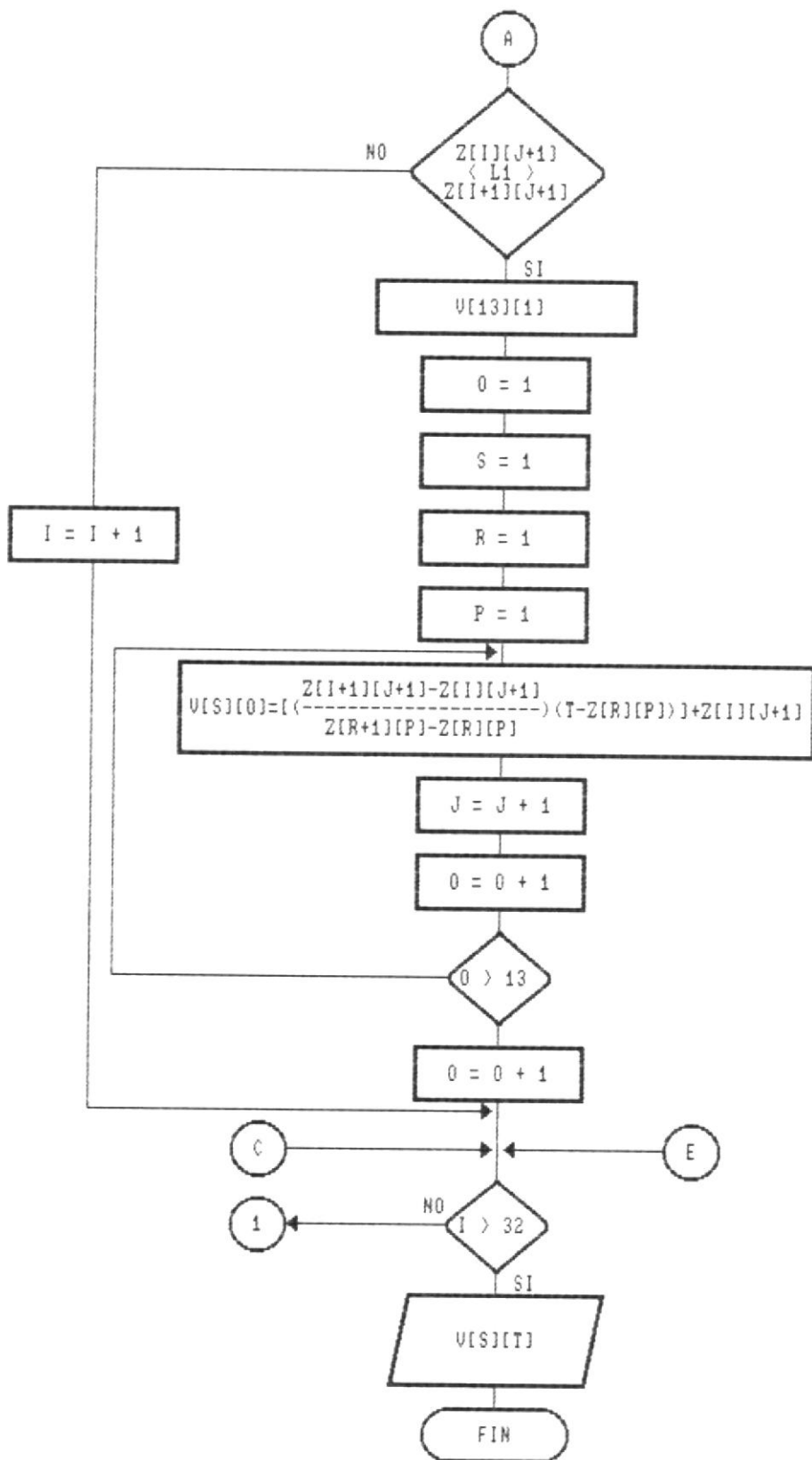
(1) Motores eléctricos. R.J. Lawrie, 1987, Pág. 7.

2.4.2 DIAGRAMA DE FLUJO Y
ALGORITMO PARA SELECCIO-
NAR LAS CARACTERISTICAS
DEL PUENTE-GRUA.

PROGRAMA PARA SELECCIONAR
LAS CARACTERISTICAS DEL PUENTE-GRUA







```

    /**SELECCION DE LAS CARACTERISTICAS***/
    /**DEL PUENTE-GRUA ELEVADO***/

```

```

#include <stdio.h>
#include <math.h>
#include "tesis.h"
#define AND &&
#define OR ||

void main()
{

char    Tc[20],L2c[20];
double  z[16][7];
int     I,J,C;
double  T=0,L2=0,Va=0,wr=0,N=0,L3=0;

clrscr();

for (I=0;I<16;I++) {
    for ( J=0;J<7;J++)
        z[I][J]=0;
}

z[0][0] = 1.0;
z[0][1] = 10.0;
z[0][2] = 40.0;
z[0][3] = 46.0;
z[0][4] = 8.0;
z[0][5] = 0.01;
z[0][6] = 4.0;
z[1][0] = 1.0;
z[1][1] = 10.0;
z[1][2] = 60.0;
z[1][3] = 46.0;
z[1][4] = 9.5;
z[1][5] = 0.01;
z[1][6] = 4.0;
z[2][0] = 1.0;
z[2][1] = 10.0;
z[2][2] = 80.0;
z[2][3] = 46.0;
z[2][4] = 12.0;
z[2][5] = 0.01;
z[2][6] = 4.0;
z[3][0] = 1.0;
z[3][1] = 10.0;
z[3][2] = 100.0;
z[3][3] = 46.0;
z[3][4] = 14.5;
z[3][5] = 0.02;
z[3][6] = 4.0;
z[4][0] = 10.0;
z[4][1] = 15.0;

```

```
z[4][2] = 40.0;  
z[4][3] = 40.0;  
z[4][4] = 9.5;  
z[4][5] = 0.014;  
z[4][6] = 4.0;  
z[5][0] = 10.0;  
z[5][1] = 15.0;  
z[5][2] = 60.0;  
z[5][3] = 40.0;  
z[5][4] = 10.0;  
z[5][5] = 0.02;  
z[5][6] = 4.0;  
z[6][0] = 10.0;  
z[6][1] = 15.0;  
z[6][2] = 80.0;  
z[6][3] = 40.0;  
z[6][4] = 12.0;  
z[6][5] = 0.02;  
z[6][6] = 4.0;  
z[7][0] = 10.0;  
z[7][1] = 15.0;  
z[7][2] = 100.0;  
z[7][3] = 40.0;  
z[7][4] = 14.5;  
z[7][5] = 0.027;  
z[7][6] = 4.0;  
z[8][0] = 15.0;  
z[8][1] = 20.0;  
z[8][2] = 40.0;  
z[8][3] = 26.0;  
z[8][4] = 10.5;  
z[8][5] = 0.02;  
z[8][6] = 4.0;  
z[9][0] = 15.0;  
z[9][1] = 20.0;  
z[9][2] = 60.0;  
z[9][3] = 26.0;  
z[9][4] = 11.0;  
z[9][5] = 0.02;  
z[9][6] = 4.0;  
z[10][0] = 15.0;  
z[10][1] = 20.0;  
z[10][2] = 80.0;  
z[10][3] = 26.0;  
z[10][4] = 11.5;  
z[10][5] = 0.034;  
z[10][6] = 4.0;  
z[11][0] = 15.0;  
z[11][1] = 20.0;  
z[11][2] = 100.0;  
z[11][3] = 26.0;  
z[11][4] = 14.5;  
z[11][5] = 0.027;  
z[11][6] = 4.0;  
z[12][0] = 20.0;
```

```

z[12][1] = 30.0;
z[12][2] = 40.0;
z[12][3] = 28.0;
z[12][4] = 11.0;
z[12][5] = 0.027;
z[12][6] = 4.0;
z[13][0] = 20.0;
z[13][1] = 30.0;
z[13][2] = 60.0;
z[13][3] = 28.0;
z[13][4] = 11.0;
z[13][5] = 0.02;
z[13][6] = 4.0;
z[14][0] = 20.0;
z[14][1] = 30.0;
z[14][2] = 80.0;
z[14][3] = 28.0;
z[14][4] = 11.5;
z[14][5] = 0.027;
z[14][6] = 4.0;
z[15][0] = 20.0;
z[15][1] = 30.0;
z[15][2] = 100.0;
z[15][3] = 28.0;
z[15][4] = 14.5;
z[15][5] = 0.034;
z[15][6] = 4.0;
        do {
printf("\n Ingrese la capacidad del elevador T en Toneladas ");
printf(" (En el rango [1,30]) \n");
printf(" ==> ");
gets(Tc);
T=es_numero(Tc);
fflush(stdin);
} while (( T < 1 ) OR ( T > 30) );
do {
printf("\n Ingrese la longitud de la viga principal L2 en pies: ");
printf(" (En el rango [40,100]) \n");
printf(" ==> ");
gets(L2c);
L2=es_numero(L2c);
fflush(stdin);
} while ( ( L2 < 40  OR  L2 > 100) );

/****SELECCION DE VALORES****/

I = 0;
J = 0;
do {
if ( T >= z[I][J]  &&  T < z[I][J+1] ) {
    C = 1;
        do {

```

```

if (L2 == z[I][J+2]) {
    Va = z[I][3];
    L3 = z[I][4];
    wr = z[I][5];
    N = z[I][6];
    C = 100.0;
}
else {
    if ( L2 > z[I][J+2] ) {
        if (L2 < z[I+1][J+2] ) {
            Va = ( ( (L2-z[I][J+2]) *
                ((z[I+1][J+3]-z[I][J+3])/(z[I+1][J+2]-z[I][J
                +2])) ) + (z[I][J+3]) );
            L3 = ( ( (L2-z[I][J+2]) *
                ((z[I+1][J+4]-z[I][J+4])/(z[I+1][J+2]-z[I][J
                +2])) ) + (z[I][J+4]) );
            wr = ( ( (L2-z[I][J+2]) *
                ((z[I+1][J+5]-z[I][J+5])/(z[I+1][J+2]-z[I][J
                +2])) ) + (z[I][J+5]) );
            N = ( ( (L2-z[I][J+2]) *
                ((z[I+1][J+6]-z[I][J+6])/(z[I+1][J+2]-z[I][J
                +2])) ) + (z[I][J+6]) );
            C = 100.0;
        }
        else {
            if ( L2 == z[I+1][J+2] ) {
                Va = z[I+1][J+3];
                L3 = z[I+1][J+4];
                wr = z[I+1][J+5];
                N = z[I+1][J+6];
                C = 100.0;
            }
            else {
                I = I+1;
            }
        }
    }
}
} while ( C <= 3 );

I = 100.0;
}
else
    I = I+4;
} while ( I <= 16.0 );

clrscr();
printf("\n LA VELOCIDAD NORMAL DEL ELEVADOR EN PIE/MIN ES\n ");
printf("====> %lf ", Va);
printf("\n\n LA DISTANCIA MINIMA ENTRE EJES DE LAS RUEDAS DEL SOPORTE
LATERAL EN PIES ES\n ");
printf("====> %lf ",L3);
printf("\n\n EL PESO DEL RIEL SOBRE LA VIGA PRINCIPAL EN KIPS/PIE
ES\n");

```

```
printf("==> %lf ",wr);
printf("\n\n EL NUMERO DE RUEDAS PARA EL PUENTE-GRUA ES\n ");
printf("==> %lf ",N);
gotoxy(18,23);printf(" [Presione cualquier tecla para continuar:]");
getch();
clrscr();
}
```

2.5 LA VIGA PRINCIPAL.

2.5.1 ANALISIS DE DEFORMACIONES Y DETERMINACION DE MAXIMOS.

Todo tipo de estructura, está constituida por materiales que sufren deformaciones, debido al efecto de diferentes tipos de cargas.

La estructura al verse sometido a una acción externa sobre una deformación general, y si los esfuerzos caen dentro del límite elástico, la deformación desaparece al cesar la acción que lo produjo.

La viga principal es considerada homogénea y simplemente apoyada (extremos empernados); y está sometida a dos tipos de cargas:

- La carga concentrada; que son las reacciones que se producen en las ruedas del carro para el trole, debido al peso de las vigas que forman el mismo, más la capacidad de la grúa.
- La carga uniformemente distribuida, debido al peso de la viga principal, más el peso del riel que esta soporta.

Aplicando el segundo Teorema de Castigliano

tenemos (3):

La deformación máxima en el centro de la viga:

$$\Delta_{MAX} = \int_0^L \frac{M \left(\frac{\delta P}{EI} \right)}{EI} dx$$

$$\Delta_{MAX} = \frac{T_2 a}{24 EI} (4a^2 - 3L^2_2) + \frac{5}{384} (W_R + W_2) \frac{L^4_2}{EI}$$

Donde:

$$T_2 = T_1 + W_1 L_1$$

$$T_1 = \frac{T + P}{4}$$

T : Capacidad de la grúa

T₁: Carga concentrada que actúa sobre el carro del trole

T₂: Carga concentrada que actúa sobre la viga principal

P : Peso del motor y la bobina

W₁: Peso propio de la viga que forma el carro del trole

W₂: Peso propio de la viga principal

W_R: Peso del riel

δ = ?

Cuando la carga es aplicada en forma brusca recibe el nombre de dinámica o de impacto (4).

La deformación elástica producida por una carga de impacto es:

$$\delta = \delta_{ST} + \sqrt{(\delta_{ST})^2 + 2\delta_{ST}h}$$

Para nuestro análisis:

$$h = 0$$

Donde:

$$\delta = 2\delta_{ST}$$

La fuerza equivalente del impacto es:

$$F = 2T$$

Los máximos se determinan realizando los diagramas de cortante y momento flector (5):

$$R_1 = R_2 = T_2 + \frac{W_{RL}L_2}{2}$$

Cortante máximo:

$$V_{MAX} = T_2 + \frac{W_{RL}L_2}{2}$$

(4) Resistencia de materiales. Singer, 1982, Pág. 434.

(5) Diseño en Ingeniería Mecánica. Shigley, 1990, Pág. 834.

Momento flector máximo:

$$M_{MAX} = \frac{1}{8} W_R L^2 z + a T z$$

Esfuerzo máximo por flexión:

$$\sigma_{MAX} = \frac{\frac{1}{8} W_R L^2 z + a T z}{S}$$

Esfuerzo cortante máximo:

$$\sigma_{MAX} = \frac{V_{MAX}}{bh^3} \left[(tf * bf) \left(\frac{bw}{2} + \frac{tf}{2} \right) + \left(\frac{bw}{2} * tw \right) \left(\frac{bw}{4} \right) \right]$$

(Σ---)bf
12

2.5.2 RECOMENDACIONES DEL MANUAL DEL A.I.S.C. PARA SELECCIONAR EL PERFIL MAS LIGERO.

Para diseñar la viga o vigueta debemos tener en cuenta lo siguiente:

- a.- Se trata de una viga sin apoyo lateral continuo, donde las alas en compresión pueden pandearse localmente; lo que deberá tomarse en cuenta en el momento del diseño.
- b.- El perfil a seleccionarse deberá ser el más económico, es decir el más ligero pero que satisfaga las condiciones.
- c.- El perfil será una viga I de ala ancha, porque estas han demostrado ser las más económicas.
- d.- La viga será diseñada para carga de impacto, por ser esta la más crítica, y su punto de aplicación será en el centro de la viga.
- e.- Para el diseño hay que considerar:
 - Momento flector
 - Fuerza cortante
 - Deformación
 - Aplastamiento
 - Pandeo local

El manual del A.I.S.C. por su parte hace las siguientes recomendaciones, para el caso de elementos sometidos a flexión:

En un diseño el esfuerzo actual de flexión debe ser menor que el esfuerzo permisible en flexión
Pág 2.6.

Vigas con apoyo lateral continuo:

Las limitaciones para las alas de una sección compacta para las alas y el alma. A.I.S.C. 1.5 - 1.4 Pág 5.20.

Para las no compactas Pág 5.21.

Para las secciones parcialmente compactas
A.I.S.C. 1.5.1.4.1.

Vigas sin apoyo lateral continuo:

El manual recomienda expresiones para determinar los esfuerzos permisibles en las vigas; estas están dadas en las fórmulas:

$$F_b = \left[\frac{2}{3} - \frac{F_y(L/RT)^2}{1530 * 10^3 C_b} \right] F_y \quad 1.5.6-a$$

$$F_b = \frac{170 * 10^3 C_b}{(L/RT)^2} \quad 1.5.6-b$$

$$F_b = \frac{12 * 10^3 C_b}{(L/RT)^2} \quad 1.5-7$$

Para el diseño por esfuerzo cortante, el manual recomienda usar

$$F_v = 0.4 F_y \quad \text{A.I.S.C. 1.5.1.2.1. - Pág 5.18}$$

Los valores del cortante máximo (V) se encuentran tabulados en Pág 2-27.

El esfuerzo de aplastamiento f_c ; fórmulas en Pág 5.35:

$$R / t(N + 2k) \leq 0.75 F_y \quad 1.10.8$$

$$R / t(N + k) \leq 0.75 F_y \quad 1.10.9$$

Deflexión en vigas:

La máxima deflexión para carga viva en Pág 5.42:

$$\Delta_{max} = \frac{1}{360} L$$

Para vigas sometidas a carga de impacto y vibraciones A.I.S.C. 1.13.1 - Pág 5.1.38.

$$L/d \leq 20$$

El módulo de la sección transversal para perfiles W. Pág 2.6 - 2.9.

FIGURA N° 6: VIGA PARA EL CARRO DEL TROLE SOMETIDA A FLEXION

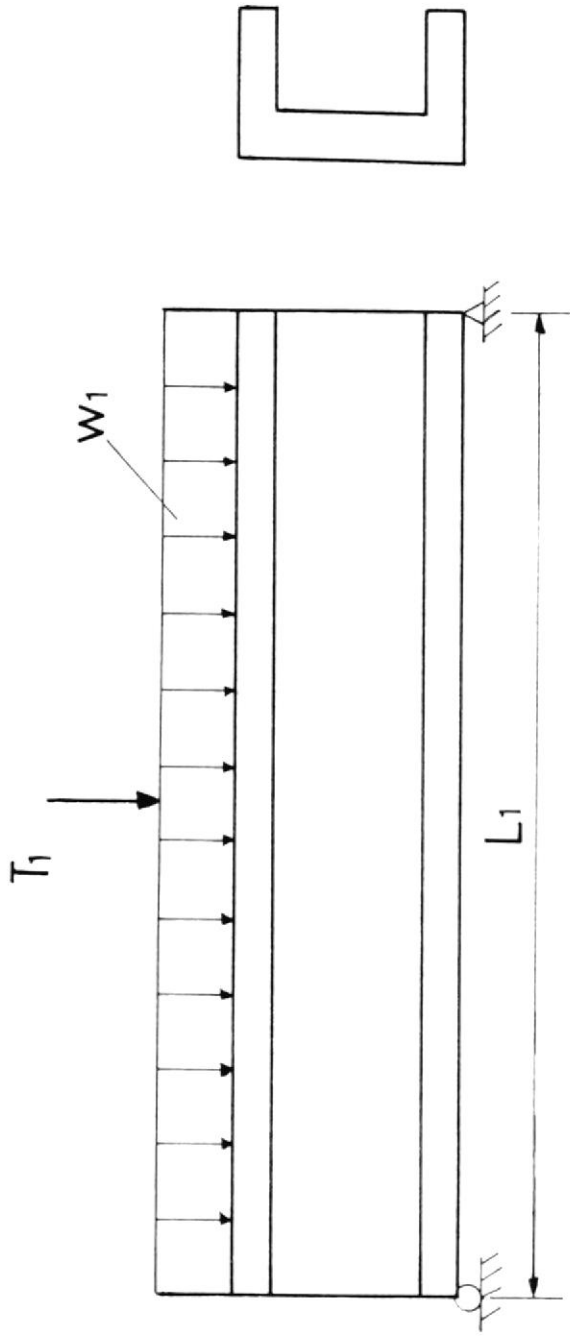
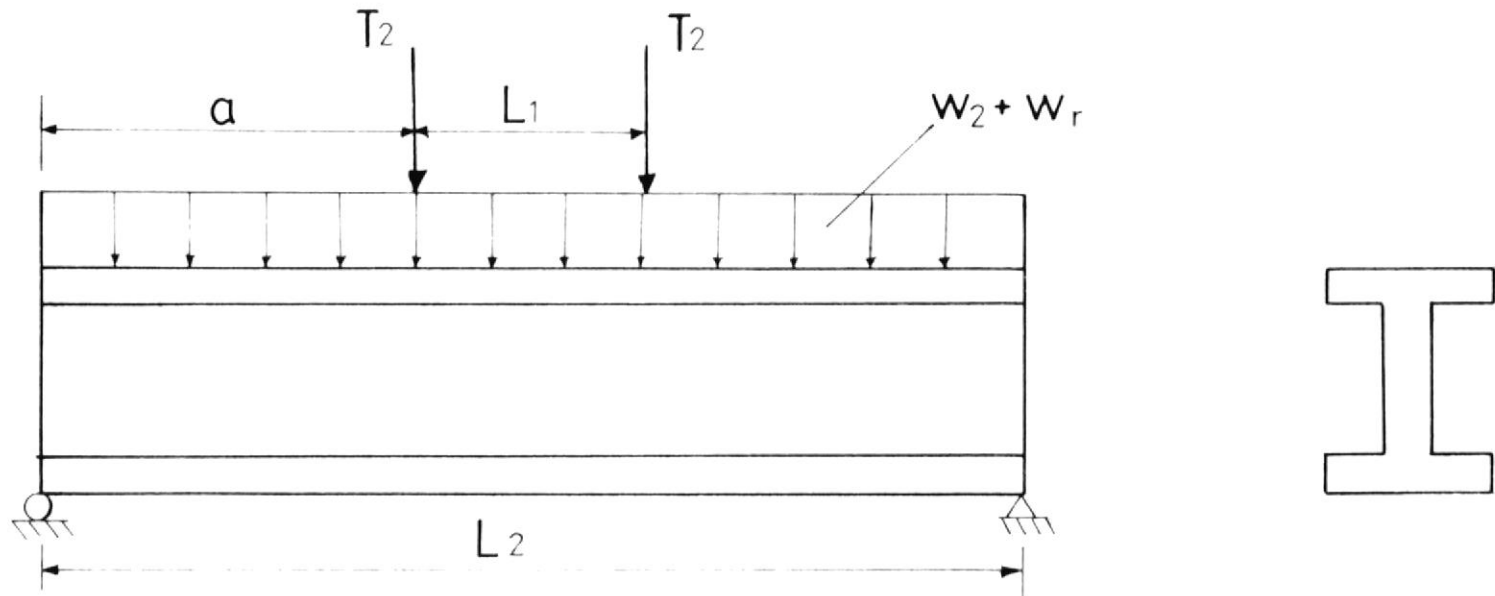


FIGURA N° 7: VIGA PRINCIPAL SOMETIDA A FLEXION



2.5.3 DIAGRAMA DE FLUJO Y ALGORITMO

DISEÑO DE LA VIGA PRINCIPAL

INTRODUCIR:

$F_y, F_u, T, W, E, P,$
 W_x, L_{na_2}, L_2, a

$$M = \frac{1}{8} W_x L_2^2 + T_2 * a$$

$$F_b = 0.66 F_y$$

$$S = M/F_b$$

1

SELECCIONE EL PERFIL
 $A, b_f, t_f, t_w, d,$
 r_t, S_x, K, W

$$A_f = b_f * t_f$$

SUB . L_c, L_u

NO
A

$L_{na} \leq L_c$

SI

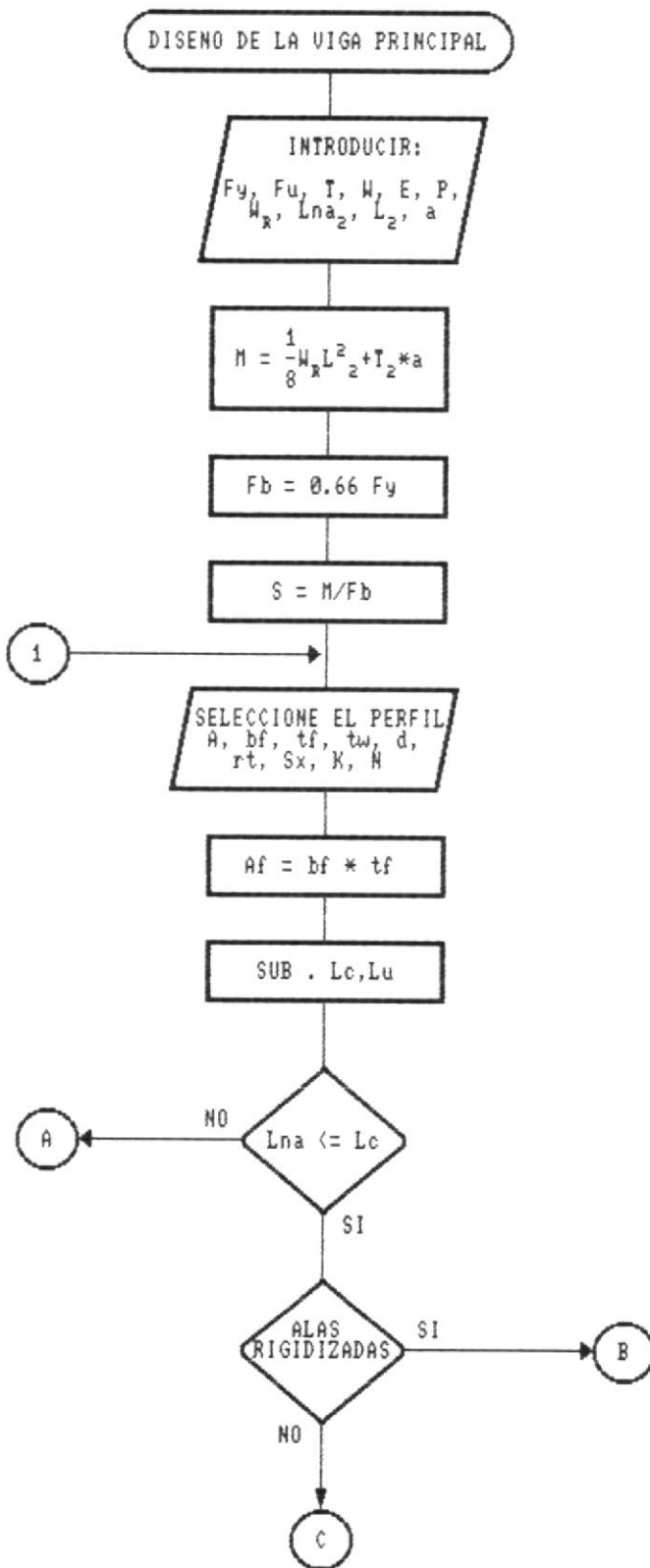
ALAS
RIGIDIZADAS

SI

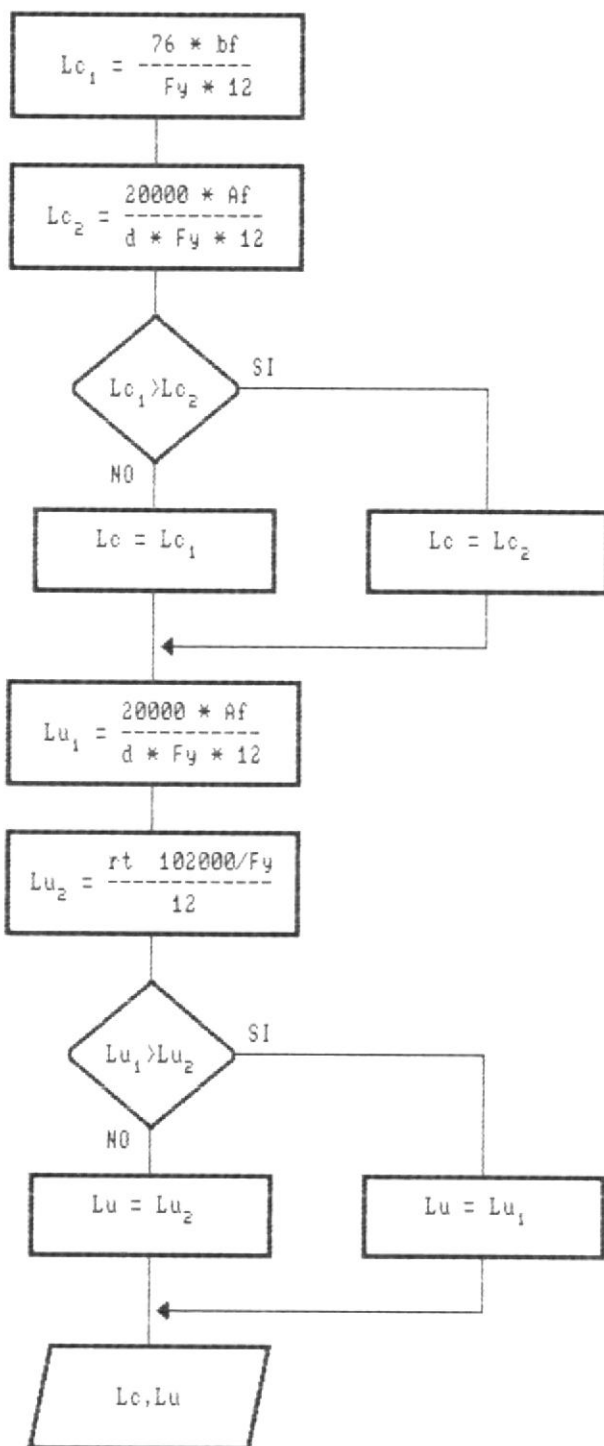
NO

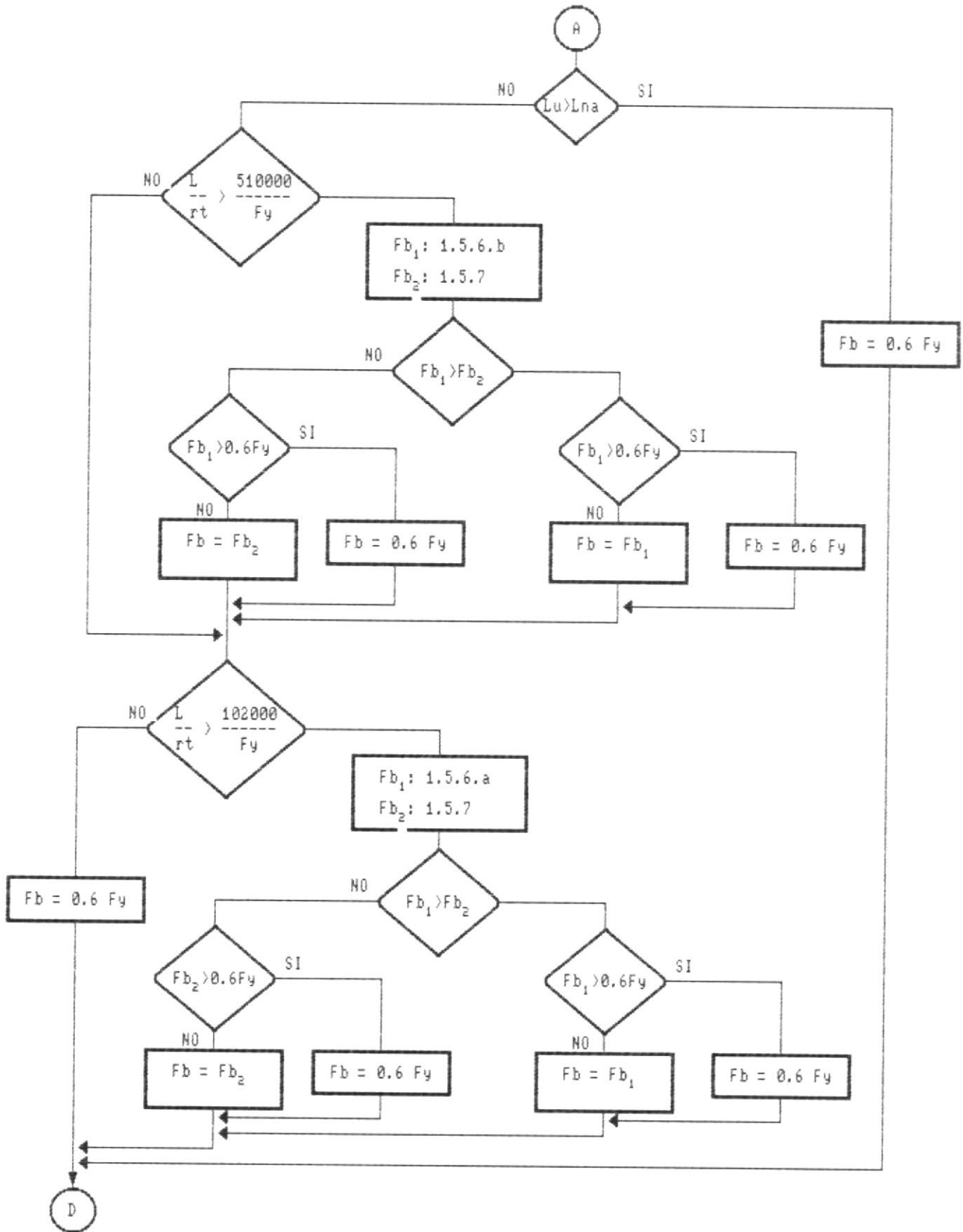
C

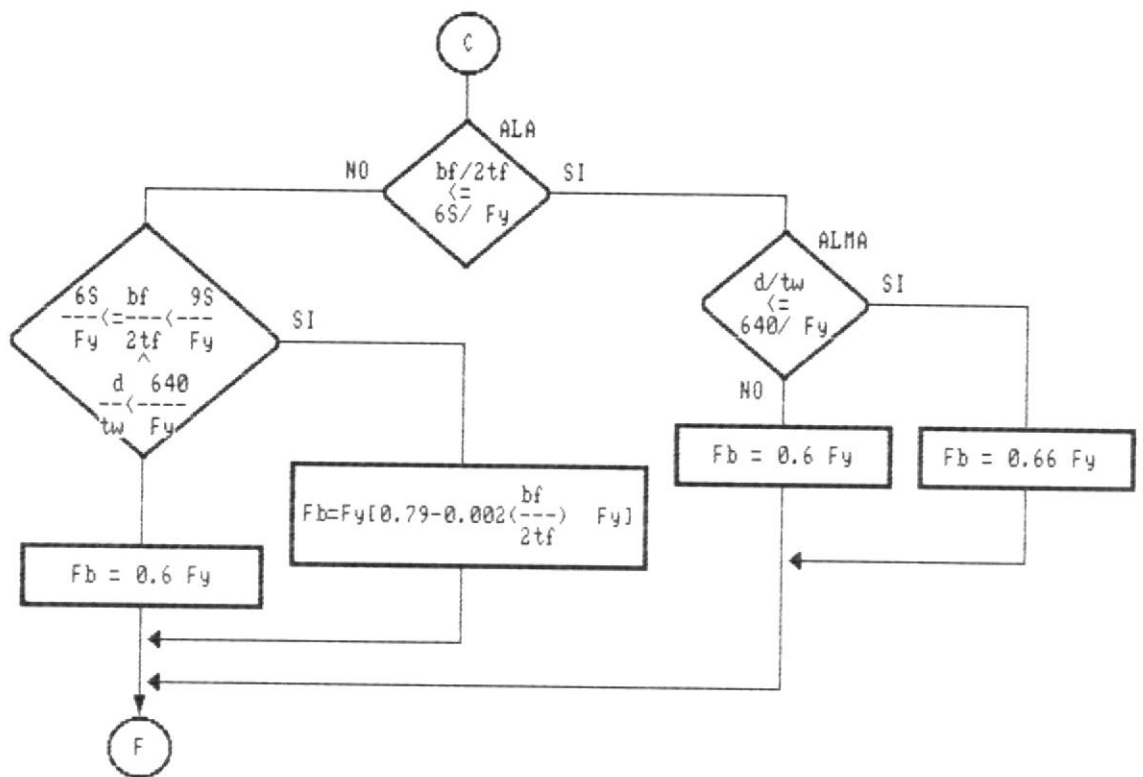
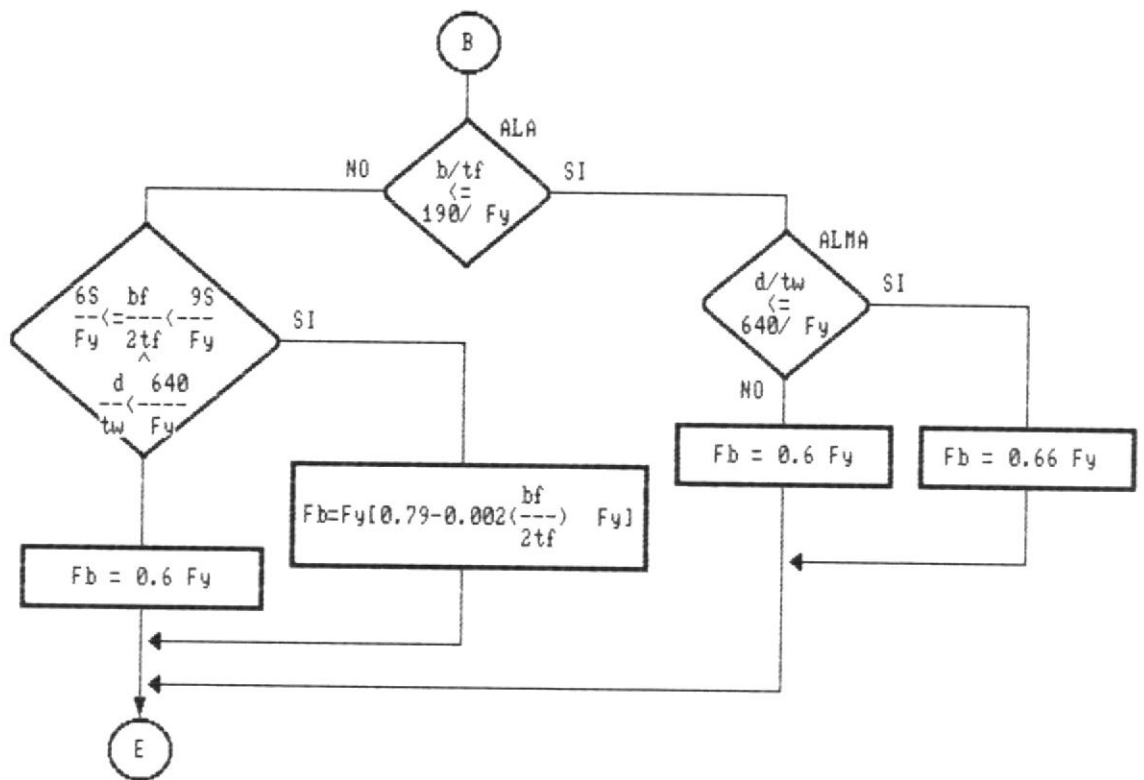
B

