

39
24



Universidad Nacional Autónoma de México

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES
ARAGON

ANESAR, PROGRAMA PARA EL ANALISIS
SISMICO DE EDIFICIOS

T E S I S

PARA OBTENER EL TITULO DE:
INGENIERO CIVIL
P R E S E N T A :
FRANCISCO REYES TELLEZ

Asesor: Ingeniero Amilcar Galindo Solorzano
**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

CAPITULO I :	INTRODUCCION.	1
CAPITULO II :	DESCRIPCION SOBRE EL LENGUAJE C.	5
2.1	HISTORIA DEL LENGUAJE C.	6
2.2	VENTAJAS DEL LENGUAJE C.	7
2.3	DESCRIPCION DEL LENGUAJE C.	11
2.4	VARIABLES DEL LENGUAJE C.	13
2.5	OPERADORES EN LENGUAJE C.	15
2.6	FUNCIONES EN LENGUAJE C.	21
2.7	SENTENCIAS DEL LENGUAJE C.	22
2.8	SENTENCIAS DE CONTROL DE PROGRAMA.	27
2.9	LIBRERIA ESTANDAR DEL LENGUAJE C.	29
CAPITULO III :	METODO GENERAL DE LAS RIGIDECES.	30
3.1	DESCRIPCION GENERAL.	31
3.1.1	Conceptos básicos sobre análisis estructural.	32
3.1.2	Historia del método de las rigideces.	34
3.2	BASES TEORICAS DEL METODO DE LAS RIGIDECES.	35
3.2.1	Expresión general del método de la rigideces.	35
3.2.2	Matriz de rigideces globales de la estructura [K].	37
3.2.3	Matriz de rigidez acoplada de barra [HRA].	38
3.3	VECTOR DE FUERZAS EXTERNAS.	44
3.4	VECTOR DE DESPLAZAMIENTOS.	45
3.5	VECTOR DE FUERZAS INTERNAS.	47
CAPITULO IV :	DESARROLLO DE UN SUBPROGRAMA PARA EL ANALISIS DE MARCOS PLANOS.	50
4.1	DESCRIPCION GENERAL.	51
4.2	DIAGRAMA DE BLOQUES.	52
4.3	ENTRADA DE DATOS.	53
4.4	CALCULO DEL NUMERO DE DESPLAZAMIENTOS.	55
4.5	CALCULO DEL ANCHO DE BANDA.	56
4.6	ENSAMBLE DE LA MATRIZ DE RIGIDEZ GLOBAL DE LA ESTRUCTURA.	58
4.7	ENSAMBLE DEL VECTOR DE FUERZAS EXTERNAS.	60
4.8	CALCULO DEL VECTOR DE DESPLAZAMIENTOS.	61
4.9	CALCULO DEL VECTOR DE FUERZAS INTERNAS.	65
4.10	SALIDA DE DATOS.	67
4.11	PROGRAMA FUENTE EN LENGUAJE C.	68
4.12	EJEMPLO.	72
CAPITULO V :	DESARROLLO DE UN SUBPROGRAMA PARA EL ANALISIS SISMICO ESTATICO.	85
5.1	DESCRIPCION GENERAL.	86
5.2	MODELO ANALITICO DE LA ESTRUCTURA.	88
5.2.1	Idealización de un edificio real como un sistema plano de n grados de libertad.	88
5.2.2	Rigideces de entrepiso.	90
5.2.3	Masas de piso.	91
5.3	RIGIDECES DE ENTREPISO EN MARCOS PLANOS.	92
5.3.1	Ecuaciones de Wilbur.	94
5.3.2	Obtención de rigideces de entrepiso con programa de análisis estructural estático.	97
5.3.3	Comparación de métodos para la obtención de rigideces de entrepiso.	99
5.3.4	Subprograma de rigideces de entrepiso.	104
5.3.4.1	Diagrama de bloques.	104
5.4	ESPECTRO DE DISEÑO.	106
5.4.1	Zona sísmica en la que se ubica la estructura.	107
5.4.2	Tipo de terreno en que se desplanta la estructura.	108
5.4.3	Clasificación de la estructura según su importancia.	109
5.4.4	Espectros de diseño para la República Mexicana.	110

5.4.5	Subprograma de espectro de diseño.	111
5.4.5.1	Variables del subprograma.	112
5.4.5.2	Diagrama de flujo.	112
5.4.5.3	Programa fuente en Lenguaje C.	114
5.5	FACTOR DE DUCTILIDAD.	115
5.5.1	Método práctico para la consideración de la ductilidad en los espectros de diseño.	115
5.5.2	Factores de ductilidad en los reglamentos mexicanos.	118
5.5.3	Reducción de fuerzas sísmicas.	120
5.5.4	Subprograma de reducción de fuerzas sísmicas.	121
5.5.4.1	Variables del subprograma.	121
5.5.4.2	Diagrama de flujo.	122
5.5.4.3	Programa fuente en Lenguaje C.	123
5.6	ANÁLISIS SÍSMICO ESTÁTICO.	124
5.6.1	Método del análisis sísmico estático.	125
5.6.2	Subprograma de análisis sísmico estático.	128
5.6.2.1	Variables del subprograma.	128
5.6.2.2	Diagrama de flujo.	129
5.6.2.3	Programa fuente en Lenguaje C.	131
5.7	EJEMPLO.	132
CAPITULO VI : DESARROLLO DE UN SUBPROGRAMA PARA EL ANÁLISIS SÍSMICO DINAMICO.		138
6.1	DESCRIPCIÓN GENERAL.	139
6.2	ANÁLISIS MODAL.	141
6.2.1	Método de Hozler.	148
6.2.1.1	Subprograma del método de Hozler.	152
6.2.1.2	Diagrama de flujo.	153
6.2.1.3	Programa fuente en Lenguaje C.	154
6.2.2	Subprograma de análisis modal.	155
6.2.2.1	Diagramas de flujo.	157
6.3	ANÁLISIS SÍSMICO DINAMICO.	160
6.3.1	Cálculo de valores característicos.	160
6.3.2	Selección del espectro de diseño.	161
6.3.3	Cálculo de las aceleraciones máximas elásticas para el cálculo de las fuerzas.	161
6.3.4	Cálculo de los factores de participación.	161
6.3.5	Desplazamientos máximos de los pisos en cada modo i.	162
6.3.6	Desplazamiento relativo de los entrepisos en cada modo.	162
6.3.7	Cortantes de entrepiso para cada modo.	163
6.3.8	Cortantes totales de entrepiso.	163
6.3.9	Fuerzas en cada piso, por sismo.	163
6.3.10	Desplazamientos relativos.	164
6.3.11	Desplazamientos totales relativos.	164
6.3.12	Diagrama de flujo.	165
6.3.13	Programa fuente en Lenguaje C.	174
6.4	EJEMPLO.	179
CAPITULO VII : DESARROLLO DE UN SUBPROGRAMA PARA LA DISTRIBUCIÓN DE CORTANTES DE ENTREPISO.		188
7.1	DESCRIPCIÓN GENERAL.	189
7.2	SUPERPOSICIÓN DE CORTANTES SÍSMICOS.	192
7.2.1	Subprograma.	192
7.3	CENTRO DE CORTANTES.	194
7.3.1	Subprograma.	195
7.4	CENTRO DE TORSIÓN.	197
7.4.1	Subprograma.	199
7.5	DISTRIBUCIÓN DE CORTANTES.	201
7.5.1	Distribución de cortantes.	201
7.5.2	Excentricidades reglamentarias.	207
7.6	EFFECTOS BIDIRECCIONALES.	209
7.6.1	Subprograma.	209

INDICE

página

7.7	SUBPROGRAMA.	211
	7.7.1 Variables del subprograma.	211
	7.7.2 Programa fuente en Lenguaje C.	212
7.8	EJEMPLO.	216
CAPITULO VIII : CONCLUSIONES.		220
APENDICE A.		223
APENDICE B.		240
BIBLIOGRAFIA.		251

CAPITULO I: INTRODUCCION

CAPITULO I

INTRODUCCION

El objetivo fundamental de la presente tesis es plantear el desarrollo de un programa de cómputo para el análisis sísmico de edificios, para este fin son presentadas las bases teóricas de los métodos empleados, así como su representación e implementación realizada en el lenguaje de programación C.

Los métodos de análisis estructural empleados son métodos comúnmente utilizados en la práctica profesional para el análisis sísmico de edificios en México, su empleo generalmente se lleva a cabo en forma manual, de aquí que se halla contemplado la factibilidad de elaborar este trabajo.

Es importante entender las bases y principios teóricos que sustentan el análisis sísmico de edificios, no obstante, el empleo de éste programa de análisis puede ser correcto sin entender completamente su teoría. A pesar de ello se requieren algunos estudios básicos para poder interpretar y emplear los resultados, de lo contrario el calculista puede perder la noción física del problema que está resolviendo, corriendo el riesgo de determinar en forma errónea los datos que alimentan al programa, además de no contar con la percepción necesaria para juzgar con criterio los resultados generados, por lo que no comprendería si estos datos son o no realistas.

Aún cuando se han realizado las verificaciones pertinentes, es importante tener en cuenta que es responsabilidad del calculista verificar los resultados que genere el programa ANESAR, debido a que como en cualquier programa, se pueden dar resultados erróneos bajo ciertas condiciones que no se hayan previsto al diseñar el programa. Sin embargo, la forma de programación estructurada con la cual se desarrolló el programa reduce considerablemente la posibilidad de errores en la programación que no hayan sido contemplados.

La estructuración del programa ANESAR está constituida por subprogramas en código fuente escritos en Lenguaje C, cada uno de ellos es tratado como subrutina independiente, si alguna persona requiere alguna subrutina en particular la podría aislar fácilmente, solo tendría que crear otra subrutina que pidiera los parámetros de entrada (INPUT) y mostrará los resultados de salida (OUTPUT), esta modificación sería muy fácil de implementar, de acuerdo a las necesidades particulares de la estructura a analizar.

La filosofía con la cual fue desarrollado el programa ANESAR cuenta con características indispensables y necesarias para el análisis de edificios complejos. En general los rasgos deseables del programa son las siguientes:

- Fácil empleo del programa por parte del usuario.
- Eficiente uso de los posibles recursos de la computadora.
- Emplear métodos modificados para acelerar el tiempo de ejecución del análisis de un edificio.
- Aumentar la velocidad de ejecución del programa.

El contenido básico del trabajo presentado lo podríamos resumir de la siguiente manera:

- **Capítulo II :** Descripción del Lenguaje C, se plantea con el fin de que el lector pueda comprender básicamente los programas a que se hacen referencia, posteriormente tendría la opción de modificarlos de acuerdo a sus necesidades particulares.
- **Capítulo III :** Desglose de las bases teóricas del método de las rigideces, no se trata de desarrollar paso a paso el origen del método, lo que se busca es dar las herramientas teóricas básicas y necesarias que representan al método con el fin de una mejor comprensión, las personas que tengan la inquietud de conocer el método de una forma más teórica se podrían remitir a la bibliografía referida. El método de las rigideces es un método analítico auxiliar que nos sirve como herramienta para realizar el análisis sísmico, aunque no es en sí mismo un método de análisis sísmico. Servirá para encontrar las rigideces de entrepiso y para calcular los elementos mecánicos que ocasiona el sismo sobre la estructura.
- **Capítulo IV :** Desarrollo del método de las rigideces desde el punto de vista de programación. A diferencia del capítulo anterior, se explican los métodos numéricos que representan a las bases teóricas del capítulo III. Se utiliza para una mejor explicación un ejemplo sencillo realizado manualmente, utilizando la misma filosofía del análisis matricial que se realiza en el programa de rigideces. Estos métodos de análisis estructural matricial empleados, ofrecen una sencillez metódica y a la vez una elegancia matemática, con la única desventaja de generar grandes sistemas de ecuaciones, lo que puede llegar a generar errores por redondeo numérico y enormes lapsos de espera. Actualmente las computadoras más modernas han reducido estas inconveniencias en gran medida, debido a procesadores y coprocesadores matemáticos mas rápidos y a compiladores numéricos mas eficientes.
- **Capítulo V :** Método del análisis sísmico estático, en general los métodos aquí empleados son métodos sugeridos por reglamentos, normas y manuales mexicanos, por lo que su empleo para otros países puede variar, sin embargo los principios son los mismos. Mediante el análisis sísmico estático se obtienen las fuerzas sísmicas estáticas. En el capítulo también se expone algunas de las bases necesarias para el análisis sísmico dinámico.

- **Capítulo VI** : Se expone el método del análisis sísmico dinámico para analizar a un modelo analítico representado por una estructura sencillamente acoplada, éste modelo idealiza al edificio real y lo simplifica.
- **Capítulo VII** : Se realiza la distribución de cortantes de entrepiso a los distintos elementos resistentes que conforman el edificio real. En ésta distribución son tomados en cuenta los efectos de torsión y efectos bidireccionales.

CAPITULO II : DESCRIPCION SOBRE EL LENGUAJE C

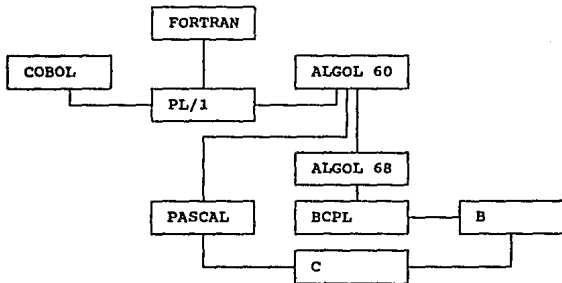


CAPITULO II

DESCRIPCION SOBRE EL LENGUAJE C

2.1 HISTORIA DEL LENGUAJE C :

El lenguaje C es un lenguaje de programación que fue desarrollado en base a los lenguajes europeos BCPL (creado por *Martin Richards*) y B (creado por *Ken Thompson*), este lenguaje fue implementado por *Dennis Ritchie* en una computadora DEC PDP-II, soportado por el sistema operativo UNIX, aunque en un principio fue elaborado para operar en este sistema, el Lenguaje C se ha hecho tan popular que actualmente esta disponible para casi todas las computadoras y sistemas operativos. En la década de los 70's tuvo su origen este lenguaje, durante algunos años el estándar de C fue proporcionado por la versión 5.00 para el sistema operativo UNIX, posteriormente comenzaron a realizarse muchas versiones de C para las distintas microcomputadoras que se crearon. El Instituto de Estándares Americano (ANSI) estableció un comité en el año de 1983, con el fin de crear un estándar que definiera totalmente el lenguaje C. Actualmente el estándar de la ANSI es el que rige en todos los principales compiladores del Lenguaje C. El siguiente diagrama muestra a los principales lenguajes que fueron básicos o influyeron en el desarrollo del lenguaje C .



2.2 VENTAJAS DEL LENGUAJE C :

Para el desarrollo del programa en que se basa el presente *trabajo de tesis* se escogió el lenguaje C debido a las **ventajas** que presenta con respecto a otros lenguajes de programación de uso común en el ámbito de la Ingeniería, como son : FORTRAN, BASIC, Pascal, FORTH y ALGOL.

Algunas de las **ventajas** del lenguaje C sobre otros lenguajes mencionados anteriormente son:

- Transportabilidad del código del lenguaje.
- Lenguaje de medio nivel.
- Lenguaje estructurado.
- No hace ningún tipo de chequeo de errores en tiempo de ejecución.
- Lenguaje especializado en permitir la manipulación directa de bits, bytes y punteros.
- Genera un código objeto más pequeño y rápido que cualquier otro compilador de alto nivel.
- Facilidad para crear una librería de funciones personales.
- Reducido número de palabras clave que forman al lenguaje.

La explicación de éstas ventajas se puede resumir de la siguiente forma:

- Debido a que en su inicio el lenguaje C fue debidamente controlado, en la actualidad está reglamentado por el ANSI y no ha sufrido grandes cambios en su estructura principal, por lo que es normal que un programa escrito en una máquina se transporte a otra computadora con un procesador distinto, otro sistema operativo o ambas cosas. A este proceso se le denomina *transportabilidad* y puede ser muy sencillo hasta extremadamente difícil, dependiendo de cómo se haya escrito originalmente el programa. Un programa no es fácilmente transportable si contiene muchas dependencias con la computadora, como lo son fragmentos de código que sólo funcionan con un procesador o sistema operativo determinado. Esta transportabilidad ahorra a su vez tiempo y dinero.
- Lenguaje de *medio nivel* no significa que sea menos potente, difícil de usar, o menos desarrollado que un lenguaje de alto nivel como lo son FORTRAN o Pascal; significa que el Lenguaje C es similar a los lenguajes de bajo nivel (ensambladores), en cuanto que tienen una representación simbólica del código máquina de una computadora; al mismo tiempo combinan elementos propios de un lenguaje de alto nivel. El lenguaje de *medio nivel* proporciona a los programadores un conjunto

mínimo de sentencias de control y de manipulación de datos, que pueden utilizarse para definir construcciones de alto nivel. Pero a diferencia de un lenguaje de alto nivel, donde se ha diseñado e incorporado previamente todas las funciones que probablemente va a requerir el programador, mientras que los lenguajes de bajo nivel obligan al programador a definirse ellos mismos todas éstas funciones, ya que ninguna va previamente incorporada.

Niveles de los lenguajes de programación :

Nivel alto	Nivel medio	Nivel bajo
Ada BASIC FORTRAN Pascal	C FORTH	Ensamblador

■ La característica que distingue a C como un lenguaje estructurado en bloques es la *compartimentalización de código* y datos. Lo que significa que el lenguaje puede separar y esconder del resto del programa toda la información y las instrucciones necesarias para realizar una tarea específica. Generalmente la compartimentación se consigue mediante subrutinas con variables locales, que son temporales. De esta manera, es posible escribir subrutinas de forma que los procesos ocurridos dentro de ellas no causen efectos laterales en otras partes del programa, de lo contrario se pueden generar errores en otras partes del programa que produzcan efectos laterales indeseables. Un lenguaje estructurado como C permite asimismo el indentar las sentencias, sin necesidad de requerir un campo estricto como ocurre con FORTRAN. Los lenguajes estructurados se consideran más modernos, llegando a ser considerada como característica de un lenguaje de programación antigua, el que no sea estructurado.

Hablando en un sentido estrictamente técnico, el término de lenguaje estructurado en bloques no se aplica al lenguaje C, dado que C no permite la creación de funciones dentro de funciones, por lo que no puede ser formalmente denominado lenguaje estructurado en bloques .

Lenguajes estructurados y no estructurados :

No estructurados	Estructurados
FORTRAN BASIC COBOL	Pascal C Modula-2 Ada

- El compilado del lenguaje C *no efectúa ningún chequeo de errores en tiempo de ejecución*, tales como revisar el límite de matrices o la compatibilidad en el tipo de los argumentos, estas tareas deben ser realizadas por el propio programador. Esta técnica de compilado hace que la ejecución sea más rápida que la de los demás compiladores, los cuales sí efectúan dichos chequeos.
- Los operadores que utiliza el lenguaje C hacen sencilla la *manipulación directa de bits, bytes y punteros*; esto permite un fácil acceso a las funciones de la computadora. Al contrario que otros lenguajes de alto nivel que operan sobre cadenas de caracteres para realizar una gran cantidad de sus funciones, mientras que el C puede operar directamente sobre los caracteres.
- La sintaxis con que se escribe un programa en lenguaje C es muy semejante a la utilizada por el lenguaje ensamblador, por lo que algunas instrucciones de C son idénticas a su similar en ensamblador, de tal forma que al compilar en C se obtienen ejecuciones rápidas y pequeñas, debido a que cada operación representa una tarea individual que ejecuta la computadora directamente en la mayoría de sus instrucciones. De hecho, un motivo para la invención del C fue proporcionar una alternativa a la programación en lenguaje ensamblador, que es un lenguaje que tiene la máxima flexibilidad, eficiencia y sobretodo rapidez, pero a su vez es más difícil trabajar con él, sobre todo cuando se desarrolla y depura un programa.
- El C permite definir subrutinas para llevar a cabo comandos de alto nivel. Estas subrutinas se llaman *funciones*. Un programador de C puede crear y mantener una librería única de funciones creada por él mismo, a la medida y con su propia personalidad. Debido a lo anterior, el lenguaje C permite crear una librería de subrutinas con funciones útiles y ya probadas, a las que se puede acceder desde cualquier programa que se escriba. Estas librerías auxilian a la creación de proyectos grandes que son administrados fácilmente, debido a que el lenguaje C permite y alienta a compilaciones separadas.
- El lenguaje C sólo está formado por *32 palabras clave*, que son las instrucciones que forman al lenguaje, debido a este reducido número de instrucciones se simplifica mucho el programar en este lenguaje, ya que hay pocas sentencias que recordar (la versión de BASIC para el IBM PC tiene 159 palabras clave).

Posiblemente la *razón más importante* por la que es utilizado el lenguaje C para la programación es porque se trata de un lenguaje para los programadores. Los programadores se ven atraídos hacia él porque no les impone restricciones y resulta ser entretenido. Debido a esto en los últimos años se ha utilizado también como lenguaje de programación

de propósito general. La sintaxis y presentación del código fuente (programa) del lenguaje C es muy legible; una vez familiarizado con él, se puede seguir fácilmente el flujo de control y la lógica de un programa, así como una fácil verificación de las operaciones de las subrutinas. Los listados de un programa en C aparecen claros; al contrario de otros escritos en BASIC o FORTRAN, que parecen estar desordenados y confusos. El C fue creado, influenciado y probado por programadores profesionales. C dota a los programadores de pocas restricciones, estructuras de bloques, funciones independientes y un compacto conjunto de palabras claves. Utilizando C, un programador puede estar muy cerca de la eficiencia del código ensamblador, y al mismo tiempo se tiene la estructuración de Pascal o ALGOL.

Todas las características y ventajas de C mencionadas anteriormente muestran el porqué se trata de un lenguaje de programación que *domina el desarrollo de software* en distintas áreas. Lo que ha llevado a C, a ser el lenguaje más popular entre los programadores profesionales de élite. Con este lenguaje se han creado paquetes comerciales muy exitosos, sobretodo programas de sistemas que hacen que la computadora sea capaz de realizar trabajo útil. Estos programas de sistemas en general abarcan :

- Procesadores de texto,
- Sistemas operativos,
- Intérpretes de lenguajes,
- Compiladores de lenguajes,
- Diseño asistido por computadora (CAD),
- Sistemas de gestión de base de datos (SGBD),
- Paquetes gráficos, etc.

En lo referente al área de *Ingeniería Civil* hasta hace pocos años se ha comenzado a utilizar este lenguaje con mucho éxito para desarrollar software en las distintas especialidades de la Ingeniería, sobretodo en el área *estructural*. Este lenguaje a venido a desplazar al lenguajes anteriormente muy utilizados en la Ingeniería, como lo son FORTRAN y BASIC, debido a las grandes ventajas mencionadas anteriormente.

2.3 DESCRIPCIÓN DEL LENGUAJE C :

Estructura del lenguaje C :

El lenguaje C utiliza una compartimentalización de código y datos. Lo que quiere decir que el lenguaje puede separar del resto del programa toda la información y las instrucciones necesarias para realizar una tarea específica. Esto se consigue mediante subrutinas con variables locales las cuales son temporales y sólo serán conocidas en la función en donde fueron declaradas. Por lo que es posible escribir subrutinas de tal forma que los procesos realizados dentro de ellas no causen efectos en otras partes del programa. El lenguaje C también permite utilizar variables globales, las cuales son reconocidas en todo el programa, aún dentro de las subrutinas, por lo que se debe de tener cuidado al manejarlas, pues éstas si pueden causar efectos indeseables en otras partes del programa.

Funciones :

El lenguaje C permite la construcción de bloques, en los cuales se realizan todos los procesos del programa. Esto permite que actividades específicas sean realizadas, definidas y codificadas por separado. Al programar de esta manera cada subrutina en C es depurada por separado, por lo que se puede confiar en que la función no tendrá errores, ni efectos laterales en otras partes del programa.

Bloque de código :

En C se utilizan los bloques de código, los cuales forman la estructura del programa. Un bloque de código es un grupo de sentencias de programas conectadas lógicamente y que se pueden tratar como una unidad. Estas son declaradas situando líneas de código (sentencias) entre una llave abierta y otra cerrada.

Punto y coma (;) :

En el lenguaje C, el punto y coma es un terminador; o sea que cada sentencia debe terminar con un punto y coma. Esto indica el final de una entidad lógica.

Llaves ({ }) :

El bloque de código definido con anterioridad es un conjunto de sentencias lógicamente relacionadas, van entre una llave abierta y otra cerrada. Considerado un bloque como un grupo de sentencias con un punto y coma después de cada una, entonces

tiene sentido el considerar que un bloque no va seguido por un punto y coma específico.

Posición :

El lenguaje C no reconoce el fin de línea como terminador. Esto significa que el C no tiene restricciones en cuanto a posiciones, lo que le permite al programador el agrupar sentencias por razones de claridad.

Comentarios :

Los comentarios en C pueden ser colocados en cualquier parte en un programa y deben ir entre dos marcas. La marca de inicio de comentario es `/*` y la de fin de comentario es `*/`.

Indentación :

Debido a que el lenguaje C acepta un formato libre, es indistinto donde se coloquen las sentencias dentro de una línea. Sin embargo, se ha tomado casi como un estándar el adoptar un estilo de indentación ya tradicional, que presenta más claros y legibles los programas de C.

Palabras reservadas clave :

El lenguaje C reserva para sí mismo las siguientes 32 palabras clave (27 del estándar de Ritchie y Kernighan, y 5 añadidas por el comité del ANSI), las cuales combinadas con su sintaxis, forman el lenguaje de programación C. Estas palabras reservadas se deben de representar siempre en minúsculas, debido a que en C las mayúsculas y minúsculas son distintas:

<i>auto</i>	<i>default</i>	<i>float</i>	<i>register</i>	<i>struct</i>	<i>volatile</i>
<i>break</i>	<i>do</i>	<i>for</i>	<i>return</i>	<i>switch</i>	<i>while</i>
<i>case</i>	<i>double</i>	<i>goto</i>	<i>short</i>	<i>typedef</i>	
<i>char</i>	<i>else</i>	<i>if</i>	<i>signed</i>	<i>union</i>	
<i>const</i>	<i>enum</i>	<i>int</i>	<i>sizeof</i>	<i>unsigned</i>	
<i>continue</i>	<i>extern</i>	<i>long</i>	<i>static</i>	<i>void</i>	

De la lista anterior de 32 palabras clave, 27 fueron definidas en la versión original de C (sistema UNIX). Las cinco siguientes fueron añadidas por el comité del ANSI: **enum**, **cont**, **signed**, **void** y **volatile**.

2.4 VARIABLES DEL LENGUAJE C :

Tipos de variables :

El lenguaje C soporta los siguientes siete tipos de estructuras de datos o variables:

Tipo de dato	Palabra clave equivalente
caracteres	char
entero corto	short int
entero	int
entero sin signo	unsigned int
entero largo	long int
punto flotante	float
doble punto flotante	double

Sintaxis de las variables :

- Los nombres de las variables deben ser cadenas de símbolos alfanuméricos de uno o más caracteres de longitud, la longitud máxima depende del compilador.
- El símbolo de subrayar (**_**) es el único permitido en el nombre de la variable (por ejemplo : *subrutina_1*).
- Las letras mayúsculas y minúsculas son diferentes (por ejemplo : *masa* y *MASA* son variables distintas en C).
- Todas las variables tienen que ser declaradas antes de ser usadas.

Combinaciones entre variables :

En C se pueden obtener combinaciones entre los tipos de datos incorporados, utilizando **struct** y **union**.

Modificadores de variables :

Existen tres modificadores de tipo en C, los cuales se emplean para alterar la forma de tratar las variables que declaran. Estos tres modificadores son: **extern**, **register** y **static**.

Matrices (Arrays) :

En C se pueden declarar matrices de cualquiera de los tipos mencionados anteriormente, también se pueden declarar matrices multidimensionales. La dimensión de la matriz se declara entre corchetes. Ejemplos :

int A[10]; sentencia que declara una variable entera llamada A, que contiene a 10 elementos.

2.5 OPERADORES EN LENGUAJE C :

Los operadores son un conjunto de símbolos o variables que señalan una operación a ser realizada; en particular designan el código de operación de una instrucción.

El lenguaje C soporta un conjunto de operadores que se pueden clasificar en las siguientes clases de operadores :

- Operadores aritméticos,
- Operadores relacionales y lógicos,
- Operadores de bajo nivel,
- Operadores de tipo puntero,
- Operadores de asignación, y
- Operadores varios.

Los operadores con el mismo nivel de precedencia se evalúan de izquierda a derecha.

Operadores aritméticos :

El lenguaje C tiene siete operadores aritméticos que le ayudan a evaluar expresiones matemáticas:

operador	acción
-	resta
+	suma
*	multiplicación
/	división
%	módulo de la división
-	decremento
++	incremento

Operadores relacionales y lógicos :

Estos operadores son utilizados para producir resultados del tipo VERDADERO/FALSO y son utilizados por lo general juntos. En C, cualquier número distinto de cero es tomado como VERDADERO. Sin embargo, un operador lógico o relacional en C produce el número 1 para VERDADERO y 0 para FALSO.

operadores relacionales	significado
>	mayor que
>=	mayor o igual que
<	menor que
<=	menor o igual que
==	igual
!=	diferente, o no igual

operadores lógicos	significado
&&	AND
	OR
!	NOT

Operadores de bajo nivel :

Este tipo de operadores es de gran ayuda, pero sólo unos cuantos lenguajes de programación los implementan. Estos operadores de bajo nivel manipulan los bits reales dentro de una cadena. Este tipo de operadores sólo se puede utilizar en variables de tipo entero o en caracteres.

operador	significado
&	AND
	OR
^	XOR
~	complemento a uno
>>	desplazamiento a la derecha
<<	desplazamiento a la izquierda

Operadores de asignación :

En el lenguaje C, el operador de asignación es el signo igual. Pero además permite una conveniente abreviatura para la asignación del tipo general :

variable1 = variable1 operador expresión;

simplificándola de la manera siguiente :

variable1 operador = expresión;

Operador ternario ? :

El operador **?** es un operador ternario que se utiliza para reemplazar la sentencia **if** de la siguiente forma general:

```
if expresión1 then variable = expresión2;  
    else variable = expresión3;
```

simplificándola de la manera siguiente :

```
variable = expresión1 ? expresión2 : expresión3;
```

Si la expresión1 es VERDADERA, entonces el valor de la expresión2 se asigna a la variable; en el caso de que la expresión1 sea FALSA entonces a la variable se le asigna el valor de la expresión3.