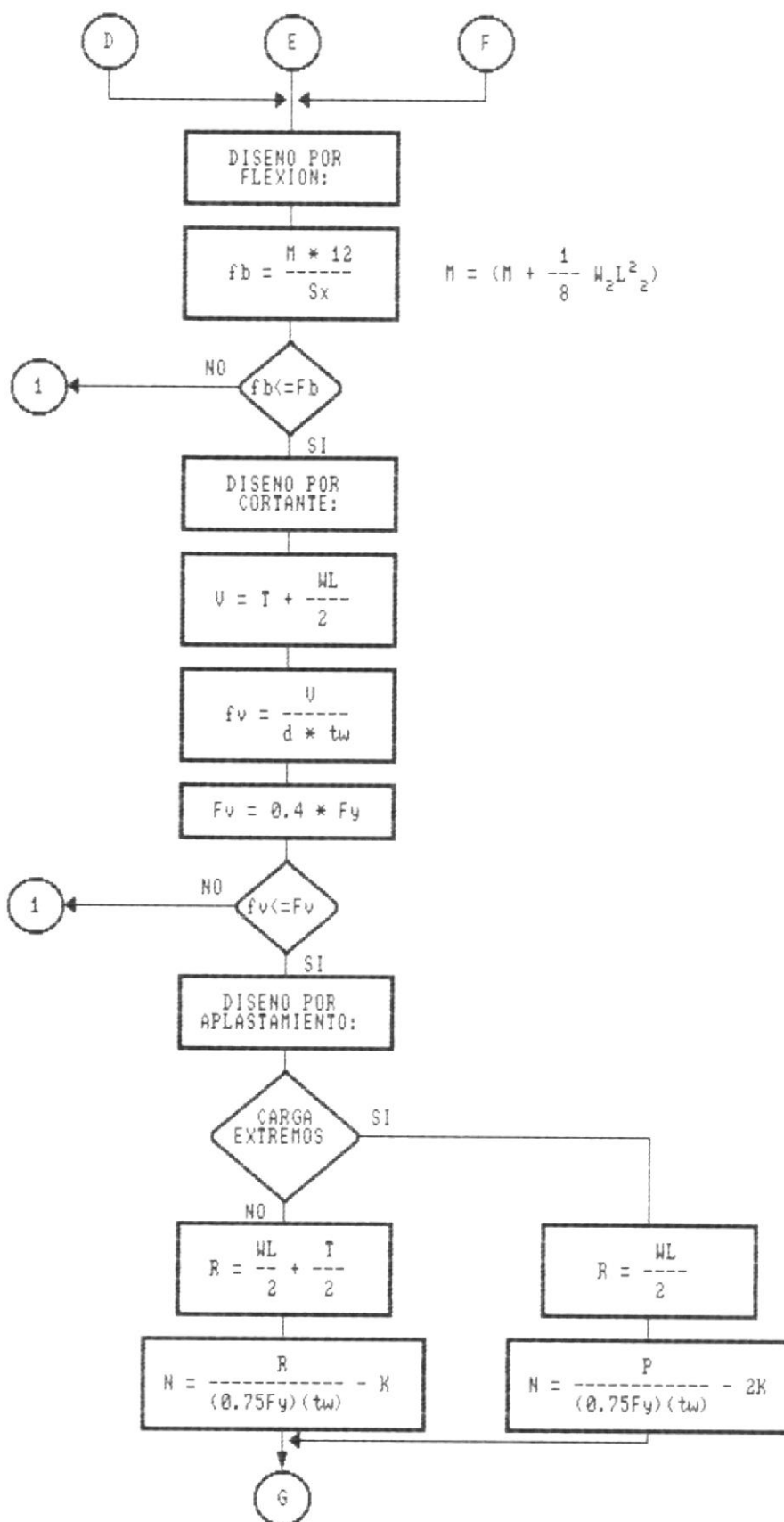


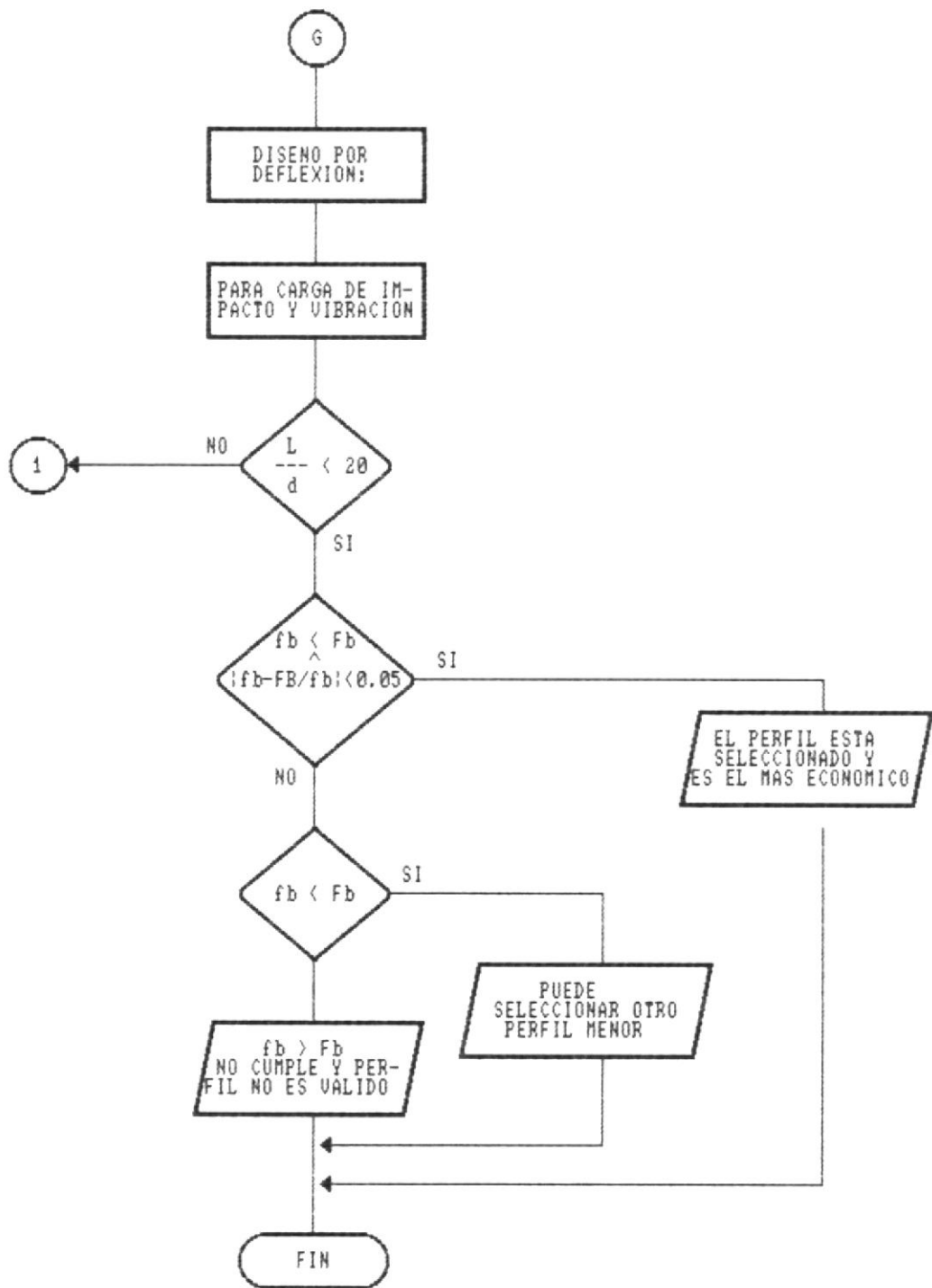
SUBROUTINA Lc,Lu











```

    /** SELECCION DEL PERFIL PARA EL CARRO DEL TROLE ***/

```

```

#include<stdio.h>
#include<math.h>
#include<ctype.h>
#include "tesis.h"
#define OR ||
#define AND &&
int salir=1;
void main()
{
char      C1,C2,C3[30],
          Fyc[20],Fuc[20],Ec[20],Tc[20],
          Fc[20],L1c[20],Lna1c[20],Ac[20],bfc[20],tfc[20],
          twc[20],dc[20],rtc[20],Sxc[20],Kc[20],wic[20];
double   T,P,T1,L1,w1,Lna1,S,d1,d2,N,N1,N2,M,Af,V,temp,
          A,bf,tf,tw,d,rt,K,Sx,Lu,Lu1,Lu2,E,M1,Fb1,
          Lc,Lc1,Lc2,R,Fv=0,fv=0,fb=0,Fb,Fu,Fy,Fb2;

clrscr();
do {
printf("\n Ingrese el esfuerzo de Fluencia Fy en Kips/pulgada
cuadrada: ");
gets(Fyc);
Fy = es_numero(Fyc);
fflush(stdin);
} while ( (Fy == 0) );
do {
printf("\n Ingrese la Resistencia ultima Fu en Kips/pulgada cuadrada:
");
gets(Fuc);
Fu = es_numero(Fuc);
fflush(stdin);
} while ( (Fu == 0) );
do {
printf("\n Ingrese el Modulo de Elasticidad E en Kips/pulg cuadrada:
");
gets(Ec);
E = es_numero(Ec);
fflush(stdin);
} while ( (E == 0) );
do {
printf("\n Ingrese la capacidad del elevador T en Kips: ");
gets(Tc);
T = es_numero(Tc);
fflush(stdin);
} while ( (T == 0) );
do {
printf("\n Ingrese el peso del motor y bobina P en Kips:
");
gets(Pc);
P = es_numero(Pc);
fflush(stdin);
} while ( (P == 0) );
do {

```

```

printf("\n ingrese la longitud del carro para el trole L1 en pies: ");
gets(L1c);
L1 = es_numero(L1c);
fflush(stdin);
} while ( (L1 == 0) );
do {
printf("\n Ingrese la longitud no arriostrada Lna1 en pies: ");
gets(Lna1c);
Lna1 = es_numero(Lna1c);
fflush(stdin);
} while ( (Lna1 == 0) );

T1 = ((T+P)/4.0);
M = ((T1*L1)/4.0);
printf("\n El valor del momento total es %lf\n",M);
Fb = (0.66*Fy);    /** ASUMO COMPACTA***/
S = ((M*12.0)/Fb);
do {
do {
do {
do {
if(fv <= Fv) {
if(fb <= Fb) {
printf("\n El modulo del perfil
requerido es %lf\n", S);
printf("\n Ingrese el perfil: ");
gets(C3);
fflush(stdin);
}
else {
clrscr();
printf("\n Falla por Flexion,ingrese un perfil
con un modulo mayor\n: ");
gets(C3);
fflush(stdin);
}
}
else {
clrscr();
printf("\n Falla por cortante,ingrese un perfil con
un modulo mayor\n: ");
gets(C3);
fflush(stdin);
}
}
do {
printf("\n Ingrese el area A en pulgada cuadrada:
");
gets(Ac);
A = es_numero(Ac);
fflush(stdin);
} while ( (A == 0) );
do {
printf("\n Ingrese el ancho del ala bf en pulgadas:
");

```

```

        gets(bfc);
        bf = es_numero(bfc);
        fflush(stdin);
    } while ( (bf == 0) );
do {
    printf("\n Ingrese el espesor del ala tf en
    pulgadas: ");
    gets(tfc);
    tf = es_numero(tfc);
    fflush(stdin);
} while ( (tf == 0) );
do {
    printf("\n Ingrese el espesor del alma tw en
    pulgadas: ");
    gets(twc);
    tw = es_numero(twc);
    fflush(stdin);
} while ( (tw == 0) );
do {
    printf("\n Ingrese la altura o peralte de la viga d
    en pulgadas: ");
    gets(dc);
    d = es_numero(dc);
    fflush(stdin);
} while ( (d == 0) );
do {
    printf("\n Ingrese el radio de giro rt en pulgadas:
    ");
    gets(rtc);
    rt = es_numero(rtc);
    fflush(stdin);
} while ( (rt == 0) );
do {
    printf("\n Ingrese el modulo de la seccion Sx en
    pulgadas cubicas: ");
    gets(Sxc);
    Sx = es_numero(Sxc);
    fflush(stdin);
} while ( (Sx == 0) );
do {
    printf("\n Ingrese la distancia entre el ala y el
    alma K en pulgadas,decimales: ");
    gets(Kc);
    K = es_numero(Kc);
    fflush(stdin);
} while ( (K == 0) );
do {
    printf("\n Ingrese el peso de la viga seleccionada
    w1 en Kips/pie: ");
    gets(w1c);
    w1 = es_numero(w1c);
    fflush(stdin);
} while ( (w1 == 0) );

Af = (bf*tf);

```

```

Lc1 = ( (352.0*0.22*bf)/(sqrt(Fy)*12.0) );
Lc2 = ( (3.0*E*0.22*bf*tf)/(Fy*d) )/12.0;
if( Lc1 > Lc2 )
    Lc = Lc2;
else
    Lc = Lc1;
Lu1 = ( (20000.0*Af)/(d*Fy*12.0) );
Lu2 = ( 0.22*bf*sqrt(102000.0/Fy) )/12.0;
if( Lu1 > Lu2 )
    Lu = Lu1;
else
    Lu = Lu2;
printf("\n Los valores de Lc y Lu son :\n");
printf("\t\t\t\t Lc = %1f\n",Lc);
printf("\t\t\t\t Lu = %1f\n",Lu);

if (Lna1 < Lc) {
    printf("\n Tiene alas rigidizadas ? S/N ==> ");
    do {
        C1=getchar();
        switch(C1) {
            case 'S' : 's' : {
                if((bf/tf) <= 190.0/sqrt(Fy)) {
                    if((d/tw) <= 640.0/sqrt(Fy))
                        Fb = (0.66*Fy);
                    else
                        Fb = (0.6*Fy);
                }
            }
            else {
                if( (((65.0/sqrt(Fy)) <= (bf/(2.0*tf))) &&
                    ((bf/(2.0*tf)) < (95.0/sqrt(Fy)))) ;: ((d/tw)
                    < (640.0/sqrt(Fy))))
                    Fb = Fy*(0.79-(0.002*(bf/(2.0*tf))*
                    sqrt(Fy)));
                else
                    Fb = (0.6*Fy);
            }
            salir=0;
            break;
        }
    }
    case 'N': 'n' : {
        if( (bf/2.0*tf) <= (65.0/sqrt(Fy)) ) {
            if((d/tw) < (640.0/sqrt(Fy)) )
                Fb = (0.66*Fy);
            else
                Fb = (0.6*Fy);
        }
        else {
            if( (((65.0/sqrt(Fy)) <= (bf/(2.0*tf)))
                && ((bf/(2.0*tf)) < (95.0/sqrt(Fy))))
                ;: ((d/tw) < (640.0/sqrt(Fy))))
                Fb = Fy*(0.79-(0.002*
                (bf/(2.0*tf))*sqrt(Fy)));
            else
                Fb = (0.6*Fy);
        }
    }
}

```

```

    }
    salir=0;
    break;
    }
    default : printf("\n Ingrese S/N ==> ");
    break;

    }
} while(salir);
}

else {
    if(Lu > Lna1)
        Fb = (0.66*Fy);
    else
        if( (L1*12.0/rt) > sqrt(510000.0/Fy) ) {
            F b 1    =    ( 1 7 0 . 0 * 1 0 0 0 . 0 ) /
            pow((L1*12.0/rt),2.0);
            Fb2 = (12.0*1000.0)/((12.0*d*L1)/Af);
            if(Fb1 > Fb2)
                if(Fb1 > (0.6*Fy))
                    Fb = (0.6*Fy);
                else
                    Fb = Fb1;
            else
                if(Fb2 > (0.6*Fy))
                    Fb = (0.6*Fy);
                else
                    Fb = Fb2;
        }
        if( ( L1*12.0/rt) > sqrt(102000.0/Fy) )
        {
            temp = pow((d*12.0/rt),2.0);
            Fb1    =    Fy*    ((2.0/3.0)-
            (Fy*temp)/(1530.0*1000.0));
            Fb2 = (12.0*1000.0)/((L1*12.0*d)/
            Af);
            if(Fb1 > Fb2)
                if(Fb1 > (0.6*Fy))
                    Fb = (0.6*Fy);
                else
                    Fb = Fb1;
            else
                if(Fb2 > (0.6*Fy))
                    Fb = (0.6*Fy);
                else
                    Fb = Fb2;
        }
        else
            Fb = (0.6*Fy);
    }
}

```

```

    /*** DISEÑO POR FLEXION ***/

    M1 = ((T1*L1)/4.0) + (w1*pow((L1),2.0)/8.0) ;
    fb = (M1*12.0)/Sx;
    } while (fb > Fb);

    /*** DISEÑO POR CORTANTE ***/

    V = (T1/2.0) + ((w1*L1)/2.0);
    fv = V/(d*tw);
    Fv = (0.4*Fy);
    } while (fv > Fv);

    /*** DISEÑO POR AFLASTAMIENTO ***/

    salir=1;
    printf("\n Tiene la viga carga interior ? S/N ==> ");
    do {
        C2 = getchar();
        switch (C2) {
            case 'S' ; 's' : {
                R = (T1/2.0) + ((w1*L1)/2.0);
                N1 = ( R/(0.75*Fy*tw))-K);
                N2 = fabs(N1);
                salir=0;
                break;
            }
            case 'N' ; 'n' : {
                R = ((w1*L1)/2.0);
                N1 = ( R/(0.75*Fy*tw))-(2.0*K));
                N2 = fabs(N1);
                salir=0;
                break;
            }
            default : printf("\n Ingrese S/N ==> ");
                break;
        }
    } while(salir);

    /*** DISEÑO POR DEFLEXION***/
    /*** PARA CARGA DE IMPACTO Y VIBRACIONES ***/

    d1 = ((L1*12.0)/d);
    d2 = 20.0;
    if(d1 <= d2)
        printf("\n El perfil satisface deflexion: ");
    else
        printf("\n Falla por deflexion,Ingrese un perfil mayor: ");
    } while (d1 > 20.0);
    clrscr();

```

```

if( (fb < Fb) && (fabs((fb-Fb)/fb) < 0.05) ) {
    printf("\n * EL PERFIL SELECCIONADO ES EL MAS ECONOMICO: \n");
    printf("\n * SE CUMPLE: ");
    printf("\n\t fb %lf <= Fb %lf", fb,Fb);
    printf("\n\t fv %lf <= Fv %lf", fv,Fv);
    printf("\n\t d1 %lf <= d2 %lf", d1,d2);
    printf("\n\n* LA LONGITUD MINIMA DE APOYO PARA QUE NO SE AFLASTE
    ES EN PULGADAS: %lf\n", N2);
    printf("\n\n* EL PERFIL SELECCIONADO ES: %s \n", C3);
    printf("\n [PRESIONE CUALQUIER TECLA PARA CONTINUAR] ");
    getch();
}
else {
    if(fb < Fb) {
        printf("\n * EL PERFIL CUMPLE PERO NO ES EL MAS
        ECONOMICO,UD PUEDE SELECCIONAR OTRO PERFIL CON UN MODULO
        Sx MENOR.\n ");
        printf("\n * SE CUMPLE: ");
        printf("\n\t fb = %lf ", fb);
        printf("\n\t Fb = %lf ", Fb);
        printf("\n\t\t Entonces : fb <= Fb");
        printf("\n\t fv = %lf ", fv);
        printf("\n\t Fv = %lf ", Fv);
        printf("\n\t\t Entonces : fv <= Fv");
        printf("\n\t d1 = %lf ", d1);
        printf("\n\t d2 = %lf ", d2);
        printf("\n\t\t Entonces : d1 <= d2");
        printf("\n\n* LA LONGITUD MINIMA DE APOYO PARA QUE NO SE
        AFLASTE EN PULGADAS ES :%lf", N2);
        printf("\n\n* EL PERFIL SELECCIONADO ES :%s \n", C3);
        gotoxy(18,23);printf("[Presione cualquier tecla para
        continuar]: ");
        getch();
    }
    else {
        printf("\n EL PERFIL NO SATISFACE: ");
        printf("\n [PRESIONE CUALQUIER TECLA PARA CONTINUAR] ");
        getch();
    }
}
} while ((fb > Fb) && (fv > Fv ));
}

```

```

    /**PROGRAMA PARA SELECCIONAR EL PERFIL***/
    /**DE LA VIGA PRINCIPAL***/

```

```

#include <stdio.h>
#include <math.h>
#include "tesis.h"
#include <ctype.h>
#define OR ||
#define AND &&
/** variables globales */
int salir=1;
/* funcion principal */

void main()
{
char    C1,C2,C3[30],
        Fyc[20],Fuc[20],Ec[20],Tc[20],Fc[20],wrc[20],
        Lna2c[20],Lic[20],L2c[20],wic[20],Ac[20],bfc[20],
        tfc[20],twc[20],dc[20],rtc[20],Sxc[20],Kc[20],
        w2c[20];

double  Fy,Fu,w1,E,L1,S,A,bf,tf,tw,d,rt,Sx,K,N2,wr,L2,a,
        Lc,Lu,M1,Af,T1,temp,Fb1,Fb2,Lc1,Lc2,Fb,Lna2,T2,
        Lu1,Lu2,N,R,V,Fv=0,fv=0,N1,W,M,fb=0,T,P,w2,d1,d2;

clrscr();
do {
printf("\n Ingrese el esfuerzo de fluencia Fy en Kips/pulgada cuadrada
: ");
gets(Fyc);
Fy = es_numero(Fyc);
fflush(stdin);
} while (( Fy == 0 ) );
do {
printf("\n Ingrese la resistencia ultima Fu en Kips/pulgada cuadrada
: ");
gets(Fuc);
Fu = es_numero(Fuc);
fflush(stdin);
} while ( ( Fu == 0 ) );
do {
printf("\n Ingrese el Modulo de Elasticidad E en Kips/pulg cuadrada :
");
gets(Ec);
E = es_numero(Ec);
fflush(stdin);
} while (( E == 0 ) );
do {
printf("\n Ingrese la capacidad del elevador T en Kips : ");
gets(Tc);
T = es_numero(Tc);
fflush(stdin);
} while ( ( T == 0 ) );
do {

```

```

printf("\n Ingrese el peso del motor y bobina P en Kips : ");
gets(Pc);
P = es_numero(Pc);
fflush(stdin);
} while (( P == 0 ) );
do {
printf("\n Ingrese el peso del riel wr en Kips/pie : ");
gets(wrc);
wr = es_numero(wrc);
fflush(stdin);
} while (( wr == 0 ) );
do {
printf("\n Ingrese la longitud entre apoyos Lna2 en pie : ");
gets(Lna2c);
Lna2 = es_numero(Lna2c);
fflush(stdin);
} while (( Lna2 == 0 ) );
do {
printf("\n Ingrese la longitud del carro del trole L1 en pie : ");
gets(L1c);
L1 = es_numero(L1c);
fflush(stdin);
} while (( L1 == 0 ) );
do {
printf("\n Ingrese la longitud de la viga principal L2 en pies : ");
gets(L2c);
L2 = es_numero(L2c);
fflush(stdin);
} while (( L2 == 0 ) );
do {
printf("\n Ingrese el peso de la viga para el carro del trole w1 en
Kips/pie : ");
gets(w1c);
w1 = es_numero(w1c);
fflush(stdin);
} while (( w1 == 0 ) );

T1 = (T+F)/4.0;
T2 = (T1 + (w1*L1));
a = fabs((L2-L1)/2.0);
M = (0.125*wr*pow((L2),2.0))+(T2*fabs((L2-L1)/2.0));
printf("\n El valor del momento total es %lf\n", M);
Fb = (0.66*Fy);    /**ASUMO COMPACTA***/
S = (M*12.0/Fb);
do{
    do {
        do {
            if(fv <= Fv) {
                if(fb <= Fb) {
                    printf("\n El modulo del perfil
requerido es %lf \n" , S);
                    printf("\n Ingrese un perfil :
");
                    gets(C3);

```

```

        fflush(stdin);
    }
    else {
        clrscr();
        printf("\n Falla por flexion, Ingrese otro perfil con un modulo mayor\n:");
        gets(C3);
        fflush(stdin);
    }
}
else {
    clrscr();
    printf("\n Falla por cortante, Ingrese otro perfil mayor\n :");
    gets(C3);
    fflush(stdin);
}
do {
    printf("\n Ingrese el área A en pulgada cuadrada : ");
    gets(Ac);
    A = es_numero(Ac);
    fflush(stdin);
} while (( A == 0) );
do {
    printf("\n Ingrese el ancho de ala bf en pulg: ");
    gets(bfc);
    bf = es_numero(bfc);
    fflush(stdin);
} while (( bf == 0 ) );
do {
    printf("\n Ingrese el espesor de ala tf en pulg : ");
    gets(tfc);
    tf = es_numero(tfc);
    fflush(stdin);
} while (( tf == 0 ) );
do {
    printf("\n Ingrese el espesor del alma tw en pulg: ");
    gets(twc);
    tw = es_numero(twc);
    fflush(stdin);
} while (( tw == 0 ) );
do {
    printf("\n Ingrese la altura o peralte de la viga d en pulg : ");
    gets(dc);
    d = es_numero(dc);
    fflush(stdin);
} while (( d == 0 ) );
do {
    printf("\n Ingrese el radio de giro rt en pulg

```

```

: ");
gets(rtc);
rt = es_numero(rtc);
fflush(stdin);
} while (( rt == 0 ) );
do {
printf("\n Ingrese el modulo de la seccion Sx
en pulgada cubica : ");
gets(Sxc);
Sx = es_numero(Sxc);
fflush(stdin);
} while (( Sx == 0 ) );
do {
printf("\n Ingrese la distancia entre el ala
y el alma K en pulg(decimales) : ");
gets(Kc);
K = es_numero(Kc);
fflush(stdin);
} while (( K == 0 ) );
do {
printf("\n ingrese el peso de la viga
seleccionada w2 en Kips/pie : ");
gets(w2c);
w2 = es_numero(w2c);
fflush(stdin);
} while(( w2 == 0 ) );

Af = (bf*tf);
Lc1 = ((76.0*bf)/(sqrt(Fy)*12.0));
Lc2 = ((20000.0*Af)/(d*Fy*12.0));
if (Lc1 > Lc2)
    Lc = Lc2 ;
else
    Lc = Lc1 ;
Lu1 = ((20000.0*Af)/(d*Fy*12.0)) ;
Lu2 = (rt*sqrt(102000.0/Fy))/12.0 ;
if(Lu1 > Lu2)
    Lu = Lu1 ;
else
    Lu = Lu2 ;
printf("\n Los valores de Lc y Lu
son %lf\n %lf\n",Lc ,Lu);
if(Lna2 < Lc) {
printf("\n Tiene alas rigidizadas
? S/N ==> ");
do{
C1=getch();
switch (C1) {

case 'S' ; 's' : {
if((bf/tf) <= 190.0/sqrt(Fy)) {
if(d/tw <= 640.0/sqrt(Fy))
    Fb = (0.66*Fy);
else
    Fb =(0.6*Fy);

```

```

    }
else {
    if( (((65.0/sqrt(Fy)) <=
        (bf/(2.0*tf))) && ((bf/(2.0*tf))
        < (95.0/sqrt(Fy)))) ;; ((d/tw) <
        (640.0/sqrt(Fy))))
        Fb = Fy*(0.79-(0.002*
            (bf/(2*tf))*sqrt(Fy)) );
    else
        Fb = (0.6*Fy);
    }

    salir=0;
    break;
}

case 'N':'n' : {
    if( (bf/(2.0*tf)) <= (65.0/sqrt(Fy)) )
    {
        if( d/tw <= 640.0/sqrt(Fy))
            Fb = (0.66*Fy);
        else
            Fb = (0.6*Fy);
    }
    else {
        if( (((65.0/sqrt(Fy)) <=
            (bf/(2.0*tf))) && ( (bf/(2.0*tf))
            < (95.0/sqrt(Fy)))) ;; ((d/tw) <
            (640.0/sqrt(Fy))) )
            Fb = Fy*(0.79-(0.002*
                (bf/(2.0*tf))*sqrt(Fy)));
        else
            Fb = (0.6*Fy);
    }
    salir=0;
    break;
}

default : printf("\n Ingrese S/N ==> ");
break;

}
} while(salir);
}
else {
    if (Lu > Lna2 )
        Fb = (0.66*Fy);
    else
        if( (L2*12.0/rt) > sqrt(510000.0/Fy) ) {
            Fb1=(170.0*1000.0)/pow((L2*12/rt),2) ;
            Fb2 = 12.0*1000.0/(12.0*d*L2/af);
            if(Fb1>Fb2)
                if(Fb1>0.6*Fy)
                    Fb=0.6*Fy ;
                else

```

```

        Fb=Fb1 ;
    else
        if(Fb2>0.6*Fy)
            Fb=0.6*Fy ;
        else
            Fb=Fb2;
        }

    if((L2*12.0/rt) > sqrt(102000.0/Fy) ) {
        temp = pow((d*12.0/rt),2);
        Fb1 = Fy*(2.0/3.0-(Fy*temp)/(1530.0*10
        00.0)) ;
        Fb2 = (12*1000/ (L2*12*d/af) );
        if( Fb1 > Fb2 )
            if(Fb1 > 0.6*Fy )
                Fb = 0.6*Fy;
            else
                Fb = Fb1;
        else
            if (Fb2 > 0.6*Fy)
                Fb = 0.6 * Fy;
            else
                Fb = Fb2;
        }
    else
        Fb = 0.6*Fy;
}

    ****DISEÑO POR FLEXION****/

M1 = ( ( (1.0/8.0)*w2*pow((L2),2) ) + (T2*a) +
((1.0/8.0)*wr*pow((L2),2)) );
fb = ( M1*12.0/Sx );

} while ( fb > Fb );

    ****DISEÑO POR CORTANTE****/

V = (T2 + ((wr*L2)/2.0) + (w2*L2)/2.0);
fv = V/(d*tw) ;
Fv = 0.4*Fy ;
} while ( fv > Fv );

    ****DISEÑO POR AFLASTAMIENTO****/

salir=1;
printf("\n Tiene la viga carga interior ? S/N ==> ");
do{
    C2 = getch();
    switch (C2) {

        case 'S';'s': {

```

```

        R = ( T2 + ((wr*L2)/2.0) + (w2*L2)/2.0);
        N1 = ( (R/(0.75*Fy*tw))-2.0*K);
        N2 = fabs(N1);
        salir=0;
        break;
    }

    case 'N';'n':{
        R = ((w2*L2)/2.0) + ((wr*L2)/2.0) ;
        N1 = ( (R/(0.75*Fy*tw))-K);
        N2 = fabs(N1);
        salir=0;
        break;
    }

    default : {
        printf("\n ingrese S/N ==> ");
        break;
    }
}
} while (salir);

        /***DISEÑO POR DEFLEXION***/
        /***PARA CARGA DE IMPACTO Y VIBRACION***/

d1 = (L2*12.0)/d;
d2 = 20.0;
if (d1 <= d2)
    printf("\n El perfil satisface por deflexion : ");
else
    printf("\n Falla por deflexion, Ingrese un perfil mayor : ");

} while ( d1 > 20.0 );
clrscr();

if( (fb < Fb) AND (fabs((fb-Fb)/fb) < 0.05) ) {
    printf("\n EL PERFIL SELECCIONADO ES EL MAS ECONOMICO : ")
    ;
    printf("\n SE CUMPLE: ");
    printf("\n fb  %1f <= Fb  %1f", fb, Fb);
    printf("\n fv  %1f <= Fv  %1f", fv, Fv);
    printf("\n d1 %1f <= d2  %1f", d1,d2);
    printf("\n LA LONGITUD MINIMA DE ASIEN TO PARA QUE NO SE
    APLASTE ES EN PULGADAS %1f\n",N2);
    printf("\n EL PERFIL SELECCIONADO ES %s\n",C3);
    printf("\n [PRESIONE CUALQUIER TECLA PARA CONTINUAR ] ");
    getch();
}
else {
    if(fb < Fb) {
        printf("\n EL PERFIL CUMPLE PERO NO ES EL MAS
        ECONOMICO, UD PUEDE SELECCIONAR UN PERFIL  CON UN
        MODULO Sx MENOR : ");
        printf("\n SE CUMPLE: ");
    }
}

```

```
printf("\n fb %1f <= Fb %1f", fb, Fb );
printf("\n fv %1f <= Fv %1f", fv, Fv );
printf("\n d1 %1f <= d2 %1f", d1, d2 );
printf("\n LA LONGITUD MINIMA DE ASIEN TO PARA QUE NO
SE AFLASTE ES EN PULGADAS %1f\n",N2);
printf("\n EL PERFIL SELECCIONADO ES %s\n",C3);
gotoxy(18,23);printf(" [Presione cualquier tecla
para continuar]: ");
getch();
}
else {
printf("\n EL PERFIL NO SATISFACE : ");
printf("\n [PRESIONE CUALQUIER TECLA PARA CONTINUAR
] ");
getch();
}
}
} while ( (fb >Fb) && ( fv > Fv));
}
```

2.6 LOS SOPORTES LATERALES.

2.6.1 ANALISIS DE DEFORMACIONES Y DETERMINACION DE MAXIMOS.

Los soportes laterales lo forman dos perfiles en C unidos mediante pernos en sus extremos; estos llevan las ruedas que permitirán el desplazamiento del puente-grúa.

Para el análisis también se considerarán como vigas homogéneas y simplemente apoyadas, y como la viga principal también está sometido a dos tipos de cargas:

- Dos cargas concentradas producto de las reacciones de las dos vigas que forman el puente.
- La carga uniformemente distribuida, que es el peso propio de la viga.

Para determinar la deformación elástica en el centro de la viga, aplicamos el segundo Teorema de Castigliano:

$$\Delta_{max} = \int_0^L \frac{M \left(\frac{\delta M}{\delta P} \right)}{EI} dx$$

$$\Delta_{MAX} = \left(\frac{T_2}{2}\right) \frac{a}{24EI} (4a^2 - 3L_3^2) + \frac{5}{384} W_3 L_3^4$$

Donde:

$$T_3 = T_2 \frac{W_R L_2}{2}$$

T_3 : Carga concentrada que actúa sobre el soporte lateral

L_2 : Longitud de la viga principal

L_3 : Longitud del soporte lateral

W_3 : Peso propio del soporte

Para impacto como en la viga principal:

$$\delta = 2\delta_{ST}$$

$$F = 2T$$

Para determinar los máximos, se lo hace del gráfico de cortante y momento flector, de donde:

$$R_1 = R_2 = \frac{T_3}{2} + \frac{W_3 L_3}{2}$$

Cortante máximo:

$$V_{MAX} = \frac{T_3}{2} + \frac{W_3 L_3}{2}$$

Momento flector máximo:

$$M_{\max} = \frac{1}{8} W_3 L^2 \bar{z} + \left(\frac{T_3}{2} \right) a$$

Esfuerzo máximo por flexión:

$$\sigma_{\max} = \frac{\frac{1}{8} W_3 L^2 \bar{z} + \left(\frac{T_3}{2} \right) a}{S}$$

$$\sigma_{\max} = \frac{V_{\max}}{bh^3} \left[(bf * tf) \left(\frac{b}{2} + \frac{tf}{2} \right) + (tw - \frac{b}{2} + \frac{tf}{2}) \left(\frac{b-tf}{2} \right) \right]$$

2.6.2 RECOMENDACIONES DEL MANUAL DEL A.I.S.C. PARA SELECCIONAR EL PERFIL MAS LIGERO.

Para diseñar los apoyos laterales debemos tener en cuenta lo siguiente algunas consideraciones:

- a.- Los apoyos son vigas de longitud pequeña y están sometidos a los mismos esfuerzos de la viga principal, por tanto deberán ser diseñados tomando en cuenta estos factores.
- b.- Al igual que en el caso de la viga principal, deberán ser diseñados como una viga sin apoyo lateral continuo tomando en cuenta el pandeo local de las alas en compresión.
- c.- El perfil puede ser una viga I, pero generalmente se utilizan otro tipo de perfiles, como la viga C, y que se acostumbra en vez de una a colocar dos en cada extremo.
- d.- También en este caso se diseñará para carga de impacto, que también tiene efecto sobre los apoyos laterales.
- e.- Para el diseño hay que considerar:
 - Momento flector
 - Fuerza cortante

- Deformación
- Aplastamiento
- Pandeo local

Las deformaciones del manual del A.I.S.C. son las mismas que para la viga principal.