Operador en tiempo de compilación sizeof :

El operador **sizeof** es un operador monario de tiempo de compilación que devuelve, en bytes, de la variable o del especificador de tipo entre paréntesis al que precede.

Operador de puntero & y * :

Un puntero es la dirección de memoria de una variable. Una variable puntero es una variable específicamente declarada para contener un puntero a su tipo específico. El conocer la dirección de una variable puede ser de gran ayuda en ciertos tipos de rutinas. En C los punteros pueden proporcionar una rápida forma de referenciar los elementos de un array. También permiten a las funciones de C modificar los parámetros de llamada. Por último, dan soporte a estructuras de datos dinámicas.

- El primer operador de punteros es **&**, se trata de un operador monario que devuelve la dirección de memoria del operando .
- El segundo operador de punteros es *****, es el complementario de **&**. También es un operador monario que devuelve el valor de la variable ubicada en la dirección que se especifica .

La coma (,) como operador :

Como operador, la coma encadena varias expresiones. La parte izquierda del operador coma siempre se evalúa como void. Esto significa que la expresión de la parte derecha se convierte en el valor de la expresión total separada por coma. Esencialmente la coma produce una secuencia de operaciones. Cuando se usa la parte derecha de una sentencia de asignación, el valor asignado es el valor de la última expresión de la lista separada por comas.

Tabla de Operadores :

La siguiente tabla muestra todos los operadores utilizados por el lenguaje C y por sus funciones, los cuales son explicados según su orden de precedencia.

Operadores primarios:

(~), func(~)
x[~], y[~][~]
st.membr
ptr -> memb

Paréntesis, operador de argumento de función.
Especifica elementos de matrices.
Especifica miembros de la estructura.
Especifica miembros de la estructura usando punteros.

Operadores unarios:

*px
&x
+x,-x
++x,--x
x++,x--
~x
(type) x
sizeof (x)

Especifica el contenido de la variable indicado por el puntero.
Dirección de variable x.
Valor de x positivo/negativo.
+1/-1 usar variable x antes.
+1/-1 usar variable x después.
Inversión del bit.
Fuerza conversión/especificación de x.
Valor de la longitud de la variable x en bytes.

Operadores binario:

x*y,x/y,x%y
x+y,x-y
x<<y,x>>y
x<y,x<=y,x>y,x>=y
x==y
x!=y
x&y
x^y
x|y
x&&y
x||y

Multiplicación, división y módulo.
Adición, sustracción.
x bits a la izquierda/y bits a la derecha.
Operadores relacionales (VERDADERO=1, FALSO=0).
Igualdad (diferente=0, igual=1).
Desigualdad (diferente=1, igual =0).
Bit AND de x y y.
Bit XOR de x y y.
Bit OR de x y y.
Lógico AND de x y y.
Lógico OR de x y y.

Operador condicional

(Operador ternario):

x?:y:z

y si x es VERDADERO, z si x es FALSO.

Operadores de asignación:

x=y
x*=y <=> x=x*y
x/=y <=> x=x/y
x%=y,x+=y,x-=y,x<<=y,x>>=y,x&=y,x|=y,x^=y.

Asignar y a variable x.
Multiplica x por y y se lo asigna a la variable x.
Divide x por y y se lo asigna a x.

Precedencia de los operadores de C :

Mayor	<code>() , [] , =></code> <code>! , ~ , ++ , -- , (tipo) , * , & , sizeof</code> <code>* , / , %</code> <code>+ , -</code> <code><< , >></code> <code><<= , >>=</code> <code>= , !=</code> <code>&</code> <code>^</code> <code>!</code> <code>&&</code> <code> </code> <code>?:</code>
Menor	<code>= , + , += , * , *= , / , /= , % = , > > = , < < = , & = , ^ = , =</code>

Los operadores con el mismo nivel de precedencia se evalúan de izquierda a derecha de la sentencia. Todos los operadores se asocian de izquierda a derecha excepto los unarios (`*` , `&` y `-`) y el operador ternario `?:` , los cuales se asocian de derecha a izquierda.

2.6 FUNCIONES EN LENGUAJE C :

Los programas en lenguaje C están formados por un conjunto de una o más funciones definidas por el usuario. En C la primera función es `main()`, y se le puede considerar como el programa principal. Todos los programas en lenguaje C consisten de una o más funciones. La única función que debe estar absolutamente presente es la denominada `main()`, siendo la primera función que es llamada cuando comienza la ejecución del programa. En un programa en C bien escrito, `main()` debe contener, en esencia, todo lo que el programa hace básicamente .

La forma general de una función es la siguiente:

```
nombre_función (lista de parámetros)
declaración de parámetros
{
    cuerpo de la función
}
```

Si la función no tiene ningún parámetro, no es necesaria ninguna declaración de parámetros. Todas las funciones terminan y retornan automáticamente a la función que hizo la llamada cuando se encuentra la llave de cierre, aunque se puede forzar el retorno de la subrutina antes de esto utilizando la sentencia `return`.

El lenguaje C relaciona a las funciones con dos tipos generales de variables: globales y locales. Una variable local está disponible para ser utilizada por todas las funciones del programa, mientras que una variable local sólo se conoce y se utiliza por la función en la que ha sido declarada. Una variable global tiene que ser declarada fuera de todas las funciones, incluyendo la función `main()`, normalmente se colocan al inicio del programa, lo que las hace más fácil de leer.

Reglas de ámbito de las funciones en C :

Las reglas de ámbito de un lenguaje son las reglas que controlan si un fragmento de código conoce o tiene acceso a otro fragmento de código o de datos. En C, cada función es un bloque de código discreto. El código de una función es privado a esa función y no se puede acceder a él mediante una expresión de otra función, a menos que se haga a través de una llamada a esa función. El código que comprende el cuerpo de una función está oculto al resto del programa y, a no ser que se usen datos o variables globales, no puede ser afectado por otras partes del programa, ni afectarlas.

2.7 SENTENCIAS DEL LENGUAJE C :

break

Sentencia que se utiliza para salir de un bucle **do**, **for**, **while** o **switch** evitando de esta manera la ejecución normal del bucle. Siempre sale del bucle más interior, sin importar como estén anidados.

case

Es un complemento a la sentencia **switch**.

char

El tipo de datos **char** declara variables de caracteres.

continue

Esta sentencia se utiliza para saltar partes del código en un bucle y forzar a que se vuelva a evaluar la condición inicial.

default

Se usa junto con la sentencia **switch** para indicar al bloque de código que se debe ejecutar por defecto si no se encuentra una comparación positiva en **switch** (ver **switch**).

do

Esta sentencia es una de las tres estructuras de bucle disponibles en lenguaje C. El bucle **do** es el único bucle en C que se ejecuta al menos una vez ya que la condición se comprueba al final del mismo.

double

El especificador de tipo de datos **double** declara variables de punto flotante como de doble precisión.

else

Complemento a la sentencia **if** (ver **if**).

entry

Esta sentencia normalmente no se implementa en los compiladores de C, aunque se reserva para un posible uso en el futuro.

extern

El modificador de tipo de datos **extern** indica al compilador que la variable está declarada en otro lugar del programa. Se utiliza con archivos compilados por separado, pero que compartan los mismos datos globales, siendo enlazados conjuntamente.

float

float es un especificador de tipo de dato que declara variables de punto flotante.

for

El bucle **for** permite una inicialización y un incremento automáticos de la variable contador. Su formato general es:

```
for(inicialización;condición;incremento) {  
    bloque de sentencias  
}
```

goto

La sentencia **goto** hace que la ejecución del programa salte a la etiqueta especificada en la sentencia **goto**. El formato general es :

```
goto etiqueta;  
..  
..  
etiqueta:
```

Las etiquetas deben terminar con dos puntos y no deben entrar en conflicto con palabras, claves o nombres de funciones. Sólo puede saltar dentro de la función en que se encuentra, sin saltar de una a otra.

if

Se trata de una sentencia condicional. El formato general de la sentencia **if** es:

```
if (condición) {  
    bloque1 de sentencias  
}  
else {  
    bloque2 de sentencias  
}
```

int

Especificador de tipo **int** declara variable enteras.

long

Se trata de un modificador de tipo de dato **long** que declara variables enteras como de doble longitud.

register

Modificador de declaración que fuerza a que un entero o un carácter se sitúe en un registro de la CPU, en lugar de la memoria, esto hace que se incremente su velocidad. **register** sólo se puede usar con variables globales.

return

Sentencia que se utiliza para que se salga de una función, pudiendo transferir un valor de vuelta a la rutina que la había llamado. Cuando se encuentra la ejecución un **return**, se sale de la función, saltando cualquier código de la función.

short

Modificador de tipo de datos **short** declara enteros de sólo un byte de longitud.

sizeof

Es un operador en tiempo de ejecución y devuelve la longitud de la variable a la que precede. Se utiliza frecuentemente para ayudar a generar código transportable cuando ese código depende del tamaño de los tipos de datos incorporados en el lenguaje C.

static

El modificador de tipo de datos **static** pide al compilador que cree una posición de almacenamiento permanente para la variable local a la que precede. El uso de **static** permite que la variable especificada mantenga su valor entre llamadas a distintas funciones.

struct

Inicializa complejas variables llamadas estructuras, que se constituyen por uno o más elementos de los siete tipos de datos básicos, de otras estructuras y de uniones, así como de tipos definidos por el usuario y arrays. En este tipo de estructura los elementos individuales se referencian mediante los operadores de punto o de flecha. El formato general de una estructura es :

```
struct nombre_estructura {  
    tipo de elemento1;  
    tipo de elemento2;  
    .  
    .  
    tipo de elementon;  
} nombre_variable_estructura;
```

switch

Se trata de la sentencia del lenguaje C de múltiple ramificación. Dirige la ejecución en una de varias formas posibles. Cada conjunto de sentencias que definen un **case** puede ser de una o varias sentencias. La última parte **default** puede ser opcional. **switch** compara la variable con todas las constantes. Si se encuentra una coincidencia, entonces se ejecuta el conjunto de sentencias correspondiente. Si se omite la sentencia **break**, la ejecución continúa hasta que termine el **switch** o se encuentre otro **break**. De esta manera cada **case** actúa como una etiqueta. El formato general de la sentencia es :

```

switch (variable) {
    case (constante1): conjunto 1 de sentencias;
        break;
    case (constante2): conjunto 2 de sentencias;
        break;
        .
        .
    case (constanten): conjunto n de sentencias;
        break;
    default: sentencias por defecto;
}

```

typedef

permite crear un nuevo nombre para un tipo de datos existente. Este tipo de datos puede ser: uno de los incorporados por el mismo lenguaje C, un nombre de una estructura o de una unión. El formato general que describe a **typedef** es:

```
typedef especificador_tipo nuevo_nombre;
```

union

asigna dos o más variables a la misma posición de memoria. El tipo de formato para esta sentencia tiene la misma estructura que para **struct**. El formato general es:

```

union nombre_union {
    tipo de elemento1;
    tipo de elemento2;
    .
    .
    tipo de elementon;
} nombre_variable_union;

```

unsigned

Sentencia que se define como modificador de tipo de dato, el cual indica al compilador que elimine el bit de signo de un entero y así use todos los bits para la aritmética. Esto dobla al máximo el valor del entero definido, pero lo limita únicamente a números positivos. El formato general es:

```
unsigned tipo_de_dato nombre_variable;
```

while

Se trata de una estructura de control de C. **while** define un bucle y tiene el siguiente formato:

```

while (condición) {
    bloque de sentencias;
}

```

while evalúa la condición al principio del bucle. De esta manera, si la condición es **FALSA** al inicio, el bucle no se ejecutará ninguna vez. Este tipo de condición inicial puede estar formada de cualquier expresión. Si el bloque de sentencias es una sola sentencia, entonces se puede omitir las llaves.

2.8 SENTENCIAS DE CONTROL DE PROGRAMA :

El lenguaje C incorpora un amplio y variado conjunto de *sentencias de control* de programa. El estándar ANSI clasifica las sentencias de C en los siguientes grupos, sentencias de:

- selección
- iteración
- salto
- etiquetado
- expresión
- bloque

Sentencias de selección :

Entre las sentencias de selección se incluyen **if** y **switch** , también son conocidas con el término de sentencias condicionales o sentencias de selección .

Sentencias de iteración :

Las sentencias de iteración son : **while**, **for** y **do while**, también son conocidas como sentencias de bucles o de ciclos. Estas sentencias permiten que un conjunto de instrucciones sean ejecutados hasta que se alcance cierta condición .

Sentencias de salto :

Las sentencias de salto son : **break**, **continue**, **goto** y **return**. Son utilizadas para llevar a cabo un salto incondicional, de ellas, se puede usar **return** y **goto** en cualquier parte del programa, mientras que **break** y **continue** se deben usar junto con una sentencia de bucle .

Sentencias de etiquetado :

Estas sentencias son : **case**, **default** y **label**.

Sentencias de expresión :

Son las sentencias compuestas por expresiones válidas en el lenguaje C .

Sentencias de bloque :

Son simplemente bloques de código .

La mayoría de estas sentencias, basan su operación en el resultado de alguna prueba lógica o condicional, la cual determina la acción que han de llevar a cabo. Una expresión condicional tiene como resultado un valor cierto o falso. A diferencia de muchos lenguajes de programación, en C cualquier valor distinto de cero es cierto, incluyendo los números negativos.

2.9 LIBRERÍA ESTÁNDAR DEL LENGUAJE C :

Cada compilador del lenguaje C maneja su propia *librería estándar* que proporciona al usuario las funciones necesarias para llevar a cabo las tareas más comunes. Este tipo de librerías abarcan muchas de las funciones de propósito general. Cuando se utiliza una función que no ha sido previamente identificada en el programa, el compilador de C retiene el nombre de la función y lo busca en la *librería estándar*, si lo encuentra entonces lo añade a su código objeto. La parte del compilador que lleva a cabo este proceso es el encadenador.

Al mencionar *librería estándar* de C no se tiene una estructura bien definida sobre ella, debido a que en el lenguaje C no hay nada que defina exactamente qué funciones deben de ir en dicha librería ni cómo debe ser su funcionamiento. Los diseñadores de cada compilador de C deciden arbitrariamente al respecto. Sin embargo la mayoría de los compiladores de C utilizan la *librería estándar* de funciones del sistema UNIX. Debido a que C fue inicialmente desarrollado bajo este sistema operativo, y aunque no existe una conexión directa, esa versión de librería de C es la que generalmente se considera como *librería estándar*.

Muchas de las funciones necesarias para programar en C, se encuentran en la *librería (biblioteca) estándar*. Se comportan como bloques constitutivos que simplemente hay que combinar. Si el programador de C escribe una función que va a usar a menudo, puede colocarla en alguna librería. Algunos compiladores permiten colocarla en la *librería estándar*; mientras que otros necesitan crear una librería adicional. De cualquier forma el código fuente de tal librería quedará listo para ser usado una y otra vez.

Las funciones que se encuentran en la librería de la mayoría de los compiladores lo están en formato reubicable. Esto significa que las direcciones de memoria de las diferentes instrucciones de código máquina no han sido definidas de forma absoluta, por lo que mantienen toda la información sobre desplazamientos. Cuando se enlaza (encadena) un programa con funciones de la biblioteca estándar, se usan esos desplazamientos de memoria para definir las direcciones reales. Sin embargo, no se necesita una mayor explicación sobre el proceso concreto de reubicación para poder programar en C.

CAPITULO III : METODO GENERAL DE LAS RIGIDECES

CAPITULO III

METODO GENERAL DE LAS RIGIDECES

3.1 DESCRIPCION GENERAL :

En el presente capítulo se dan nociones sobre la teoría del método general de las rigideces tratado en forma matricial, con el fin de darle un enfoque más práctico que teórico, debido a que actualmente existen una gran variedad de libros y publicaciones sobre las bases teóricas de esta materia, algunos de los cuales son referidos en la bibliografía, para aquellas personas interesadas en profundizar más en el estudio de las bases teóricas del método.

El método general de las rigideces es un método de análisis estructural que nos permite analizar una estructura para determinar sus respuestas internas por efectos de la acción de fuerzas externas (condiciones de carga).

Para comprender los métodos matriciales de análisis estructural se requieren de los fundamentos que han servido de base para éstos. Por tanto, el objetivo del capítulo es la introducción a los métodos matriciales como un medio para el análisis estructural

Los métodos de análisis estructural matricial aquí tratados ofrecen una sencillez metódica y a su vez elegancia matemática, con la única desventaja de generar grandes sistemas de ecuaciones, lo que puede llegar a generar errores por redondeo numérico y enormes lapsos de espera. Actualmente las computadoras más modernas han reducido estas inconveniencias en gran medida, debido a procesadores más rápidos y a compiladores numéricos más eficientes.

La finalidad de plantear en este trabajo el método de las rigideces es el de realizar el análisis estructural de marcos planos, para la obtención de las rigideces de entrepiso y los elementos mecánicos de diseño, para posteriormente realizar el análisis sísmico de edificios. Se puede emplear este método para el análisis estructural de cualquier tipo de estructura reticular definida en el plano.

3.1.1 Conceptos básicos sobre Análisis Estructural:

Para comprender mejor este método, se inicia el capítulo dando algunas definiciones básicas de carácter general en el Análisis Estructural.

Topología Estructural:

Es la descripción de la ubicación geométrica de una estructura, en un plano o en el espacio. Se considera que una estructura está compuesta, para efectos de análisis matricial, de un conjunto de miembros o elementos estructurales (barras: columnas o traveses) conectados entre sí en puntos llamados nodos. Debido a que los nodos son los puntos de referencia de la topología estructural, se seleccionan los desplazamientos en los nodos para este fin. La descripción topológica de la estructura se puede hacer con referencia a un sistema coordenado global o general, o bien a varios sistemas coordenados locales.

Propósito principal de una estructura:

El fin de una estructura es el de transmitir y resistir eficazmente las cargas externas (cargas permanentes, variables y accidentales). Estas cargas o fuerzas externas producen traslaciones y giros en los nodos, llamados desplazamientos externos, además de esfuerzos y deformaciones en los miembros, llamados fuerzas internas y desplazamientos internos.

Estructuras Reticulares:

Son aquellas estructuras que están constituidas por un conjunto de miembros cuya característica es que su sección longitudinal es grande en comparación con las dimensiones de su sección transversal.

Tipos de Estructuras Reticulares:

Las estructuras reticulares se pueden clasificar, según su tipo de estructuración, entre las más comunes están: armaduras planas, armaduras espaciales, vigas, marcos planos, marcos espaciales, parrillas, etc.

Marco Plano:

El marco plano es una estructura reticular que está formada por miembros en un solo plano y cuyo eje de simetría está en el mismo plano. Se considera que los nodos entre los miembros son conexiones rígidas, aunque dichas conexiones pueden tener un estado elástico o plástico. Estas fuerzas y desplazamientos que se presentan en un marco están en el mismo plano de la estructura, mientras que los momentos que actúan sobre el marco tienen sus vectores de momento normales al plano. Los esfuerzos internos resultantes (elementos mecánicos) que actúan en cualquier

sección de un miembro del marco plano pueden consistir, en general, de fuerza axial, fuerza cortante y momento flexionante. En el análisis de los marcos planos no son considerados los efectos de los *momentos torsionantes*.

Convención de sistemas coordenados de referencia :

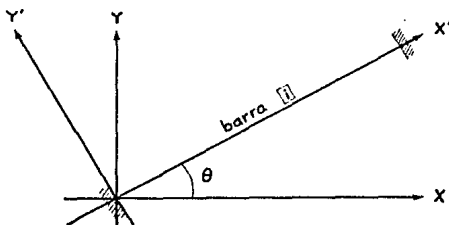
Los sistemas empleados en las construcciones del método de las rigideces, únicamente se utilizan para referir la rotación de ejes en cada concepto, por lo que la traslación que también puede representar un sistema de referencia coordenada no es tomada en cuenta.

En el sistema global X-Y de la estructura, están referenciados los siguientes conceptos del método de las rigideces:

- Ubicación geométrica de la estructura.
- Matriz de rigideces global de la estructura.
- Vector de fuerzas externas.
- Desplazamientos en los nodos.
- Matriz de rigidez acoplada de barra $[a]^T[k][a]$.
- Matriz de continuidad.

Los distintos sistemas locales X'-Y' de cada barra están en función de la dirección nodo origen a nodo destino de cada elemento de la estructura, los conceptos del método de rigideces que tienen como referencia a éstos sistemas son los siguientes:

- Vector de fuerzas internas (P_x, P_y, M_z) o elementos mecánicos (N,V,M) de cada elemento.
- Matriz de continuidad.
- Matriz de rigidez de barra $[k]$.
- Deformaciones de barra (desplazamientos locales).



3.1.2 Historia del Método de las Rigideces:

En el año de 1915, el profesor George A. Manay presentó en una publicación sobre Ingeniería Estructural de la Universidad de Minesota el método de análisis de estructuras hiperestáticas por deformaciones angulares y lineales, posteriormente conocido como el **Método de las Rigideces**. Después de su publicación este método dominó el análisis por más de una década, hasta la aparición de los métodos iterativos aproximados de análisis estructural, principalmente los métodos de Cross y de Kani, los cuales fueron mas utilizados, debido a que tenían una metodología más simplificada y sencilla que la del Método de las Rigideces. Este método resurgió cuando las computadoras estuvieron accesibles, debido a que es más fácil programar sus algoritmos en forma matricial, además representaba un modelo mas general y exacto que todos los anteriores métodos iterativos.

El Método de las Rigideces se conoce también como método de los desplazamientos, debido a que un conjunto de fuerzas externas aplicadas a la estructura, sirven para hallar a los desplazamientos internos correspondientes.

En el Método de las Rigideces, se consideran acciones producidas por desplazamientos externos unitarios, los cuales considerados en el plano son: traslaciones o rotaciones unitarias. Estas acciones causadas por los desplazamientos unitarios son conocidas como **rigideces**, y en las acciones son tomadas: fuerzas o momentos externos.

3.2 BASES TEORICAS DEL METODO DE LAS RIGIDECES:

3.2.1 Expresión General del Método de las Rigideces:

El método de las rigideces está basado en los tres principios mencionados a continuación, los cuales son expresados mediante las siguientes ecuaciones:

Para un sistema estructural, se tiene:

$$\begin{aligned} \{ e \} &= [a] \{ d \}, && \text{Ecuación de Continuidad.} \\ \{ P \} &= [k] \{ e \}, && \text{Ecuaciones Constitutivas (Ley de Hooke).} \\ 0 &= \{ F \} + [a]^T \{ P \}, && \text{Ecuaciones de Equilibrio.} \end{aligned}$$

en donde:

$\{ e \}$	Vectores de deformaciones.
$\{ d \}$	Vector de desplazamientos.
$\{ P \}$	Vector de fuerzas internas.
$\{ F \}$	Vector de fuerzas externas.
$[a]$	Matriz de continuidad o matriz operador.
$[k]$	Matriz de rigidez del sistema.

Sustituyendo la ecuación de continuidad en la ecuación constitutiva nos da la ecuación (3-1), y sustituyendo esta ecuación (3-1) en la ecuación de equilibrio, se tiene la ecuación (3-2):

$$\begin{aligned} \{ P \} &= [k] [a] \{ d \} && (3-1) \\ 0 &= \{ F \} + [a]^T [k] [a] \{ d \} && (3-2) \end{aligned}$$

La expresión (3-2) nos define convenientemente el **Método de las Rigideces**, aunque representa a un sistema de ecuaciones diferenciales, podemos solucionar dicho sistema discretizando estas ecuaciones para un caso práctico (debido a que se discretiza al sistema estructural al tomar como incógnitas a los desplazamientos en los nodos), por lo que se reduce el problema a trabajar únicamente con un método discreto en forma matricial.

Tomando a $[K]$ como:

$$[K] = [a]^T [k] [a] \quad (3-3)$$

en donde:

[K]	Matriz de rigideces globales de la estructura. (Matriz acoplada)
[k]	Matriz de rigideces de la estructura.
[a]	Matriz de continuidad o matriz operador.

Sustituyendo la anterior ecuación (3-3) en la ecuación (3-2), nos queda:

$$\{ F \} + [K] \{ d \} = 0 \quad (3-4)$$

$$-\{ F \} = [K] \{ d \} \quad (3-5)$$

El signo negativo en el vector $\{ F \}$ de la ecuación (3-5) indica que se tratan fuerzas internas. Debido a que las que interesan obtener son las fuerzas externas, se cambia el signo a $\{ F \}$:

$$\{ F \} = [K] \{ d \} \quad (3-6)$$

Esta última expresión (3-6) es la que nos define el método de las rigideces, y es la expresión que se va a manejar de aquí en adelante para representar el método. Después de haber encontrado la anterior ecuación, nos podemos dar cuenta que la única incógnita de esta ecuación son los desplazamientos en los nodos, ya que el vector columna de fuerzas externas es ya conocido (condiciones de carga), y la **Matriz de Rigideces Globales de la Estructura [K]** está en función de las características geométricas y mecánicas de cada elemento de dicha estructura y además de la relación fuerza-deformación para la estructura global.

Por tanto, para efectuar el método es necesario contar con la Matriz $[K]$ y con el vector columna de fuerzas externas $\{ F \}$, la definición y obtención de éstos parámetros son descritos en los siguientes temas.

3.2.2 Matriz de Rigideces Globales de la Estructura [K]:

Como los desplazamientos son las incógnitas en los nodos, la noción de indeterminación estática se reemplaza por el concepto de indeterminación cinemática, definido por el número de desplazamientos en los nodos, independientes y desconocidos. Aunque la formulación es diferente, las condiciones que se deben satisfacer (equilibrio estático y compatibilidad de deformaciones) son las mismas. Para este fin, la Matriz de Rigidez de la estructura la podemos descomponer en dos sistemas:

- Sistema básico, este sistema se obtiene restringiendo los extremos de cada uno de los miembros del sistema inicial, previendo sus desplazamientos (inmovilizando el sistema), pero conservando las cargas aplicadas. La fijación de los extremos (nodos) convierte a las estructuras en un sistema de miembros fijos (barras doblemente empotradas), con un alto grado de indeterminación.
- Sistema complementario, se obtiene a partir del sistema básico, retirando las cargas aplicadas e introduciendo deformaciones en los extremos (redundantes) desconocidos para cada uno de los miembros, satisfaciendo las restricciones naturales del sistema (diferentes tipos de condiciones de continuidad y de frontera).

La elección de estos sistemas componentes es arbitraria, siempre que cada uno de ellos esté independientemente en un estado de deformación compatible (consistente con la otra parte del mismo sistema) y sea, además, geoméricamente estable.

Para encontrar el sistema de la matriz global de rigideces [K] de la estructura analizada, se deben de obtener para cada miembro de dicha estructura, una matriz conocida como: **Matriz de rigidez acoplada de barra [MRA]**, la cuál estará dada en un sistema básico, y posteriormente (conforme las condiciones de restricciones en los extremos de cada miembro) se modificará para así obtener la matriz de rigidez acoplada [MRA] en un sistema complementario. Con las matrices [MRA] de cada barra se ensambla la matriz [K], de acuerdo a las condiciones de continuidad en cada miembro.

Por otra parte, al obtener las rigideces de una estructura, se hace evidente que no están en función de las fuerzas que actúan sobre ella, por esta razón la Matriz de Rigidez [K] se puede utilizar para distintas condiciones de carga ó vectores de fuerzas externas de fijación { F }, lo cual es muy usual en la práctica debido a los distintos factores de carga que se dan en los reglamentos de construcción.

3.2.3 Matriz de Rigidez Acoplada de Barra [MRA]:

La matriz de rigidez acoplada de barra tiene como bases a los principios del método de rigideces, en términos generales esta compuesta para cada barra i de la estructura analizada, de la siguiente forma:

$$[a]^T [k] [a] = [MRA] \quad (3-7)$$

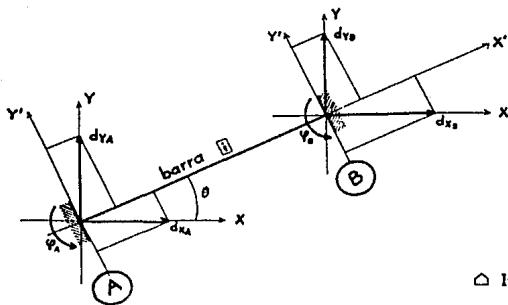
En donde:

$[a]$	Matriz de continuidad de barra.
$[k]$	Matriz de rigidez de barra (sistema local).
$[MRA]$	Matriz de rigidez acoplada de barra (sistema global).

Matriz de continuidad [a]:

La matriz de continuidad o matriz operador es una matriz de transformación que asocia las propiedades de continuidad de la barra referida a su sistema local transformándolo al sistema global de la estructura, se obtiene al ensamblar la siguiente matriz:

$$[a] = \begin{bmatrix} c & s & 0 & 0 & 0 & 0 \\ -s & c & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & c & s & 0 \\ 0 & 0 & 0 & -s & c & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (3-8)$$



En donde:

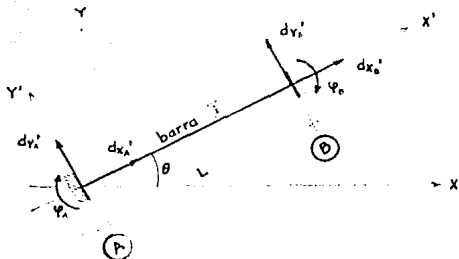
X -Y	Sistema coordenado global de la estructura.
X'-Y'	Sistema coordenado local de la barra i.
barra i	Elemento reticular que forma parte de la estructura analizada.
A	Nodo origen de la barra i.
B	Nodo final de la barra i.
Θ	Angulo que forma la barra i con el eje coordenado X-Y.
c	cos Θ
s	sen Θ
dx_A	Traslación sobre el eje X del nodo A.
dy_A	Traslación sobre el eje Y del nodo A.
φ_A	Giro normal al plano X-Y del nodo A.
dx_B	Traslación sobre el eje X del nodo B.
dy_B	Traslación sobre el eje Y del nodo B.
φ_B	Giro normal al plano X-Y del nodo B.