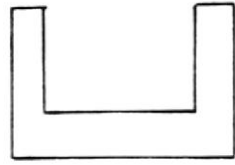
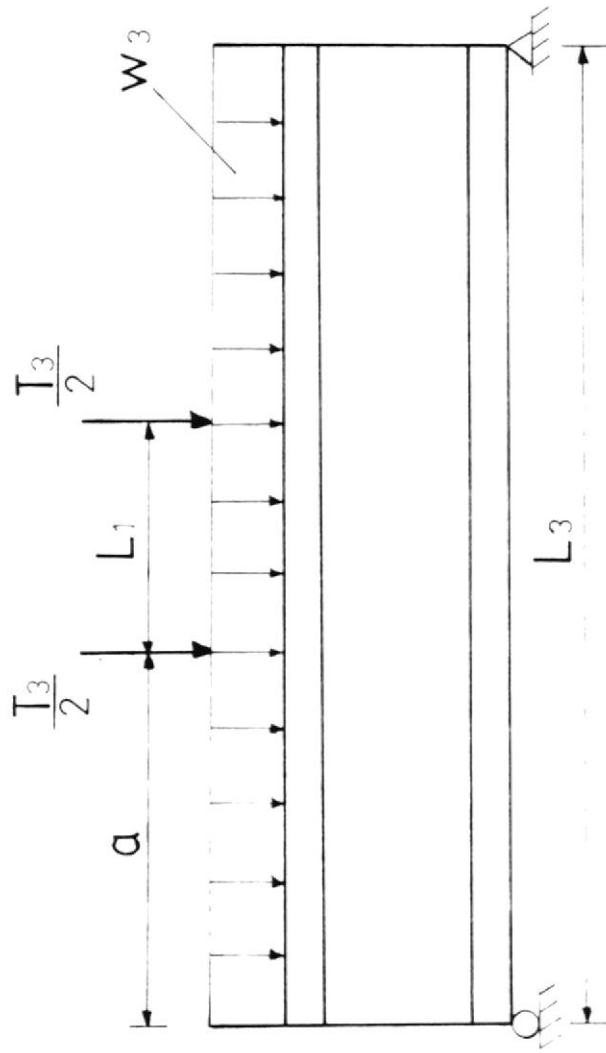
Longitud mínima de aplastamiento:

A.I.S.C. Pág 5.35.

$$\frac{R}{t(N + 2K)} \leq 0.75 F_y \quad 1.10-8$$

$$\frac{R}{t(N + K)} \leq 0.75 F_y \quad 1.10-9$$

FIGURA N° 8 : SOPORTE LATERAL SOMETIDO A FLEXION



2.6.3 DIAGRAMA DE FLUJO Y ALGORITMO

DISEÑO DE LOS SOPORTES LATERALES

$F_y; F_u; E, W_1, I$
 $T_2, T_1; W_2; Lna_1, L_2L_1$

$$T_3 = T_2 + \frac{W_2 L_2^2}{2}$$

$$M = \left(\frac{1}{8} W_3 L_3^2 \right) + \frac{T_3}{2} a$$

$$F_b = 0.66 * F_y$$

$$S = M / F_b$$

1

$A; b_f; t_f; t_w;$
 $d; r_t; S_x; K$

$$A_f = b_f * t_f$$

A

NO
PERFIL
W

SUB, L

B

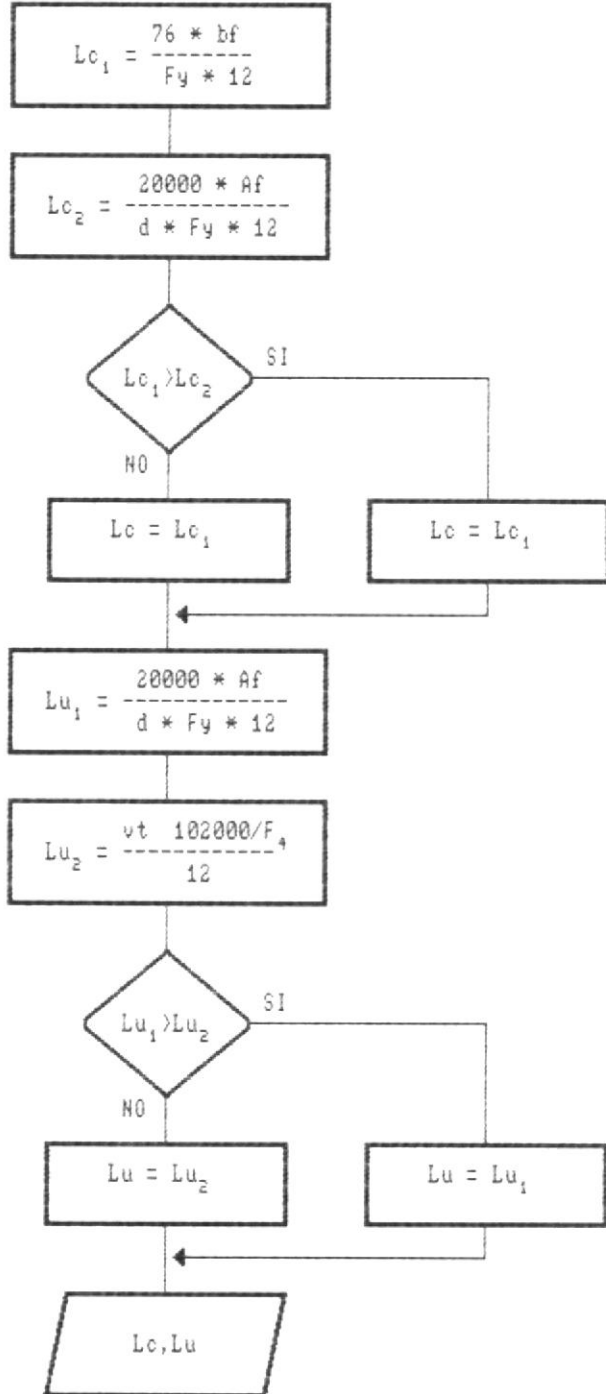
C

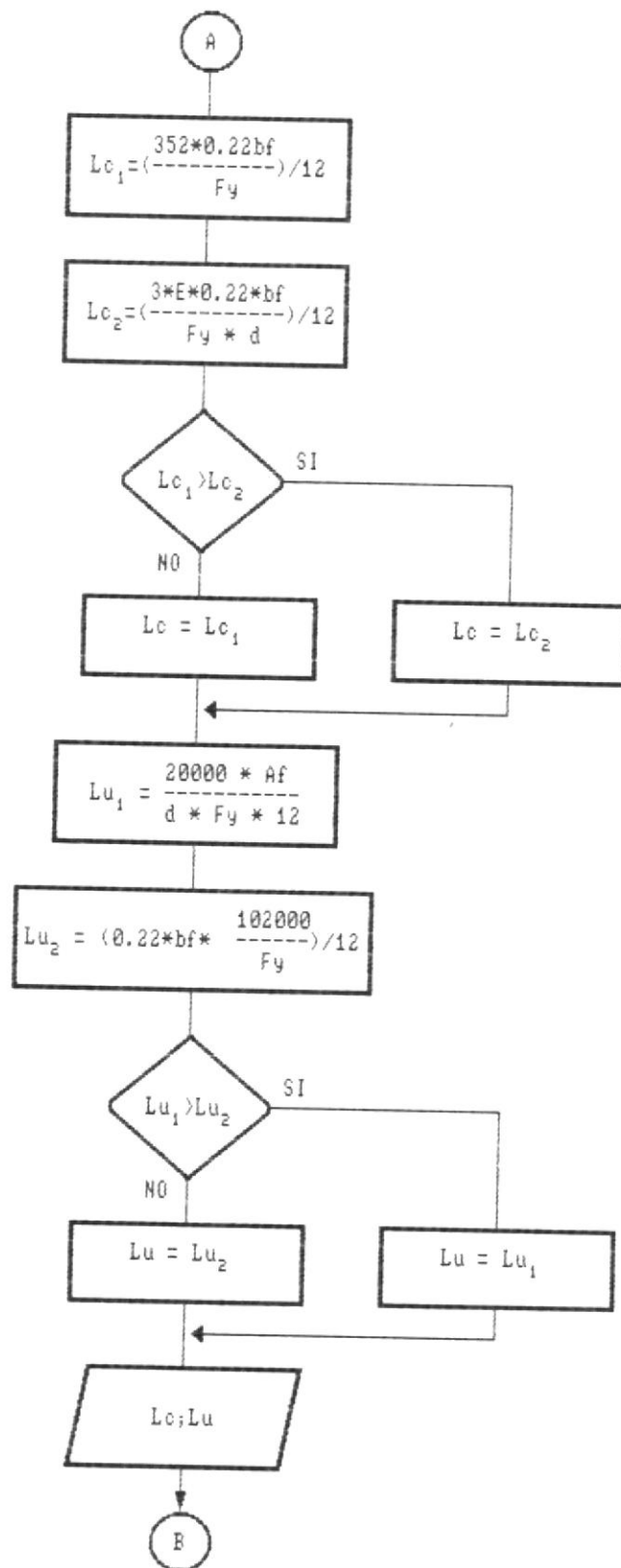
$Lna_2 \leq Lc$

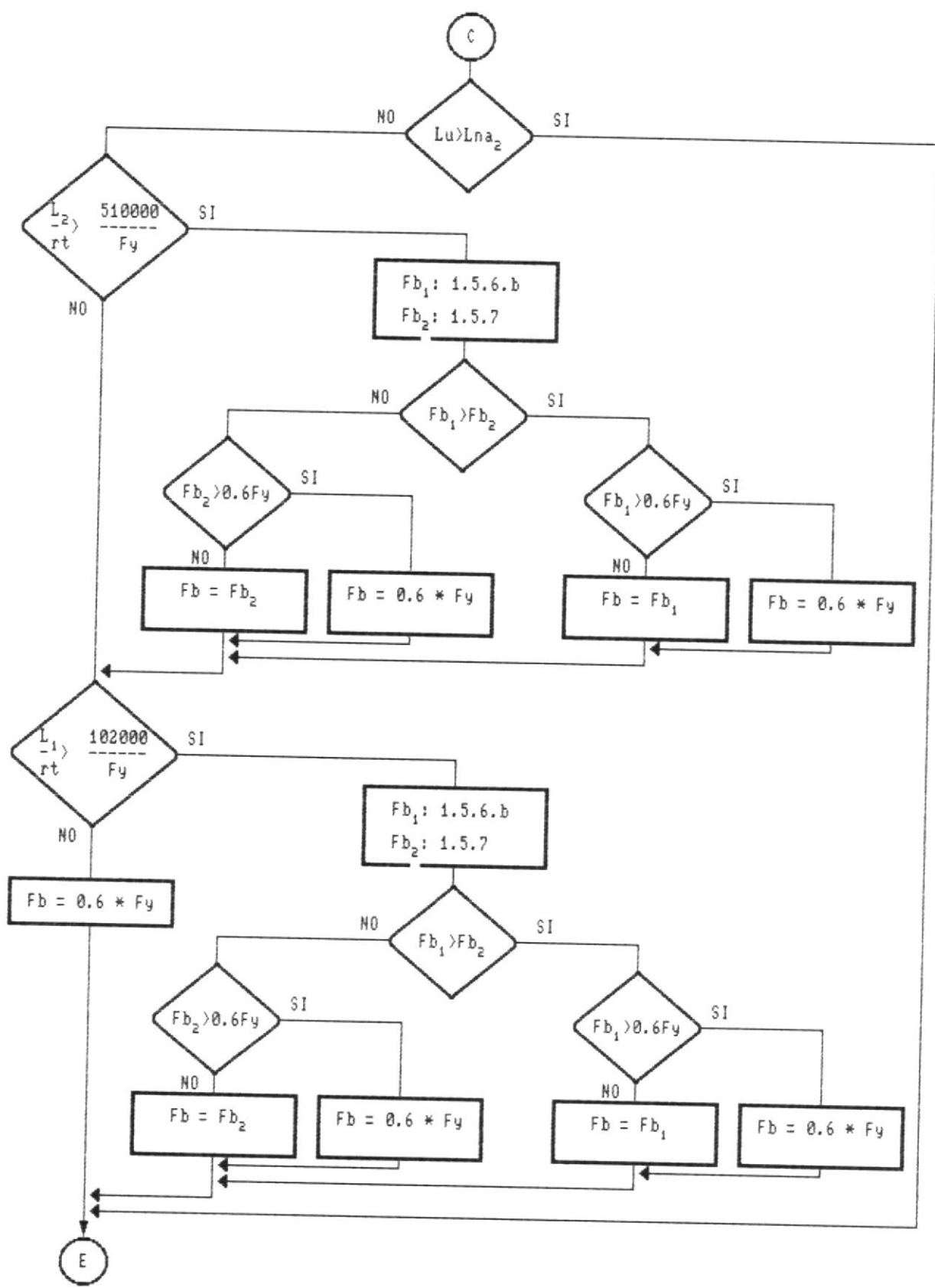
D

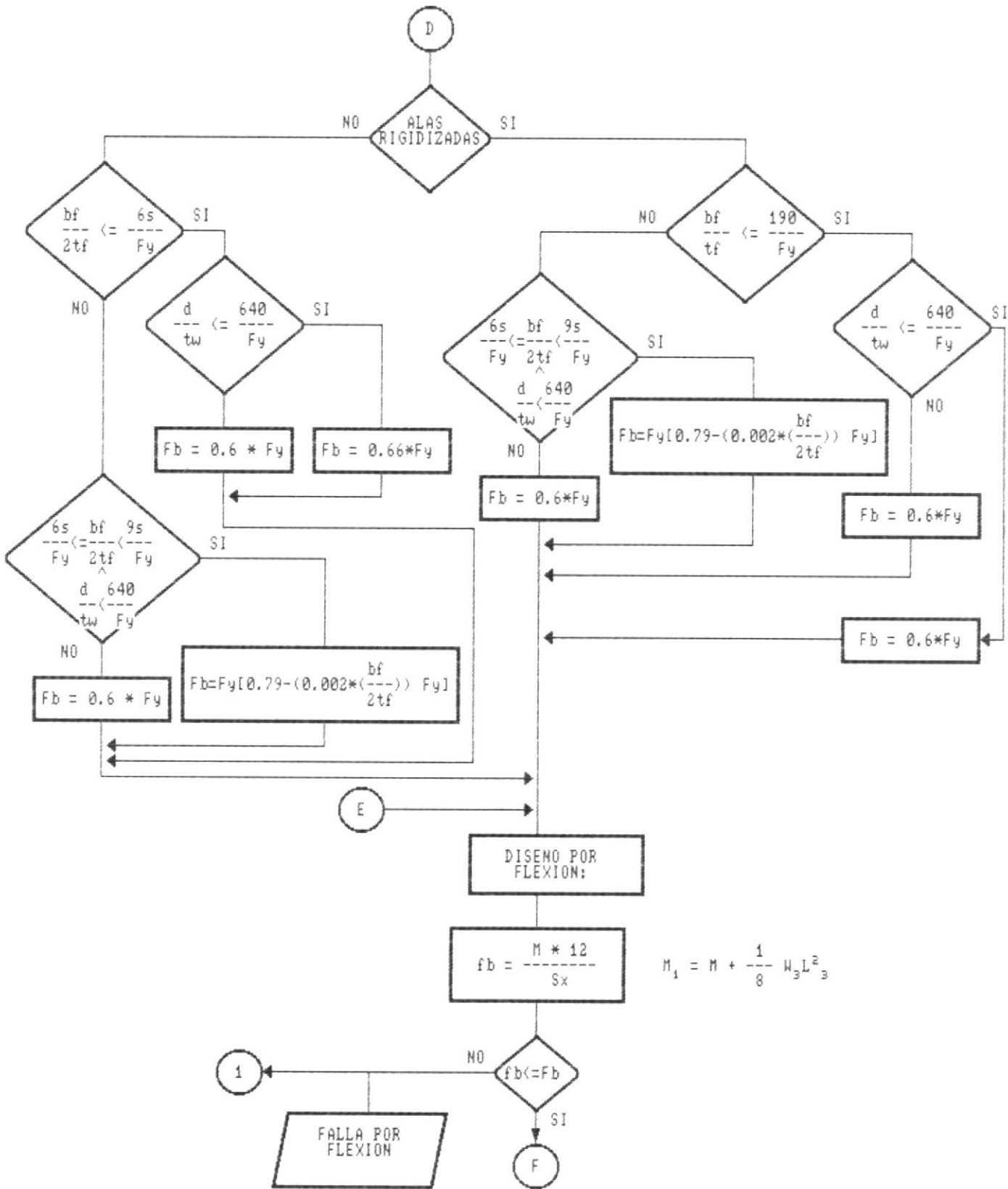


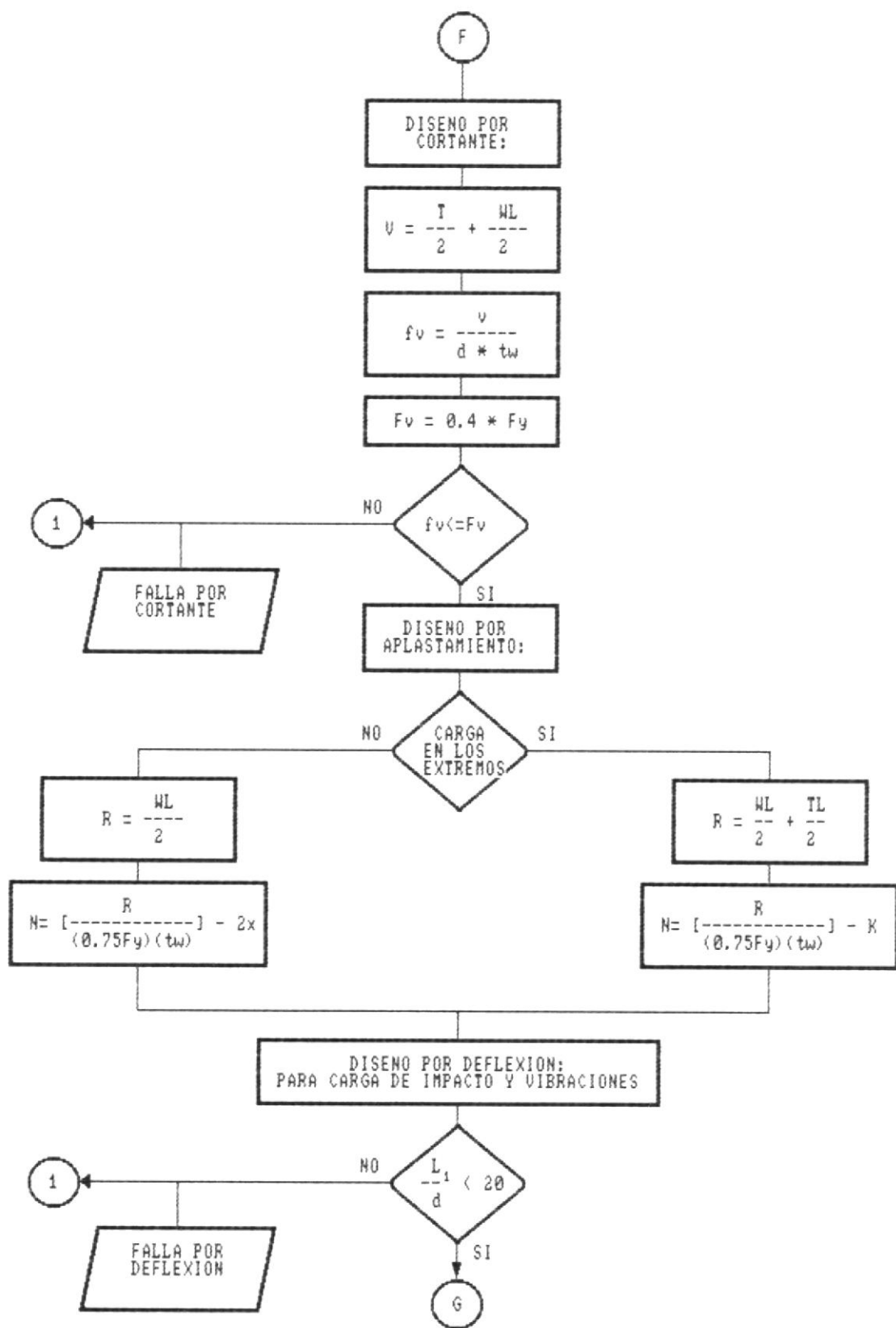
SUBROUTINA Lc,Lu

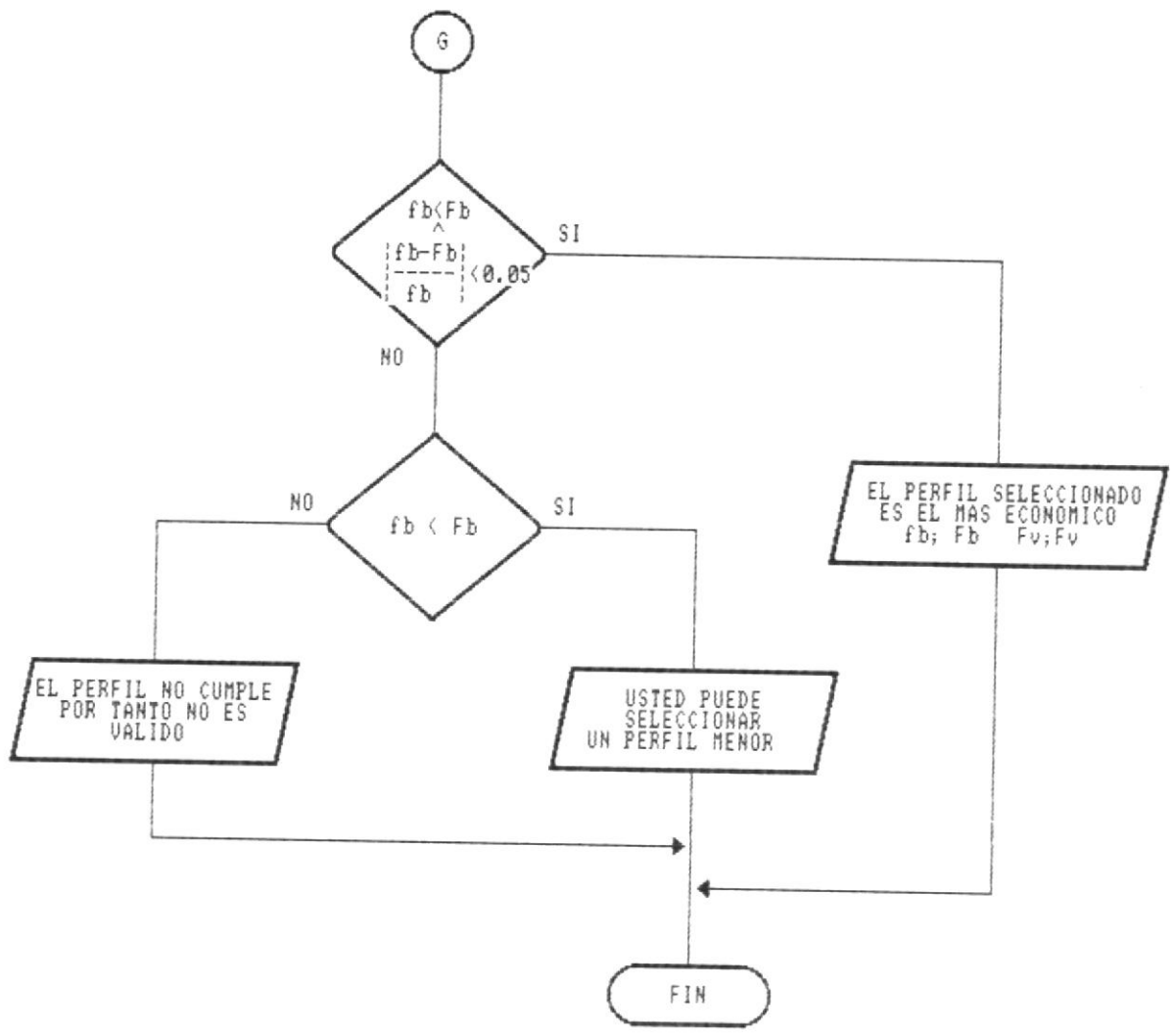












```

                /*** PROGRAMA PARA SELECCIONAR EL PERFIL ***/
                /*** DE LOS SOPORTES LATERALES ***/

#include <stdio.h>
#include <math.h>
#include <ctype.h>
#include "tesis.h"
#define OR  ||
#define AND  &&
/* variables globales */
int salir=1;
/* funcion principal */

void main()
{

char      C1,C2,C3,C4[30],

Fyc[20],Ec[20],Tc[20],wrc[20],wic[20],w2c[20],L1c[20],

L2c[20],L3c[20],Lna3c[20],Pc[20],Ac[20],bfc[20],tfc[20],
twc[20],Kc[20],Sxc[20],rtc[20],w3c[20],dc[20];
double    Fy,T,w1,w2,L1,L2,E,A,bf,tf,tw,d,rt,Sx,K,M1,S,T2,W,M,Af,

N,N1,N2,Lc,Lc1,Lc2,Lu,Lu1,Lu2,R,V,Fb,Fb1,Fb2,fb=0,Fv=0,fv=0,temp,
d1,d2,wr,L3,Lna3,a,w3,Tt,P,T1,T3;

clrscr();
do{
printf("\n Ingrese el esfuerzo de fluencia Fy en Kips/pulgada
cuadrada: ");
gets(Fyc);
Fy = es_numero(Fyc);
fflush(stdin);
} while (( Fy == 0 ) );
do{
printf("\n Ingrese el Modulo de Elasticidad E en Lb/pulgada cuadrada:
");
gets(Ec);
E = es_numero(Ec);
fflush(stdin);
} while ( (E == 0) );
do {
printf("\n Ingrese la capacidad del elevador T en Kips: " );
gets(Tc);
T = es_numero(Tc);
fflush(stdin);
} while ( (T == 0) );
do {
printf("\n Ingrese el peso del riel wr en Kips/pie: ");
gets(wrc);
wr = es_numero(wrc);
fflush(stdin);
} while ( (wr == 0) );
do{

```

```

printf("\n Ingrese el peso de la viga para el carro del trole w1 en
Kips/pie: ");
gets(w1c);
w1 = es_numero(w1c);
fflush(stdin);
} while ( (w1 == 0) );
do {
printf("\n Ingrese el peso de la viga principal w2 en Kips/pie: ");
gets(w2c);
w2 = es_numero(w2c);
fflush(stdin);
} while ( (w2 == 0) );
do{
printf("\n Ingrese la longitud de la viga para el carro del trole L1
en pie: ");
gets(L1c);
L1 = es_numero(L1c);
fflush(stdin);
} while ( (L1 == 0) );
do{
printf("\n Ingrese la longitud de la viga principal L2 en pie: ");
gets(L2c);
L2 = es_numero(L2c);
fflush(stdin);
} while ( (L2 == 0) );
do{
printf("\n Ingrese la longitud del soporte lateral L3 en pie: ");
gets(L3c);
L3 = es_numero(L3c);
fflush(stdin);
} while ( (L3 == 0) );
do{
printf("\n Ingrese la longitud entre apoyos del sopotrte lateral Lna3
en pie: ");
gets(Lna3c);
Lna3 = es_numero(Lna3c);
fflush(stdin);
} while ( (Lna3 == 0) );
do {
printf("\n Ingrese el peso del motor y bobina F en Kips: ");
gets(Fc);
F = es_numero(Fc);
fflush(stdin);
} while ( (F == 0) );

a = ((L3-L1)/2.0);
T1 = ((T+F)/4.0);
T2 = (T1+(w1*L1));
T3 = (T2+((wr*L2)/2.0));
Tt = (T3/2.0);
M = ( (1.0/8.0)*wr*pow((L3),2) + (Tt*a) );
printf("\n El valor del momento total es %lf\n", M) ;
Fb = 0.66*Fy; /* ASUMO COMPACTA*/
S =( M*12.0/Fb );
do {

```

```

do {
    do {
        do {
            if(fv <= Fv) {
                if(fb <= Fb ) {
                    printf("\n El modulo del perfil
                    requerido es %lf\n", S);
                    printf("\n Ingrese un perfil: ");
                    gets (C4);
                    fflush(stdin);
                }
                else {
                    clrscr();
                    printf("\n Falla por Flexion
                    ,Ingrese otro perfil con un
                    modulo mayor\n: ");
                    gets(C4);
                    fflush(stdin);
                }
            }
            else {
                clrscr();
                printf("\n Falla por cortante, Ingrese
                otro perfil mayor \n: ");
                gets(C4);
                fflush(stdin);
            }
        }
        do {
            printf("\n Ingrese el area A en pulgada
            cuadrada: ");
            gets(Ac);
            A = es_numero(Ac);
            fflush(stdin);
        } while ( (A == 0) );
        do {
            printf("\n Ingrese el ancho de ala bf en
            pulgada: ");
            gets(bfc);
            bf = es_numero(bfc);
            fflush(stdin);
        } while ( (bf == 0) );
        do {
            printf("\n Ingrese el espesor de ala tf en
            pulgada: ");
            gets(tfc);
            tf = es_numero(tfc);
            fflush(stdin);
        } while ( (tf == 0) );
        do {
            printf("\n Ingrese el espesor del alma tw en
            pulgada: ");
            gets(twc);
            tw = es_numero(twc);
            fflush(stdin);
        } while ( (tw == 0) );

```

```

do {
printf("\n Ingrese la altura o peralte de la
viga d en pulgada: ");
gets(dc);
d = es_numero(dc);
fflush(stdin);
} while ( (d == 0) );
do {
printf("\n Ingrese el radio de giro rt en
pulgada: ");
gets(rtc);
rt = es_numero(rtc);
fflush(stdin);
} while ( (rt == 0) );
do {
printf("\n Ingrese el modulo de la seccion Sx
en pulgada cubica: ");
gets(Sxc);
Sx = es_numero(Sxc);
fflush(stdin);
} while ( (Sx == 0) );
do {
printf("\n Ingrese la distancia entre el ala
y el alma K en pulgada,decimales: ");
gets(Kc);
K = es_numero(Kc);
fflush(stdin);
} while ( (K == 0) );
do {
printf("\n Ingrese el peso de la viga
seleccionada w3 en Kips/pie: ");
gets(w3c);
w3 = es_numero(w3c);
fflush(stdin);
} while ( (w3 == 0) );

Af = (bf*tf) ;
printf("\n Perfil W ? S/N ==> ");
do {
C1 = getch();
switch (C1) {
case 'S':'s': {
Lc1 = ( (76.0*bf)/(sqrt(Fy)*12.0) );
Lc2 = ( (20000.0*Af)/(d*Fy*12.0) );
if( Lc1 > Lc2 )
Lc = Lc2 ;
else
Lc = Lc1 ;
Lu1 = ( (20000.0*Af)/(d*Fy*12.0) );
Lu2 = rt*(sqrt(102000.0/Fy))/12.0 ;
if( Lu1 > Lu2 )
Lu = Lu1 ;
else
Lu = Lu2 ;
salir = 0;
}
}
}

```

```

        break;
    }
    case 'N': {
        Lc1 = ( (352.0*0.22*bf)/(sqrt(Fy)*12.0) );
        Lc2 = ( (3.0*E*0.22*bf*tf)/(Fy*d) )/12.0;
        if( Lc1 > Lc2 )
            Lc = Lc2 ;
        else
            Lc = Lc1 ;
        Lu1 = ( (20000.0*Af)/(d*Fy*12.0) );
        Lu2 = (0.22*bf*sqrt(102000.0/Fy))/12.0;
        if( Lu1 > Lu2 )
            Lu = Lu1 ;
        else
            Lu = Lu2 ;
        salir = 0;
        break;
    }
    default : {
        printf("\n Ingrese S/N ==> ");
        break;
    }
}
} while (salir);
printf("\n Los valores de Lc y Lu son : %lf\n",Lc,Lu);
salir = 1;
if( Lna3 <= Lc ) {
    printf("\n Tiene alas rigidizadas ? S/N ==> ");
    do{
        C2 = getch();
        switch (C2) {
            case 'S': {
                if( bf/tf <= 196.0/sqrt(Fy) ) {
                    if( d/tw <= 640.0/sqrt(Fy) )
                        Fb = 0.06*Fy ;
                    else
                        Fb = 0.6*Fy ;
                }
            }
            else {
                if((( (65.0/sqrt(Fy)) <= (bf/(2.0*tf))) &&
                    ((bf/(2.0*tf)) < (95.0/sqrt(Fy))) ) || (
                    (d/tw) < (640.0/sqrt(Fy))) )
                    Fb = Fy*(0.79-(0.002*(bf/(2.0*tf))*
                    sqrt(Fy))) ;
                else
                    Fb = 0.6*Fy ;
            }
        }
        salir = 0;
        break;
    }
}
case 'N': {

```

```

if( 65.0 <= bf/(2*tf) ) {
    if( d/tw <= 640.0/sqrt(Fy) )
        Fb = 0.66*Fy ;
    else
        Fb = 0.6*Fy ;
}
else {
    if((( (65.0/sqrt(Fy)) <= (bf/(2.0*tf)))
    && ((bf/(2.0*tf)) < (95.0/sqrt(Fy))) )
    || ( (d/tw) < (640.0/sqrt(Fy)) ) )
        Fb = Fy*(0.79-(0.002*
        (bf/(2.0*tf))*sqrt(Fy))) ;
    else
        Fb = 0.6*Fy ;
}
salir = 0;
break;
}
default :{
    printf("\n Ingrese S/N ==> ");
    break;
}
}
} while(salir);
}
else {
    if( Lu > Lna3 )
        Fb = 0.6*Fy ;
    else
        if( ((L3*12.0)/rt) > sqrt(510000.0/Fy)) {
            Fb1 = ((170.0*1000.0)/(L3*12.0/rt));
            Fb2 = ( (12.0*1000.0)/(L3*12.0*d/Af));
            if( Fb1 > Fb2 )
                if( Fb1 > 0.6*Fy )
                    Fb = 0.6*Fy ;
                else
                    Fb = Fb1 ;
                    if (Fb2 > 0.6*Fy )
                        Fb = 0.6*Fy ;
                    else
                        Fb = Fb2 ;
        }

        if( ((L3*12.0)/rt) > sqrt(102000.0/Fy)) {
            temp = pow((d*12.0/rt),2);
            Fb1 = Fy*(2.0/3.0-(Fy*temp)/
            (1530.0*1000.0)) ;
            Fb2 = (12.0*1000.0/(L3*12.0*d/Af));
            if( Fb1 > Fb2 )
                if( Fb1 > 0.6*Fy )
                    Fb = 0.6*Fy ;
                else
                    Fb = Fb1 ;
            else
                if(Fb2 > 0.6*Fy)

```

```

                Fb = (0.6*Fy) ;
            else
                fb = Fb2 ;
        }
    else
        Fb = 0.6*Fy ;
    }

    /*** DISEÑO POR FLEXION ***/

M1 = ( ((1.0/8.0)*wr*pow((L2),2)) + ((1.0/8.0)*w3*pow((L3),2)) +
(Tt*a) );
fb = (M1*12.0/Sx) ;
} while ( fb > Fb );

    /*** DISEÑO POR CORTANTE ***/

V = ( ((w3*L3)/2.0) + ((wr*L3)/2.0) + Tt );
fv = V/(d*tw) ;
Fv = 0.4*Fy ;
} while ( fv > Fv ) ;

    /*** DISEÑO POR APLASTAMIENTO ***/

salir = 1;
printf("\n Tiene la viga carga interior ? S/N ==> ");
do {
C3 = getch();
switch (C3) {
case 'S' : 's' : {
R = ( ((w3*L3)/2.0) + ((wr*L3)/2.0) + Tt );
N1 = ( (R/(0.75*Fy*tw))-2.0*K) ;
N2 = fabs(N1) ;
salir = 0;
break;
}
case 'N' : 'n' : {
R = ((wr*L3)/2.0) + ((w3*L3)/2.0);
N1 = ( (R/(0.75*Fy*tw))-K) ;
N2 = fabs(N1) ;
salir = 0;
break;
}
default : {
printf("\n Ingrese S/N ==> ");
break;
}
}
} while(salir);

```

```

                /*** DISEÑO POR DEFLEXION***/
                /*** PARA CARGA DE IMPACTO Y VIBRACION ***/

d1 = (L3*12.0)/d;
d2 = 20.0;
if( d1 <= d2 )
    printf("\n EL PERFIL SATISFACE DEFLEXION: ");
else
    printf("\n Falla por deflexion: ");
} while ( d1 > 20.0 );
clrscr();

if( (fb < Fb) AND ((fabs(fb-Fb)/fb) < 0.05) ) {
    printf("\n EL PERFIL SELECCIONADO ES EL MAS ECONOMICO: ");
    printf("\n SE CUMPLE: ");
    printf("\n fb %1f <= Fb %1f ",fb,Fb);
    printf("\n fv %1f <= Fv %1f ",fv,Fv);
    printf("\n d1 %1f <= d2 %1f ",d1,d2);
    printf("\n LA LONGITUD MINIMA DE ASIEN TO PARA QUE NO SE
    APLASTE ES: %1f\n",N2);
    printf("\n EL PERFIL SELECCIONADO ES: %s \n",C4);
    printf("\n [ PRESIONE CUALQUIER TECLA PARA CONTINUAR] " );
    getch();
}
else {
    if (fb < Fb ) {
        printf("\n EL PERFIL CUMPLE PERO NO ES EL MAS
        ECONOMICO, UD PUEDE ELEGIR UN PERFLI MENOR: " );
        printf("\n SE CUMPLE: ");
        printf("\n fb %1f <= Fb %1f ",fb,Fb);
        printf("\n fv %1f <= Fv %1f ",fv,Fv);
        printf("\n d1 %1f <= d2 %1f ",d1,d2);
        printf("\n LA LONGITUD MINIMA DE ASIEN TO PARA QUE NO
        SE APLASTE ES: %1f\n",N2);
        printf("\n EL PERFIL SELECCIONADO ES: %s \n",C4);
        gotoxy(18,23);printf(" [Presione cualquier tecla
        para continuar:]");
        getch();
    }
    else {
        printf("\n EL PERFIL NO CUMPLE Y NO ES VALIDO\n ");
        printf("\n [PRESIONE CUALQUIER TECLA PARA
        CONTINUAR]");
        getch();
    }
}
} while ( (fb > Fb) AND (fv > Fv)) ;
}

```

2.7 LA VIGA DE ASIENTO PARA EL RIEL Y SUS PLACAS DE APOYO.

2.7.1 ANALISIS DE LA DEFORMACION Y DETERMINACION DE REACCIONES Y MOMENTOS.

En el caso de las guías el análisis es muy diferente, ya que se trata de una viga estáticamente indeterminada, y por tanto su tratamiento es diferente.