Matriz de rigidez de barra [k]:

La matriz de rigidez de una barra, referida al sistema local y considerando deformaciones axiales, cortantes y flexionantes, es la siguiente:

$$[k]_i = \begin{pmatrix} Y_N & 0 & 0 & -Y_N & 0 & 0 \\ 0 & \left(\frac{Y_{AA} + Y_{BB} + 2Y_{AB}}{L^2} \right) & \left(\frac{Y_{AA} + Y_{AB}}{L} \right) & 0 & -\left(\frac{Y_{AA} + Y_{BB} + 2Y_{AB}}{L^2} \right) & \left(\frac{Y_{BB} + Y_{AB}}{L} \right) \\ 0 & \left(\frac{Y_{AA} + Y_{AB}}{L} \right) & Y_{AA} & 0 & -\left(\frac{Y_{AA} + Y_{AB}}{L} \right) & Y_{AB} \\ -Y_N & 0 & 0 & Y_N & 0 & 0 \\ 0 & -\left(\frac{Y_{AA} + Y_{BB} + 2Y_{AB}}{L^2} \right) & -\left(\frac{Y_{AA} + Y_{AB}}{L} \right) & 0 & \left(\frac{Y_{AA} + Y_{BB} + 2Y_{AB}}{L^2} \right) & -\left(\frac{Y_{BB} + Y_{AB}}{L} \right) \\ 0 & \left(\frac{Y_{BB} + Y_{AB}}{L} \right) & Y_{AB} & 0 & -\left(\frac{Y_{BB} + Y_{AB}}{L} \right) & Y_{BB} \end{pmatrix}$$

(3-9)



Para la anterior matriz $[k]$ definida en la ecuación (3-9) de la barra i , se tiene:

$X'-Y'$	Sistema coordinado local de la barra i .
$X-Y$	Sistema coordinado global de la estructura.
γ_N	Rigidez a la fuerza axial.
$\gamma_{AA}\gamma_{BB}\gamma_{AB}$	Rigideces angulares.
A	Nodo origen de la barra i .
B	Nodo final de la barra i .
L	Longitud de la barra i .
dx_A'	Traslación sobre el eje X' del nodo A .
dy_A'	Traslación sobre el eje Y' del nodo A .
ϕ_A	Giro normal al plano $X-Y$ del nodo A .
dx_B'	Traslación sobre el eje X' del nodo B .
dy_B'	Traslación sobre el eje Y' del nodo B .
ϕ_B	Giro normal al plano $X-Y$ del nodo B .

Ensamble de la matriz de rigidez acoplada de barra $[MRA]$:

Después de definir las matrices de continuidad y de rigidez de barra, se ensambla la matriz de rigidez acoplada de barra. Utilizando la definición de rigideces, se tiene:

$$[a]^T [k] [a] = [MRA] \quad (3-10)$$

En donde:

$[a]$	Matriz de continuidad de barra.
$[k]$	Matriz de rigidez de barra (sistema local).
$[MRA]$	Matriz de rigidez acoplada de barra (sistema global).

Realizando el producto matricial $[a]^T [k] [a]$, se tiene:

$$[MRA]_i = \begin{bmatrix} Y_6 & Y_7 & Y_1 & -Y_6 & -Y_7 & Y_2 \\ Y_7 & Y_6 & -Y_3 & -Y_7 & -Y_6 & -Y_4 \\ Y_1 & -Y_3 & Y_{AA} & -Y_1 & Y_3 & Y_{AB} \\ -Y_6 & -Y_7 & -Y_1 & Y_6 & Y_7 & -Y_2 \\ -Y_7 & -Y_6 & Y_3 & Y_7 & Y_6 & Y_4 \\ Y_2 & -Y_4 & Y_{BA} & -Y_2 & Y_4 & Y_{BB} \end{bmatrix} \quad (3-11)$$

En donde:

Y_N	Rigidez a la fuerza axial.
$Y_{AA} Y_{BB} Y_{AB}$	Rigideces angulares.
Θ	Angulo que forma la barra i sobre el eje coordenado X-Y.
c	$\cos \Theta$
s	$\sin \Theta$
L	Longitud de la barra i.

Las siguientes ecuaciones (3-12) a (3-22), son ecuaciones simplificadorias auxiliares (sin significado estructural).

$$w = Y_{AB} + Y_{AA} \quad (3-12)$$

$$x = Y_{AB} + Y_{BB} \quad (3-13)$$

$$y = (w + x) / L^2 \quad (3-14)$$

$$z = c^2 (Y_N - y) \quad (3-15)$$

$$Y_1 = w * s / L \quad (3-16)$$

$$Y_2 = x * s / L \quad (3-17)$$

$$Y_3 = w * c / L \quad (3-18)$$

$$Y_4 = x * c / L \quad (3-19)$$

$$Y_5 = y + z \quad (3-20)$$

$$Y_6 = Y_N - z \quad (3-21)$$

$$Y_7 = c * s (Y_N - y) \quad (3-22)$$

Con los procedimientos anteriores se obtuvo la matriz $[MRA]_i$ de cualquier miembro de la estructura analizada, referida al sistema coordenado global. Con dichas matrices $[MRA]_i$ se puede ensamblar la matriz $[K]$, solo faltaría definir como encontrar las variables necesarias para definir la matriz $[MRA]_i$, estas variables son: las rigideces angulares y la rigidez a la fuerza axial, por cada miembro o barra. Las cuales se representan por convención con la siguiente simbología:

Rigideces angulares:

- Y_{AA} Rigidez angular del nodo A al nodo A,
- Y_{AB} Rigidez angular del nodo A al nodo B,
- Y_{BA} Rigidez angular del nodo B al nodo A,
- Y_{BB} Rigidez angular del nodo B al nodo B.

Rigidez a la fuerza axial:

- Y_N Rigidez a la fuerza axial.

Las rigideces angulares y axiales, son las variantes que faltan por definir y están a su vez, en función de las flexibilidades de cada miembro, dichas flexibilidades pueden ser calculadas de la siguiente forma, para cada barra:

$$\Delta = f_{22} f_{33} - f_{23}^2 \quad (3-23)$$

$$Y_{AA} = f_{22}/\Delta \quad (3-24)$$

$$Y_{AB} = -(f_{23} L + f_{22})/\Delta \quad (3-25)$$

$$Y_{BA} = Y_{AB} \quad (3-26)$$

$$Y_{BB} = (f_{23} L^2 + 2 f_{23} + f_{22})/\Delta \quad (3-27)$$

$$Y_N = 1/f_{11} \quad (3-28)$$

En donde:

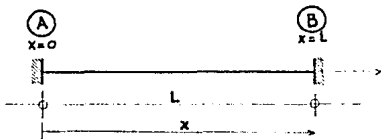
$$f_{11} = \int \frac{dx}{EA} \quad (3-29)$$

$$f_{22} = \int \frac{x^2 dx}{EI} + \int \frac{dx}{GAC} \quad (3-30)$$

$$f_{23} = -\int \frac{x dx}{EI} = f_{32} \quad (3-31)$$

$$f_{33} = \int \frac{dx}{EI} \quad (3-32)$$

Estas flexibilidades se obtienen al integrar a lo largo de la sección de la barra:



Debido al planteamiento anterior, la sección del miembro puede presentar las siguientes propiedades geométricas y mecánicas:

I	$I(x)$ Momento de inercia (variable).
A	$A(x)$ Area de la sección transversal (variable).
A_c	$A_c(x)$ Area de cortante de la sección transversal (variable).
E	Módulo de elasticidad (constante).
G	Módulo de cortante (constante).

Por consiguiente, se pueden analizar en términos generales barras que consideren las siguientes características:

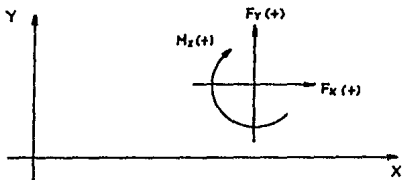
- miembros de sección constante o sección prismática,
- miembros de sección variable, por lo que puede considerar marcos con muros de cortante,
- distintos módulos de elasticidad en la estructura, pero por lo general un solo módulo por miembro, y
- deformaciones por esfuerzos axiales, flexionantes y cortantes.

3.3 VECTOR DE FUERZAS EXTERNAS:

El vector de fuerzas externas de fijación está representado por las fuerzas que se aplican a la estructura para mantener el equilibrio. Estas fuerzas son aplicadas a la estructura como acciones de las condiciones de carga a las que se somete el sistema estructural, en general las fuerzas externas son debidas a las cargas muertas, cargas vivas y cargas accidentales.

Debido a que en el método de las rigideces se discretiza el medio continuo de la estructura a través de los puntos nodales, por consiguiente cualquier tipo de carga, sea puntual o distribuida se debe de tomar en cuenta mediante su aplicación puntual en los nodos.

El vector de fuerzas externas $\{ F \}$ debe de estar referido al sistema global de la estructura, por convención los signos de éstas fuerzas se toman de la siguiente manera:



En donde:

X-Y	Sistema coordinado global de la estructura.
$F_y(+)$	Fuerzas externa vertical positiva del nodo i.
$F_x(+)$	Fuerzas externa horizontal positiva del nodo i.
$M_z(+)$	Momento flexionante positivo del nodo i (siguiendo el recorrido de las manecillas del reloj).

3.4 VECTOR DE DESPLAZAMIENTOS:

El vector de desplazamientos obtenidos $\{d\}$, por lo general no es tomado en cuenta, salvo en el análisis de marcos planos, donde se revisen condiciones de servicio, en esta tesis los desplazamientos horizontales se utilizan para encontrar a las rigideces de entrepiso del marco analizado, con lo cual podemos simplificar posteriormente el análisis sísmico.

Después de obtener la matriz global de rigideces de la estructura $[K]$ y el vector de fuerzas de fijación $\{F\}$, resolvemos el sistema de ecuaciones lineales que representa al Método de las Rigideces para obtener los desplazamientos en los nodos.

$$[K] \{d\} = \{F\} \quad (3-33)$$

Debido a las propiedades de $[K]$, el método comúnmente utilizado para resolver el sistema representado por la ecuación (3-33), es el Método de Cholesky para resolver ecuaciones lineales simultáneas, debido a que la matriz $[K]$ es positiva definida y por tanto, simétrica. Los procedimientos de este método se presentan en el capítulo posterior.

Otra opción que puede ser planteada para obtener el vector de desplazamientos, es la siguiente:

Retomando la teoría sobre operaciones de matrices, nos menciona que si una matriz $[K]$ tiene la propiedad de ser cuadrada (no necesariamente positiva definida), entonces existe otra matriz $[K]^{-1}$, llamada la matriz inversa de $[K]$, la cual tiene las siguientes propiedades:

$$[K] [K]^{-1} = [K]^{-1} [K] = [I] \quad (3-34)$$

También esta teoría sobre operaciones matriciales demuestra que utilizando la matriz inversa se pueden resolver una serie de ecuaciones simultáneas, de la siguiente forma:

$$[K]^{-1} \{F\} = \{d\} \quad (3-35)$$

Debido a las propiedades de la ecuación (3-35), la aplicación de la Matriz Inversa es eficiente cuando es necesario resolver varios sistemas de ecuaciones lineales representados por la ecuación (3-33). En los cuales, lo que únicamente difieren, es en el vector de términos independientes $\{F\}$, manteniendo en todos los demás sistemas de ecuaciones lineales a la misma matriz $[K]$. En el análisis estructural de edificios, es común someter a una estructura a distintas condiciones de carga, representados por distintos vectores $\{F\}$, por lo que es factible aplicar también este último procedimiento.

En lugar de resolver cada sistema de ecuaciones por separado para encontrar las distintas soluciones de estos sistemas lineales, la alternativa diferente consistiría en determinar:

la inversa de la matriz de coeficientes $[K]$, y simplemente multiplicando la matriz $[K]^{-1}$ por los distintos vectores de términos independientes correspondientes a $\{F\}$ se obtendrían directamente las soluciones de los vectores de desplazamientos $\{d\}$.

Debido a que la multiplicación matricial es mucho más rápida que la inversión matricial, el consumo de tiempo se lleva a cabo una sola vez, y después se obtienen las soluciones adicionales de una manera más eficiente.

A esta opción se le llama: Método de las Flexibilidades ó Método de las Fuerzas, en función del Método de las Rigideces, debido a que la inversa de la matriz de rigidez $[K]$ es la matriz de flexibilidades $[F]$.

$$[K]^{-1} = [F]$$

(3-36)

3.5 VECTOR DE FUERZAS INTERNAS:

El objetivo de utilizar un método de análisis estructural para marcos planos es el siguiente: dada una condición de carga { F }, encontrar como se distribuyen estas fuerzas externas a lo largo de cada uno de sus miembros para encontrar la magnitud de fuerzas internas, sin embargo los resultados que obtenemos con el método de rigideces solo son desplazamientos { d }, por lo que es necesario complementar este método, para que encuentre la distribución de fuerzas sobre cada miembro de la estructura, dicho complemento está basado en la misma teoría del método de rigideces, y es precisamente el proceso que se describe en esta parte del capítulo.

A partir del vector de desplazamientos { d } de los nodos de toda la estructura, se obtienen por separado los vectores de desplazamientos para cada nodo (nodo-A,nodo-B) que conforman al miembro en sus extremos. Estos desplazamientos son representados por los siguientes vectores:

$$\{d_A\}_i = \begin{Bmatrix} dx_A \\ dy_A \\ \varphi_A \end{Bmatrix} \quad (3-37)$$

$$\{d_B\}_i = \begin{Bmatrix} dx_B \\ dy_B \\ \varphi_B \end{Bmatrix} \quad (3-38)$$

Estos vectores conforman el vector { d } de la barra i analizada:

$$\{d\}_i = \begin{Bmatrix} dx_A \\ dy_A \\ \varphi_A \\ dx_B \\ dy_B \\ \varphi_B \end{Bmatrix} \quad (3-39)$$

Una vez aislados los vectores de desplazamientos correspondientes a los nodos de cada miembro (3-39), se calculan las fuerzas internas { P } utilizando las ecuaciones constitutivas, de acuerdo a las siguientes relaciones, para cada barra i:

$$\{ P \} = [k] \{ e \} \quad (3-40)$$

$$\{ P \} = [k] \{ a \} \{ d \} \quad (3-41)$$

En donde:

$\{ P \}$	Vector de fuerzas internas de la barra i .
$[k]$	Matriz de rigideces de la barra i (en sistema local).
$\{ a \}$	Matriz de continuidad de la barra i .
$\{ d \}$	Vector de desplazamientos (en sistema global) de los nodos que discretizan a la barra i .

Hay que recordar que éstas matrices de continuidad y rigidez de cada barra i son las mismas que se utilizaron para ensamblar la matriz $[MRA]$ de cada barra.