Las estructuras indeterminadas se estudian basándose en la teoría de deformaciones elásticas. Podemos convertir una estructura hiperestática, cualquiera que esta sea, en una estáticamente determinada; esto lo conseguimos al suprimir las acciones redundantes o sobrantes.

Aplicando el principio de superposición, la deformación elástica bajo el efecto del peso propio en cada apoyo del riel será (6).

Primer apoyo: $\Delta_2 = \Delta_{2w_1} + S_{22}R_2 + \dots + S_{29}R_9$

Segundo apoyo: $\Delta_9 = \Delta_{9w_1} + S_{92}R_2 + \dots + S_{99}R_9$

Luego para determinar las reacciones, se pone estas ecuaciones en forma matricial:

(6) Teoría Elemental de Estructuras. Yuan-Yu Ksiek, Pág. 58.

$$\begin{bmatrix} R_1 \\ R_2 \\ \cdot \\ \cdot \\ R_m \end{bmatrix} = \begin{bmatrix} - \Delta_{1W1} \\ - \Delta_{2W1} \\ \cdot \\ \cdot \\ - \Delta_{mW1} \end{bmatrix} * \begin{bmatrix} S_{11} & \cdot & \cdot & S_{1n} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ S_{m1} & \cdot & \cdot & S_{mn} \end{bmatrix}^{-1}$$

De igual forma, para el caso del riel sometido bajo el efecto de una carga concentrada aplicada gradualmente, tenemos:

$$\begin{bmatrix} R_1 \\ R_2 \\ \cdot \\ \cdot \\ R_m \end{bmatrix} = \begin{bmatrix} - \Delta_{1T} \\ - \Delta_{2T} \\ \cdot \\ \cdot \\ - \Delta_{mT} \end{bmatrix} * \begin{bmatrix} S_{11} & \cdot & \cdot & S_{1n} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ S_{m1} & \cdot & \cdot & S_{mn} \end{bmatrix}^{-1}$$

Para el caso del riel sometido bajo el efecto de una carga concentrada aplicada bruscamente (carga dinámica o de impacto), tenemos (7):

$$\begin{bmatrix} R_1 \\ \cdot \\ \cdot \\ \cdot \\ R_m \end{bmatrix} = \begin{bmatrix} - \Delta_{1Timp} \\ \cdot \\ \cdot \\ \cdot \\ - \Delta_{mTimp} \end{bmatrix} * \begin{bmatrix} S_{imp11} & \cdot & \cdot & S_{imp1n} \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ S_{impm1} & \cdot & \cdot & S_{impmn} \end{bmatrix}^{-1}$$

(7) Resistencia de materiales. Singer 1982, Pág. 436.

En las estructuras estáticamente indeterminadas o hiperestáticas, ocurren pequeños desplazamientos de los apoyos, cuyo resultado es un cambio completo en el diagrama de momentos flectores. El análisis se realiza en base a los valores relativos de los asentamientos, esto para cuando todos los apoyos o una parte de ellos ceden.

Si los apoyos ceden diferentes cantidades, pero permanecen en una línea recta, los esfuerzos no cambian; ahora si todos los apoyos ceden un mismo valor, la viga no sufre cambio.

Si los apoyos ceden cantidades desiguales y tampoco están en líneas rectas, las condiciones de los esfuerzos cambian, debido a que la viga se deforma.

Para nuestro caso, vamos a asumir que los apoyos no se asentarán, y de este modo la viga no se deformará. Además en los apoyos se colocarán placas base de acero para distribuir la resistencia de la viga.

La razón principal para calcular las reacciones en una viga continua es la de trazar el diagrama de fuerza cortante. Existen dos métodos para determinar estas reacciones. En el primero se

aplica la definición de momento flexionante, y en el segundo, que es el método que se utilizará en este diseño, la reacción se divide en partes a partir de las cuales se traza fácilmente el diagrama de cortante. Para esto es necesario calcular previamente los momentos de continuidad en los apoyos.

Para determinar los momentos de continuidad se utilizará la ecuación de los tres momentos, a los puntos sobre los apoyos (8):

$$\text{Tramo 1: } M_1 L_1 + 2M_2 (L_1 + L_2) + M_3 L_2 + \frac{\delta A_1 \bar{a}_1}{L_1} + \frac{\delta A_2 \bar{b}_2}{L_2} = 0$$

$$\text{Tramo 9: } M_9 L_9 + 2M_{10} (L_9 + L_{10}) + M_{11} L_{10} + \frac{\delta A_2 \bar{a}_2}{L_9} + \frac{\delta A_2 \bar{b}_2}{L_{10}} = 0$$

Las reacciones en un claro vienen dadas por:

$$R^1 = \frac{M_M - M_m}{L}$$

(8) Resistencia de materiales. Singer 1982, Pág. 259.

Ahora se puede trazar fácilmente el diagrama de fuerza cortante. Como la fuerza cortante a la izquierda de un apoyo (derecha del claro) es igual y opuesta a la reacción, se ha de cambiar el signo obtenido en el apoyo derecho de cada claro para tener el valor de la fuerza cortante. El valor de la reacción total en cada apoyo es la suma, con su signo, de los valores de las reacciones a la izquierda y a la derecha de cada apoyo.

El valor del mayor momento se lo debe encontrar, analizando varias posiciones de la carga actuante; para esto se analiza cuando la fuerza concentrada actúa en cada uno de los tramos; para cada uno se escoge el mayor, y luego se compara entre todos ellos y con el mayor se procede al diseño.

El procedimiento anteriormente expuesto, lo realiza el mismo programa que seleccionará el perfil, y no se incluye en esta sección debido a su extensión.

Las cargas que actúan sobre la viga porta-grúa son:

- La carga concentrada que es la reacción que resulta por efecto del soporte lateral.

- La carga uniformemente distribuida, que es el peso de la viga, más el peso del riel que esta soporta.

2.7.2 RECOMENDACIONES DEL MANUAL DEL A.I.S.C. PARA SELECCIONAR EL PERFIL MAS LIGERO Y LAS DIMENSIONES DE LAS PLACAS.

Cunado se trata de diseñar vigas con más de dos apoyos, recurrimos a la Teoría del Diseño Plástico. El manual del A.I.S.C. recomienda que para vigas continuas de sección compacta, el diseño se realice sobre la base de los $9/10$ de los momentos negativos máximos ocasionados por cargas verticales, y los momentos positivos se aumenten en $1/10$ del promedio de los momentos negativos en los apoyos adyacentes.

A.I.S.C. 1.5.14.1 - Pág. 5.20.

Para el diseño por flexión, aplastamiento, deflexión, cortante, se utilizarán las mismas recomendaciones expuestas en la sección 3.3.1 de este trabajo.

Como se trata de una viga continua con más de dos apoyos, se utilizará una subrutina para calcular las reacciones, y así poder resolver sin complicaciones este problema estáticamente indeterminado.

En las vigas continuas con más de dos apoyos, las reacciones no se pueden encontrar por las reacciones de la estática. Es posible calcular

los valores de estas reacciones hiperestáticas aplicando las condiciones de deformación. Sin embargo, es más conveniente considerar, como desconocidos o hiperestáticos, los momentos flectores en los apoyos. Una vez determinados estos momentos, que se suelen llamar momentos de continuidad, se pasan a calcular las reacciones.

Cálculo de los momentos de continuidad:

Mecánica de sólidos, Singer 8.2 - Pág. 294.

Cálculo de las reacciones en las vigas continuas:

Mecánica de sólidos, Singer 8.5 - Pág. 307.

En este punto hay que hacer una aclaración, y es que la subrutina es válida sólo para el caso de que existan diez reacciones a cuya separación sea la misma para todas; si existieran más o menos reacciones o la separación no sea igual, esta subrutina ya no es válida.

El programa general no se verá afectado, en todo caso cuando se trate de más o menos reacciones, se procederá sólo a cambiar la subrutina.

En los casos en que los extremos de la viga se asientan directamente sobre concreto o mampostería, se suelen colocar placas de acero para distribuir la resistencia.

La resistencia se distribuirá uniformemente a través de la placa de apoyo, a la vez que la mampostería tendrá que ejercer una reacción sobre la placa, tratando de flexarla hacia arriba.

El manual del A.I.S.C. recomienda como una sección crítica desde el borde n libre y medida hasta el borde del filete:

$$t = \frac{3fpn^2}{0.75 F_y}$$

Longitud mínima de aplastamiento de la viga:

$$\frac{R}{t(N + 2K)} \leq 0.75 F_y \quad 1.10.8$$

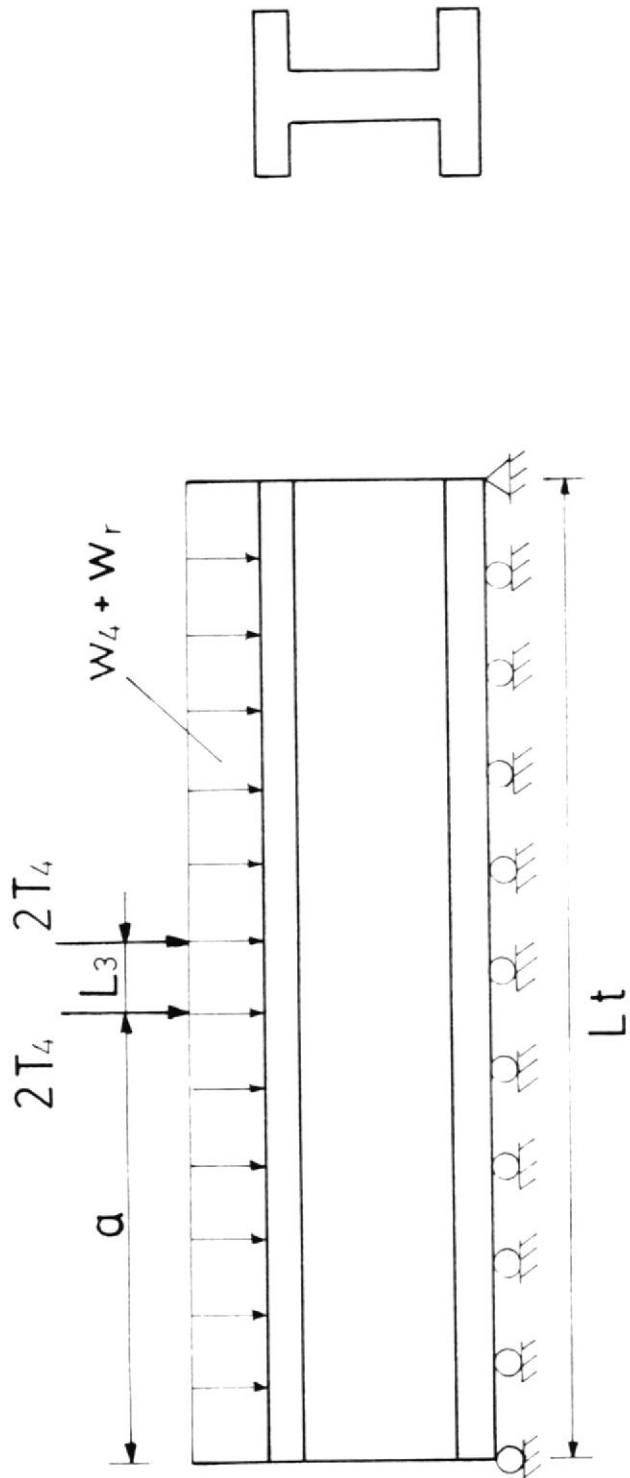
$$\frac{R}{t(N + K)} \leq 0.75 F_y \quad 1.10.9$$

Para una placa plana:

A.I.S.C. 1.5.1.4.3 - Pág. 5.21.

$$F_b = 0.75 F_y$$

FIGURA Nº 9: VIGA PORTAGRUA SOMETIDA A FLEXION



2.7.3 DIAGRAMA DE FLUJO Y ALGORITMO

SUBROUTINA R: PARA
CALCULAR LAS REACCIONES
SOBRE LA VIGA PORTAGRUA

W_3 H
 L_4 (total)
 L_4 (no arriostrada)
N (numero reacciones)

$$A = \left\langle \frac{R}{L} \left\langle \frac{L+H}{2} \right\rangle \right\rangle - \left\langle \frac{R}{L} \left\langle \frac{L-H}{2} \right\rangle [L^2 - \left\langle \frac{L}{2} - \frac{H}{2} \right\rangle] \right\rangle$$

$$B = \left\langle \frac{R}{L} \left\langle \frac{L-H}{2} \right\rangle \right\rangle - \left\langle \frac{R}{L} \left\langle \frac{L+H}{2} \right\rangle [L^2 - \left\langle \frac{L}{2} + \frac{H}{2} \right\rangle] \right\rangle$$

$$M_3 = \frac{0.020A - 5.69W_3L^3 - 0.072B}{15.92L}$$

$$M_8 = \frac{W_3L^2}{2} - 4M_3$$

$$M_7 = \frac{-0.4W_3L^3 - 0.072A + 0.27B - M_8L}{3.73L}$$

$$M_6 = \frac{-\frac{15}{38}W_3L^3 - M_7L + \frac{56}{209}A - B}{\frac{780}{209}L}$$

$$M_5 = \frac{-\frac{11}{28}W_3L^3 - M_6L - \frac{A}{L}}{\frac{780}{209}L}$$

$$M_4 = -\frac{15}{56}M_5 - \frac{3}{28}W_3L^2$$

A

A

$$M_3 = -\frac{1}{10}W_3L^2 - \frac{4M_4}{15}$$

$$M_2 = -\frac{M_3}{4} - \frac{W_3L^2}{8}$$

$$V_1 = V'_2 = -\frac{W_3L}{2}$$

$$V''_2 = V'_3 = \frac{WL}{2}$$

$$V''_3 = V'_4 = -\frac{W_3L}{2}$$

$$V''_4 = V'_5 = -\frac{W_3L}{2}$$

$$V''_5 = -\frac{W_3L^4}{2} + T_3$$

$$V''_6 = -\frac{W_3L^4}{2} + T_3$$

$$V''_7 = V'_7 = -\frac{W_3L}{2}$$

$$V''_7 = V'_8 = -\frac{W_3L}{2}$$

B

B

$$U''_0 = U'_0 = -\frac{W_0 L}{2}$$

$$U''_1 = U'_{10} = -\frac{W_0 L}{2}$$

$$A = 1$$

$$U_{AM} = \frac{M_{A+1} - M_A}{L}$$

$$A = A + 1$$

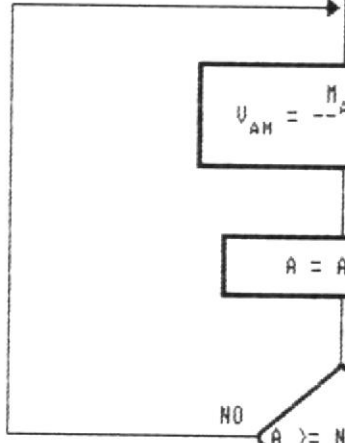
NO
 $A \geq N - 1$
SI

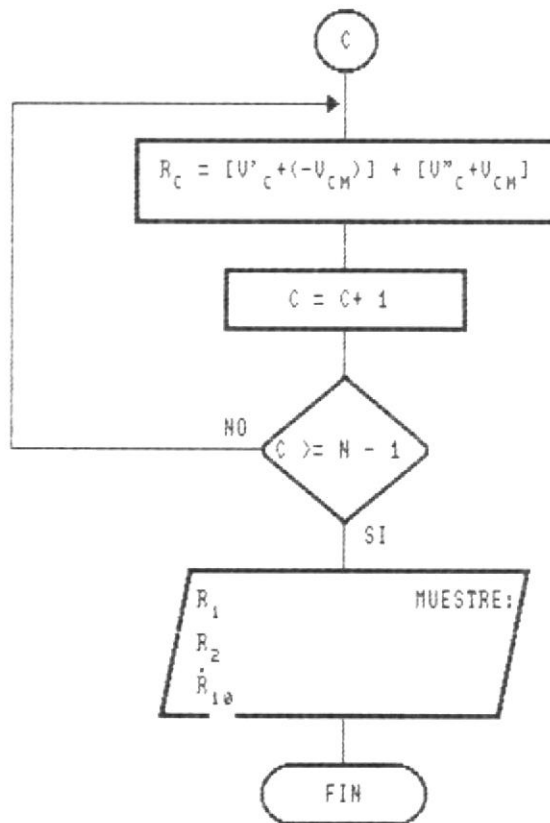
$$R_1 = U_1 + U_{1M}$$

$$R_{10} = U_{10} + (-U_{0N})$$

$$C = 2$$

C





- W_3 : Peso propio del riel
- L_4 : Longitud total de la viga porta-grua
- L : Longitud no arriostrada
- N : Numero de apoyos (reacciones)
- M_N : Momentos de continuidad
- R_C : Reacciones

DISEÑO DE LAS PLACAS DE APOYOS PARA LOS RIELES

bf; tw; d; tf; k
Fp; Fy; G

SUB. R

Rt

NO
CARGA EXTREMOS
SI

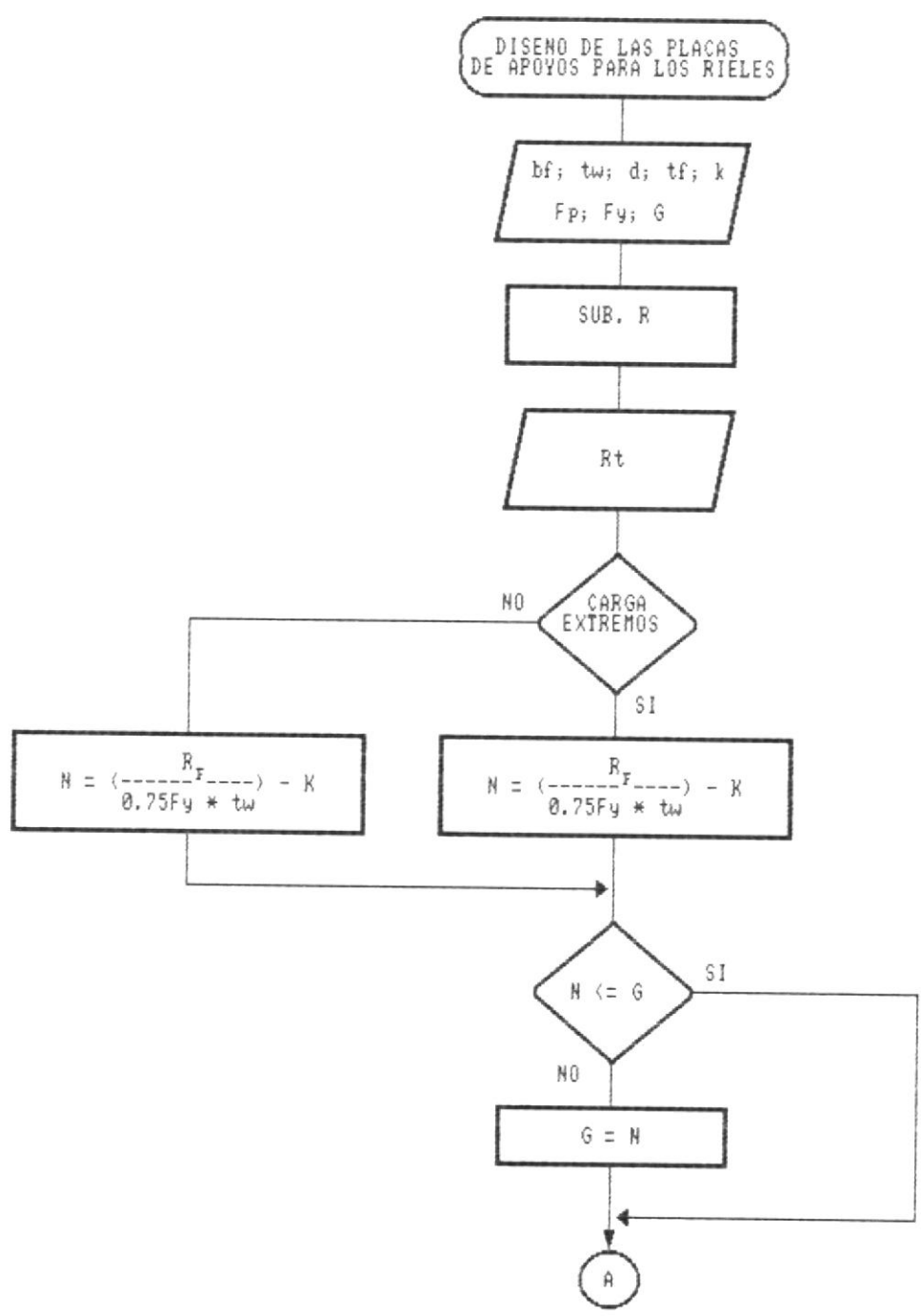
$$N = \left(\frac{R_f}{0.75F_y * tw} \right) - K$$

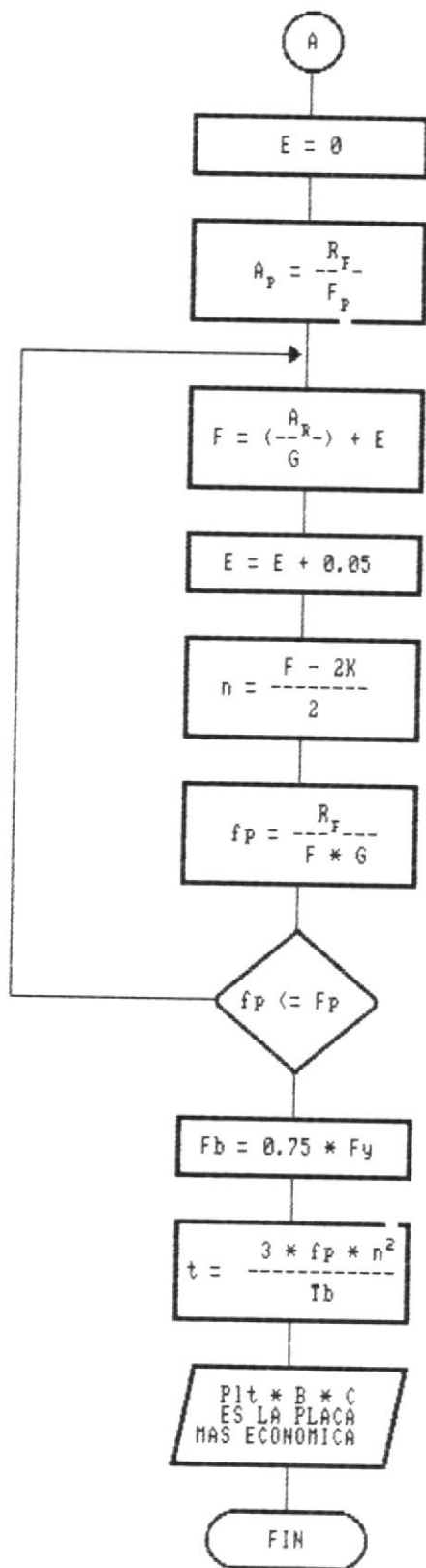
$$N = \left(\frac{R_f}{0.75F_y * tw} \right) - K$$

NO
N <= G
SI

G = N

A





A

$$E = 0$$

$$A_p = \frac{R_f}{F_p}$$

$$F = \left(\frac{A}{G}\right) + E$$

$$E = E + 0.05$$

$$n = \frac{F - 2K}{2}$$

$$f_p = \frac{R_f}{F * G}$$

$f_p \leq F_p$

$$F_b = 0.75 * F_y$$

$$t = \frac{3 * f_p * n^2}{F_b}$$

Plt * B * C
ES LA PLACA
MAS ECONOMICA

FIN

```

                /**PROGRAMA PARA SELECCIONAR EL PERFIL***/
                /**DE LA VIGA PORTAGRUAA***/

#include<math.h>
#include<stdio.h>
#include<ctype.h>
#include "tesis.h"
#define AND &&
#define OR ||
#define Pi 3.1415926
int salir =1.0;

void main()
{
char      C1,C2,C4[30],C5[30],
          Tc[20],Pc[20],L1c[20],L2c[20],Hc[20],vpc[20],
          wrc[20],w1c[20],w2c[20],w3c[20],L3c[20],Fyc[2],
          Fuc[20],L6c[20],Fpc[20],Fusc[20],Gc[20],Ac[20],
          bfc[20],tfc[20],twc[20],rtc[20],Kc[20],Sxc[20],
          w6c[20],dc[20];
          double Fy,Fu,L6,Tt,T1,d,a,B,C,D,temp2,Fp,fp,t,n,E,Ap,F,G,

T,L1,w1,wr,T2,L2,w2,T3,L3,w3,Lna6,T4,T5,P,R,cb1,S,Sx,

A,Af,bf,tf,tw,rt,K,M,M1,Lc,Lc1,Lc2,Lu,Lu1,Lu2,cb2,cb,

fb=0,Fb,fv=0,Fv=0,Fb1,Fb2,M11,M12,M13,M14,M15,M16,M17,

M18,M19,M110,V11,V12,R11,R12,R1m,M1m,temp1,M21,M22,M23,

M24,M25,M26,M27,M28,M29,M210,V21,V22,V23,R2m,M2m,R21,R22,

M31,M32,M33,M34,M35,M36,M37,M38,M39,M310,V32,V33,V34,R31,

R32,R3m,M3m,M41,M42,M43,M44,M45,M46,M47,M48,M49,M410,V43,

V44,V45,R41,R42,R4m,M4m,M51,M52,M53,M54,M55,M56,M57,M58,
M59,M510,V54,V55,V56,R51,R52,R5m,M5m,M2,Mt,w7,w6,

M11a,M12a,M13a,M14a,M15a,M16a,M17a,M18a,M19a,M110a,V11a,

V12a,R11a,R12a,R1ma,M1ma,temp3,M21a,M22a,M23a,M24a,M25a,

M26a,M27a,M28a,M29a,M210a,V21a,V22a,V23a,R2ma,R21a,R22a,

M2ma,M31a,M32a,M33a,M34a,M35a,M36a,M37a,M38a,M39a,M310a,

V32a,V33a,V34a,R31a,R32a,R3ma,M3ma,M41a,M42a,M43a,M44a,

M45a,M46a,M47a,M48a,M49a,M410a,V43a,V44a,V45a,R41a,R42a,

R4ma,M4ma,M51a,M52a,M53a,M54a,M55a,M56a,M57a,M58a,M59a,

M510a,V54a,V55a,V56a,R51a,R52a,R5ma,M5ma,N1,N2,V1,V2,V3,

```

```

V4,V5,d2,d1,Rt,Rt1,Rt2,Mt1,Mt2,Mmax,Mneg,Mpos,V,as,Lmin,
Fus,fs,fbs,fr,I,te,Ap1,Ap3,Ap2,Apt,Wt,H,vp;

clrscr();
do {
    printf("\n Ingrese la capacidad del elevador T en Kips: ");
    gets(Tc);
    T = es_numero(Tc);
    fflush(stdin);
} while ( (T == 0) );

do {
    printf("\n Ingrese el peso del motor y bobina P en Kips: ");
    gets(Pc);
    P = es_numero(Pc);
    fflush(stdin);
} while ( (P == 0) );

do {
    printf("\n Ingrese la longitud del carro para el trole L1 en
pies: ");
    gets(L1c);
    L1 = es_numero(L1c);
    fflush(stdin);
} while ( (L1 == 0) );

do {
    printf("\n Ingrese la longitud de la viga principal L2 en pies:
");
    gets(L2c);
    L2 = es_numero(L2c);
    fflush(stdin);
} while ( (L2 == 0) );

do {
    printf("\n Ingrese la potencia del motor H en hp: ");
    gets(Hc);
    H = es_numero(Hc);
    fflush(stdin);
} while ( (H == 0) );

do {
    printf("\n Ingrese la velocidad del puente vp en pie/minuto: ");
    gets(vpc);
    vp = es_numero(vpc);
    fflush(stdin);
} while ( (vp == 0) );

do {
    printf("\n Ingrese el peso del riel wr en Kips/pie: ");
    gets(wrc);
    wr = es_numero(wrc);
    fflush(stdin);
} while ( (wr == 0) );

do {
    printf("\n Ingrese el peso de la viga del carro para el trole w1
en Kips/pie: ");
    gets(w1c);
    w1 = es_numero(w1c);
    fflush(stdin);
}

```

```

    } while ( (w1 == 0) );
do {
    printf("\n Ingrese el peso de la viga principal w2 en Kips/pie:
");
    gets(w2c);
    w2 = es_numero(w2c);
    fflush(stdin);
    } while ( (w2 == 0) );
do {
    printf("\n Ingrese la longitud del soporte lateral L3 en pies:
");
    gets(L3c);
    L3 = es_numero(L3c);
    fflush(stdin);
    } while ( (L3 == 0) );
do {
    printf("\n Ingrese el peso del soporte lateral w3 en Kips/pie:
");
    gets(w3c);
    w3 = es_numero(w3c);
    fflush(stdin);
    } while ( (w3 == 0) );
do {
    printf("\n Ingrese el Esfuerzo de Fluencia Fy en Kips/pulgada
cuadrada: ");
    gets(Fyc);
    Fy = es_numero(Fyc);
    fflush(stdin);
    } while ( (Fy == 0) );
do {
    printf("\n Ingrese la Resistencia ultima Fu en Kips/pulgada
cuadrada: ");
    gets(Fuc);
    Fu = es_numero(Fuc);
    fflush(stdin);
    } while ( (Fu == 0) );
do {
    printf("\n Ingrese la longitud no arriostrada L6 de la viga
portagrua en pies: ");
    gets(L6c);
    L6 = es_numero(L6c);
    fflush(stdin);
    } while ( (L6 == 0) );
do {
    printf("\n Ingrese el Fp del concreto en Kips/pulgada cuadrada:
");
    gets(Fpc);
    Fp = es_numero(Fpc);
    fflush(stdin);
    } while ( (Fp == 0) );
do {
    printf("\n Ingrese la resistencia ultima a la tension del
electrodo Fus En Ksi: ");
    gets(Fusc);
    Fus = es_numero(Fusc);

```

```

        fflush(stdin);
    } while ( (Fus == 0) );
do {
    printf("\n Ingrese el ancho de la columna de concreto o de la
    placa da apoyo G en pulgadas: ");
    gets(Gc);
    G = es_numero(Gc);
    fflush(stdin);
    } while ( (G == 0) );

    T1 = ((T+P)/4.0);
    T2 = (T1+(w1*L1));
    T3 = (T2+((wr*L2)/2.0));
    T4 = ((T3/2.0)+((w3*L3)/2.0));
    Tt = ((2.0*T4));
    a = ((L6/2.0)-(L3/2.0));
    A = ( ((a+L3)/L6)*(pow((L6),2.0)-pow((a+L3),2.0)) );
    B = ( (a/L6)*(pow((L6),2.0)-pow((a+L3),2.0)) );
    C = (Tt/L6)*(A+B);
    temp1 = ((wr*pow((L6),2.0))/16.0);    /** Kips por pie***/
    D = (C/4.0);                        /**Kips por pie***/
    Wt = (33.0*H/vp);    /**Kips***/

        /** ANALISIS DEL PRIMER TRAMO ***/

M19 = ( ((-17137.0*temp1)+D)/(40545.0/4.0)); /**Kips/pie***/
M18 = ( ( ((-2911.0*M19)/4.0)-(4591.0*temp1)-D)/2716.0);
M17 = ( ( (-195.0*M18)-(1231.0*temp1)+D)/(2911.0/4.0));
M16 = ( ( ((-209.0*M17)/4.0)-(329.0*temp1)-D)/195.0);
M15 = ( ( (-14.0*M16)-(89.0*temp1)+D)/(209.0/4.0));
M14 = ( ( ((-15.0*M15)/4.0)-(23.0*temp1)-D)/14.0);
M13 = ( ((-M14)-(7.0*temp1)+D)/(15.0/4.0) );
M12 = ( (-M13/4.0)-(temp1)-D);
M11 = 0.0;
V11 = ((M12-M11)/L6);    /**Kips***/
V12 = ((M13-M12)/L6);
R11 = V11+Tt+((wr*L6)/2.0) ;    /**Kips***/
R12 = (( (Tt+((wr*L6)/2.0))-(V11) )+( ((wr*L6)/2.0)+V12 ));
M1m = (((R11-Tt)*(L6/2.0))/2.0)+(Tt*a); /**Kips por pie***/
if(R11 > R12)
    R1m = R11;

else
    R1m = R12;

        /** ANALISIS DEL SEGUNDO TRAMO ***/

M29 = ( ((-17133.0*temp1)-(3.0*D))/(40545.0/4.0) );
M28 = ( ( ((-2911.0*M29)/4.0)-(4595.0*temp1)+(3.0*D))/2716.0 );
M27 = ( ( ((-195.0*M28)-(1227.0*temp1)-(3.0*D))/2911.0/4.0) );
M26 = ( ( ((-209.0*M27)/4.0)-(333.0*temp1)+(3.0*D))/195.0);
M25 = ( ((-14.0*M26)-(85.0*temp1)-(3.0*D))/(209.0/4.0) );

```

```

M24 = ( ( (-15.0*M25)/4.0)-(27.0*temp1)+(3.0*D))/14.0);
M23 = ( (-M24)-(3.0*temp1)-(3.0*D))/(15.0/4.0));
M22 = ( (-M23/4.0)-temp1-D );
M21 = 0.0;
V21 = ((M22-M21)/L6);
V22 = ((M23-M22)/L6);
V23 = ((M24-M23)/L6);
R21 = ( (((wr*L6)/2.0)-(V21))+(Tt+((wr*L6)/2.0)+V22));
R22 = ( (Tt+((wr*L6)/2.0)-(V22))+(((wr*L6)/2.0)+(V23)));
M2m = (((Tt+((wr*L6)/2.0)+V22)-Tt)*(L6/2.0))/2.0+(Tt*a);
if(R21 > R22)
    R2m = R21;

else
    R2m = R22;

    /** ANALISIS DEL TERCER TRAMO **/

M39 = ( ((-8566.0*temp1*2.0)+(11.0*D))/(40545.0/4.0) );
M38 = ( ( (-2911.0*M39)/4.0)-(2298.0*2.0*temp1)-(11.0*D))/2716.0);
M37 = ( ((-195.0*M38)-(613.0*2.0*temp1)+(11.0*D))/(2911.0/4.0));
M36 = ( ( (-209.0*M37)/4.0)-(167.0*2.0*temp1)-(11.0*D))/195.0);
M35 = ( ((-14.0*M36)-(42.0*2.0*temp1)+(11.0*D))/(209.0/4.0));
M34 = ( ( (-15.0*M35)/4.0)-(14.0*2.0*temp1)-(11.0*D))/14.0);
M33 = ( ((-M34)-(2.0*temp1)-(4.0*D))/(15.0/4.0));
M32 = ((-2.0*temp1)-(M33/4.0));
V32 = ((M33-M32)/L6);
V33 = ((M34-M33)/L6);
V34 = ((M35-M34)/L6);
R31 = ( (((wr*L6)/2.0)-(V32))+(Tt+((wr*L6)/2.0)+V33) );
R32 = ( (Tt+((wr*L6)/2.0)-V33)+(((wr*L6)/2.0)+V34) );
M3m = ( ( (((Tt+((wr*L6)/2.0)+V33)-Tt)*(L6/2.0))/2.0)+(Tt*a));
if( R31 > R32 )
    R3m = R31;

else
    R3m = R32;

    /** ANALISIS DEL CUARTO TRAMO **/

M49 = ( ((-17095.0*temp1)-(41.0*D))/(40545.0/4.0) );
M48 = ( ( (-2911.0*M49)/4.0)-(4633.0*temp1)+(41.0*D))/2716.0);
M47 = ( ((-195.0*M48)-(1189.0*temp1)-(41.0*D))/(2911.0/4.0) );
M46 = ( ( (-209.0*M47)/4.0)-(371.0*temp1)+(41.0*D))/195.0);
M45 = ( ((-14.0*M46)-(47.0*temp1)-(41.0*D))/(209.0/4.0) );
M44 = ( ( (-15.0*M45)/4.0)-(9.0*temp1)-(15.0*D))/14.0);
M43 = ( ((-M44)-(3.0*temp1*2.0))/(15.0/4.0) );
V43 = ((M44-M43)/L6);
V44 = ((M45-M44)/L6);
V45 = ((M46-M45)/L6);
R41 = ( (((wr*L6)/2.0)-(V43))+( Tt+((wr*L6)/2.0)+V44) );
R42 = ( (Tt+((wr*L6)/2.0)-V44)+(((wr*L6)/2.0)+V45) );
M4m = ( ( (((Tt+((wr*L6)/2.0)+V45)-Tt)*(L6/2.0))/2.0)+(Tt*a));
if(R41 > R42 )