Si existen cargas intermedias $\{ P_e \}^{A/B}$ a lo largo de la barra i , se tienen que considerar para el cálculo de las fuerzas internas $\{ P \}$, de la siguiente forma:

$$\{ P \} = [k] \{ e \} + \{ P_e \} \quad (3-42)$$

En donde:

$\{ P_e \}$	Vector de fuerzas internas de la barra i , debido a cargas intermedias. Se debe de considerar a este vector de fuerzas como reacciones apoyo-barra (A/B) y además debe de tener como referencia al sistema local de la barra i analizada.
-------------	---

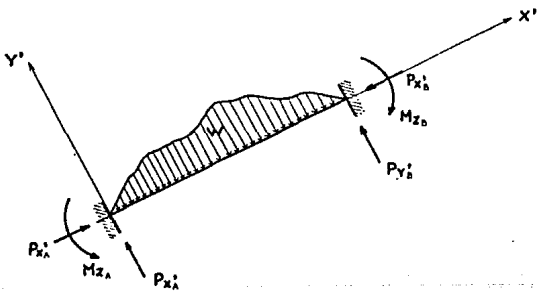
El vector de fuerzas internas $\{ P \}$ se obtiene para cada barra i de la estructura analizada, este vector contiene a los *elementos mecánicos* de la barra i referidos a los nodos de los extremos de la barra, por lo que en general el vector $\{ P \}$ esta compuesto de la siguiente manera:

$$\{ P \}_i = \begin{Bmatrix} P'_{x_A} \\ P'_{y_A} \\ M'_{z_A} \\ P'_{x_B} \\ P'_{y_B} \\ M'_{z_B} \end{Bmatrix}$$

En donde:

$X'-Y'$	Sistema coordenado local de la barra i .
$P_{x_A}^i$	Fuerzas interna horizontal (fuerza normal) en el nodo A.
$P_{y_A}^i$	Fuerzas interna vertical (fuerza cortante) en el nodo A.
$M_{z_A}^i$	Momento flexionante en el nodo A.
$P_{x_B}^i$	Fuerzas interna horizontal (fuerza normal) en el nodo B.
$P_{y_B}^i$	Fuerzas interna vertical (fuerza cortante) en el nodo B.
$M_{z_B}^i$	Momento flexionante en el nodo B.

Los vectores $\{ P \}$ de cada barra i son calculados como reacciones apoyo-barra (A/B) y es el principal objetivo que se persigue en el método de las rigideces.



**CAPITULO IV :
DESARROLLO DE UN
SUBPROGRAMA PARA EL
ANALISIS DE MARCOS PLANOS**

CAPITULO IV

DESARROLLO DE UN SUBPROGRAMA PARA EL ANALISIS DE MARCOS PLANOS

4.1 DESCRIPCION GENERAL :

En el presente capítulo se describen los aspectos principales sobre el desarrollo del subprograma de análisis de marcos planos. En la descripción de cada tema en específico, se explican sus principios y como realizan su función, pero sin exponer en detalle su programación (instrucción por instrucción). Las personas que tengan interés en conocer la mecánica de la programación de los métodos expuestos podrán utilizar el código del programa escrito en Lenguaje C, apoyándose en las descripciones y diagramas de flujo que se indican a lo largo del capítulo. Esta forma de desarrollar la explicación, se debe a que las técnicas de programación empleadas en la construcción del subprograma, están fuera de los objetivos y alcances del trabajo de tesis aquí expuesto, aunque son indispensables como herramientas auxiliares para la solución de problemas estructurales complejos.

El subprograma fue creado como una sola unidad, independiente de las demás. Esta basado en el método de las rigideces. Se proyectó para que fuera el subprograma mas eficiente del programa general ANESAR, debido a que requiere de un número elevado de operaciones aritméticas, por esta razón el subprograma cuenta con algoritmos relativamente complejos.

El planteamiento del subprograma de análisis de marcos planos está construido con un enfoque descendente, de hecho el lenguaje C empleado, se orienta por si mismo hacia este tipo de construcción. El método descendente genera un código claro y legible al que se le puede dar mantenimiento fácilmente, simplifica la clarificación de la estructura general y la operación del programa antes de empezar a codificar las funciones de bajo nivel, minimizando el tiempo que se emplearía en comienzos erróneos. El inicio del método descendente es una descripción general del problema y trabajar hacia lo específico, se debe de definir exactamente en los niveles superiores lo que se necesita realizar.

4.2 DIAGRAMA DE BLOQUES :

La descripción general del subprograma de análisis de marcos planos, estará representada en el nivel superior mediante el siguiente diagrama de bloques secuencial, de ésta forma se puede tener una mejor idea del cómo trabaja esta subrutina. Mediante el diagrama de bloques se divide el problema general (análisis estructural de marcos planos) en varios problemas más sencillos, de tal forma de ir paso a paso hacia lo específico. Posteriormente serán descritos con mayor detalle cada una de las estructuras que conforman al diagrama.

- **INICIO** del subprograma de: análisis de marcos planos.
- Entrada de datos (INPUT).
- Cálculo del número de desplazamientos (d_s) de la estructura.
- Cálculo del ancho de banda (ab) de la matriz de rigidez $[K]$.
- Ensamble de la matriz $[K]$.
- Factorización de la matriz $[K]$:
$$[K] = [L][L]^T$$
- Ensamble de los vectores de fuerzas externas $\{F\}_i$.
- Cálculo de los vectores de desplazamiento $\{d\}_i$, correspondientes a los vectores $\{F\}_i$:
$$\{F\} = [L]\{Z\}$$
$$\{Z\} = [L]^T\{d\}$$
- Cálculo de los vectores de fuerzas internas $\{P\}_i$, correspondientes a los vectores de fuerzas externas $\{F\}_i$.
- Salida de datos (OUTPUT).
- **FIN** del subprograma.

4.3 ENTRADA DE DATOS:

Todos los datos de entrada (INPUT) necesarios para el subprograma de análisis de marcos planos, se obtienen de archivos secuenciales ASCII de sólo lectura. Estos archivos deben de ser generados fuera de la subrutina, como producto de los datos de entrada del programa ANESAR o como el cálculo de los análisis sísmicos aquí realizados.

En el diseño de este subprograma no se emplea la posibilidad de entrada de datos utilizando los parámetros de la subrutina, debido a que se continúa con la optimización de memoria de la computadora y además los datos para el análisis de marcos planos no son requeridos en los análisis posteriores, por estos motivos no es necesario mantenerlos en la memoria virtual de la computadora.

Los datos de entrada están agrupados en los dos siguientes archivos:

- Archivo *a_proy* : Contiene a los datos relacionados con la obtención de la matriz de rigidez global [K] de la estructura.
- Archivo *a_carg* : Contiene a los datos necesarios para el ensamble del vector de fuerzas externas {F}, éste archivo puede contener a distintas condiciones de carga.

Para poder emplear en forma correcta al subprograma de análisis de marcos planos, se requiere seguir con toda exactitud el orden en que el subprograma emplea la secuencia de entrada de datos en cada archivo. Esta secuencia está definida para cada archivo (INPUT) de la forma siguiente:

- Archivo *a_proy* :

nn	Número total de nodos en la estructura.
nnr	Número total de nodos restringidos (según el tipo de apoyo o conexión).
nb	Número total de elementos barra (miembros reticulares) en la estructura.
X{1..nn}	Coordenada del nodo i sobre el eje X (horizontal).
Y{1..nn}	Coordenada del nodo i sobre el eje Y (vertical).
NNR{1..nnr}	Número (dirección) del nodo restringido.
GNR{1..nnr}	Grado de libertad del nodo restringido.
A{1..nb}	Nodo origen A de la barra j.
B{1..nb}	Nodo final B de la barra j.
G{0..4 x nb}	Rigidez a la fuerza axial y rigideces angulares de la barra j (Y_N, Y_{AA}, Y_{AB} y Y_{BB}).

■ Archivo *a_carg* :

nc Número de condiciones de carga (1..*nc*)

Para cada condición de carga (*nc*) se requiere al siguiente grupo de datos:

<i>nnc</i>	Número de nodos cargados (cargas nodales).
DNC[1.. <i>nnc</i>]	Número (dirección) del nodo cargado.
FEN [3][1.. <i>nnc</i>]	Fuerzas externas (Fx, Fy y Mz) referidas al sistema coordenado global, concentradas en el nodo <i>i</i> .

Este subprograma de análisis de marcos planos es aprovechado como subprograma auxiliar para determinar a las rigideces de entrepiso de los distintos marcos planos (marcos tipo) que conforman al edificio real. El subprograma de rigideces de entrepiso genera a los archivos de entrada (*a_proy* y *a_carg*) de acuerdo a las secuencias anteriormente mencionadas para emplear correctamente el subprograma aquí tratado.

4.4 CALCULO DEL NUMERO DE DESPLAZAMIENTOS:

La obtención del número de desplazamientos (ds) se refiere al cálculo del número total de desplazamientos globales que existen en la estructura. Estos desplazamientos incluyen posibles traslaciones (dx,dy) y rotaciones (ϕ) por cada nodo de la estructura general. Por lo tanto los desplazamientos estarán en función de las condiciones de continuidad y de las condiciones de frontera de todos los nodos de la estructura.

Las condiciones de continuidad entre los miembros están en función del detalle constructivo con el que se resuelve la conexión, por lo que la suposición que se tome en el modelo analítico debe de ser congruente con el tipo de conexión que se especifique en el diseño estructural para el miembro en cuestión.

Los tipos de conexiones que se pueden dar en las condiciones de continuidad y de frontera de cada nodo, son las combinaciones entre las traslaciones y rotación del nodo. Estas combinaciones representadas en el plano y referidas al sistema global de la estructura son las siguientes:

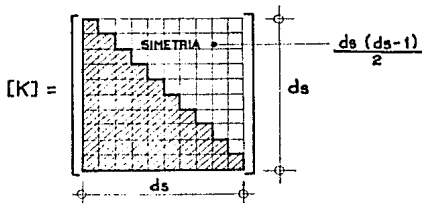
						Tipo de apoyo
						Rotación en Z
						Traslación en X
						Traslación en Y
						No. desplaz.
						Representación
0	0	0	0	0		
1	0	0	1	1		
2	0	1	0	1		
3	0	1	1	2		
4	1	0	0	1		
5	1	0	1	2		
6	1	1	0	2		
7	1	1	1	3		

4.5 CALCULO DEL ANCHO DE BANDA :

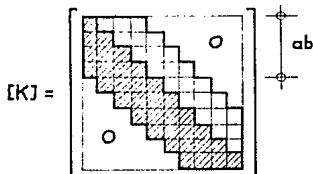
El ancho de banda (ab) es un artificio del método de rigideces, con lo cual se logra mayor eficiencia en el uso de la memoria de la computadora y acelera en gran medida el método de Choleski, porque se procesa a un menor número de elementos de la matriz de rigidez global $[K]$.

El ancho de banda se obtiene mediante la modificación externa de la matriz de rigidez $[K]$ de orden original $d_s \times d_s$. Debido a que ésta matriz tiene las siguientes características, se tiene un ahorro considerable de memoria:

- La matriz $[K]$ es simétrica, por lo que no es necesario almacenar una mitad de esta matriz ($d_s(d_s-1)/2$ elementos simétricos innecesarios).

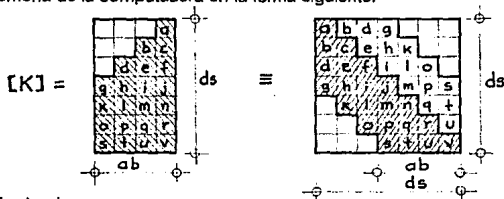


- La matriz $[K]$ puede contener a submatrices triangulares inferior y superior, formadas por elementos con valor cero. En las diagonales centrales (entorno de la diagonal principal) se tienen elementos diferentes de cero.



Las anteriores características pueden ser utilizadas para ahorrar memoria utilizada por elementos innecesarios para el método de rigideces. Se modifica el método de resolución de ecuaciones lineales de Choleski, de tal manera de que solo se procese a los elementos que no sean nulos o simétricos.

La matriz de rigidez $[K]$ utilizando el ancho de banda es representada internamente en la memoria de la computadora en la forma siguiente:



En donde:

$ab = 4$ (ancho de banda).

$ds = 7$ (número de desplazamientos).

Utilizando a la matriz $[K]$ completa para el ejemplo anterior, se tiene el siguiente número de elementos:

$$ds \times ds = 49 \text{ (elementos)} = 100\%$$

Utilizando a la matriz $[K]$ en forma bandeada para el ejemplo anterior, se tiene el siguiente número de elementos:

$$ds \times ab = 28 \text{ (elementos)} = 57\%$$

Se tiene para este ejemplo en particular, un ahorro de memoria del 43%.

El ahorro de memoria y la velocidad del método de Choleski son mas significativos en marcos planos con un gran número de pisos, debido a que generan un ancho de banda (ab) reducido en comparación con el número de desplazamientos (ds).

Es importante señalar que el ancho de banda (ab) está en función de la numeración de los nudos y su relación con las barras. Se deben de numerar de tal forma que la diferencia máxima entre el número de nudos relacionados entre sí por un miembro, sea la mínima posible, si esto se logra se reduce el ancho de banda.

La mayoría de los programas comerciales no realizan una optimización del ancho de banda, al igual que el programa ANESAR, debido a que dicha optimización es compleja y consume bastante tiempo de ejecución, en algunos casos no se justifica.

Es responsabilidad de la persona que crea el modelo analítico de la estructura, numerar sus nodos correctamente, de acuerdo con el criterio descrito anteriormente. El peor caso que se podría dar, es que el ancho de banda sea igual al número de desplazamientos ($ab=ds$), en dicho caso no habría ningún ahorro de memoria y de velocidad.

4.6 ENSAMBLE DE LA MATRIZ DE RIGIDEZ GLOBAL DE LA ESTRUCTURA:

Asignar memoria:

En este proceso se le asigna memoria del tipo dinámico a los elementos de la matriz [K] bandeada, dicha memoria es eliminada cuando se termina el subprograma.

Debido a las características del lenguaje C, la matriz [K] fue linealizada para aprovechar al máximo las características del recurso mas importante de éste lenguaje, los punteros. La utilización de éstos aumenta aún mas la velocidad del método de Choleski.

Al linealizar la matriz [K], también se modifican todos los algoritmos referentes a ella, de tal forma que tomen en cuenta dicha linealización. Los algoritmos que en general se ven más beneficiados por la modificación son los algoritmos del ensamble de la matriz [K] y el método de Choleski.

Ensamble:

En este proceso únicamente se ensambla la matriz triangular inferior de la matriz global de rigidez [K] en forma bandeada, con orden $n \times n$:

En el proceso de ensamble se superponen las matrices de rigidez acoplada de cada barra i ([MRA] _{i}) con la matriz de rigidez global de la estructura [K], esta superposición puede describirse de la siguiente forma:

- Se obtiene la matriz de rigidez acoplada [MRA] de la barra i . En este proceso teóricamente se deberían calcular 36 elementos (rigideces de barra), debido a que el programa únicamente utiliza a la matriz triangular inferior solo se calculan 21 elementos, en donde 3 de ellos son datos de entrada (Y_{AA} , Y_{AB} Y Y_{BB}), de los restantes 18 elementos únicamente se precisa definir a 7 elementos. Como resultado se tiene que para definir a la matriz [MRA] de cada barra i solo se requiere calcular 7 rigideces en lugar de 36.
- De acuerdo a los tipos de conexiones (apoyos) en los extremos de cada barra i , se determina si existen o no desplazamientos en sus extremos, en el nodo A o en el nodo B (dx_A , dy_A , ϕ_A , dx_B , dy_B Y ϕ_B).
- Si existen desplazamientos en el nodo B o en el nodo A, entonces se llevan al cabo los siguientes pasos para cada desplazamiento en los extremos de la barra i :

- Se calcula el número de desplazamientos que le corresponde en la numeración global.
- Se superpone la columna y el renglón de la matriz [MRA] sobre la columna y renglón que le corresponde en la matriz [K].
- Si no existen posibilidades de desplazamientos en los extremos de la barra i , entonces la matriz de la barra calculada [MRA] no alteran a la matriz [K] de la estructura.