```

```

R4m = R41;

else
R4m = R42;

    /** ANALISIS DEL QUINTO TRAMO ***/

M59 = ( ((-17289.0*temp1)+(153.0*D))/(40545.0/4.0));
M58 = ( ( ((-2911.0*M59)/4.0)-(4439.0*temp1)-(153.0*D))/2716.0);
M57 = ( ( (-195.0*M58)-(1383.0*temp1)+(153.0*D))/(2911.0/4.0));
M56 = ( ( ((-209.0*M57)/4.0)-(177.0*temp1)-(153.0*D))/195.0);
M55 = ( ( (-14.0*M56)-(16.0*temp1*2.0))/(209.0/4.0));
M54 = ( ( ((-15.0*M55)/4.0)-(12.0*2.0*temp1))/14.0);
V54 = ((M55-M54)/L6);
V55 = ((M56-M55)/L6);
V56 = ((M57-M56)/L6);
R51 = ( (((wr*L6)/2.0)-V54)+(Tt+((wr*L6)/2.0)+V55) );
R52 = ( (Tt+((wr*L6)/2.0)-V55)+(((wr*L6)/2.0)+V56) );
M5m = ( ( (((Tt+((wr*L6)/2.0)+V55)-Tt)*(L6/2.0))/2.0)+(Tt*a) );
if (R51 > R52)
    R5m = R51;

else
    R5m = R52;

    /** DETERMINACION DEL MOMENTO MAXIMO ***/

M = M1m;
M1 = M11;
M2 = M12;
R = R1m;
if (M < M2m) {
    M = M2m;
    M1 = M22;
    M2 = M23;
    R = R2m;
}
if (M < M3m) {
    M = M3m;
    M1 = M33;
    M2 = M34;
    R = R3m;
}
if (M < M4m) {
    M = M4m;
    M1 = M44;
    M2 = M45;
    R = R4m;
}
if (M < M5m) {
    M = M5m;
    M1 = M55;
    M2 = M56;
    R = R5m;
}

```

```

    }
    printf("\n El valor del momento maximo M es %lf",M);
    printf("\n el valor de R es %lf",R);
    cb1 = (1.75+(1.05*(M1/M2))+(0.3*pow((M1/M2),2.0)));
    if (cb1 > 2.3)
        cb2 = 2.3;

else
    cb2 = cb1;

printf("\n El valor de cb2 es %lf",cb2);
Fb = (0.66*Fy); /**ASUMO COMPACTA ***/
S = ((M*12.0)/Fb);

do {
do {
do {
do {
if(fv <= Fv) {
    if(fb <= Fb) {
        printf("\n El modulo del perfil requerido es %lf",S);
        printf("\n Ingrese un perfil: ");
        gets(C4);
        fflush(stdin);
    }
    else {
        clrscr();
        printf("\n Falla por Flexion,Ingrese otro perfil con un
modulo mayor\n:");
        gets(C4);
        fflush(stdin);
    }
}
else {
    clrscr();
    printf("\n Falla por cortante,Ingrese otro perfil con un modulo
mayor\n:");
    gets(C4);
    fflush(stdin);
}
}
do {
    printf("\n Ingrese el area A en pulgada cuadrada: ");
    gets(Ac);
    A = es_numero(Ac);
    fflush(stdin);
} while ( (A == 0) );
do {
    printf("\n Ingrese el ancho del ala bf en pulgadas: ");
    gets(bfc);
    bf = es_numero(bfc);
    fflush(stdin);
} while ( (bf == 0) );
do {
    printf("\n Ingrese el espesor del ala tf en pulgadas: ");
    gets(tfc);

```

```

    tf = es_numero(tfc);
    fflush(stdin);
} while ( (tf == 0) );
do {
    printf("\n Ingrese el espesor del alma tw en pulgadas: ");
    gets(twc);
    tw = es_numero(twc);
    fflush(stdin);
} while ( (tw == 0) );
do {
    printf("\n Ingrese la altura o peralte de la viga d en pulgadas:
");
    gets(dc);
    d = es_numero(dc);
    fflush(stdin);
} while ( (d == 0) );
do {
    printf("\n Ingrese el radio de giro rt en pulgadas: ");
    gets(rtc);
    rt = es_numero(rtc);
    fflush(stdin);
} while ( (rt == 0) );
do {
    printf("\n Ingrese el modulo de la seccion Sx en pulgadas: ");
    gets(Sxc);
    Sx = es_numero(Sxc);
    fflush(stdin);
} while ( (Sx == 0) );
do {
    printf("\n Ingrese la distancia entre el ala y el alma K en
pulgadas(decimales): ");
    gets(Kc);
    K = es_numero(Kc);
    fflush (stdin);
} while ( (K == 0) );
do {
    printf("\n Ingrese el peso de la viga seleccionada w6 en
Kips/pie: ");
    gets(w6c);
    w6 = es_numero(w6c);
    fflush(stdin);
} while ( (w6 ==0) );

if ( (M > M1) && (M >M2) )
    cb = 1.0;

else
    cb = cb2;

Af = (bf*tf);
Lc1 = ((76.0*bf)/(sqrt(Fy)*12.0));
Lc2 = ((20000.0*Af)/(d*Fy*12.0));
if (Lc1 > Lc2)
    Lc = Lc2;
else

```

```

Lc = Lc1;
Lu1 = ((20000.0*Af)/(d*Fy*12.0));
Lu2 = (rt*sqrt(102000.0/Fy))/12.0;
if (Lu1 > Lu2)
    Lu = Lu1;
else
    Lu = Lu2;
printf("\n Los valores de Lc y Lu son
%lf\n%lf\n",Lc,Lu);
if(Lna6 < Lc ) {
    printf("\n Tiene alas rigidizadas ? S/N ==>
");
do {
    C1 = getch();
    switch(C1) {
    case 'S' | 's' : {
        if( (bf/tf) <= (190.0/sqrt(Fy)) ) {
            if(d/tw <= 640.0/sqrt(Fy))
                Fb = (0.66*Fy);
            else
                Fb = (0.6*Fy);
        }
    else {
        if( (((65.0/sqrt(Fy)) <= (bf/(2.0*tf)))
        && ((bf/(2.0*tf)) < (95.0/sqrt(Fy))))
        || ((d/tw) < (640.0/sqrt(Fy))))
            Fb = Fy*(0.79-(0.002*
            (bf/(2.0*tf))*sqrt(Fy)));
        else
            Fb = (0.6*Fy);
        }
        salir=0;
        break;
    }
    case 'N' | 'n' : {
        if((bf/(2.0*tf)) <= (65.0/sqrt(Fy))) {
            if(d/tw <= 640.0/sqrt(Fy))
                Fb = (0.66*Fy);
            else
                Fb = (0.6*Fy);
        }
    else {
        if( (((65.0/sqrt(Fy)) <= (bf/(2.0*tf)))
        && ((bf/(2.0*tf)) < (95.0/sqrt(Fy))))
        || ((d/tw) < (640.0/sqrt(Fy))))
            Fb = Fy*(0.79-(0.002*
            (bf/(2.0*tf))*sqrt(Fy)));
        else
            Fb = (0.6*Fy);
        }
        salir=0;
        break;
    }
}
default : {
    printf("\n Ingrese S/N ==> ");
}
}

```

```

        break;
    }
}
} while(salir);
}
else {
    if(Lu > Lna6)
        Fb = (0.66*Fy);
    else
        if( ((L6*12.0)/rt) >
            sqrt((510000.0*cb)/Fy) ) {
            Fb1 = (170000.0*cb)/
                pow(((L6*12.0)/rt),2.0);
            Fb2 = (12000.0*cb)/(L6*12.0*d/Af);
            if(Fb1 > Fb2)
                if(Fb1 > 0.6*Fy)
                    Fb = (0.6*Fy);
                else
                    Fb = Fb1;
            else
                if(Fb2 > (0.6*Fy))
                    Fb = (0.6*Fy);
                else
                    Fb = Fb2;
        }
        if( (L6*12.0/rt) > sqrt(102000.0*cb/Fy)
        ) {
            temp2 = pow((L6*12.0/rt),2.0);
            Fb1 = Fy*((2.0/3.0) -
                ((Fy*temp2)/(1530000.0*cb)));
            Fb2 = (12000.0*cb)/
                (L6*12.0*d/Af);
            if(Fb1 > Fb2)
                if(Fb1 > (0.6*Fy))
                    Fb = (0.6*Fy);
                else
                    Fb = Fb1;
            else
                if(Fb2 > (0.6*Fy))
                    Fb = (0.6*Fy);
                else
                    Fb = Fb2;
        }
        else
            Fb = (0.6*Fy);
    }
    printf("\n EL VALOR DE Fb ES %1f",Fb);
}

```

/**/ DETERMINACION DEL MOMENTO REAL MAXIMO /**/

/**/ ANALISIS DEL PRIMER TRAMO/**/

w7 = (wr+w6);

```

temp3 = ((w7*pow((L6),2.0))/16.0);
M19a = (((-17137.0*temp3)+D)/(40545.0/4.0));
M18a = ( ( ( (-2911.0*M19a)/4.0)-(4591.0*temp3)-D)/2716.0);
M17a = ( ( ( (-195.0*M18a)-(1231.0*temp3)+D)/(2911.0/4.0));
M16a = ( ( ( (-209.0*M17a)/4.0)-(329.0*temp3)-D)/195.0);
M15a = ( ( ( (-14.0*M16a)-(89.0*temp3)+D)/(209.0/4.0));
M14a = ( ( ( (-15.0*M15a)/4.0)-(23.0*temp3)-D)/14.0);
M13a = ( ( (-M14a)-(7.0*temp3)+D)/(15.0/4.0));
M12a = ( (-M13a/4.0)-(temp3)-D);
M11a = 0.0;
V11a = ((M12a-M11a)/L6);
V12a = ((M13a-M12a)/L6);
R11a = V11a+Tt+((w7*L6)/2.0);
V1 = V11a+Tt+((w7*L6)/2.0);
R12a = ((Tt+((w7*L6)/2.0)-(V11a))+((w7*L6)/2.0+V12a));
M1ma = (((R11a-Tt)*(L6/2.0))/2.0)+(Tt*a);
if(R11a > R12a)
    R1ma = R11a;
else
    R1ma = R12a;

```

/** ANALISIS DEL SEGUNDO TRAMO **/

```

M29a = ( ((-17133.0*temp3)-(3.0*D))/(40545.0/4.0));
M28a = ( ( ( (-2911.0*M29a)/4.0)-(4595.0*temp3)+(3.0*D))/2716.0);
M27a = ( ( ( (-195.0*M28a)-(1227.0*temp3)-(3.0*D))/(2911.0/4.0));
M26a = ( ( ( (-209.0*M27a)/4.0)-(333.0*temp3)+(3.0*D))/195.0);
M25a = ( ( ( (-14.0*M26a)-(85.0*temp3)-(3.0*D))/(209.0/4.0));
M24a = ( ( ( (-15.0*M25a)/4.0)-(27.0*temp3)+(3.0*D))/14.0);
M23a = ( ( (-M24a)-(3.0*temp3)-(3.0*D))/(15.0/4.0));
M22a = ( (-M23a/4.0)-temp3-D);
M21a = 0.0;
V21a = ((M22a-M21a)/L6);
V22a = ((M23a-M22a)/L6);
V23a = ((M24a-M23a)/L6);
R21a = ( (((w7*L6)/2.0)-(V21a))+Tt+((w7*L6)/2.0)+(V22a));
R22a = ( (Tt+((w7*L6)/2.0)-V22a)+(((w7*L6)/2.0)+(V23a));
V2 = Tt+((w7*L6)/2.0)+V22a;
M2ma = (((Tt+((w7*L6)/2.0)+(V22a)-Tt)*(L6/2.0))/2.0)+(Tt*a);
if(R21a > R22a)
    R2ma = R21a;
else
    R2ma = R22a;

```

/** ANALISIS DEL TERCER TRAMO **/

```

M39a = ( ((-8566.0*temp3*2.0)+(11.0*D))/(40545.0/4.0));
M38a = ( ( ( (-2911.0*M39a)/4.0)-(2298.0*2.0*temp3)-(11.0*D))/2716.0);
M37a = ( ( ( (-195.0*M38a)-(613.0*2.0*temp3)+(11.0*D))/(2911.0/4.0));
M36a = ( ( ( (-209.0*M37a)/4.0)-(167.0*2.0*temp3)-(11.0*D))/195.0);
M35a = ( ( (-14.0*M36a)-(42.0*2.0*temp3)+(11.0*D))/(209.0/4.0));

```

```

M34a = ( ( (-15.0*M35a)/4.0)-(14.0*2.0*temp3)-(11.0*D))/14.0);
M33a = ( ( (-M34a)-(2.0*temp3)-(4.0*D))/(15.0/4.0));
M32a = ((-2.0*temp3)-(M33a/4.0));
V32a = ((M33a-M32a)/L6);
V33a = ((M34a-M33a)/L6);
V34a = ((M35a-M34a)/L6);
R31a = ( (((w7*L6)/2.0)-(V32a))+(Tt+((w7*L6)/2.0)+V33a));
R32a = ( (Tt+((w7*L6)/2.0)-V33a)+(((w7*L6)/2.0)+V34a));
M3ma = ( ( (((Tt+((w7*L6)/2.0)+V33a)-Tt)*(L6/2.0))/2.0)+(Tt*a));
V3 = (Tt+((w7*L6)/2.0)+V33a);
if(R31a > R32a)
    R3ma = R31a;
else
    R3ma = R32a;

```

/** ANALISIS DEL CUARTO TRAMO **/

```

M49a = ( ((-17095.0*temp3)-(41.0*D))/(40545.0/4.0));
M48a = ( ( ((-2911.0*M49a)/4.0)-(4633.0*temp3)+(41.0*D))/2716.0);
M47a = ( ((-195.0*M48a)-(1189.0*temp3)-(41.0*D))/(2911.0/4.0));
M46a = ( ( ((-209.0*M47a)/4.0)-(371.0*temp3)+(41.0*D))/195.0);
M45a = ( ((-14.0*M46a)-(47.0*temp3)-(41.0*D))/(209.0/4.0));
M44a = ( ( ((-15.0*M45a)/4.0)-(9.0*temp3)-(15.0*D))/14.0);
M43a = ( ((-M44a)-(3.0*2.0*temp3))/(15.0/4.0));
V43a = ((M44a-M43a)/L6);
V44a = ((M45a-M44a)/L6);
V45a = ((M46a-M45a)/L6);
R41a = ( (((w7*L6)/2.0)-(V43a))+(Tt+((w7*L6)/2.0)+V44a));
R42a = ( (Tt+((w7*L6)/2.0)-V44a)+(((w7*L6)/2.0)+V45a));
M4ma = ( (((Tt+((w7*L6)/2.0)+V45a)-Tt)*(L6/2.0))/2.0)+(Tt*a));
V4 = (Tt+((w7*L6)/2.0)+V44a);
if(R41a > R42a)
    R4ma = R41a;
else
    R4ma = R42a;

```

/** ANALISIS DEL QUINTO TRAMO **/

```

M59a = ( ((-17289.0*temp3)+(153.0*D))/(40545.0/4.0));
M58a = ( ( ((-2911.0*M59a)/4.0)-(4439.0*temp3)-(153.0*D))/2716.0);
M57a = ( ((-195.0*M58a)-(1383.0*temp3)+(153.0*D))/(2911.0/4.0));
M56a = ( ( ((-209.0*M57a)/4.0)-(177.0*temp3)-(153.0*D))/195.0);
M55a = ( ((-14.0*M56a)-(16.0*temp3*2.0))/(209.0/4.0));
M54a = ( ( ((-15.0*M55a)/4.0)-(12.0*2.0*temp3))/14.0);
V54a = ((M55a-M54a)/L6);
V55a = ((M56a-M55a)/L6);
V56a = ((M57a-M56a)/L6);
R51a = ( (((w7*L6)/2.0)-V54a)+(Tt+((w7*L6)/2.0)+V55a));
R52a = ( (Tt+((w7*L6)/2.0)-V55a)+(((w7*L6)/2.0)+V56a));
M5ma = ( (((Tt+((w7*L6)/2.0)+V55a)-Tt)*(L6/2.0))/2.0)+(Tt*a));
V5 = (Tt+((w7*L6)/2.0)+V55a);

```

```

if(R51a > R52a)
    R5ma = R51a;

else
    R5ma = R52a;

    Mt = M1ma;
    Mt1 = M11a;
    Mt2 = M12a;
    Rt = R1ma;
    V = V1;
    if(Mt < M2ma) {
        Mt = M2ma;
        Mt1 = M22a;
        Mt2 = M23a;
        Rt = R2ma;
        V = V2;
    }
    if(Mt < M3ma) {
        Mt = M3ma;
        Mt1 = M33a;
        Mt2 = M34a;
        Rt = R3ma;
        V = V3;
    }
    if(Mt < M4ma) {
        Mt = M4ma;
        Mt1 = M44a;
        Mt2 = M45a;
        Rt = R4ma;
        V = V4;
    }
    if(Mt < M5ma) {
        Mt = M5ma;
        Mt1 = M55a;
        Mt2 = M56a;
        Rt = R5ma;
        V = V5;
    }
    printf("\n EL VALOR DE Mt1 ES %1f",Mt1);
    printf("\n EL VALOR DE Mt2 ES %1f",Mt2);
    printf("\n EL VALOR DE Rt ES %1f",Rt);

    /** DISEÑO POR FLEXION ***/

Mneg = (9.0/10.0)*((-1.0)*(Mt1+Mt2));
Mpos = Mt+((1.0/10.0)*(-1.0)*(Mt1+Mt2));
if(Mneg > Mpos)
    Mmax = Mneg;
else
    Mmax = Mpos;
if(Fb == (0.66*Fy))
    fb = ((Mmax*12.0)/Sx);

```

```

        else
            fb = ((Mt*12.0)/Sx);
    } while ( fb > Fb );

        /** DISEÑO POR CORTANTE ***/

fv = V/(d*tw);
Fv = (0.4*Fy);
} while ( fv > Fv );

        /** DISEÑO POR AFLASTAMIENTO ***/

N1 = ((Rt/(0.75*Fy*tw))-2.0*K);
N2 = fabs(N1);

        /** DISEÑO POR DEFLEXION ***/
        /**PARA CARGA DE IMPACTO Y VIBRACION ***/

d1 = (L6*12.0)/d;
d2 = 20.0;
if (d1 <= d2)
    printf("\n EL PERFIL SATISFACE DEFLEXION: ");
else
    printf("\n Falla por Deflexion: ");
} while ( d1 > d2 );

        /** DISEÑO DE LAS PLACAS DE APOYO ***/

if ( N2 > G ) {
    G = N2;
}
E = 0.0;
Ap = (Rt/Fp);
do {
    E = (E + 0.05);
    F = ((Ap/G)+E);
    n = ((F/2.0)-K);
    fp = (Rt/(F*G));
} while ( fp > Fp );
    printf("\n el valor de fp es %lf",fp);
    t = sqrt((3.0*fp*pow((n),2.0))/(0.75*Fy));

        /** DISEÑO DE LAS UNIONES SOLDADAS ***/

I = pow((d-(2.0*tf)),3.0)/12.0;
fs = V/((d-(2.0*tf))+(2.0*(bf-tw)));

```

```

fbs = Mt*((d-(2.0*tf))/2.0)/I;
fr = sqrt(pow((fs),2.0)+pow((fbs),2.0));
te = fr/(0.3*Fus);
as = te/0.707;
Lmin = 4.0*as;

      /** DISEÑO DE LOS PERNOS DE ANCLAJE ***/

Apt = (V/(0.66*Fy));
Ap2 = (Wt/(0.3*Fu));
Ap3 = (Wt/(0.75*Fy));
Apt = Ap1;
if(Apt < Ap2)
    Apt = Ap2;
if(Apt < Ap3)
    Apt = Ap3;
clrscr();

if( (fb < Fb) && (fabs((fb-Fb)/fb) < 0.05)) {
    printf("\n EL PERFIL SELECCIONADO ES EL MAS ECONOMICO: ");
    printf("\n SE CUMPLE: ");
    printf("\n fb %1f <= Fb %1f",fb,Fb);
    printf("\n fv %1f <= Fv %1f",fv,Fv);
    printf("\n d1 %1f <= d2 %1f",d1,d2);
    printf("\n LA LONGITUD MINIMA DE ASIEN TO PARA QUE NO SE APLASTE
ES EN PULGADAS %1f\n",N2);
    printf("\n EL PERFIL SELECCIONADO ES %s",C4);
    printf("\n LAS DIMENSIONES DE LA PLACA SON: ");
    printf("\n EL ESPESOR DE LA PLACA t EN PULGADAS ES %1f",t);
    printf("\n EL LARGO DE LA PLACA G EN PULGADAS ES %1f",G);
    printf("\n EL ANCHO DE LA PLACA F EN PULGADAS ES %1f ",F);
    printf("\n LAS DIMENSIONES DE LA SOLDADURA SON: ");
    printf("\n ANCHO DE SOLDADURA as EN PULGADAS ES %1f",as);
    printf("\n LA LONGITUD MINIMA Lmin EN PULGADAS ES %1f",Lmin);
    printf("\n EL AREA OPTIMA DE LOS PERNOS DE ANCLAJE EN PULGADAS
ES : %1f",Apt);
    printf("\n [PRESIONE CUALQUIER TECLA PARA CONTINUAR] ");
    getch();
}
else {
    if(fb < Fb) {
        printf("\n EL PERFIL CUMPLE PERO NO ES EL MAS ECONOMICO,UD
PUEDE SELECCIONAR UN PERFIL      MENOR: ");
        printf("\n fb %1f <= Fb %1f",fb,Fb);
        printf("\n fv %1f <= Fv %1f",fv,Fv);
        printf("\n d1 %1f <= d2 %1f", d1,d2);
        printf("\n LA LONGITUD MINIMA DE ASIEN TO PERA QUE NO SE
APLASTE ES EN PULGADAS %1f\n",N2);
        printf("\n EL PERFIL SELECCIONADO ES %s",C4);
        printf("\n LAS DIMENSIONES DE LA PLACA SON: ");
        printf("\n EL ESPESOR DE LA PLACA t EN PULGADAS ES
%1f",t);
        printf("\n EL LARGO DE LA PLACA G EN PULGADAS ES %1f",G);
        printf("\n EL ANCHO DE LA PLACA F EN PULGADAS ES %1f",F);
    }
}

```

```
printf("\n LAS DIMENSIONES DE LA SOLDADURA SON: ");
printf("\n EL ANCHO DE LA SOLDADURA as EN PULGADAS ES
%lf",as);
printf("\n LA LONGITUD MINIMA Lmin EN PULGADAS ES
%lf",Lmin);
printf("\n EL AREA OPTIMA DE LOS PERNOS DE ANCLAJE EN
PULGADAS ES : %lf",Apt);
gotoxy(18,23);printf(" [Presione cualquier tecla para
continuar]: ");
getch();
}
else {
printf("\n EL PERFIL NO SATISFACE: ");
printf("\n [PRESIONE CUALQUIER TECLA PARA CONTINUAR] ");
getch();
}
}
} while ((fb > Fb) && (fv > Fv));
}
```

2.8 EL GANCHO DEL APAREJO ELEVADOR.

2.8.1 ANALISIS DE ESFUERZOS.

Los elementos sometidos a flexión no siempre son barras rectas. A veces como en el caso de los ganchos de las grúas, la línea media de la barra es una curva en el plano de aplicación de las cargas y momentos flexionantes.

El gancho del aparejo elevador para el puente-grúa está sometido a una carga concentrada, que es la capacidad del mismo.

Si la curvatura es grande, o el radio de curvatura pequeño, la distribución de esfuerzos difiere de la que se dedujo para vigas rectas.

Para el análisis del gancho, haremos su estudio como una viga curva, donde el esfuerzo en las fibras extremas se puede calcular mediante la expresión (9):

$$\sigma = K \frac{Mc}{I}$$

Donde K depende de $\frac{R}{C}$

Para este análisis se analizará la sección más crítica.

(9) Resistencia de materiales. Singer 1982, Pág. 457.

El momento flexionante producido por la carga T, en la sección AB (ver figura No.5) con respecto a su centro de gravedad, es:

$$M = T * (C + r_{AB})$$

El esfuerzo de flexión será:

- Tensión

$$\sigma_A = K_i \frac{T * (C + r_{AB})}{I} . C$$

- Compresión

$$\sigma_B = K_e \frac{T * (C + r_{AB})}{I} . C$$

El esfuerzo axial uniforme será:

$$\sigma_a = \frac{T}{A}$$

Donde:

A: área sección transversal

Resultan como valores finales:

$$\sigma_B = - \left[K_i \frac{T * (C + r_{AB})}{I} . C \right] + \frac{T}{A}$$

$$\sigma_A = \left[K_e \frac{T * (C + r_{AB})}{I} . C \right] + \frac{T}{A}$$

Para una sección cualquiera CD:



$$M = T(C + r_{CD}) \cdot \cos \theta$$

El esfuerzo flexionante será:

$$\sigma = \frac{N}{A} \pm K \frac{Mc}{I}$$

Donde la componente de T normal a CD es:

$$N = T \cos \theta$$

Donde:

$$V_{AB} = R - C$$

2.8.2 RECOMENDACIONES PARA DETERMINAR EL AREA OPTIMA Y DIMENSIONES.

Para encontrar el esfuerzo circunferencial:

$$\sigma_{\theta} = \frac{N}{A} \pm \frac{(PR)(A - r_B A_m)}{A r_B (R A_m - A)}$$

También se puede utilizar la siguiente ecuación para el esfuerzo circunferencial:

$$\sigma_{\theta} = \frac{P}{A} + \frac{PR}{AR} \left[1 + \frac{1}{Z} \frac{r_B}{R + r_B} \right]$$

El esfuerzo circunferencial no puede ser mayor que el límite de resistencia

$$\sigma_{\theta} \leq \frac{S_y}{F.S.}$$

La porción del eje neutro se lo encuentra por las siguientes ecuaciones:

$$R_n = \frac{APR}{A_m M + N(A - R A_m)}$$

Para encontrar la distancia entre el eje neutro y el eje baricéntrico:

$$y_o = - \frac{ZR}{(Z + 1)}$$

La constante Z es una característica geométrica de la sección, que guarda cierta similitud con el

momento de inercia en la fórmula de las vigas rectas:

$$Z = - \frac{1}{A} \int \frac{y}{R + y} dA$$

La exentricidad se la puede encontrar por la recomendación de Singer, Pág. 462:

$$e = \frac{R}{A} \int \frac{dA}{v}$$

Donde:

F.S.: Factor de seguridad

A : Area de la sección transversal

R : Radio de curvatura

v : Distancia del eje de la curvatura al elemento diferencial considerado.

F : Carga concentrada actuante

N : Componente normal de la carga

Para determinar la longitud en función del área, Searly Fred propone una tabla, Pág 148:

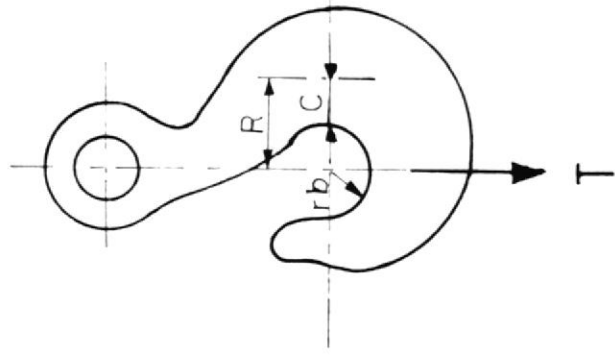
Para la sección G:

$$A = 1.05 b^2$$

$$C = 0.70 b$$

$$I = 0.18 b^4$$

FIGURA N° 10 : EL GANCHO DEL APAREJO ELEVADOR



2.8.3 DIAGRAMA DE FLUJO Y ALGORITMO

PROGRAMA PARA DISEÑAR
GANCHO DEL APAREJO ELEVADOR

$T, S_y, F.S.$
 $K, R/C, \theta$

$$B = R/C$$

$$\text{temp} = \frac{S_y}{F.S.}$$

$$Q = K * B$$

$$H = T \text{ Cos } \theta$$

$$A = \frac{H * [1 + (2.85Q)]}{\text{temp}}$$

$$b = \frac{A}{1.05}$$

$$C = 0.70 b$$

$$R = B * C$$

$$I = 0.18 b^4$$

A

A

$$r_B = R - C$$

$$x_1 = b$$

$$x_2 = 1.24 b$$

$$x_3 = 1.625 b$$

$$x_4 = 0.345 b$$

$$R_1 = 0.256 b$$

$$R_2 = 0.535 b$$

$$A_1 = \frac{I * F.S.}{4}$$

$$r = \frac{A_1}{P_i}$$

PASADOR

B

B

$$M = \frac{T * L}{4}$$

$$D = (M * 12)^2$$

$$E = \frac{32 * 1.8}{\pi * S_y}$$

$$F = D^{0.5}$$

$$d = (E * F)^{0.34}$$

A, b, c, I, R₁, R₂
x₂, x₃, x₄, r_B
At, d

FIN

```

    /*** PROGRAMA PARA DISEÑAR EL GANCHO DEL APAREJO ELEVADOR ***/

#include<stdio.h>
#include<math.h>
#include<ctype.h>
#include "tesis.h"
#define OR ||
#define AND &&
#define Pi 3.1415926

void main()
{
double    T,Sy,FS,K,b,A,c,R,rb,B,temp,r,M,d,P,x,
          D,F,E,X1,X2,X3,A1,Sy1,L,I,R1,R2,Q,hh;

char      Tc[20],Kc[20],Bc[20],Syc[20],Sy1c[20],
          Lc[20],FSc[20],xc[20];

int       k;

clrscr();
do {
    printf("\n Ingrese la capacidad del elevador T en Kips: ");
    gets(Tc);
    T = es_numero(Tc);
    fflush(stdin);
    } while ( (T == 0) );
do {
    printf("\n Ingrese el factor de correccion K: ");
    gets(Kc);
    K = es_numero(Kc);
    fflush(stdin);
    } while ( (K == 0) );
do {
    printf("\n Ingrese el coeficiente B: ");
    gets(Bc);
    B = es_numero(Bc);
    fflush(stdin);
    } while ( (B == 0) );
do {
    printf("\n Ingrese la Resistencia de Fluencia Sy del gancho en
    Kpsi: ");
    gets(Syc);
    Sy = es_numero(Syc);
    fflush(stdin);
    } while ( (Sy == 0) );
do {
    printf("\n Ingrese la Resistencia de Fluencia del pasador Sy1 en
    Kpsi: ");
    gets(Sy1c);
    Sy1 = es_numero(Sy1c);
    fflush(stdin);

```

```

    } while ( (Sy1 == 0) );
do {
    printf("\n Ingrese la longitud del pasador L en pies: ");
    gets(Lc);
    L = es_numero(Lc);
    fflush(stdin);
    } while ( (L == 0) );
do {
    printf("\n Ingrese un factor de seguridad FS: ");
    gets(FSc);
    FS = es_numero(FSc);
    fflush(stdin);
    } while ( (FS == 0) );
do {
    printf("\n Ingrese el angulo de inclinacion de la seccion
deseada x en grados: \n");
    printf(" En el rango [0,135]");
    printf(" ==>:");
    gets(xc);
    x = es_numero(xc);
    fflush(stdin);
    } while ( (x > 135.0) OR (x < 0) );

```

/** SECCION TRAPEZOIDAL **/

```

temp = (Sy/FS);  /**Kpsi**/
Q = (K*B);
hh = T*cos(x);
A = fabs( (T*cos(x)*(1.0+(2.85*Q)))/temp );
b = sqrt(A/1.05);
c = fabs(0.70*b);
I = fabs(0.18*pow((b),4.0));
R = fabs(B*c);
rb = fabs(R-c);
X1 = (1.625*b);
X2 = (1.24*b);
X3 = (0.345*b);
R1 = (0.256*b);
R2 = (0.535*b);

```

/** DIAMETRO DEL PASADOR **/

```

M = (((T*L)/4.0)*1000.0);  /**Lb por pie**/
D = pow((M*12.0),2.0);
E = ((32.0*1.8)/(Pi*Sy*1000.0));
F = pow((D),0.5);
d = pow((E*F),0.34);
clrscr();

```

```

printf("\n EL AREA OPTIMA A EN PULGADA CUADRADA ES: %lf",A);
printf("\n EL ANCHO MAXIMO b DE LA SECCION TRANSVERSAL EN PULGADAS ES:
%lf",b);
printf("\n LA DISTANCIA c DE LA FIBRA INTERIOR AL EJE NEUTRO ES EN

```

```
PULGADAS: %1f",c);
printf("\n EL MOMENTO DE INERCIA I EN PULGADAS A LA CUARTA ES :
%1f",I);
printf("\n EL RADIO MENOR R1 EN PULGADAS ES: %1f",R1);
printf("\n EL RADIO MAYOR R2 EN PULGADAS ES: %1f",R2);
printf("\n LA LONGITUD X1 DE LA SECCION TRANSVERSAL EN PULGADAS ES:
%1f",X1);
printf("\n LA LONGITUD X2 DE LA SECCION TRAPEZOIDAL EN PULGADAS ES:
%1f",X2);
printf("\n LA LONGITUD X3 DEL CASQUETE ELIPTICO EN PULGADAS ES:
%1f",X3);
printf("\n EL RADIO INTERIOR DEL GANCHO rb EN PULGADAS ES: %1f",rb);
printf("\n EL DIAMETRO DEL PASADOR d EN PULGADAS ES : %1f\n",d);
gotoxy(18,23);printf("[Presione cualquier tecla para continuar]: ");
getch();
}
```

2.9 LAS UNIONES SOLDADAS.

El tipo más extendido de unión de las estructuras metálicas es la soldada. Es una unión inseparable ejecutada por soldadura. Los tipos básicos de uniones soldadas son:

- A tope
- En ángulo
- En T
- A solape

La parte de la unión soldada que se forma con el metal fundido durante la soldadura y, posteriormente, solidificado, se llama costura soldada.

Antes de soldar, los bordes de las piezas a unir se cortan frecuentemente bajo un ángulo recto, u oblicuo. En dependencia del espesor del metal y del tipo de soldadura, el corte de los bordes puede ser unilateral, bilateral o en forma de copa. Al haber biseles, como regla, se deja el testigo para evitar la quemadura del metal al soldarlo (10).

En este proyecto se diseñara tres uniones soldadas:

- La placa de alojamiento del rodamiento del eje de las ruedas del puente, con el perfil del soporte lateral.

(10) Producción de estructuras metálicas. Mir 1982, Pág. 84.

Esta unión será diseñada para torsión pura, que resulta por el efecto del momento de rotación de la potencia de transmisión.

Se considera esta situación para un eventual caso que falle el rodamiento, y de alguna manera la pista externa gire, y con él la placa. Aunque esto resulta difícil que suceda, el diseño se basas en esta consideración.

- La placa de alojamiento del rodamiento del eje de las ruedas del carro del trole, con el perfil que forma el mismo.

También se diseñara para las mismas condiciones que la unión anterior.

- Las uniones de los diferentes tramos que forman la viga porta-grúa.

El diseño se hará para cortante y flexión, producto de las fuerzas y momentos que se producen en las columnas.

Las uniones se harán para tramos de 6 metros, esto nos da seguridad, ya que en un caso de falla de la soldadura, la viga no se verá afectada por estar descansando sobre las columnas.

Todas las soldaduras se harán del tipo de filete.

2.9.1 RECOMENDACIONES DEL MANUAL DEL A.I.S.C. PARA DETERMINAR LAS DIMENSIONES.

El manual del A.I.S.C. recomienda que para una soldadura de filete:

$$F_t = 0.6 F_y \quad \text{Tensión o compresión}$$

$$F_v = 0.3 F_u \quad \text{Cortante}$$

Recomendaciones para el tamaño mínimo de soldadura de filete: A.I.S.C. 1.17.2.

Recomendaciones para el tamaño máximo de soldadura de filete: A.I.S.C. 1.17.3.

Longitud de soldadura del filete:

A.I.S.C. 1.17.4.

Recomendaciones para longitudes intermitentes:

A.I.S.C. 1.17.5.