En el desarrollo en Lenguaje C de este proceso se utilizan punteros, lo que complica la lógica y comprensión del programa fuente, pero a su vez acelera en gran medida a los algoritmos aquí tratados, aunque se tiene la desventaja de un aumento en el código fuente. La optimización realizada en el ensamble de la matriz [K] se debe a que este proceso es la parte más crítica de la subrutina del análisis de marcos planos.

Salida de datos:

Después de terminar de ensamblar la matriz de rigidez [K], el proceso que a continuación seguiría sería resolver el sistema $\{K\}\{d\} = \{F\}$ empleando el método de Choleski. Debido a la optimización de memoria, el método de Choleski factoriza progresivamente a la matriz [K] sobre ella misma, al final de dicha factorización la matriz [L] ocupa los mismos elementos que la matriz [K], por lo que los valores de las rigideces de la matriz [K] se pierden. Por esta razón, la matriz de rigidez [K] es guardada (si así lo desea el usuario) en un archivo secuencial tipo ASCII, para una posterior consulta.

Para respaldar el análisis estructural realizado en algunas memorias de cálculo, la matriz de rigidez [K] es necesaria, también sirve para comprobar los resultados generados por otros programas de análisis de marcos planos o por cálculos efectuados manualmente.

4.7 ENSAMBLE DEL VECTOR DE FUERZAS EXTERNAS:

El vector de fuerzas externas está compuesto totalmente por datos de entrada (INPUT), los cuales deben de ser obtenidos o calculados externamente a éste subprograma.

Lo que se elabora en el proceso de ensamble es la superposición de fuerzas puntuales en los nodos. No son consideradas las fuerzas externas inducidas por cargas intermedias en las barras, debido a que las fuerzas externas que resultan del análisis sísmico únicamente son fuerzas nodales.

Las fuerzas externas son introducidas como acciones. El sistema coordinado de referencia de cada fuerza externa esta basado en los siguientes aspectos:

- Sistema coordinado global, si las fuerzas externas son fuerzas concentradas en los nodos.

En función de éstos aspectos, los datos de entrada para definir el vector de fuerzas externas, están dados por:

- Número de nodos cargados.
- Para cada nodo i cargado se deben de incluir sus respectivas fuerzas nodales (puntuales): F_{x_i} , F_{y_i} y M_{z_i} , referidas en sistema global.

4.8 CALCULO DEL VECTOR DE DESPLAZAMIENTOS:

Los desplazamientos $\{d\}$ son obtenidos al resolver el sistema de ecuaciones simultáneas representado por:

$$[K]\{d\} = \{F\}$$

El método utilizado para obtener la solución del anterior sistema es el método de Choleski. Este método aprovecha la factorización de la matriz $[K]$ en dos matrices: $[L][L]^T$, mediante las cuales se facilita la solución de ecuaciones simultáneas empleando las ventajas de las matrices triangulares. En forma general se tiene:

$$[K]\{d\} = \{F\} \quad (4-1)$$

Factorizando $[K]$:

$$[K] = [L][L]^T \quad (4-2)$$

Sustituyendo la ecuación (4-2) en la ecuación (4-1):

$$[L][L]^T\{d\} = \{F\} \quad (4-3)$$

Utilizando:

$$\{Z\} = [L]^T\{d\} \quad (4-4)$$

Sustituyendo la ecuación (4-4) en la ecuación (4-3):

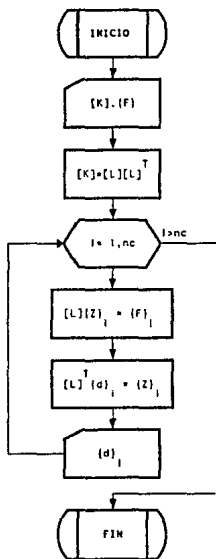
$$[L]\{Z\} = \{F\} \quad (4-5)$$

La obtención de la solución complementaria $\{Z\}$ es resolver la ecuación (4-5), la cual representa a una matriz triangular inferior, por lo que sólo se necesita realizar una sustitución progresiva. Después de encontrar al vector $\{Z\}$ se resuelve la ecuación (4-4) y se obtiene el vector $\{d\}$ que representa a la solución general del sistema $[K]\{d\} = \{F\}$, la obtención del vector $\{d\}$ se realiza al substituir en forma regresiva, debido a que la matriz $[L]^T$ es triangular superior.

El planteamiento para la programación de este método lo podemos dividir en tres etapas principales:

- Factorización de la matriz $[K] = [L][L]^T$.
- Cálculo de la solución complementaria $\{Z\}$, $[L]\{Z\} = \{F\}$.
- Cálculo de la solución general $\{d\}$, $[L]^T\{d\} = \{Z\}$.

En las etapas anteriores se hace evidente que si se requiere analizar un marco plano sujeto a varias condiciones de carga (nc) se realizará una sola factorización de la matriz $[K]$ y posteriormente se calcularán los correspondientes vectores $\{d\}$ para cada condición de carga. Como se ilustra en el siguiente diagrama:



En la factorización de la matriz $[K] = [L][L]^T$, para efectos de programación únicamente se requiere conocer la matriz $[L]$ en forma bandeada, los elementos de ésta matriz son obtenidos de la siguiente forma:

$$l_{ii} = k_{ii} - \sum_{m=1}^{i-1} l_{im}^2 \quad (4-6)$$

$$l_{ij} = \frac{k_{ij} - \sum_{m=1}^{j-1} l_{jm} l_{im}}{l_{jj}} \quad (4-7)$$

En donde:

$$[K] = \begin{bmatrix} k_{11} & k_{12} & k_{13} & \dots & k_{1n} \\ k_{21} & k_{22} & k_{23} & \dots & k_{2n} \\ k_{31} & k_{32} & k_{33} & \dots & k_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & k_{n3} & \dots & k_{nn} \end{bmatrix} \quad (4-8)$$

$$[L] = \begin{bmatrix} l_{11} & 0 & 0 & \dots & 0 \\ l_{21} & l_{22} & 0 & \dots & 0 \\ l_{31} & l_{32} & l_{33} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ l_{n1} & l_{n2} & l_{n3} & \dots & l_{nn} \end{bmatrix} \quad (4-9)$$

El programa fuente en lenguaje C guarda gradualmente a los elementos de la matriz $[L]$ en la matriz $[K]$. Esta característica es una de las ventajas del método de Choleski programado.

Se toma en cuenta el ancho de banda (ab) calculado anteriormente, por lo que el programa genera a la matriz $[L]$ en forma bandeda, de acuerdo al mismo orden ($dis \times ab$) de la matriz $[K]$.

Después de calcular la matriz $[L]$, el programa utiliza únicamente a ésta matriz para obtener a la solución complementaria $\{Z\}$ y a la solución general $\{d\}$, por lo que el algoritmo de Choleski toma en cuenta a la matriz $[L]$ como la matriz $[L]^T$ al encontrar la solución general $\{d\}$.

El método de Choleski empleado sólo se aplica a matrices positivas definidas. La matriz de rigidez global $[K]$ es positiva definida, siempre y cuando la estructura sea estable. Si la estructura es mecánicamente inestable, el método de Choleski conduce a un error del tipo: raíz negativa. Este tipo de error podría ser generado por una incorrecta entrada de datos (INPUT).

4.9 CALCULO DEL VECTOR DE FUERZAS INTERNAS:

El vector de fuerzas internas {P} es calculado para cada barra i, en función del siguiente producto matricial basado en el método de las rigideces.

$$\{ P \} = [k] [a] \{ d \} \quad (4-10)$$

En donde:

{ P }	Vector de fuerzas internas de la barra i.
[k]	Matriz de rigideces de la barra i (en sistema local).
[a]	Matriz de continuidad de la barra i.
{ d }	Vector de desplazamientos (en sistema global) de los nodos que discretizan a la barra i.

El subprograma toma en cuenta los siguientes aspectos para el cálculo del vector de fuerzas internas {P} de cada barra i. De las matrices y vectores que definen a la ecuación (4-10), los únicos que se conoce es el vector de desplazamientos {d}, la matriz de rigidez de barra [k] y la matriz de continuidad [a] de cada barra i fueron utilizadas implícitamente cuando se obtuvo la matriz de rigidez acoplada de barra [MRA], para ensamblar la matriz [K], por lo que éstas matrices no se mantuvieron en memoria o en algún archivo de respaldo.

Para obtener el vector de fuerzas internas {P} de acuerdo al planteamiento matricial de la ecuación (4-10), se tendría que calcular nuevamente, la matriz de rigidez de barra [k] junto con su respectiva matriz de continuidad [a], posteriormente se tendría que realizar el producto matricial [k][a]{d}. Lo que en realidad realiza el subprograma es encontrar en una forma matricialmente desarrollada (elemento a elemento) cada uno de los componentes del vector {P}, por lo que numéricamente no se realiza ningún producto matricial, sólo se sustituyen los valores que definen a las matrices [k] y [a] en las siguientes ecuaciones que definen a los elementos de las fuerzas internas {P}:

Las matrices [k] y [a] de cualquier barra, están en función de las siguientes variables:

$$[k] = f(C, S, L, \gamma_N, \gamma_{AB}, \gamma_{AA}, \gamma_{BB}) \quad (4-11)$$

$$[a] = f(C, S) \quad (4-12)$$

Variables auxiliares:

$$w = dx_B - dx_A \quad (4-13)$$

$$x = dy_B - dy_A \quad (4-14)$$

$$y = (x^*C - w^*S) / L + \phi_B \quad (4-15)$$

$$z = (x^*C - w^*S) / L + \phi_A \quad (4-16)$$

Componentes de las fuerzas internas {P}:

$$N = (X^*S + W^*C) * \gamma_N \quad (4-17)$$

$$Mz_A = Z^*\gamma_{AA} + Y^*\gamma_{AB} \quad (4-18)$$

$$Mz_B = Z^*\gamma_{AB} + Y^*\gamma_{BB} \quad (4-19)$$

En donde:

w,x,y,z	Variables simplificadoras, sin ninguna interpretación estructural. Ecuaciones de (4-13) a (4-16).
N	Fuerzas normal (A/B) de la barra i (ecuación 4-17).
Mz _A	Momento flexionante (A/B) hiperestático en el extremo A de la barra i (ecuación 4-18).
Mz _B	Momento flexionante (A/B) hiperestático en el extremo B de la barra i (ecuación 4-19).
γ_N	Rigidez a la fuerza axial.
$\gamma_{AA}, \gamma_{AB}, \gamma_{BB}$	Rigideces angulares.
dx_A, dy_A, ϕ_A	Desplazamientos del nodo A de la barra i.
dx_B, dy_B, ϕ_B	Desplazamientos del nodo B de la barra i.
Θ	Angulo que forma la barra i sobre el sistema coordenado global de la estructura X-Y.
c	cos Θ
s	sen Θ
L	Longitud de la barra i.

El cálculo aquí empleado para obtener las fuerzas internas {P} es más eficiente que el realizar el producto matricial. Debido a las características de una barra en forma aislada, sólo es necesario calcular la fuerza normal (N) con una ecuación, en lugar de calcular Px_A y Px_B porque tienen la misma magnitud, pero signo contrario. El cálculo de las fuerzas cortantes se realiza por equilibrio estático.

4.10 SALIDA DE DATOS:

El subprograma de análisis de marcos planos tiene la opción de controlar la salida de datos (OUTPUT), este control se efectúa mediante una variable de entrada (INPUT) llamada: **opcion**, ésta variable es una variable de decisión con la cual se maneja distintas posibilidades de salidas de datos:

- (opcion&1) Almacenar matriz de rigidez global [K].
- (opcion&2) Almacenar vector de desplazamientos globales {d}.
- (opcion&4) Almacenar vectores de fuerzas internas {P}.

Los datos de salida son guardados en un archivo secuencial ASCII (a_gen). La matriz [K] se guarda en forma bandeada (orden ds x ab).

A continuación se lista en forma secuencial las variables de salida que contiene el archivo ASCII de escritura (a_gen):

opcion	Variable de decisión para almacenar distintos resultados.
nn	Número de nodos que constituyen al marco plano.
nb	Número de barras que constituyen al marco plano.
(opcion&1)	
ds	Número de desplazamientos globales del marco plano.
ab	Ancho de banda de la matriz [K].
[K]	Matriz de rigidez global [K] de la estructura, de orden ds x ab.
(opcion&2)	
{d}	Vector de desplazamientos globales de la estructura, 3nn elementos.
(opcion&3)	
{P}	Vector de fuerzas internas de cada barra, 3nb elementos.