El manual del A.I.S.C. recomienda que:

El esfuerzo actual debe ser menor o igual al esfuerzo admisible

$$f_r \leq f_w$$

Los esfuerzos horizontal y vertical para torsión son:

$$f_h = \frac{\delta * v}{J}$$

$$f_v = \frac{\delta * h}{J}$$

Esfuerzo cortante directo:

$$f_s = \frac{V}{A_{\text{soldadura}}}$$

Esfuerzo de flexión:

$$f_b = \frac{MC}{I}$$

2.9.2 DIAGRAMA DE FLUJO Y ALGORITMO

PROGRAMA PARA DISEÑAR
LAS UNIONES SOLDADAS

$F_y, F_u, T_m,$
 L_s, A_s, t_e, h
 L, a

TORSION PURA

$$X_c = \frac{L_s}{2}$$

$$J = \frac{(2L_s)^3}{6}$$

$$A_s = 1.414 t_e (2L_s)$$

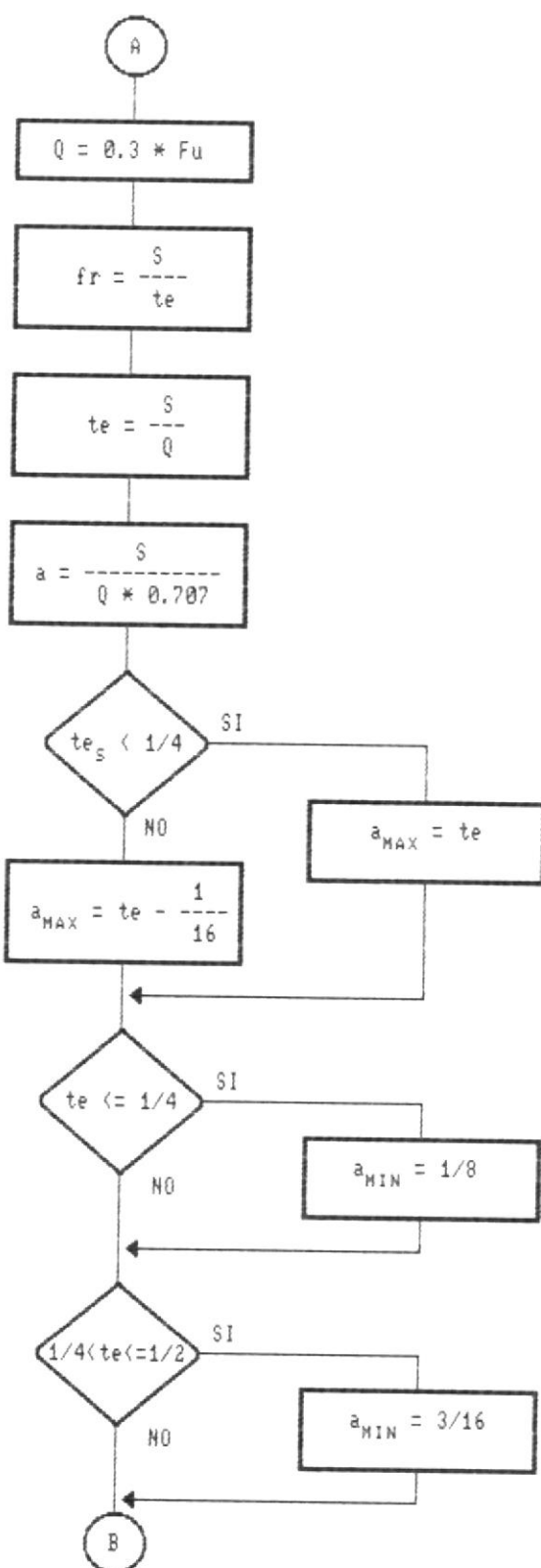
$$f_v = \frac{(T_m * L_s/2)}{J}$$

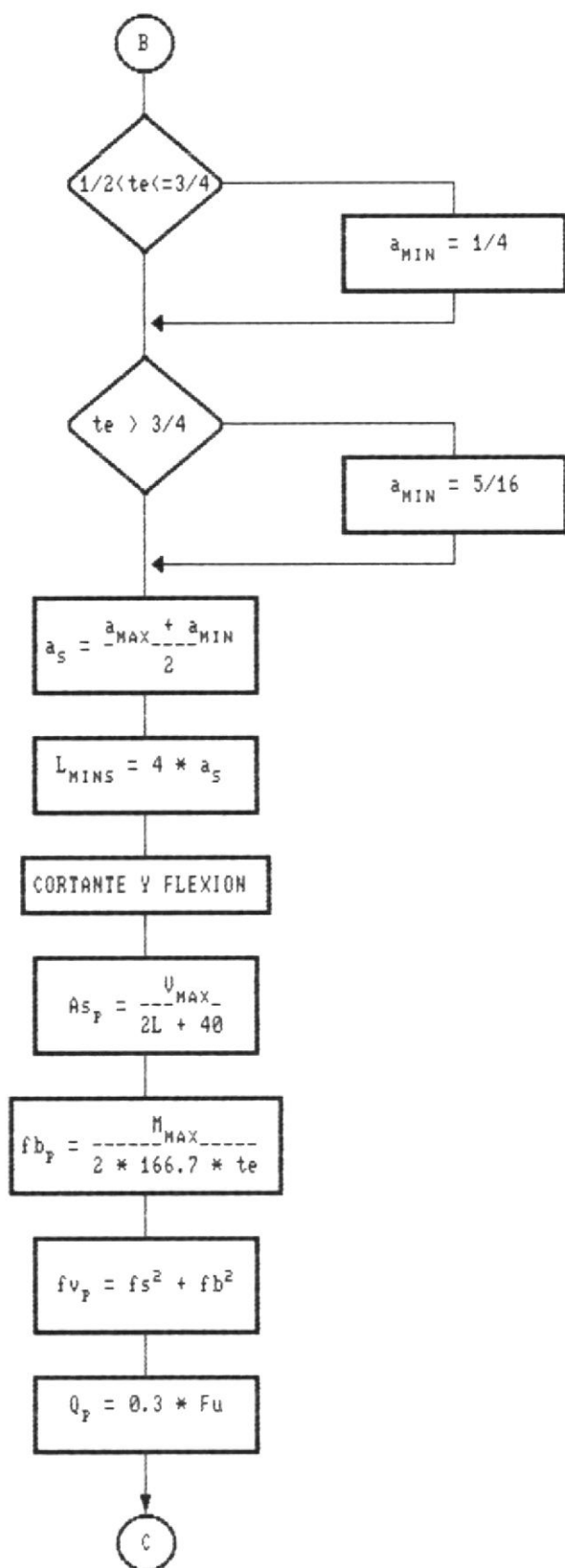
$$f_n = \frac{(T_m * L_s/2)}{J}$$

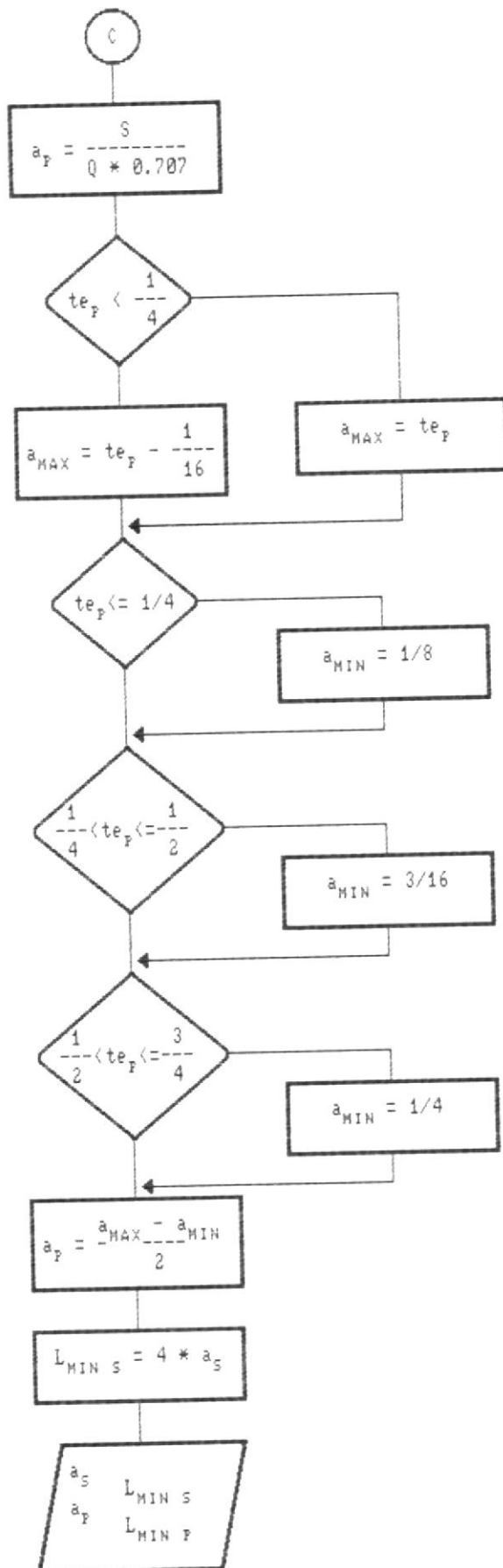
$$f_r = f_v^2 + f_n^2$$

A

Fu: Resistencia ultima a la
tension del material del
electrodo Ksi







```
/** PROGRAMA PARA DISEÑAR LOS PERNOS DE CONEXION ***/
```

```
#include<stdio.h>
#include<math.h>
#include<ctype.h>
#include "tesis.h"
#define OR    ||
#define AND  &&
#define Pi   3.1415829

void main()
{
double    T,T1,T2,T3,T4,L1,L2,L3,L4,L5,wr,w1,w2,
          w3,P,Fy,Fu,Atc,Att,Ata,dt,Auc,Asc,Ast,
          Asa,ds,A1,A2,tt,ts,d1,d2,d3,Aut,Aua,
          dua,A3;

char      c1,
          Tc[20],Pc[20],L1c[20],L2c[20],L3c[20],
          w1c[20],w2c[20],w3c[20],Fyc[20],Fuc[20],
          ttc[20],tsc[20],wrc[20];

clrscr();
do {
    printf("\n Ingrese la capacidad del elevador T en Kips: ");
    gets(Tc);
    T = es_numero(Tc);
    fflush(stdin);
} while ( (T == 0) );
do {
    printf("\n Ingrese el peso del motor y bobina P en Kips: ");
    gets(Pc);
    P = es_numero(Pc);
    fflush(stdin);
} while ( (P == 0) );
do {
    printf("\n Ingrese la longitud del carro del trole L1 en pies:
    ");
    gets(L1c);
    L1 = es_numero(L1c);
    fflush(stdin);
} while ( (L1 == 0) );
do {
    printf("\n Ingrese el peso del carro del trole w1 en Kips/pie:
    ");
    gets(w1c);
    w1 = es_numero(w1c);
    fflush(stdin);
} while ( (w1 == 0) );
do {
    printf("\n Ingrese la longitud de la viga principal L2 en pies:
    ");
```

```

    gets(L2c);
    L2 = es_numero(L2c);
    fflush(stdin);
    } while ( (L2 == 0) );
do {
    printf("\n Ingrese el peso de la viga principal w2 en Kips/pie:
    ");
    gets(w2c);
    w2 = es_numero(w2c);
    fflush(stdin);
    } while ( (w2 == 0) );
do {
    printf("\n Ingrese la longitud del soporte lateral L3 en pies:
    ");
    gets(L3c);
    L3 = es_numero(L3c);
    fflush(stdin);
    } while ( (L3 == 0) );
do {
    printf("\n Ingrese el peso del soporte lateral w3 en Kips/pie:
    ");
    gets(w3c);
    w3 = es_numero(w3c);
    fflush(stdin);
    } while ( (w3 == 0) );
do{
    printf("\n Ingrese el peso del riel wr en Kips/pie: ");
    gets(wrc);
    wr = es_numero(wrc);
    fflush(stdin);
    } while ( (wr == 0) );
do {
    printf("\n Ingrese la Resistencia de Fluencia Fy del perno en
    Kpsi: ");
    gets(Fyc);
    Fy = es_numero(Fyc);
    fflush(stdin);
    } while ( (Fy == 0) );
do {
    printf("\n Ingrese la Resistencia Ultima Fu del perno en Kpsi:
    ");
    gets(Fuc);
    Fu = es_numero(Fuc);
    fflush(stdin);
    } while ( (Fu == 0) );
do {
    printf("\n Ingrese el espesor de la placa menor tt en pulgadas:
    ");
    gets(ttc);
    tt = es_numero(ttc);
    fflush(stdin);
    } while ( (tt == 0) );
do{
    printf("\n Ingrese el espesor de la placa menor ts en pulgadas:
    ");

```

```

    gets(tsc);
    ts = es_numero(tsc);
    fflush(stdin);
} while ( (ts == 0) );

T1 = ((T+F)/4.0);
T2 = (T1+(w1*L1));
T3 = (T2+((wr*L2)/2.0));
T4 = ((T3/2.0)+((w3*L3)/2.0));

    /*** CARRO DEL TROLE ***/

    /*** DISEÑO POR CORTANTE ***/

Atc = (T1/(0.40*Fy));

    /*** DISEÑO POR TENSION PURA ***/

Att = (T1/(0.6*Fy));

    /*** DISEÑO POR AFLASTAMIENTO ***/

dt = (T1/(0.75*Fy*tt));
Ata = ((Pi*pow((dt),2.0))/4.0);

A1 = Atc;
if(Att > A1)
    A1 = Att;
if(Ata > A1)
    A1 = Ata;

    /*** UNION DE LA VIGA PRINCIPAL CON EL SOPORTE ***/

    /*** DISEÑO POR TENSION PURA ***/

Auc = (T3/(0.6*Fy));

    /*** DISEÑO POR CORTANTE ***/

Aut = ((T3*0.18)/(0.40*Fy));

    /*** DISEÑO POR AFLASTAMIENTO ***/

dua = ((T3*0.18)/(0.75*Fy*tt));
Aua = ((Pi*pow((dua),2.0))/4.0);

A3 = Auc;
if(Aut > A3)
    A3 = Aut;

```

```

if(Aua > A3)
    A3 = Aua;

    /*** SOPORTE LATERAL ***/

    /*** DISEÑO POR CORTANTE ***/

Asc = ((2.0*T4)/(0.40*Fy));

    /*** DISEÑO POR TENSION PURA ***/

Ast = ((2.0*T4)/(0.6*Fy));

    /*** DISEÑO POR AFLASTAMIENTO ***/

ds = ((2.0*T4)/(0.75*Fy*ts));
Asa = ((Pi*pow((ds),2.0))/4.0);

A2 = Asc;
if(Ast > A2)
    A2 = Ast;
if(Asa > A2)
    A2 = (Asa);

d1 = sqrt(4.0*A1/Pi);
d2 = sqrt(4.0*A2/Pi);
d3 = sqrt(4.0*A3/Pi);
clrscr();

printf("\n EL DIAMETRO DE LOS PERNOS PARA EL CARRO DEL TROLE EN
PULGADAS ES: %1f",d1);
printf("\n EL DIAMETRO DE LOS PERNOS PARA LA CONEXION VIGA
PRINCIPAL-SOPORTE EN PULGADAS ES: %1f",d3);
printf("\n EL DIAMETRO DE LOS PERNOS PARA EL SOPORTE LATERA EN
PULGADAS ES: %1f",d2);
printf("\n EL DIAMETRO DE LOS PERNOS DE LOS RIGIDIZADORES DE LA
VIGA PRINCIPAL EN PULGADAS ES: %1f\n",d3);
gotoxy(18,23);printf(" [Presione cualquier tecla para
continuar]: ");
getch();
}

```

2.10 LOS EJES DE LAS RUEDAS Y CHAVETAS.

2.10.1 ANALISIS DE DEFORMACIONES Y DETERMINACION DE MAXIMOS.

En cualquier eje rotatorio cargados por momentos estacionarios de flexión y torsión actuarán esfuerzos por flexión completamente invertida, debido a la rotación del árbol, pero el esfuerzo torsional permanecerá estable.

Los ejes de las ruedas están sometidos a este tipo de esfuerzo, debido a la acción de las fuerzas que actúan sobre él, y por el momento de torsión que resulta de la transmisión de potencia.

Las cargas concentradas actuantes son:

- La reacción que se produce en el soporte lateral, por efecto de las cargas que actúan sobre este más el peso propio de él.
- La componente radial de la fuerza que se transmite por medio del engranaje.
- El peso propio de la rueda.

El torque se produce por efecto de la componente tangencial de la fuerza que se

transmite por medio del engranaje.

Los ejes serán considerados como simplemente apoyados y homogéneos.

La deformación elástica que se produce en el plano vertical, y en el centro del eje es (11):

$$\delta = \frac{1}{48} \left(\frac{T_t + W_{RAD}}{EI} \right)$$

Donde:

$$T_t = 2T_4 + T_s$$

T_s : peso de la rueda

$$W_{RAD} = wt * \tan(\theta)$$

W_{RAD} = componente radial de la fuerza en el engranaje.

θ : ángulo de presión (20°)

$$T_4 = \frac{T_3}{2} + \frac{W_3 L_3}{2}$$

$$T_3 = T_2 + \left[\frac{(W_R + W_2)L_2}{2} \right]$$

$$T_2 = T_1 + W_1 L_1$$

$$T_1 = \frac{T + F}{4}$$

(11) Resistencia de materiales. Singer 1982, Pág. 199.

T: capacidad del elevador

P: peso del motor y bobina

El máximo momento de diseño se encuentra sumando los momentos máximos que se producen en cada plano en que actúan las cargas:

$$M_{max} = \left[\left(\frac{Tt + Wt}{4} \right) L_d \right]^2 + \left(\frac{W_R}{4} L_d \right)^2$$

2.10.2 RECOMENDACIONES PARA DETERMINAR EL DIAMETRO.

En cualquier eje rotatorio cargado por momentos estacionarios de flexión y torsión actuarán esfuerzos por flexión completamente invertida, debido a la rotación del árbol, pero el esfuerzo torsional permanecerá estable.

El eje de las ruedas del puente-grúa será diseñado considerando fatiga, y se determinarán los diámetros tanto de la sección donde va alojada la rueda, de la sección del cojinete y de la sección donde irá acoplado el engranaje, que transmitirá la potencia y movimiento.

Para determinar el factor de seguridad:
Shigley, Pág. 803.

$$\frac{1}{n} = \frac{32}{\pi d^3} \left[\left(\frac{M_a}{S_c} \right)^2 + \left(\frac{T_m}{S_{ut}} \right)^2 \right]^* \quad 18.27$$

Para determinar el factor el límite de resistencia a la fatiga: Shigley, Pág. 317.

$$S_c = K_a * K_b * K_c * K_d * K_e * S_c \quad 7.13$$

Factores que modifican el límite de resistencia a la fatiga: Shigley 7.13.

Factor de superficie K_a Pág. 3.17.

Factor de tamaño K_b Pág. 3.18.

Factor de carga K_c Pág. 3.20.

Factor de temperatura K_d Pág. 3.20.

Resistencia última a la tensión:

Shigley A.20 Pág. 859, Tabla A.20.

Para diseñar el diámetro para fatiga:

Shigley 18.28.

$$d = \left\{ \frac{32n}{\pi} \left[\left(\frac{K_f M_a}{S_c} \right)^2 + \left(\frac{T_m}{S_{ut}} \right)^2 \right]^{1/2} \right\}^{1/3} \quad 18.28$$

Las chavetas son elementos que se usan para poder transmitir momento de rotación desde un eje hasta el elemento que soporta dicho árbol.

Existen varios tipos de chavetas, pero para nuestro diseño se elegirá una de sección cuadrada (también disponible de sección rectangular), debido a que no tiene una velocidad muy elevada.

La longitud de la chaveta se basa en la longitud del cubo y la carga torsional por transmitir:

Shigley 8.17 Pág. 415.

Tamaño de la chaveta y profundidad del chavetero: Shigley, "Unthreaded Fasteners", Cap. 22, Shigley y Mischke (E.D.S.), Standard

Handbook of Machine Design, McGraw-Hill.

Shigley 8.17 Tabla 8.15 Pág. 416.

Resistencia de fluencia al cortante por la teoría de la energía de distorsión:

Shigley 6.9 Pág. 282.

$S_{sy} = 0.5S_y$ Teoría esfuerzo cortante máximo

$S_{sy} = 0.577S_y$ Teoría energía de distorsión

Una recomendación importante, es que la longitud del cubo de un engranaje por lo general es mayor que el diámetro del eje, para que haya estabilidad.

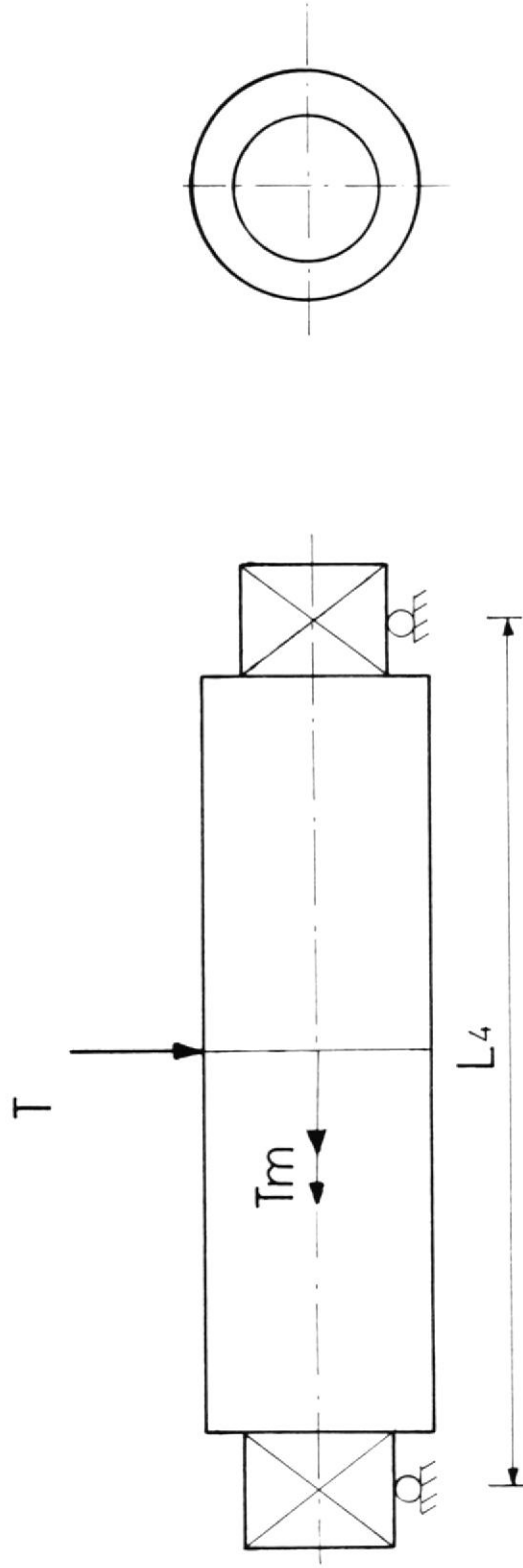
Si la chaveta tuviera una longitud igual a la del cubo, tendría por consiguiente amplia resistencia.

Shigley 8.17 Pág. 420.

Longitud de la chaveta:

$$l = \frac{2nF}{S_y t}$$

FIGURA Nº11: ELEJE DE LAS RUEDAS SOMETIDO A FLEXION Y TORSION



2.10.3 DIAGRAMA DE FLUJO Y ALGORITMO

DISEÑO DE LOS
EJES DE LAS RUEDAS

$$T_4, T_5, L_4, L_5$$
$$K_a, K_b, K_c, K_d, K_e$$
$$K_f, n$$

$$S_e = K_a * K_b * K_c * K_d * K_e * K_f$$

$$w_t = \frac{33,000 H}{V_e}$$

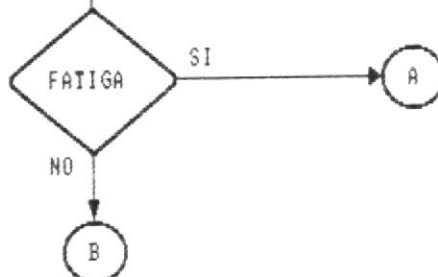
$$T_m = \frac{w_t * d_{\text{ENGRANE}}}{2}$$

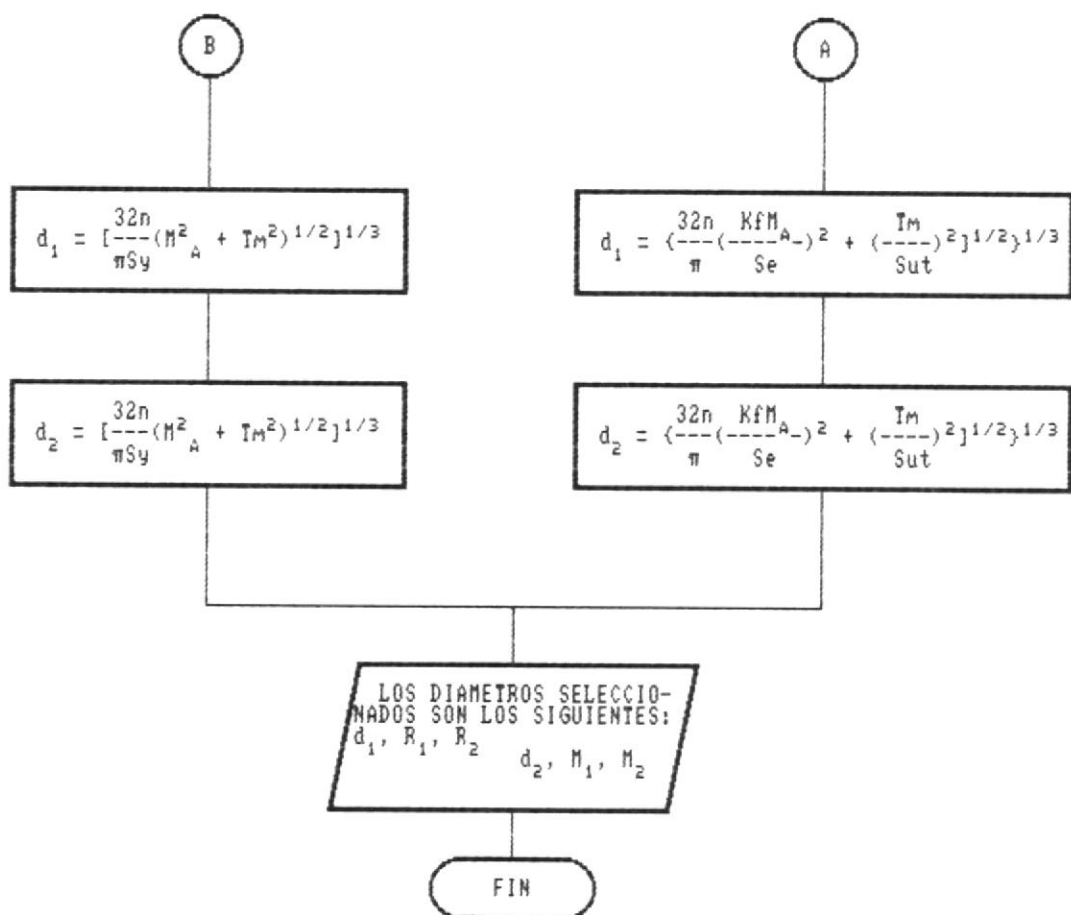
$$R_1 = \frac{T_t + w_t}{2}$$

$$R_2 = \frac{T_t + w_t}{2}$$

$$M_1 = R_1 * \frac{L_4}{2}$$

$$M_2 = R_1 * \frac{L_4}{2}$$





PROGRAMA PARA DISEÑAR
LAS CHAQUETAS:

INT [9] [5]

I, H, n₁, n₂,
d₃, s_y

I = 1

J = 1

C = 1

Z[I][J] <=
d₂ <
Z[9][J]

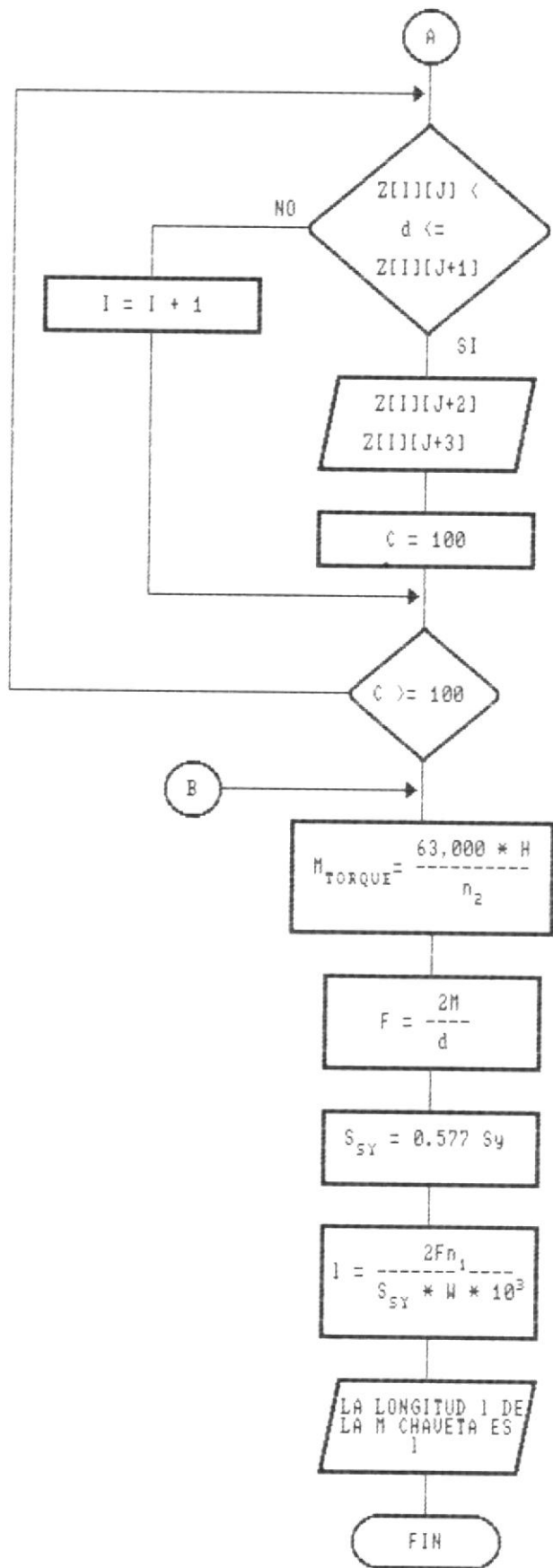
SI

A

NO

EXTRAPOLACION

B



```

                /**PROGRAMA PARA DISEÑAR EL EJE DE LAS RUEDAS***/
                /** Y CHAVETAS***/

#include <stdio.h>
#include <math.h>
#include <ctype.h>
#include "tesis.h"
#define OR ||
#define AND &&
#define Pi 3.141592654

/**variables globales***/

int salir = 1;

/**FUNCION PRINCIPAL***/

void main()
{
double      T,T1,w1,L1,T2,L2,w2,wr,T3,L3,w3,T5,Tm,Tt,l,P,F,Fu,
            Ht,Sut,Sy,L4,T4,Ka,L5,d1,d2,r,Se,Se1,A,B,n1,D1,Dt2,
            Dp2,Mt,Mp,w,h,p,Ssy,Wtt,Wrt,Tmt,dt,Wtp,Wrp,Tmp,dp,

Ttp,Ttt,amaxt,amint,amaxp,aminp,at,ap,Lt,Lp,tet,tep,
v1,D2,Hs,v2;
char        C1,
            Tc[20],Fc[20],wrc[20],wic[20],L1c[20],L2c[20],
            w2c[20],Htc[20],Hsc[20],L3c[20],w3c[20],Sutc[20],
            Syc[20],Fyc[20],L4c[20],Ac[20],Bc[20],D1c[20],
            D2c[20],Dt2c[20],Dp2c[20],n1c[20],Fuc[20];

            int      I,J,C;
            double   z[9][5];

            clrscr();
            for(I=0;I<9;++I) {
            for(J=0;J<5;++J)
            z[I][J] = 0;
            }
            z[0][0]=0.3125;
            z[0][1]=0.4375;
            z[0][2]=0.09375;
            z[0][3]=0.09375;
            z[0][4]=0.046875;
            z[1][0]=0.4375;
            z[1][1]=0.5625;
            z[1][2]=0.125;
            z[1][3]=0.09375;
            z[1][4]=0.046875;
            z[2][0]=0.5625;
            z[2][1]=0.875;
            z[2][2]=0.1875;
            z[2][3]=0.125;

```

```

z[2][4]=0.0625;
z[3][0]=0.875;
z[3][1]=1.25;
z[3][2]=0.25;
z[3][3]=0.1875;
z[3][4]=0.09375;
z[4][0]=1.25;
z[4][1]=1.375;
z[4][2]=0.3125;
z[4][3]=0.25;
z[4][4]=0.125;
z[5][0]=1.375;
z[5][1]=1.75;
z[5][2]=0.375;
z[5][3]=0.25;
z[5][4]=0.125;
z[6][0]=1.75;
z[6][1]=2.25;
z[6][2]=0.5;
z[6][3]=0.375;
z[6][4]=0.1875;
z[7][0]=2.25;
z[7][1]=2.75;
z[7][2]=0.625;
z[7][3]=0.4375;
z[7][4]=0.21875;
z[8][0]=2.75;
z[8][1]=3.25;
z[8][2]=0.75;
z[8][3]=0.5;
z[8][4]=0.25;

do {
    printf("\n Ingrese la capacidad del elevador T en Kips:
");
    gets(Tc);
    T = es_numero(Tc);
    fflush(stdin);
} while ( (T == 0) );

do {
    printf("\n Ingrese el peso del motor y bobina P en Kips:
");
    gets(Pc);
    P = es_numero(Pc);
    fflush(stdin);
} while ( (P == 0) );

do {
    printf("\n Ingrese el peso del riel wr en Kips/pulgada
cuadrada: ");
    gets(wrc);
    wr = es_numero(wrc);
    fflush(stdin);
} while ( (wr == 0) );

do {
    printf("\n Ingrese el peso del carro para el trole wi en

```

```

        Kips/pie: ");
        gets(w1c);
        w1 = es_numero(w1c);
        fflush(stdin);
    } while ( (w1 == 0) );
do {
    printf("\n Ingrese la longitud del carro para el trole L1
    en pies: ");
    gets(L1c);
    L1 = es_numero(L1c);
    fflush(stdin);
    } while ( (L1 == 0) );
do {
    printf("\n Ingrese la longitud de la viga principal L2 en
    pies: ");
    gets(L2c);
    L2 = es_numero(L2c);
    fflush(stdin);
    } while ( (L2 == 0) );
do {
    printf("\n Ingrese el peso de la viga principal w2 en
    Kips/pie: ");
    gets(w2c);
    w2 = es_numero(w2c);
    fflush(stdin);
    } while ( (w2 == 0) );
do {
    printf("\n Ingrese la potencia del motor Ht para el trole
    en hp: ");
    gets(Htc);
    Ht = es_numero(Htc);
    fflush(stdin);
    } while ( (Ht == 0) );
do {
    printf("\n Ingrese la potencia del motor Hs para el
    soporte en hp: ");
    gets(Hsc);
    Hs = es_numero(Hsc);
    fflush(stdin);
    } while ( (Hs == 0) );
do {
    printf("\n Ingrese la longitud del soporte lateral L3 en
    pies: ");
    gets(L3c);
    L3 = es_numero(L3c);
    fflush(stdin);
    } while ( (L3 == 0) );
do {
    printf("\n Ingrese el peso del soporte lateral w3
    Kips/pie: ");
    gets(w3c);
    w3 = es_numero(w3c);
    fflush(stdin);
    } while ( (w3 == 0) );
do{

```

```

printf("\n Ingrese la resistencia ultima a la tension Sut
en kpsi: ");
gets(Sutc);
Sut = es_numero(Sutc);
fflush(stdin);
} while ( (Sut == 0) );
do {
printf("\n Ingrese la resistencia de fluencia Sy en Kpsi:
");
gets(Syc);
Sy = es_numero(Syc);
fflush(stdin);
} while ( (Sy == 0) );
do {
printf("\n Ingrese la resistencia ultima a la tension del
electrodo Fu en Kips/pulgada cuadrada: ");
gets(Fuc);
Fu = es_numero(Fuc);
fflush(stdin);
} while ( (Fu == 0) );
do {
printf("\n Ingrese la longitud del eje entre cojinetes L4
en pies: ");
gets(L4c);
L4 = es_numero(L4c);
fflush(stdin);
} while ( (L4 == 0) );
do {
printf("\n Ingrese el factor A para el acabado de
superficie en Kpsi: ");
gets(Ac);
A = es_numero(Ac);
fflush(stdin);
} while ( (A == 0) );
do {
printf("\n ingrese el exponente B (positivo) para el
acabado de superficie: ");
gets(Bc);
B = es_numero(Bc);
fflush(stdin);
} while ( (B == 0) );
do {
printf("\n Ingrese el radio del pinon D1 del trole en
pulgadas: ");
gets(D1c);
D1 = es_numero(D1c);
fflush(stdin);
} while ( (D1 == 0) );
do {
printf("\n Ingrese el radio del pinon D2 del soporte en
pulgadas: ");
gets(D2c);
D2 = es_numero(D2c);
fflush(stdin);
} while ( (D2 == 0) );

```

```

do {
    printf("\n Ingrese el diametro del engranaje Dt2 del carro
del trole en pulgadas: ");
    gets(Dt2c);
    Dt2 = es_numero(Dt2c);
    fflush(stdin);
} while ( (Dt2 == 0) );

do {
    printf("\n Ingrese el diametro del engranaje Dp2 del
puente en pulgadas: ");
    gets(Dp2c);
    Dp2 = es_numero(Dp2c);
    fflush(stdin);
} while ( (Dp2 == 0) );

do {
    printf("\n Ingrese el numero de revoluciones de entrada n1
del carro del trole y para el soporte: ");
    gets(n1c);
    n1 = es_numero(n1c);
    fflush(stdin);
} while ( (n1 == 0) );

Se1 = (0.504*Sut);
v1 = ((Pi*D1*n1)/(12.0));
v2 = ((Pi*D2*n1)/(12.0));
Lt = 4.0;
Lp = 9.0;
T1 = ((T+P)/4.0);
T2 = (T1 + (w1*L1));
T3 = (T2 + ((wr*L2)/2.0) + ((w2*L2)/2.0) );
T4 = ((T3/2.0) + ((w3*L3)/2.0)); /** Carga puntual en Kips ***/
Ttp = (2.0*T4*1000.0); /** Carga total concentrada en Lb ***/
Ttt = (2.0*T1*1000.0);
Ka = A/pow(Sut,B);
Se = (Ka*Se1);
Wtt = ( (33000.0*Ht*1.2)/v1); /** Carga transmitida en lb ***/
Wtp = ( (33000.0*Hs*1.2)/v2);
Wrt = (Wtt*tan(20));
Wrp = (Wtp*tan(20));
Tmt = Wtt*Dt2/2.0; /** Lb por pulgada ***/
Tmp = Wtp*Dp2/2.0;/** Momento de torsion en Lb por pulgada ***/
Mt = sqrt(pow(((Wtt+Ttt)*(L4*12.0/4.0)),2.0) +
pow((Wrt*L4*12.0/4.0),2.0)); /** lb por pulg***/
Mp = sqrt(pow( ((Wtp+Ttp)*(L4*12.0/4.0)),2.0) +
pow((Wrp*L4*12.0/4.0),2.0) );
printf("\n\n El momento maximo para el trole es: %lf",Mt);
printf("\n El momento maximo para el soporte es: %lf\n",Mp);
printf("\n Diseno para fatiga ? S/N ==> ");
do {
    C1 = getchar();
    switch (C1){
    case 'S'/'s': {
        dt = pow( ((32.0*1.8)/Pi)*pow((pow((Mt/
(Se*1000.0)),2.0) + pow((Tmt/(Sut*1000.0)),
2.0)),0.5),0.33333) ;

```

```

        dp = pow(((32.0*1.8)/Pi)*pow((pow((Mp/(Se*1000.0)),2.0) + pow((Tmp/(Sut*1000.0)),2.0)),0.5)),0.33333) ;
        salir = 0 ;
        break;
    }

    case 'N':'n': {
        dt = pow(((32.0*1.8)/(Pi*Sy*1000.0))*pow((pow((Mt),2.0)+pow((Tmt),2.0)),0.5)),0.33333) ;
        dp = pow(((32.0*1.8)/(Pi*Sy*1000.0))*pow((pow((Mp),2.0)+pow((Tmp),2.0)),0.5)),0.33333) ;
        salir = 0 ;
        break;
    }

    default: {
        printf("\n Ingrese S/N ==> ");
        break;
    }
}
} while(salir);

    /**SELECCION DE LA CHAVETA***/

I = 0;
J = 0;
C = 0;

if( dp > z[I][J] && dp < z[8][J] ) {
    do {
        if( dp > z[I][J] && dp <= z[I][J+1] ) {
            w = z[I][J+2];
            h = z[I][J+3];
            p = z[I][J+4];
            C = 100.0;
            printf("\n valores directos: ");
        }
        else {
            ++I;
        }
    } while ( C < 100.0 );
}
else {
    w = ( ((( z[8][J+2] - z[0][J+2]) / (z[8][0] - z[0][0])) * (z[0][J+2] - z[0][0]) ) + dp );
    h = ( ((( z[8][J+3] - z[0][J+3]) / (z[8][0] - z[0][0])) * (z[0][J+3] - z[0][0]) ) + dp );
    p = ( ((( z[8][J+4] - z[0][J+4]) / (z[8][0] - z[0][0])) * (z[0][J+4] - z[0][0]) ) + dp );
    C = 100.0;
}

```

```

        printf("\n Extrapolo: ");
        printf("\n Extrapolo: ");
    }
    Tm = ((63000.0*Hs*1.2)/n1);
    F = ((2.0*Tm)/dt);  /*** lb ***/
    Ssy = (0.577*Sy);  /*** Kpsi ***/
    l = ( (2.8*F)/(Ssy*w*1000.0) );  /*** pulg ***/

        /*** DISEÑO DE LAS UNIONES SOLDADAS ***/

    tet = ((0.26*Tmt)/(pow((Lt),2.0)*0.3*Fu*1000.0));
    tep = ((0.26*Tmp)/(pow((Lp),2.0)*0.3*Fu*1000.0));
    at = (tet/0.707);
    ap = (tep/0.707);
    if(tet < 0.25)
        amaxt = tet;
    else
        amaxt = fabs(tet-0.0625);
        if(tep < 0.25)
            amaxp = tep;
        else
            amaxp = fabs(tep-0.0625);
            if(tet <= 0.25)
                amint = 0.125;
            if((tet > 0.25) && (tet <= 0.5))
                amint = 0.1875;
            if((tet > 0.5) && (tet <= 0.75))
                amint = 0.25;
            if(tet > 0.75)
                amint = 0.3125;
            if(tep <= 0.25)
                aminp = 0.125;
            if((tep > 0.25) && (tep <= 0.5))
                aminp = 0.1875;
            if((tep > 0.5) && (tep <= 0.75));
                aminp = 0.25;
            if(tep > 0.75)
                aminp = 0.1875;
            at = fabs((amaxt-amint)/2.0);
            ap = fabs((amaxp-aminp)/2.0);
            Lt = 4.0*at;
            Lp = 4.0*ap;
            clrscr();

        printf("\n* EL DIAMETRO DEL EJE (dt) PARA EL CARRO
        DEL TROLE ES EN PULGADAS\n ");
        printf("===> %1f\n", dt);
        printf("\n* EL DIAMETRO DEL EJE (dp) PARA EL SOPORTE
        LATERAL ES EN PULGADAS \n ");
        printf("===> %1f\n", dp);
        printf("\n* EL ANCHO DE LA CHAVETA (w) SELECCIONADA
        ES EN PULGADAS\n ");
        printf("===> %1f\n", w);
        printf("\n* LA ALTURA DE LA CHAVETA      (h)

```

```
SELECCIONADA ES EN FULGADAS \n");
printf("===> %lf\n", h);
printf("\n* LA PROFUNDIDAD DEL CHAVETERO (p)
SELECCIONADA ES EN FULGADAS \n");
printf("===> %lf\n", p);
printf("\n* LA LONGITUD DE LA CHAVETA (l) ES EN
FULGADAS \n");
printf("===> %lf\n", l);
printf("\n* LAS DIMENSIONES DE LA SOLDADURA PARA LA
FLACA DEL CARRO DEL TROLE SON: ");
printf("\n Lado (pulg) at = %lf",at);
printf(",Longitud mínima (pulg) Lt = %lf\n",Lt);
printf("* LAS DIMENSIONES DE LA SOLDADURA PARA EL
SOPORTE LATRAL SON: ");
printf("\n Lado (pulg) ap = %lf",ap);
printf(",Longitud mínima (pulg) Lp = %lf\n",Lp);
gotoxy(20,25);printf("[Presione cualquier tecla para
continuar]:");
getch();
}
```

2.11 SELECCION DE LOS RODAMIENTOS PARA LOS EJES DE LAS RUEDAS.

Los rodamientos que se emplearan para los ejes, están sometidos a cargas radiales y velocidades relativamente bajas.

Para cumplir estas condiciones se pueden emplear rodamientos rígidos a bolas como de rodillos cilíndricos, aunque estos últimos se utilizan más bien para velocidades elevadas.

El catálogo de rodamientos de la NTN, sugiere encontrar la capacidad de carga, para entrar a las tablas y seleccionar el más óptimo. La capacidad de carga se la encuentra con la ecuación (12):

$$C_R = F_D \left\{ \frac{L_D \cdot n_D}{L_R \cdot n_R} \right\}^{1/a} \left[0.02 + 4.439 [\ln(1/R)]^{1/1.483} \right]$$

Donde:

F_D : Carga de diseño requerida (Lb)

n_D : Velocidad de diseño requerida (RPM)

L_D : Duración requerida

R : Confiabilidad

$L_R = 3$ KH

$n_R = 500$ RPM

(12) Diseño de Ingeniería Mecánica. Shigley 1990, Pág. 520.

$a = 3$ para rodamientos de bolas

$a = \frac{10}{3}$ para rodamientos de rodillos

Los rodamientos rígidos a bolas con tapas de protección son los que utilizaremos en este diseño. Los profundos canales de rodadura, sin ranura de entrada, posibilitan la absorción de cargas radiales y axiales de ambas direcciones, así como cargas combinadas.

Gracias a la sencilla construcción de estos rodamientos es posible su fabricación en alta precisión; además son diseñados para compensar las diferencias de dilatación térmica (13).

El rodamiento se selecciona con la capacidad de carga mayor, y con el diámetro, que será algo menos que el del eje.

(13) Rodamientos de bolas y de rodillos. NTN, Catálogo 2647, 1989, Pág. 8.

2.12 PROTECCION DEL PUENTE-GRUA.

Antes de aplicar recubrimientos anticorrosivos, la superficie de las estructuras deberá prepararse: eliminar de la superficie del metal los productos de la corrosión (cascarilla y herrumbre), manchas de grasa y demás sedimentos, dándole a la superficie cierta aspereza, a fin de mejorar la adhesión del revestimiento protector de ella.

La calidad de la protección en las estructuras metálicas dependerá, en su mayor grado, de la calidad lograda al reparar la superficie antes de pintarla.

Los defectos que queden aun después de limpiar las superficies pueden resultar tan sólo encubiertos, siendo después causa de una prematura corrosión del metal.

Las estructuras antes de pintarse se pueden limpiar de dos maneras:

- Mecánico, consiste en limpiar la superficie con herramientas manuales y mecanizadas, a saber, cepillos de alambre, materiales abrasivos, granalla.

- Térmico, calentamiento local de los artículos, recurriendo a la llama oxiacetilénica.

La limpieza, empleando herramientas manuales o mecanizadas es la más accesible debido a que es sencilla, barata y, además, por ser posible adquirir el equipo. La desventaja en el empleo de herramientas manuales y mecanizadas reside en su rápido desgaste y en la necesidad de reemplazar con frecuencia las partes con las que se trabaja.

El método fundamental para proteger las estructuras metálicas contra la destrucción por corrosión consiste en aplicar pinturas y barnices, y que se las aplica en el terreno para edificar.

Para dar la mano de fondo se utiliza, la imprimación oleosa a base de sanguínea en aceite, también se utiliza barniz, y pocas veces minio disuelto en aceite natural.

Se recomienda la imprimación sobre la estructura de 15-28°C, y el espesor de la capa será de 12-25µm; la duración del secado a este intervalo de temperatura es de 48 horas.

La imprimación se aplica recurriendo a pistolas neumáticas, que hace que la pintura a esa temperatura por medio de aire comprimido, se convierta en una masa dispersa que se aplica sobre la superficie.

La calidad del revestimiento aplicado, empleando este

método, resulta elevada.

Al trabajar con pistolas pulverizadas, estas deberán disponerse perpendicularmente a la superficie que se pinta y a una distancia constante respecto a esta; el desplazamiento de la pistola tendrá que hacerse a velocidad constante, la pintura se aplicará en bandas paralelas, sin pasar dos veces por el mismo sitio.

Es necesario tener presente que las pérdidas de pintura dependen no sólo de la configuración de la estructura, sino que también de la distancia desde la pistola hasta la superficie que se pinta y de la presión del aire comprimido.

La presión efectiva del aire comprimido para la pulverización no deberá exceder de 0.5 MPa.

CAPITULO III

RESULTADOS Y ANALISIS DE COSTOS

3.1 RESULTADOS OBTENIDOS.

El programa requiere de cierto número de variables que deben ser ingresadas desde el teclado, y son las siguientes:

VARIABLES:

Estas pueden variar según los requerimientos:

- L_1 : Longitud del carro para el trole (pies)
- L_2 : Longitud de la viga principal (pies)
- L_3 : Longitud del soporte lateral (pies)
- L_4 : Longitud de los ejes de las ruedas (pies)
- L_6 : Longitud no arriestrada de la viga porta-grúa (pies)
- T : Capacidad del elevador (kips)
- P : Peso del motor para el aparejo elevador (kips)
- $\overset{s}{F}_y$: Resistencia de fluencia del material (kpsi)
- $\overset{s}{F}_u$: Resistencia última del material (kpsi)

CONSTANTES:

Estos valores dependen de las variables y no pueden ser tomados arbitrariamente:

- W_r : Peso del riel (kip/pie)
- W_1 : Peso de la viga para el carro del trole
(kip/pie)
- W_2 : Peso de la viga para la viga principal
(kip/pie)
- W_3 : Peso de la viga para el soporte lateral
(kip/pie)
- A : Area de la sección transversal del perfil
(pulgadas²)
- bf : Ancho del ala del perfil (pulgadas)
- tf : Espesor del ala del perfil (pulgadas)
- tw : Espesor del alma del perfil (pulgadas)
- r_T : Radio de giro (pulgadas)
- S_x : Módulo del perfil (pul³).

A continuación los resultados de cada uno de los elementos, y las cargas y momentos máximos para los que fueron diseñados.

CARACTERISTICAS DEL PUENTE-GRUA:

Los valores han sido tomados de la Tabla No.2 de Marks (Dresser Industries, Inc.).

Para $T = 10$ Ton (considerando impacto) y $L_1 = 47.25$

pies, el programa da los siguientes resultados:

- Velocidad normal de elevación. $V_p = 40$ pie/minuto
- Distancia entre las ruedas del soporte lateral.
 $H = 9.68$ pies
- Peso del riel sobre la viga principal. $W_r = 0.016175$ Kip/pie
- Número de ruedas del puente. $N = 4$

POTENCIA DEL MOTOR DEL APAREJO ELEVADOR:

Para:

$F = 11,000$ Kips; capacidad del elevador sin impacto

$v = 40$ pie/min; velocidad normal de elevación

Potencia = $(11,000 \text{ lb}) * (0.66 \text{ pie/seg})$

donde $1 \text{ hp} = 550 \text{ pies-lb/seg}$ --> Factor de conversión

$P = 13.2 \text{ HP}$ (9.84 KW).

Seleccionamos un motor jaula de ardilla - 4 polos

Tipo 2SST (160 M) cuyas características son:

Peso = 88 Kg

Potencia = 11 KW (14.75 HP)

$N = 1,800 \text{ RPM}$

Se escoge un motor con una potencia mayor, ya que no existe en el mercado uno con la requerida.

POTENCIA DEL MOTOR PARA LAS RUEDAS DEL TROLE Y DEL SOPORTE LATERAL.

Para el carro del trole se colocará un motor y mediante un par de engranajes transmitirá el movimiento a la rueda.

$$\text{Para: } T = F \cdot v$$

$$F = \mu_k N \text{ (por dinámica de partículas)}$$

$$F = \mu_k * T_1$$

$$T_1 = \left(\frac{T + P}{4} \right)$$

$$\mu_k = 0.18 \quad (1)$$

$$F = (0.18) (2.75 \text{ Kip})$$

$$v = 150 \frac{\text{pie}}{\text{min}} \text{ (velocidad recomendada)}$$

$$P = 2.25 \text{ HP (1.68 KW)}$$

Seleccionamos un motor jaula de ardilla - 4 polos

Tipo 182T (112S) cuyas características son:

$$\text{Peso} = 17 \text{ Kg}$$

$$\text{Potencia} = 2.2 \text{ KW (2.95 HP)}$$

Para el soporte lateral, se utilizarán dos motores uno para cada uno; el cual dará movimiento a una rueda de cada soporte.

(1) Manual de Fórmulas Técnicas. Kurtgieck, 18va edición, tabla Z-20, 1981.

Para:

$$F = \mu_k N$$

$$T_2 = T_1 + W_1 L_1$$

$$F = \mu_k \cdot T_3$$

$$T_3 = T_2 + \frac{W_r L_2}{2}$$

$$v = 250 \text{ pie/min}$$

$$\mu_k = 0.18$$

$$F = (0.18) (4.14 \text{ Kip})$$

$$P = 6.77 \text{ HP (5.05 KW)}$$

Seleccionamos un motor jaula de ardilla - 4 polos
Tipo 213T (132S) cuyas características son:

$$\text{Peso} = 43 \text{ Kg}$$

$$\text{Potencia} = 5.5 \text{ KW (7.37 HP)}$$

Todos los motores son seleccionados en base a catálogos
y que existen en el mercado (2).

PERFIL DEL CARRO PARA EL TROLE:

Para:

$$T_1 = \frac{T + P}{4}$$

$$T = 22 \text{ Kips (con impacto)}$$

$$P = 0.1606 \text{ Kips (peso del motor)}$$

$$T_1 = 5.54 \text{ Kips}$$

$$L_1 = 4.27 \text{ pies}$$

$$F_y = 36 \text{ kpsi}$$

$$F_u = 58 \text{ kpsi}$$

(2) Motores y ventiladores, Siemens, Catálogo. Sección 1, 1989.

El programa da los siguientes resultados:

$$M_{\max} = 5.91 \text{ kip-pie}$$

$$S = 2.986 \text{ pulg}^3$$

- Perfil C 6 * 8.2

$$f_b = 16.25 \text{ kpsi}$$

$$F_b = 21.6 \text{ kpsi}$$

- Perfil C 5 * 9

$$f_b = 20.004 \text{ kpsi}$$

$$F_b = 21.6 \text{ kpsi}$$

- Perfil C 5 * 6.7

Falla por flexión

Podemos concluir que el más económico es el perfil;

A.I.S.C. C 5 * 9

PERFIL DE LA VIGA PRINCIPAL:

Para:

$$L_2 = 47.25 \text{ pies}$$

$$T = 22 \text{ kips}$$

$$P = 0.1606 \text{ kips}$$

$$F_y = 36 \text{ kpsi}$$

$$F_u = 58 \text{ kpsi}$$

$$W_r = 0.016175 \text{ kip/pie (peso del riel)}$$

$$L_1 = 4.27 \text{ pies}$$

$$W_1 = 0.099 \text{ kip/pie}$$

El programa da los siguientes resultados:

$$M_{max} = 124.39 \text{ kip-pie}$$

$$S = 62.82 \text{ pulg}^3$$

- Perfil W 30 * 108

$$fb = 6.202 \text{ kpsi}$$

$$Fb = 7.76 \text{ kpsi}$$

- Perfil W 30 * 99

$$fb = 6.7818 \text{ kpsi}$$

$$Fb = 7.789 \text{ kpsi}$$

- Perfil W 27 * 178

$$\frac{L * 12}{d} = 20.38$$

Falla por deflexión

$$\text{porque: } \frac{L * 12}{d} \leq 20$$

El perfil más económico es:

A.I.S.C. W 30 * 99

PERFIL DEL SOPORTE LATERAL:

Para:

$$L_3 = H + 2L_e$$

$$L_3 = 9.68 + 1.5$$

$$L_3 = 11.18 \text{ pies}$$

$$T_1 = 22 \text{ kips}$$

$$L_1 = 4.27 \text{ pie}$$

$$L_2 = 47.25 \text{ pie}$$

$$W_1 = 0.009 \text{ kip/pie}$$

$$W_2 = 0.099 \text{ kip/pie}$$

$$W_r = 0.016175 \text{ kip/pie}$$

El programa da los siguientes resultados:

$$M_{\max} = 10.549 \text{ kip-pie}$$

$$S = 5.32 \text{ pulg}^3$$

- Perfil C 6 * 13

Falla por deflexión

$$\frac{L * 12}{d} = 22.3$$

- Perfil C 10 * 30

$$fb = 8.857 \text{ kpsi}$$

$$Fb = 11.828 \text{ kpsi}$$

- Perfil C 10 * 25

$$fb = 10.023 \text{ kpsi}$$

$$Fb = 11.25 \text{ kpsi}$$

- Perfil C 10 * 20

Falla por deflexión

Todos los perfiles C con d menores a 6.7 fallan por deflexión

- Perfil C 10 * 15.3

Falla por flexión

El perfil más económico es:

A.I.S.C. C 10 * 25

PERFIL, PERNOS DE ANCLAJE Y SOLDADURA DE LA VIGA PORTA-

GRUA

T = 22 Kips
 P = 0.1606 Kips
 L₁ = 4.27 pies
 L₂ = 47.25 pies
 H = 2.95 HP (potencia del motor)
 V_{punto} = 150 pie/minuto
 W_r = 0.016175 kip/pie
 W₁ = 0.009 kip/pie
 W₂ = 0.099 kip/pie
 L₃ = 11.18 pie
 W₃ = 0.025 kip/pie
 L₆ = 19.686 pies (6 metros)
 F_p = 0.25 Kpsi (concreto)
 F_{us} = 60 Kpsi (electrodo)
 G = 13.78 pul (ancho de la mampostería)

El programa da los siguientes resultados:

M_{max} = 30.39 kip-pie

S = 15.34 pulg³

- Perfil W 12 * 19

$$f_b = 17.57 \text{ Kpsi}$$

$$F_b = 23.76 \text{ Kpsi}$$

- Perfil W 12 * 16

$$f_b = 21.79 \text{ Kpsi}$$

$$F_b = 23.76 \text{ Kpsi}$$

- Perfil W 12 * 14

Falla por flexión

El perfil más económico es

A.I.S.C. W 12 * 16

Todos los perfiles con d menores a 11.8 fallan por deflexión.

- Las dimensiones de la placa de apoyo son:

$$\text{Largo de la placa} \quad G = 13.78 \text{ pulgadas}$$

$$\text{Ancho de la placa} \quad F = 2.22 \text{ pulgadas}$$

$$f_p = 0.247$$

$$F_p = 0.25$$

$$\text{Espesor de la placa} \quad t = 0.059 * \frac{16}{16} = \frac{0.94}{16} \approx \frac{1}{16}$$

Se tomará una placa de espesor 1/4" (existe en el mercado).

Es importante señalar que se puede variar las dimensiones de G y F, siempre que se cumpla que:

$$F * G = \frac{R}{F_p}$$

$$F * G = 29.92 \text{ pulg}^2$$

- Las dimensiones de la soldadura para la unión de los tramos de la viga porta-grúa son:

$$a = 0.114760 * \frac{16}{16} = \frac{1.8}{16} \approx 1/8''$$

$$L_{\text{MIN}} = 0.459 * \frac{16}{16} = \frac{7.34}{16} = 1/2''$$

- Pernos de anclaje y pernos de sujeción del riel con la viga porta-grúa:

Están diseñados para:

Cortante, tensión, aplastamiento, de la que se ha escogido el área mayor:

$$A = 0.27 \text{ pulgadas}^2$$

$$d = 0.58 * \frac{16}{16} = \frac{9.4}{16} \approx 5/8''$$

AREA CRITICA DEL GANCHO DEL APAREJO ELEVADOR:

Para comenzar se asumió un factor R/C, en base a la dimensión que debe llevar un gancho (3).

Con la relación R/C, entramos a la tabla y nos da el valor de K que es el factor de corrección para las vigas curvas (4).

(3) Prontuario de Máquinas. Paraninfo, 1990, Pág. 308.

(4) Curso Superior de Resistencia de materiales. Searly Fred.

Para:

$T = 22$ (para carga de impacto)

$R/C = 1.99$

$K = 1.69$

$S_y = 77$ Kpsi (acero SAE 104S-CD-gancho)

$S_{y_1} = 77$ Kpsi (acero SAE 104 S-CD- pasador)

$L_{\text{pasador}} = 0.5249$ pies (16 cm)

F.S = 1.5 (factor conservador)

$\theta = 0^\circ$ (sección más crítica)

El programa da los siguientes resultados:

- Area óptima en la sección; $A = 4.557$ pulg²
- Ancho de la sección transversal; $b = 2.08$ pulg
- Radio de curvatura casquete circular; $R_1 = 0.53$ pulg
- Radio de curvatura casquete elipzoidal; $R_2 = 1.11$ pulg
- Longitud de la sección transversal; $X_1 = 3.38$ pulg
- Radio interior del gancho; $r_b = 1.45$ pulg
- Diámetro del pasador; $d = 2.049$ pulg

Si comparamos estas medidas con las que recomienda el prontuario, veremos que se asemejan mucho.

LOS EJES DE LAS RUEDAS Y CHAVETAS:

$T = 22$ Kips

$F = 0.1606$ Kips

$W_r = 0.016175$ kip/pie

W_1	= 0.009 kip/pie
L_1	= 4.27 pies
L_2	= 47.25 pies
W_2	= 0.099 kip/pie
H_t	= 2.95 HP (trole)
H_m	= 5.36 HP (soporte)
L_3	= 11.18 pie
W_3	= 0.025 kip/pie
S_{ut}	= 80 kpsi (Acero SAE 1035-CD)
S_y	= 67 Kpsi (Acero SAE 1035-CD)
F_u	= 60 kpsi (electrodo)
L_4	= 0.89 pies (10.7 pulg) longitud del eje (rodamiento = 20 mm)
A	= 1.34 kpsi factor acabado superficie
B	= - 0.085
D_1	= 2.5 pulg
D_2	= 3.34 pulg
D_{t_1}	= 11.85 pulg
D_{t_2}	= 11.85 pulg

El programa da los siguientes resultados:

- Diámetro del eje para el carro del trole;
d = 2.45 pulg
- Diámetro del eje para el soporte lateral;
d = 2.25 pulg

Las dimensiones de la chaveta son:

- Ancho, $W = 0.625$ pulgadas
- Altura, $h = 0.437$ pulgadas
- Profundidad del chavetero; $p = 0.218$ pulgadas

Las dimensiones de la soldadura para la placa de alojamiento del rodamiento son:

Soporte lateral:

$$a = 0.1249 = \frac{1.99}{16} \approx 1/8''$$

$$L_{MIN} = 0.49 = \frac{7.84}{16} = 1/2''$$

Trole:

$$a = 0.062 = 1/6''$$

$$L_{MIN} = 0.248 = \frac{3.9}{16} = 1/4''$$

PERNOS DE CONEXION:

Los pernos para el carro del trole están diseñados para: cortante, tensión y aplastamiento para la mayor carga T_1 .

Los pernos para unión viga-soporte están diseñados para tensión pura y para cortante; incluido aplastamiento.

Para:

$$T = 22 \text{ kips}$$

$$P = 0.1606 \text{ kips}$$

$$L_1 = 4.27 \text{ pies}$$

$$L_2 = 47.25 \text{ pies}$$

$$L_3 = 11.18 \text{ pies}$$

$$F_y = 67 \text{ kpsi (Acero SAE 1035-CD)}$$

$$F_y = 80 \text{ kpsi (Acero SAE 1035-CD)}$$

$$t_t = 0.320 \text{ pulg}$$

$$t_s = 0.436 \text{ pulg}$$

El resultado del programa es:

- Diámetro de los pernos para el carro del trole:

$$d = 0.513 = \frac{8.20}{16} = 9/16" = 5/8"$$

- Diámetro de los pernos de conexión viga-soporte:

$$d = 0.42 = \frac{6.72}{16} = 1/2"$$

- Diámetro de los pernos del soporte lateral:

$$d = 0.52 = \frac{8.32}{16} = 5/8"$$

- Diámetro de los pernos para los rigidizadores de la viga principal:

$$d = 0.42 = \frac{6.72}{16} = 1/2"$$

- Número de pernos para cada conexión

$$N = 4$$

CALCULO DE ENGRANAJES:

- Carro del trole: con

$$H = 2.95 * 1.2 = 3.54 \text{ HP}$$

$$n_1 = 1800 \text{ RPM}$$

En la Tabla No.4 Apéndice B

$$Z = 20 \text{ (número de dientes)}$$

$$\text{paso} = 8$$

De donde:

$$D_1 = 2.5 \text{ pulgadas}$$

$$Z_2 = 11.85 * 8$$

$$Z_2 = 95 \text{ dientes}$$

$$D_2 = 11.85 \text{ pulgadas}$$

- Soporte lateral: con

$$HP = 7.37 * 1.2 = 8.84 \text{ HP}$$

$$n_1 = 1800 \text{ RPM}$$

En la Tabla No.4 Apéndice B

$$Z = 20 \text{ (número de dientes)}$$

$$\text{paso} = 8$$

$$D_1 = 3.34 \text{ pulgadas}$$

$$Z_2 = 11.85 * 6$$

$$Z_2 = 72 \text{ dientes}$$

$$D_2 = 11.85 \text{ pulgadas}$$

De donde:

$$\text{Factor de servicio} = 1.2$$

$$\text{Angulo de presión} = 20^\circ$$

Cables flexibles de acero:

Según su aplicación, considerando cargas y servicio:

Seleccionamos grupo II (5):

$$S = 7$$

El diámetro del cable es:

$$d = K\sqrt{T}$$

T: Carga a que está sometido el cable

K: Coeficiente según el grupo

De donde elegimos:

$$K = 0.34 \text{ (Grupo II)}$$

$$d = 0.34 \sqrt{10,000}$$

$$d = 34 \text{ mm (1.34 pulgadas)}$$

CARGA DE ROTURA:

$$Tr = 10,000 * 7 = 70,000 \text{ Kg}$$

Escogemos un cable con las siguientes características:

$$d = 35 \text{ mm}$$

Con una resistencia a la rotura de 71,400 Kg (160 Kg/mm²)

$$\text{Peso} = 4,24 \text{ Kg/metro}$$

Diámetro de poleas y tambor:

$$D = S\sqrt{T}$$

$$D = 7\sqrt{10,000}$$

$$D = 700 \text{ mm (27.5 pulgadas)}$$

(5) Prontuario de Máquinas. Paraninfo, 1990, Pág. 308.

TABLA DE RESULTADOS

CARACTERISTICAS DEL PUENTE-GRUA							
CAPACIDAD (TON)		VELOCIDAD DE ELEVACION PIE/MIN		DISTANCIA H PIES		No. RUEDAS	
10		40		9.68		4	
POTENCIA DE MOTORES (HP)							
APAREJO ELEVADOR		CARRO DEL TROLE		SOPORTE LATERAL			
14.75		2.95		7.37			
PERFILES MAS ECONOMICOS							
CARRO DEL TROLE		VIGA PRINCIPAL		SOPORTE LATERAL		VIGA PORTA-GRUA	
C 5 * 9		W30 * 99		C10 * 25		W12 * 16	
EJES DE LAS RUEDAS Y CHAVETAS							
CARRO DEL TROLE (pulg)				SOPORTE LATERAL (pulg)			
d	w	h	p	d	w	h	p
2.45	0.625	0.43	0.21	2.25	0.625	0.43	0.21
UNIONES SOLDADAS							
CARRO DEL TROLE (pulg)		SOPORTE LATERAL (pulg)		VIGA PORTA-GRUA (pulg)			
a = 1/16		a = 1/8		a = 1/8			
L = 1/4		L = 1/2		L = 1/2			
UNIONES ATORNILLADAS (DIAMETROS)							
CARRO DEL TROLE (pulg)		SOPORTE LATERAL (pulg)		UNION VIGA-SOPORTE (pulg)		RIGIDIZADOR VIGA PRINCIPAL (pulg)	
5/8		5/8		1/2		1/2	

TABLA DE RESULTADOS

ENGRANAJES	
CARRO DEL TROLE	
PIÑON (pulg)	ENGRANAJE (pulg)
D = 2.5	D = 11.8
Z = 20	Z = 95
P = 8	P = 8
SOPORTE LATERAL	
PIÑON (pulg)	ENGRANAJE (pulg)
D = 3.35	D = 11.85
Z = 20	Z = 72
P = 6	P = 6
APAREJO ELEVADOR	
PIÑON (pulg)	ENGRANAJE (pulg)
D = 5	D = 11.85
Z = 20	Z = 48
P = 4	P = 4
CABLES FLEXIBLES DE ACERO (pulg)	
d = 1.34	
POLEAS Y TAMBOR (pulg)	
d = 27.5	

3.2 COSTOS DE CADA ELEMENTO DEL PUENTE-GRUA.

- Carro del trole:		
2 perfiles C 5 * 9 DE 6 METROS	S/.	276,000.00
- Viga principal:		
3 perfiles W 30 * 99 DE 6 METROS	S/.	8,751,442.00
- Soporte lateral:		
4 perfiles W C 10 * 25 DE 6 METROS	S/.	1,415,880.00
- Viga porta-grúa:		
9 perfiles W 12 * 16	S/.	16,767,000.00
- Uniones:		
64 pernos de 1/2 * 3"	S/.	24,320.00
64 pernos de 5/8 * 3"	S/.	30,720.00
108 pernos de 1/2 * 2"	S/.	45,360.00
80 pernos de 1/2 * 5"	S/.	57,600.00
- Motores:		
1 motoreductor de 15 HP (polipasto)	S/.	5,430,000.00
1 motoreductor de 3 HP (trole)	S/.	562,000.00
2 motoreductores de 6 HP (puente)	S/.	1,828,000.00
- 8 ruedas rieles	S/.	720,000.00
- 1 gancho para el aparejo (10 ton)	S/.	147,000.00
- Rieles (137 metros)	S/.	1,843,335.00
- 1 plancha 1/4" * 1.22 * 2.44	S/.	1,026,240.00

- 6 galones de pintura anticorrosiva	S/. 1,104,000.00
- Botoneras y protecciones:	
1 para un motor de 15 HP	S/. 680,000.00
1 para un motor de 6 HP	S/. 400,000.00
1 para un motor de 3 HP	S/. 250,000.00
- Breaker:	
1 para 30 Amperios	S/. 70,000.00
1 para 8 Amperios	S/. 35,000.00
1 para 5 Amperios	S/. 27,000.00
- Cable tipo TW 3 * 8 AWG	S/. 135,000.00

3.3 COSTO TOTAL DEL PUENTE-GRUA.

- El total de los elementos constitutivos	S/. 40,508,697.00
- Costos indirectos	S/. 3,950,000.00
- Mano de obra	S/. 5,926,000.00
- Costo total	S/. 50,589,697.00
	=====

Esta cotización es válida hasta la fecha (Noviembre/93), luego se vera afectada por concepto de inflación.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES:

- 1.- Cuando se hace un análisis estructural de un acero, implica calcular las cargas y momentos máximos y las propiedades requeridas para resistirlas.
- 2.- El diseño en base a las cargas máximas, seleccionará el perfil más ligero y que cumpla con ciertas propiedades; este perfil es el más económico, que es un factor fundamental en el criterio del diseñador.
- 3.- El manual del A.I.S.C. garantiza la consideración de factores de seguridad que son fundamentales para un diseño óptimo y por tanto seguro.
- 4.- El manual A.I.S.C. recomienda para el diseño esfuerzos permisibles, que deben ser iguales a cierto porcentaje del esfuerzo de fluencia del acero.
- 5.- Para el diseño de los ejes móviles, se aplica la teoría del esfuerzo cortante máximo para pronosticar el esfuerzo de daño y la línea de Goodman modificada para predecir la resistencia significativa; el eje se diseña para fatiga.
- 6.- Para el diseño de las vigas por flexión, se debe considerar primeramente que resistan la flexión, y

luego: cortante, aplastamiento y deflexión.

- 7.- Se ha escogido para la viga principal perfiles W, ya que estos poseen un mayor momento de inercia y momentos resistentes para una misma carga.
- 8.- Se ha escogido un puente principal compuesto, es decir dos vigas W unidas en sus extremos para formar una caja, para darle mayor rigidez torsional y lateral.
- 9.- El diseño de todos los elementos está hecho para la carga más crítica; que para este caso es dos veces la capacidad de la grúa (carga de impacto).

RECOMENDACIONES:

- 1.- El perfil para cada elemento deberá ser diseñado para que cumpla con ciertas características y de acuerdo a las condiciones disponibles.

El perfil debe ser el más económico, y si no existe en el mercado habrá que fabricarlos.

- 2.- Siguiendo las recomendaciones que nos presenta la Dresser Industries, Inc., este tipo de puente-grúa está hecho para cargas de hasta 60 toneladas y para claros mayores de 40 pies (hasta un límite de 100 pies); así que el programa cumple estas condiciones.

- 3.- El programa que calcula la viga porta-grúa (viga estáticamente indeterminada); contiene una subrutina que encuentra las reacciones y momentos máximos, la misma que se ha hecho para diez apoyos; si tiene más o menos, la subrutina no será válida y habrá que cambiarla por otra para el número requerido.

El hecho de cambiar la subrutina, no afecta en absoluto el programa global.

- 4.- A pesar de que se ha diseñado el gancho, lo recomendable es seleccionarlo directamente.

APENDICE A

MANUAL DEL USUARIO

El programa para el diseño del puente-grúa ha sido realizado en turbo C, debido a que este lenguaje tiene una cuidadosa inclusión de las estructuras correctas de control, lo que hace que su empleo sea significativo y casi ilimitado. Además es de aprendizaje rápido.

Este lenguaje tiene un compilador ("Compilador C" o "Compilador CC"), que traduce el programa o código fuente a código objeto o lenguaje de máquina que es ejecutable.

Para poner en marcha el programa, se tiene que cargar el turbo C, que es el lenguaje utilizado.

EJECUCION DEL PROGRAMA.-

1. Para poner en marcha el sistema, se introduce el disco del dos en la unidad A.
2. Se retira este disco y se introduce el que contiene el turbo C en la misma unidad, y en la unidad B, el disco que contiene el programa de diseño.
3. Dentro del directorio de B se encuentra un fichero denominado Fabián; que es el que contiene el programa.
4. Una vez cargado el programa, se ejecuta el mismo pulsando R (RUN); para detener la ejecución se pulsa CTRL-PAUSE.

PROGRAMA:

Al inicio de la ejecución del programa se presenta un Menú de Opciones para diseñar cada elemento constitutivo del puente-grúa; el menú es el siguiente:

1. Selección de características del puente-grúa
2. Selección del perfil para el carro del trole
3. Selección del perfil para la viga principal
4. Selección del perfil para el soporte lateral
5. Selección del perfil para la viga porta-grúa
6. Diseño del gancho del aparejo elevador
7. Diseño de los ejes de las ruedas y chavetas
8. Diseño de las uniones atornilladas

Se escoge la opción, posicionando el cursor y se pulsa la barra espaciadora.

Es importante señalar que si bien se puede elegir cualquier opción, es recomendable (y así lo requiere este diseño) seguir el orden establecido, ya que cada diseño necesita datos del anterior, a excepción del gancho del aparejo elevador.

Una vez que se eligió el elemento a diseñar, el programa solicita datos referente a cargas, longitudes, y del tipo de acero.

Luego se introduce los datos referentes al perfil que están en el Manual del A.I.S.C.; para por último chequear todas

las especificaciones y recomendaciones de diseño y nos da el resultado más óptimo.

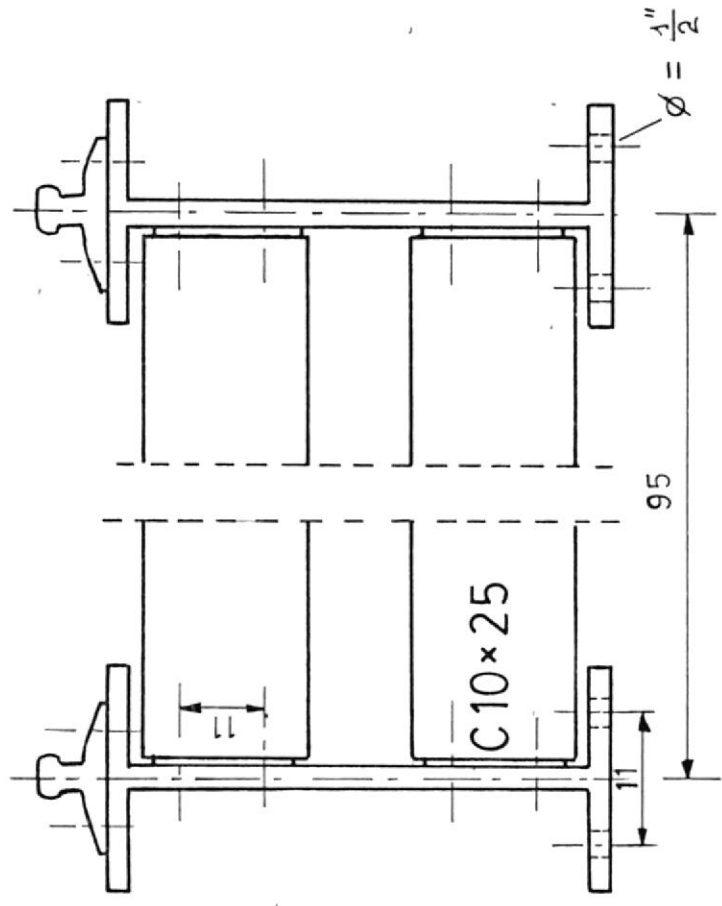
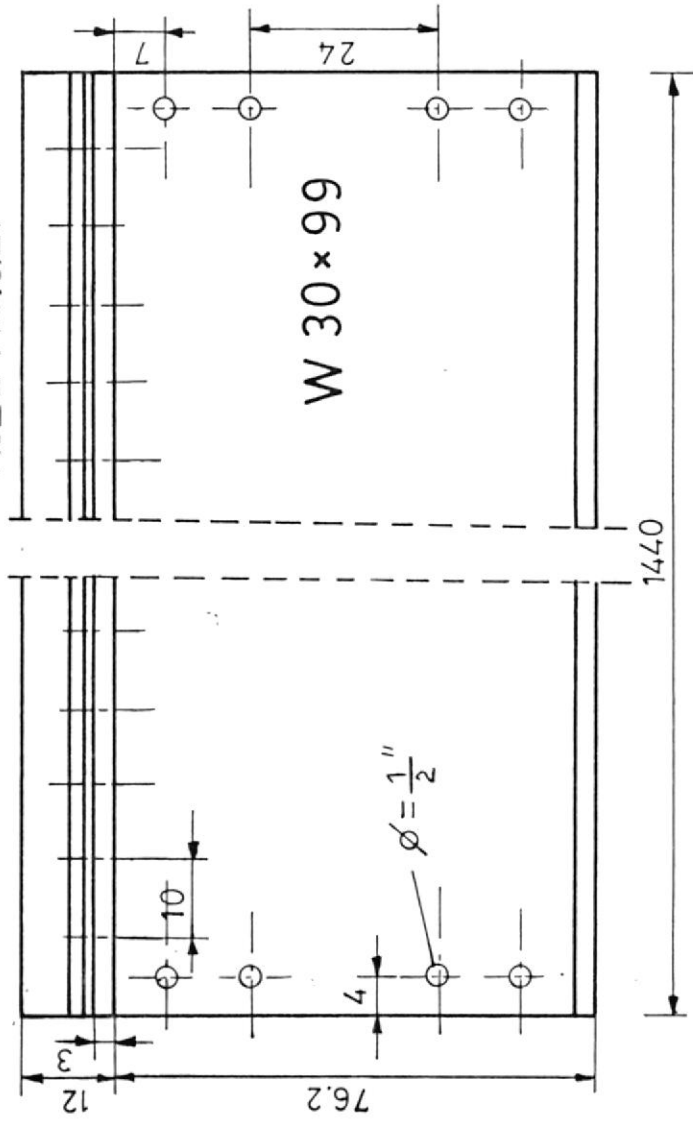
Todos los datos, ya sean ingresados o resultados, son reales de doble precisión.



Para abandonar el programa se pulsa ESC y se regresa al Menú Principal; para regresar o salir al DOS se pulsa ALT-X.

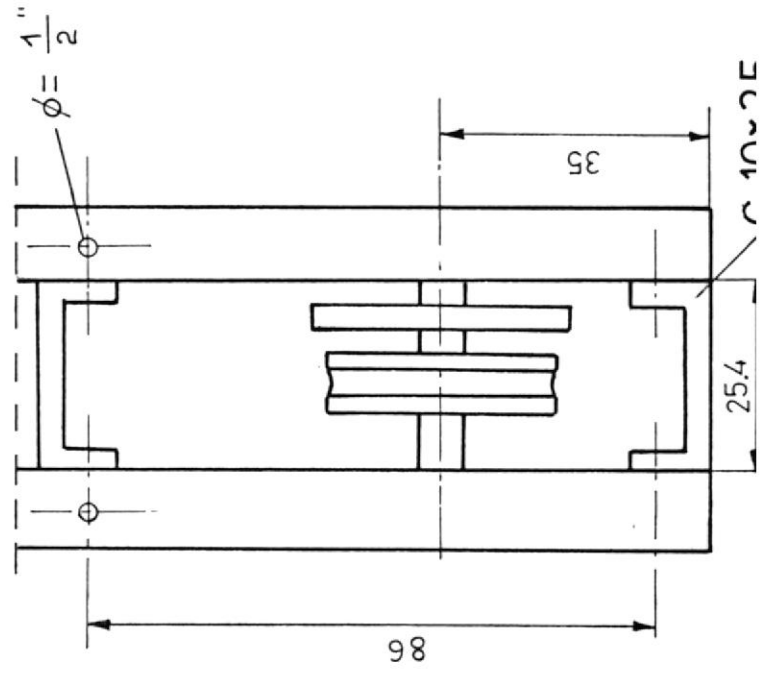
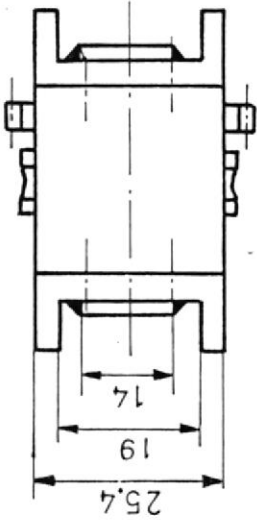
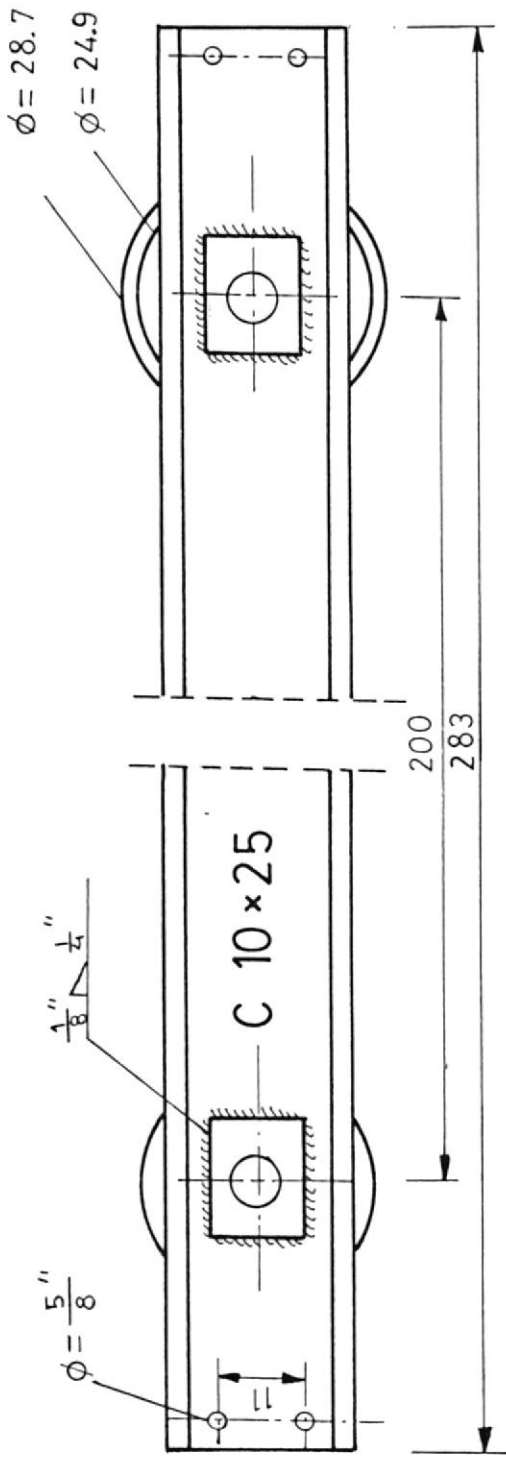
APENDICE B



P L A N O S

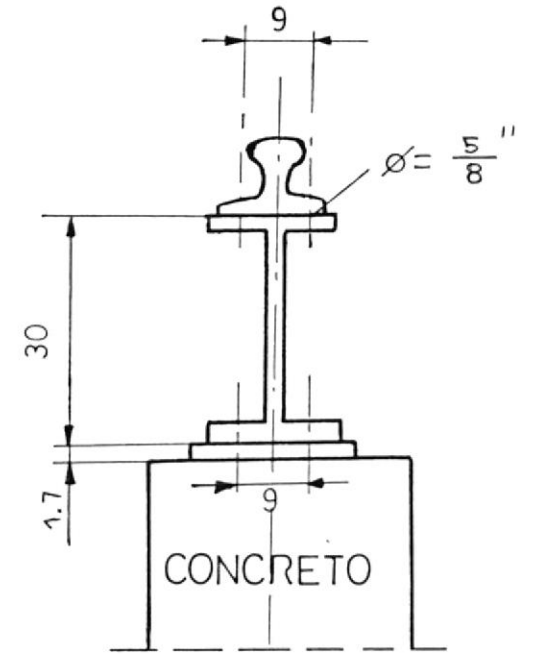
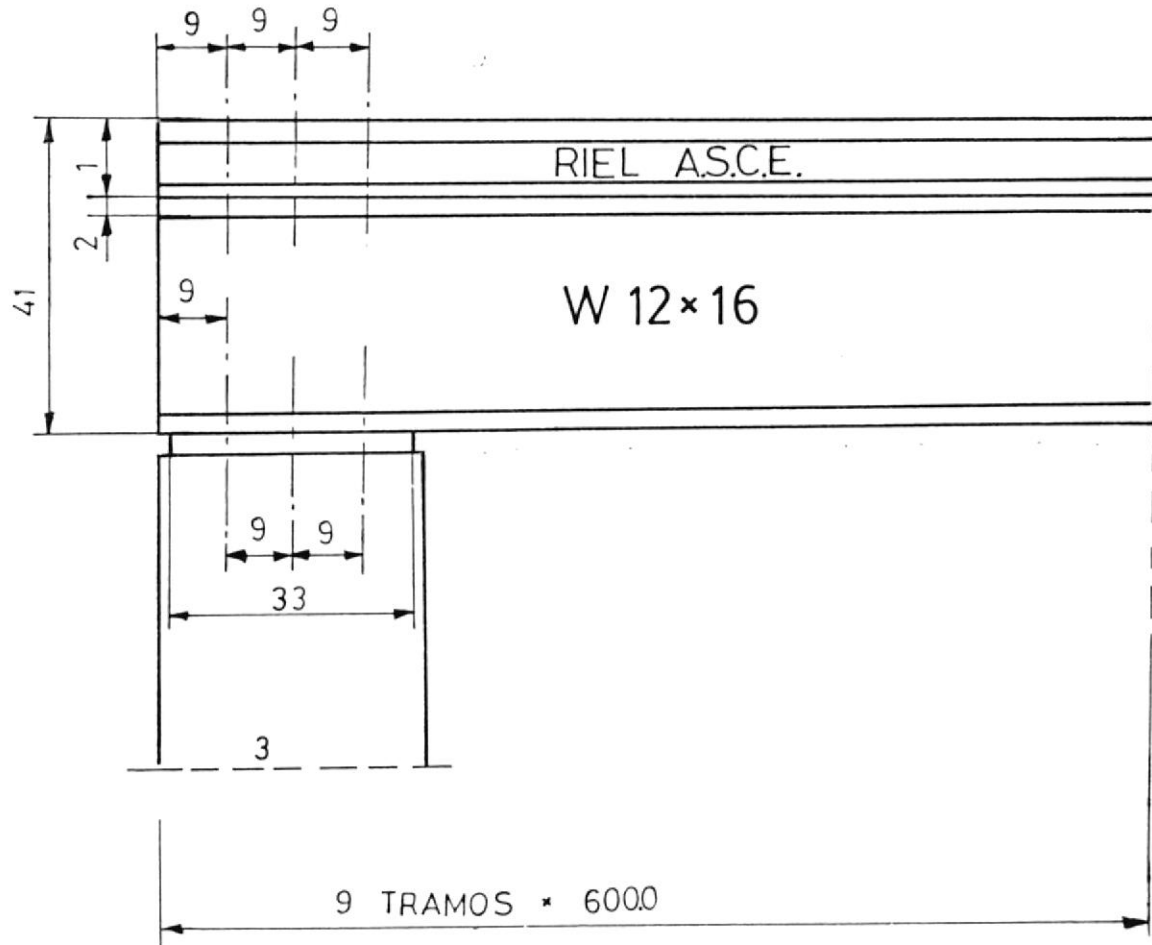
RIEL A.S.C.E.



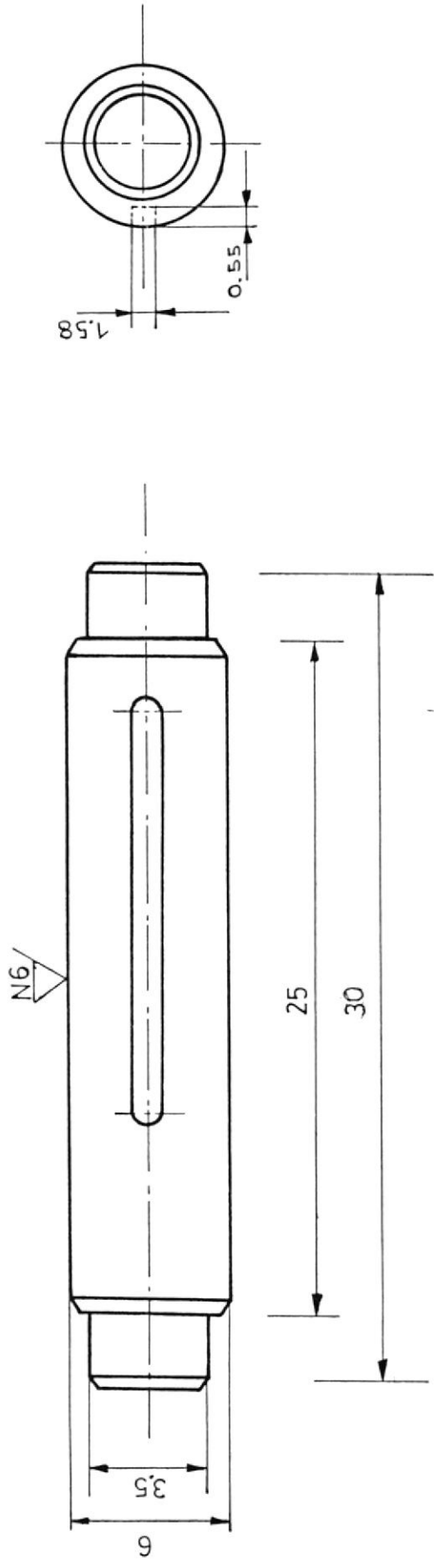
	ESPOL	
2	VIGA PRINCIPAL	



ESPOL		
3	SOPORTE LATERAL	



		ESPOL
4	VIGA PORTAGRUA	
1:10		



		ESPOL
5	EJE DE LAS RUEDAS	
1-25		

BIBLIOGRAFIA

- 1.- A.I.S.C., Manual de Construcción en Acero, 8ava. Edición.
- 2.- DIAS MOSTO Jorge, Problemas de Resistencias de Materiales.
- 3.- KERNIGHAN Brian, The C Programming Language, Prentice Hall, New Jersey 1988.
- 4.- LARBURU Nicolas, Máquinas Prontuario, 2da. Edición, Paraninfo, Madrid 1990.
- 5.- LAWRIE Robert, Motores Eléctricos, Océano/Centrum, Barcelona 1987.
- 6.- MARKS, Manual del Ingeniero Mecánico, 8ava. Edición, McGraw-Hill, México 1986.
- 7.- MCCORMAC Jack, Diseño de Estructuras de Acero, Alfaomega, México 1991.
- 8.- N.T.N. Toyo Bearing Co.Ltd., Rodamientos de bolas y rodillos, Catálogo No.2647, 1990.
- 9.- PESHKOUSKI O, Producción de Estructuras Metálicas, Mir, Moscu 1982.

- 10.- SEARLY Fred, Curso Superior de Resistencia de Materiales, Nigar S.R.L., Buenos Aires 1967.
- 11.- SHIGLEY Joseph, Diseño en Ingeniería Mecánica, 5ta. edición, McGraw-Hill, México 1990.
- 12.- ROSALER Robert, Manual de Mantenimiento Industrial, 1ra. edición, McGraw-Hill, México 1990.

Dimensiones, cargas y velocidades. Grúas tipo industrial a, e, f, g

Capacidad elevador principal, ton cortas, 2 000 lb	Elevación normal, elevador principal, pies ^b	Velocidad normal de elevador, pie/min	Dimensiones (véase la Fig. 13)						Carga máxima por rueda, lb ^d	Rieler, lb/yarda	X, pulg	Número de ruedas del puente
			A	B	C	D	E	F				
5	40	35	46	2'9"	3'1"	3'0"	2'6"	8'0"	14,100	30	6	4
	60	53	46	3'9"	3'1"	3'0"	2'6"	9'6"	17,000	30	6	4
	75	86	46	3'9"	3'1"	3'0"	2'6"	12'0"	20,400	30	6	4
	100	118	46	3'9"	3'1"	3'0"	2'6"	14'6"	24,100	60	6	4
10	40	53	40	4'4"	4'5"	2'10"	3'5"	9'6"	19,700	40	6	4
	60	71	40	4'4"	4'5"	2'10"	3'5"	10'0"	23,000	60	6	4
	78	80	40	4'4"	4'5"	2'10"	3'5"	12'0"	26,600	60	6	4
	100	120	40	4'4"	4'5"	2'10"	3'5"	14'6"	30,800	80	6	4
15	40	46	26	4'4"	5'0"	2'11"	3'4"	10'6"	24,800	60	12	4
	60	46	26	4'4"	5'0"	2'11"	3'4"	11'0"	28,500	60	12	4
	75	46	26	4'4"	5'0"	2'11"	3'4"	11'6"	32,200	100	12	4
	100	80	26	4'6"	5'0"	2'11"	3'4"	14'6"	37,400	80	12	4
20, auxiliar 5 ton cortas	40	34	28	4'5"	5'0"	3'0"	3'4"	11'0"	39,500	80	18	4
	60	34	28	4'6"	5'0"	3'0"	3'4"	11'0"	33,600	60	18	4
	80	34	28	4'6"	5'0"	3'0"	3'4"	11'6"	37,800	80	18	4
	100	60	28	4'6"	5'0"	3'0"	3'4"	14'6"	43,700	100	18	4
30, auxiliar 5 ton cortas	40	34	17	5'1"	5'10"	3'11"	3'5"	11'6"	51,600	100	18	4
	60	34	17	5'3"	5'10"	3'11"	3'5"	11'6"	55,600	100	18	4
	80	34	17	5'3"	5'10"	3'11"	3'5"	12'0"	59,500	100	18	4
	100	54	17	5'0"	5'10"	4'0"	3'5"	14'5"	33,600	60	18	8
40, auxiliar 5 ton cortas	40	31	13	5'1"	5'11"	3'11"	3'5"	11'6"	52,600	100	18	4
	60	31	13	5'3"	5'11"	3'11"	3'5"	11'6"	57,500	100	18	4
	80	31	13	5'3"	5'11"	3'11"	3'5"	12'0"	62,000	100	18	4
	100	49	13	5'0"	5'11"	4'1"	3'5"	14'5"	34,892	60	18	8
50, auxiliar 10 ton cortas	40	27	11	5'6"	6'9"	4'5"	3'8"	12'0"	61,000	100	18	4
	60	27	11	5'3"	6'9"	4'6"	3'8"	11'10"	33,700	60	18	8
	80	27	11	5'3"	6'9"	4'6"	3'8"	12'1"	36,800	80	18	8
	100	40	11	5'3"	6'9"	4'6"	3'8"	14'5"	40,400	100	18	8
60, auxiliar 10 ton cortas	40	27	9	5'4"	6'9"	4'6"	3'8"	11'10"	35,100	60	18	8
	60	27	9	5'4"	6'9"	4'6"	3'8"	12'1"	38,800	80	18	8
	80	27	9	5'4"	6'9"	4'6"	3'8"	12'1"	42,300	100	18	8
	100	40	9	5'5"	6'9"	4'5"	3'8"	14'9"	46,500	80	18	8

^a División de Grúas y Elevadores, Dresser Industries, Inc.

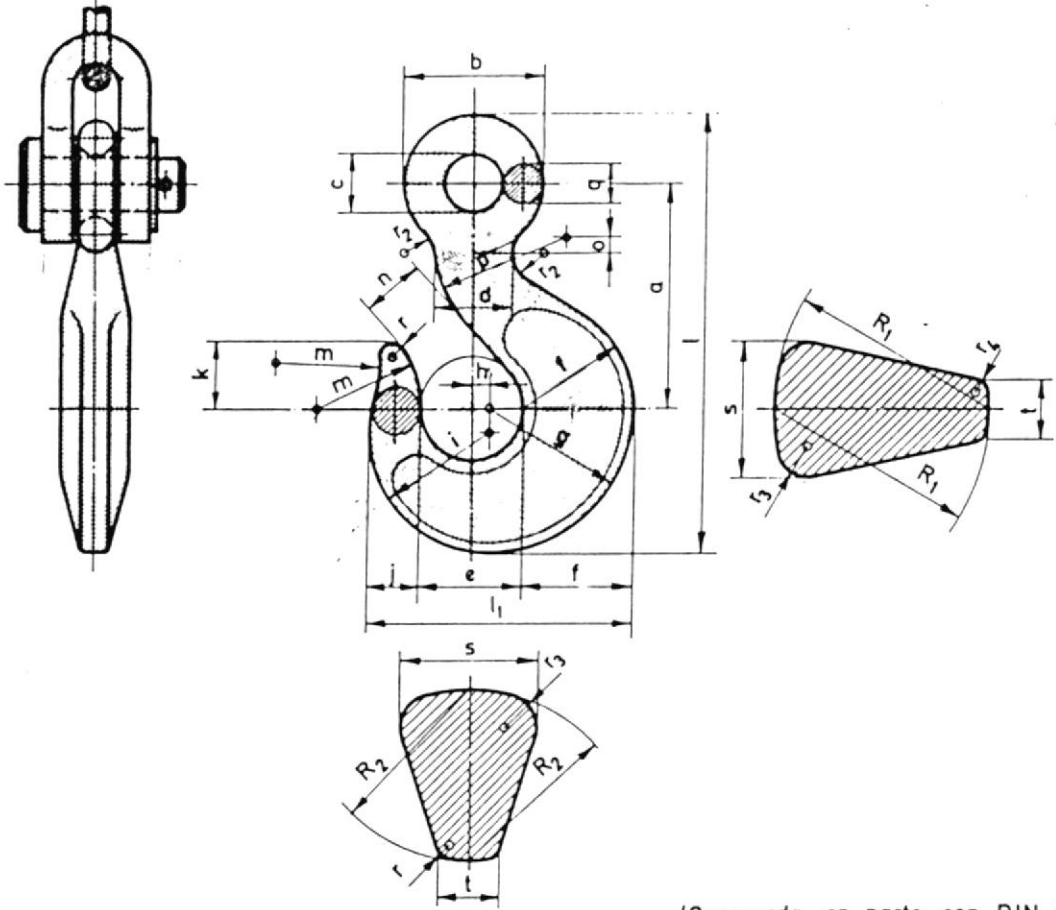
^b Por cada 10 pies de elevación adicional, aumente H, por X.

^c Gancho principal, si no hay gancho auxiliar.

^d Cargas directas, sin impacto.

^e Basado en velocidad normal de 100 a 150 pie/min; la velocidad del puente es de 175 a 300 pie/min. Para 40 a 60 ton cortas, carro 100 pie/min; puente 250 pie/min.

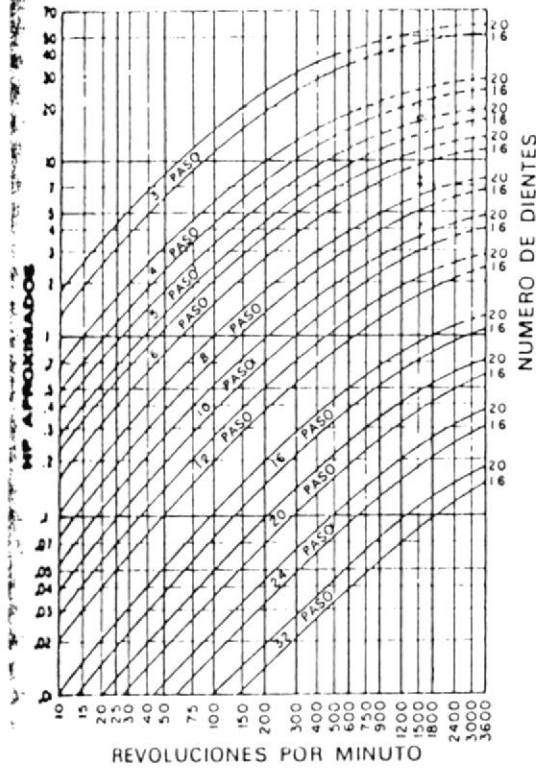
^f Para grúas de 30 ton cortas o menos, la velocidad normal del carro es de 100 a 150 pie/min; la velocidad del puente es de 175 a 300 pie/min. Para 40 a 60 ton cortas, carro 100 pie/min; puente 250 pie/min.



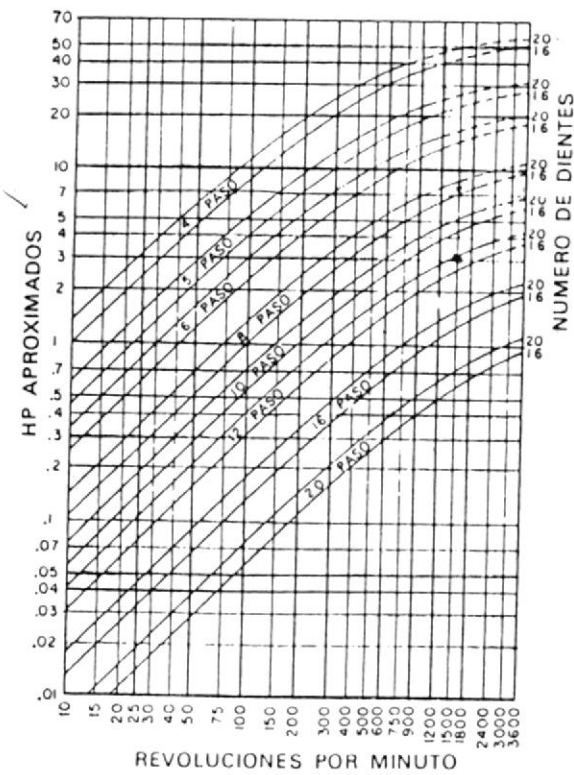
(Concuerta, en parte, con DIN 689)

Valores	CARGA UTIL EN KILOGRAMOS													
	250	500	1000	1600	2500	4000	5000	6300	8000	10000	12500	16000	20000	25000
a	62	82	113	133	167	211	236	265	299	334	373	422	472	528
b	24,5	32,5	44	52,5	65,5	82	92	103	117	131	146	165	184	206
c	9,5	12,5	17	20,5	25,5	32	36	40	46	51	57	64	72	80
d	15	20	27	32	40	50	56	64	72	80	89	100	113	126
e	23	31	42	50	62	79	88	99	112	125	140	158	176	198
f	19,5	26	36	42,5	53,5	67,5	75	84,5	96	106	118	135	151	168
g	28,5	38	52,5	62	77,5	98	109	123	139	155	173	196	219	245
h	2,5	3,5	4,5	5,5	7	9	10	11	13	13,5	15	18	20	22
i	25	33,5	46	54,5	68,5	86,5	96,5	109	123	137	153	173	193	216
j	11	14,5	20	23,5	29,5	37	41,5	46,5	52,5	58,5	65,5	74	83	93
k	13	17,5	24	28,5	36	45	50	57	64	72	80	90	101	113
l	86,5	114,5	157,5	185,5	233	294	329	369	417	466	520,5	588,5	658	736
l ₁	53,5	71,5	98,5	116,5	146	184,5	205,5	232	266	292	326	369	412	461
m	21	28	38	45	57	72	80	90	100	115	125	145	160	180
n	18	24	33	39	49	62	69	78	88	98	109	124	138	155
o	2,3	3	4,3	5	6,4	8	9	10	11,5	12,5	14	16	18	20
p	22	30	41	48	60	76	85	96	108	121	135	152	170	194
q	7,5	10	13,5	16	20	25	28	31,5	35,5	40	44,5	50,5	56	63
s	13	17	23,5	28	35	44,5	49,5	56	63	70	78	89	99	110
t	5,5	7	10	11,5	14,5	18,5	20,5	23	26	29	33	37	41	46
R ₁	19,5	26	36	42,5	53,5	67,5	75	84,5	96	106	118	135	151	168
R ₂	17	22,5	31	36,5	46	58	65	73	82,5	92	103	116	130	145
r ₁	3	4	5,5	6,5	8	10,5	11,5	13	14,5	16,5	18	20,5	23	26
r ₂	6	8	11	13	16	20,5	23	26	29	32,5	36,5	41	46	51
r ₃	2,5	3	4,5	5	6,5	8	9	10	12	13	15	16,5	18	20
r ₄	1	1,5	2	2,5	3	4	4,5	5	6	6,5	7	8	9	10
Cadena #	5y6	7y8	10	13	16	18y20	23	26	28y30	33	33	39y42	45	48y51

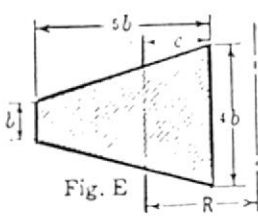
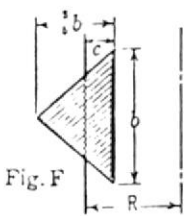
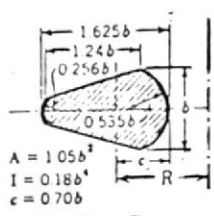
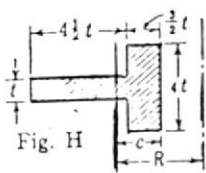
ENGRANAJES DE $14\frac{1}{2}^\circ$ DE 16 Y 20 DIENTES


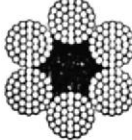
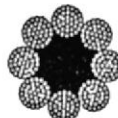


ENGRANAJES DE 20° DE 16 Y 20 DIENTES



VALORES DE K PARA DIFERENTES SECCIONES Y DIFERENTES RÁDIOS DE CURVATURA

Sección	$\frac{R}{c}$	Factor K		y_o
		Fibra interior	Fibra exterior	
 <p>Fig. E</p>	1.2	3.14	0.52	0.352R
	1.4	2.29	0.54	0.243R
	1.6	1.93	0.62	0.179R
	1.8	1.74	0.65	0.138R
	2.0	1.61	0.68	0.110R
	3.0	1.34	0.76	0.050R
	4.0	1.24	0.82	0.028R
	6.0	1.15	0.87	0.012R
	8.0	1.12	0.91	0.0060R
	10.0	1.10	0.93	0.0039R
 <p>Fig. F</p>	1.2	3.26	0.44	0.361R
	1.4	2.39	0.50	0.251R
	1.6	1.99	0.54	0.186R
	1.8	1.78	0.57	0.144R
	2.0	1.66	0.60	0.116R
	3.0	1.37	0.70	0.052R
	4.0	1.27	0.75	0.029R
	6.0	1.16	0.82	0.013R
	8.0	1.12	0.86	0.0060R
	10.0	1.09	0.88	0.0039R
 <p>Fig. G</p>	1.2	3.65	0.53	0.269R
	1.4	2.50	0.59	0.184R
	1.6	2.08	0.63	0.136R
	1.8	1.85	0.66	0.106R
	2.0	1.69	0.69	0.085R
	2.5	1.49	0.74	0.0535R
	3.0	1.38	0.78	0.0373R
	4.0	1.27	0.83	0.0209R
	6.0	1.19	0.90	0.0091R
	8.0	1.14	0.93	0.00524R
10.0	1.12	0.96	0.00329R	
 <p>Fig. H</p>	1.2	3.63	0.58	0.418R
	1.4	2.54	0.63	0.299R
	1.6	2.14	0.67	0.229R
	1.8	1.89	0.70	0.183R
	2.0	1.73	0.72	0.149R
	3.0	1.41	0.79	0.069R
	4.0	1.29	0.83	0.040R
	6.0	1.18	0.88	0.018R
	8.0	1.13	0.91	0.010R
	10.0	1.10	0.92	0.0065R

Órganos de tracción y de sustentación	CABLES NORMALIZADOS DE ACERO COMPOSICIÓN Y RESISTENCIA A LA ROTURA (1.º)							
	Composición del cable	Diámetro del cable d mm.	Diámetro de los alambres d' mm.	Sección total de los alambres A mm ²	Peso por metro de cable g kg.	Resistencia de los alambres kg/mm ²		
						130	160	180
						Resistencia del cable a la rotura kg.		
 6 x 19 = 114 alambres y un alma de cáñamo	6,5	0,4	14,3	0,135	1860	2300	2550	
	8	0,5	22,4	0,21	2900	3600	4050	
	9,5	0,6	32,2	0,30	4200	5150	5800	
	11	0,7	43,9	0,41	5700	7000	7900	
	12,5	0,8	57,3	0,54	7450	9150	10300	
	14	0,9	72,5	0,68	9450	11600	13050	
	16	1,0	89,5	0,85	11650	14300	16100	
	17	1,1	108,3	1,02	14100	17350	19500	
	19	1,2	128,9	1,22	16750	20600	23200	
	20	1,3	151,3	1,43	19650	24200	27250	
	22	1,4	175,5	1,66	22800	28050	31600	
	 6 x 37 = 222 alambres y un alma de cáñamo	9	0,4	27,9	0,26	3650	4450	5000
		10	0,45	35,3	0,34	4600	5650	6350
11		0,5	43,6	0,41	5650	7000	7850	
12		0,55	52,7	0,50	6850	8450	9500	
13		0,6	62,8	0,59	8150	10050	11300	
14		0,65	73,7	0,70	9600	11800	13250	
15		0,7	85,4	0,81	11100	13650	15350	
16		0,75	98,1	0,93	12750	15700	17650	
18		0,8	111,6	1,06	14500	17850	20100	
20		0,9	141,2	1,34	18350	22600	25400	
22		1,0	174,4	1,65	22650	27900	31400	
24		1,1	211,0	2,00	27450	33750	38000	
27		1,2	251,1	2,38	32650	40200	45200	
29		1,3	294,7	2,80	38300	47150	53050	
31		1,4	341,7	3,24	44400	54650	61500	
33		1,5	392,3	3,72	51000	62750	70600	
35	1,6	446,4	4,24	58050	71400	80350		
37	1,7	503,9	4,78	65500	80600	90700		
40	1,8	564,9	5,36	73450	90400	101700		
42	1,9	629,4	5,97	81800	100700	113300		
44	2,0	697,4	6,62	90650	111600	125550		
 8 x 37 = 296 alambres y un alma de cáñamo	16	0,6	83,7	0,84	10900	13400	15050	
	19	0,7	113,9	1,14	14800	18200	20500	
	20	0,75	130,8	1,31	17000	20950	23550	
	21	0,8	148,8	1,49	19350	23800	26800	
	23	0,85	168,0	1,68	21850	26900	30250	
	25	0,95	209,8	2,10	27250	33550	37750	
	27	1,0	232,5	2,32	30250	37200	44850	
	30	1,1	281,3	2,81	36550	45000	50650	
	32	1,2	334,8	3,35	43500	53550	60250	
	35	1,3	392,9	3,93	51050	62850	70700	
	37	1,4	455,7	4,56	59200	72900	82000	
	40	1,5	523,1	5,24	68000	83700	94150	
	43	1,6	595,1	5,95	77350	95200	107100	
	45	1,7	671,9	6,72	87350	107500	120950	
	48	1,8	752,2	7,52	97800	120350	135400	
	51	1,9	839,2	8,39	109100	134300	151050	
54	2,0	929,9	9,30	120900	148800	167400		
58	2,2	1125,1	11,25	146250	180000	202500		