4.11 PROGRAMA FUENTE EN LENGUAJE C:

El subprograma de análisis estructural de marcos planos presentado en esta sección tiene como bases teóricas y metodológicas a los capítulos III y IV respectivamente.

```
/* SUBPROGRAMA DEL METODO GENERAL DE LAS RIGIDECES */

void rig(char *fproy,int opcion)

/*
    fproy.est (a_proy),      archivo secuencial de solo lectura ASCII .
    Define las características geométricas y mecánicas de la estructura :

        nn      : Número total de nodos en la estructura .
        nnr     : Número total de nodos restringidos en la estructura .
        nb      : Número total de elementos barras .
        X[1..nn] : Coordenadas en el eje X del nodo i .
        Y[1..nn] : Coordenadas en el eje Y del nodo i .
        NNR[1..nnr] : Número de nodo restringido .
        GNR[1..nnr] : Grado de libertad de nodos restringidos .
        A[1..nb]  : Nodo inicial A al cual se conecta la barra j .
        B[1..nb]  : Nodo final B al cual se conecta la barra j .
        G[D..nb*4] : Rigideces angulares y rigidez axial del elemento barra j .

    fproy.car (a_carg),      archivo secuencial de solo lectura ASCII .
    Contiene todas las condiciones de carga (Fuerzas Externas) :

        nc      : Número de condiciones de carga .

    Los siguientes datos se requieren para cada condición de carga (nc):

        nnc     : Número de nodos cargados .
        DNC[1..nnc] : Direcciones de los nodos cargados .
        FEN[3][1..nnc] : Fuerzas externas concentradas en los nodos .

    fproy.gen (a_gen),      archivo secuencial de solo escritura ASCII .
    Almacena los resultados generados por las distintas fuerzas externas :

        opcion  : Variable de decisión, para almacenar distintos resultados .
        nn      : Número de nodos .
        nb      : Número de barras .
        ds      : Número de desplazamientos externos por nodo .
        ab      : Ancho de banda, para reducir la matriz [K] .
        K[ab*ds+1] : Matriz de Rigidez Global de la Estructura .
        nc      : Número de condiciones de carga .
        D[1..3*nn] : Desplazamientos externos por nodo .
        P[1..6*nb] : Elementos mecánicos por cada miembro .

*/

C

FILE *a_proy,*a_carg,*a_gen;
int nn,nnr,nb,*NNR,*GNR,*A,*B,nc;
float *X,*Y,*G,*p;
```

```

double *K,*ap,*ccp,*ok,*qt[11],dd,ee,l,c,s,w,x,y,z;

int *L,{i,i,j,k,a,b,ab+1,ds=1,
    ll,kk,cc,e,d,*nd,*nk,*N},N[7],
M[22]={0,8,10,1,8,10,2,9,3,10,9,4,7,1,3,6,8,10,2,9,4,5},
S[22]={0,1,1,1,-1,-1,1,1,-1,-1,-1,-1,-1,1,1,1,-1,1,1,1,1};

char nom[200];

/* Lectura de información general de la estructura */
strcpy(nom,fproy);strcat(nom,".est");
a_proy=fopen(nom,"r");
fscanf(a_proy,"%d%d%d",&n,&nrr,&nb);

/* Lectura de datos de nodos en la estructura */
K = calloc(nm+1,4);
Y = calloc(nm+1,4);
for(i=1;i<=nm;i++) fscanf(a_proy,"%f%f",&K[i],&Y[i]);

/* Lectura de datos de nodos restringidos */
NRR=calloc(nnr+1,2);
GNR=calloc(nnr+1,2);
for(i=1;i<=nnr;i++) fscanf(a_proy,"%d%d",&NRR[i],&GNR[i]);

/* Cálculo del número global de desplazamientos (ds) en la estructura */
L = calloc(nm+1,2);
nd = calloc(nm+2,2);
for(i=1;i<=nm;i++)L[i]=7;
for(i=1;i<=nnr;i++)L[NRR[i]]=GNR[i];
free(NRR);free(GNR);
for(i=1;i<=nm;i++)
    ( nd[i]=ds+*(L[i]&4&&1)+*(L[i]&2&&1)+*(L[i]&1) );

/* Lectura de información de elementos barra */
A=calloc(nb+1,2);
B=calloc(nb+1,2);
C=calloc(nb+1,4);
for(i=1;i<=nb;i++)
    fscanf(a_proy,"%d%d%f%f%f",&A[i],&B[i],&G[4*i-4],&G[4*i-3],&G[4*i-2],&G[4*i-1]);
fclose(a_proy);

/* Cálculo del ancho de banda (ab) */
nd[1]=ds;
for(i=1;i<=nb;i++) (
    a=A[i];b=B[i];
    if(a>b) (d=s+a;b=d);
    a=d+(b+1)-nd[a];if(a>ab)ab=a;
)

K=calloc(--ds*ab+1,8);
if(!K) { printf("\n memoria insuficiente !!!");exit(); }

/* Ensamble de la Matriz de Rigidez Global de la Estructura */
ab--;
for(i=1;i<=nb;i++) (

/* Calculo de la matriz [MRA] */
p=&G[4*i-2];q=t;
a=A[i];b=B[i];c=X[b]-X[a];s=Y[b]-Y[a];l=sqrt(c*c+s*s);w=l/c;w/l;
x=w*p-l;
y=(w+x)/l;
**+q*w*s/l; **+q*x*s/l;
**+q*w*c/l; **+q*x*c/l;
**+q*s**+p; **+q*s**-p; **+q*x**-p;
z=c*c*(**+p-y);
**+qz-z+y;
**+qz-z**+p;
**+qz-c*s*(**+p-y);
end=0;
for(j=0;j<3;j++){

```

```

        N[j+1]=(L[a]&1<=j)?nd(a)+e++:0;
        N[j+4]=(L[b]&1<=j)?nd(b)+d++:0;
    }
    cc=0;Nj=N;
    for(j=1;j<7;j++) {
        if(("*+Nj")) {
            Nk=Nj-1;
            for(k=1;k<7;k++){if(("*+Nk")) {
                if((!k*"Nj"))>>(!l*"Nk")) { kk=1;ll="Nj";
                    K[ll*"ab*kk"]+=S[d+cc+k-j+1]*t[K(d)];
                }
            }
        }
        cc+=7-j;
    }
}
ab++;

```

```

/* Prepara archivo de salida de datos (OUTPUT) */
strcpy(nom,fproy);strcat(nom,".gen");
a_gen=fopen(nom,"w");
fprintf(a_gen,"%d %d %d %d",opcion,nn,nb);

```

```

/* Imprime matriz de rigidez global [K] de la estructura */
if(opcion&1) {
    fprintf(a_gen,"\n\n\n%d %d\n",ds,ab);
    for(i=0;i<ab*ds+1;i++) fprintf(a_gen,"\nX[%d]",K[i]);
    fflush(a_gen);
}

```

```

/* Factoriza a la matriz de rigidez global [K] en la matriz [L] */
for(i=1;i<=ds;i++) {
    kki=ab;cc=i*ab-ab;ll=(kki<0)?7-kk+1:1;
    for(j=1;j<=ab;j++) {
        y=0;d=(j+kk)*ab;ccp=K+ll*cc;ok=K+d-1;
        for(sp=K+ll*d-j;sp<ok;y=*ccp+**sp++)
            x=K[cc+j]-y;
        K[cc+j]=(j==ab)?sqrt(x):x/K(d);
    }
}

```

```

/* EVALUACION DE LAS DISTINTAS CONDICIONES DE CARGA */

```

```

{
    int nc,nnc,DNC;
    double FEN[3],*F,*D,O[3];

    strcpy(nom,fproy);strcat(nom,".car");
    a_carg=fopen(nom,"r");
    fscanf(a_carg,"%d",&nc);
    fprintf(a_gen,"\n\n\n%d ",nc);
    D = calloc(ds+1,B);

    for(i=1;ll<=nc;ll++) {
        F = calloc(ds+1,B);
        fscanf(a_carg,"%d",&nnc);

        /* Vector de fuerzas externas concentradas en los nodos */
        if(nnc>0)
            for(i=1;i<=nnc;i++) {
                fscanf(a_carg,"%dX[%d]X[%d]X[%d]&DNC,&FEN[0],&FEN[1],&FEN[2]);
                d=0;for(j=0;j<3;j++) if(L[DNC]&1<=j) F[nd[DNC]+d]=FEN[j];
            }
    }
}

```

```

/* METODO DE CHOLESKY PARA LA SOLUCION DE ECUACIONES SIMULTANEAS */
/* Solución complementaria (Z) mediante sustitución progresiva */
for(i=1; i<=ds; i++) {
    kk=i-ab; cc=i*ab-ab; l=(kk<0)?-kk+1:1;
    ccp=k+l*cc; ok=D+kk*ab-1; y=0;
    for(ap=D+kk+1; ap<=ok; y=*ccp+*ap+);
    D[i]=(F[i]-y)/K[i*ab];
}
/* Solución general (D) mediante sustitución regresiva */
for(i=ds; i>0; i--) {
    kk=i-ab; cc=i*ab-ab; l=(kk<0)?-kk+1:1;
    x=D[i]/K[i*ab]; ccp=k+l*cc; ok=D+kk*ab-1;
    for(ap=D+1+kk; ap<=ok; *ap+*=*ccp+*x);
}

/* Almacena el vector de desplazamientos (D) en archivo secuencial a_gen */
if(opcionA2) {
    fprintf(a_gen, "\n\n\n");
    a=0;
    for(i=1; i<=m; i++)
        for(j=D; j<3; j++) {
            w=(L[i]&i<<j)?D[+a]:0;
            fprintf(a_gen, "\nXlf", w);
        }
}

fprintf(a_gen, "\n\n\n");
for(i=1; i<=ds; i++) fprintf(a_gen, "\nXlf", F[i]);

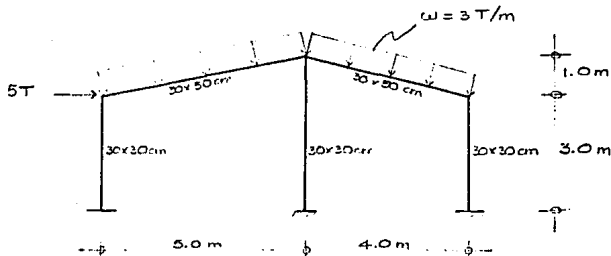
/* Calcula y almacena los elementos mecánicos de cada barra */
if(opcionA4) {
    fprintf(a_gen, "\n\n\n");
    for(i=1; i<=nb; i++) {
        a=A[i]; b=B[i]; e=d=0;
        c=X[b]-X[a]; s=Y[b]-Y[a]; l=sqrt(c*c+a*a); a=l; c/=l;
        for(j=D; j<3; j++) {
            N[j+1]=(L[a]&i<<j)?nd[a]+e+*0;
            N[j+4]=(L[b]&i<<j)?nd[b]+d+*0;
        }
        D[0]=0; p=a*g*(4*i-4);
        w=D[N[6]]-D[N[1]];
        x=D[N[5]]-D[N[2]];
        y=(x*c-w*a)/f+D[N[6]];
        z=y+D[N[3]]-D[N[6]];
        Q[0]=(c*x+a*y*c)**e;
        Q[1]=x*p[1]+y*p[2];
        Q[2]=z*p[2]+y*p[3];
        fprintf(a_gen, "\nXlf Xlf Xlf", Q[0], Q[1], Q[2]);
    }
}
free(F);
fclose(a_carg); fclose(a_gen);
}
}

```

4.12 EJEMPLO :

Para el siguiente marco plano, emplear el método general de las rigideces para obtener:

- Diagramas de fuerzas cortantes, normales y momentos flexionantes.
- Reacciones en los apoyos (condiciones de frontera).



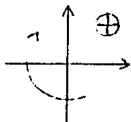
$$E = 1'300,000 \text{ T/m}^2$$

# nodos (nn)	: 6
# barras (nb)	: 5
# nodos restringidos	: 3
# nodos móviles	: 1

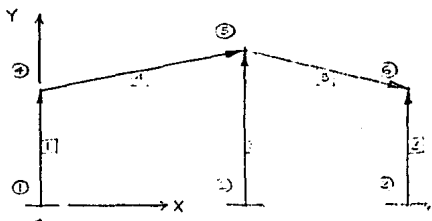
La topología empleada para modelar el marco plano es la siguiente:

- ⊕ Los números encerrados por círculos indican nodos.
- ⊕ Los números encerrados por rectángulos indican barras.
- ⊕ La dirección de la barra (nodo inicio-nodo final) está referida de izquierda a derecha y de abajo hacia arriba.

La convención de signos empleada es la siguiente:



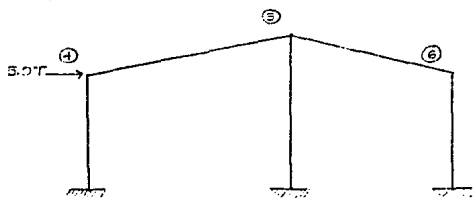
MODELO ANALITICO:



origen del sistema coordenado global X-Y.

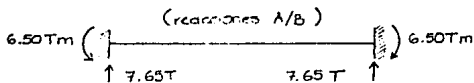
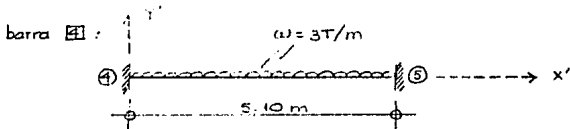
● CALCULO DE FUERZAS EXTERNAS:

FUERZAS EXTERNAS CONCENTRADAS EN LOS NODOS (fuerzas nodales):

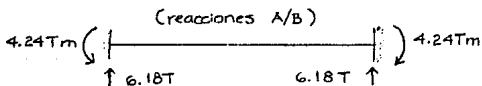
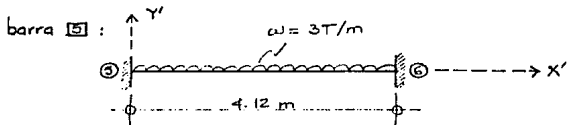


$$\{F\} = \begin{Bmatrix} F_{x4} \\ F_{y4} \\ M_{z4} \\ F_{x5} \\ F_{y5} \\ M_{z5} \\ F_{x6} \\ F_{y6} \\ M_{z6} \end{Bmatrix} = \begin{Bmatrix} 5.0T \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix}$$

FUERZAS EXTERNAS INDUCIDAS POR CARGAS INTERMEDIAS EN LAS BARRAS :



$$\{P_e\}_{\text{B}}^{A/B} = \begin{Bmatrix} P_{x'4} \\ P_{y'4} \\ M_{z4} \\ P_{x'5} \\ P_{y'5} \\ M_{z5} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 7.65 \text{ T} \\ -6.50 \text{ Tm} \\ 0 \\ 7.65 \text{ T} \\ 6.50 \text{ Tm} \end{Bmatrix}$$



$$\{P_e\}_{\text{B}}^{A/B} = \begin{Bmatrix} P_{x'5} \\ P_{y'5} \\ M_{z5} \\ P_{x'6} \\ P_{y'6} \\ M_{z6} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 6.18 \text{ T} \\ -4.24 \text{ Tm} \\ 0 \\ 6.18 \text{ T} \\ 4.24 \text{ Tm} \end{Bmatrix}$$

SUPERPOSICION DE FUERZAS EXTERNAS NODALES CON LAS FUERZAS EXTERNAS INDUCIDAS POR CARGAS INTERMEDIAS EN LAS BARRAS :

Esta superposición es llevada al cabo mediante la siguiente relación:

$$\{F\} - \{F\}^{A/B} = [K]\{d\}$$

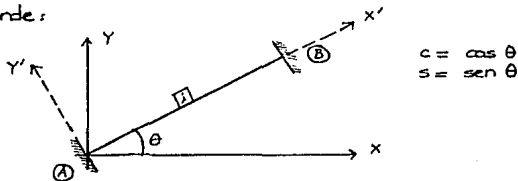
En donde:

$$\{F\}^{A/B} = [a]^t \{P_e\}^{A/B}$$

Desarrollando el anterior producto matricial, se tiene para cada elemento barra, su respectivo vector de fuerzas externas $\{F\}^{A/B}$ de la siguiente forma:

$$\{F\}_{\square}^{A/B} = \begin{Bmatrix} F_{xA} \\ F_{yA} \\ M_{zA} \\ F_{xB} \\ F_{yB} \\ M_{zB} \end{Bmatrix} = \begin{Bmatrix} c P_{x'A} - s P_{y'A} \\ s P_{x'A} + c P_{y'A} \\ M_{zA} \\ c P_{x'B} - s P_{y'B} \\ s P_{x'B} + c P_{y'B} \\ M_{zB} \end{Bmatrix}$$

En donde:



Para el ejemplo aquí tratado se tiene:

barra \square : $\theta = 11.31^\circ$ $c = \cos \theta = 0.9806$
 $s = \text{sen } \theta = 0.1961$

$$\{F\}_{\square}^{A/B} = \begin{Bmatrix} -1.5002 \text{ T} \\ 7.5016 \text{ T} \\ -6.50 \text{ Tm} \\ -1.5002 \text{ T} \\ 7.5016 \text{ T} \\ 6.50 \text{ Tm} \end{Bmatrix} = \begin{Bmatrix} F_{x4} \\ F_{y4} \\ M_{z4} \\ F_{x5} \\ F_{y5} \\ M_{z5} \end{Bmatrix}$$

Barra [5]: $\theta = 345.7638^\circ$

$c = \cos \theta = 0.4701$
 $s = \sin \theta = -0.2155$

$$\{F\}_{5i}^{A/B} = \begin{Bmatrix} -1.5002 \text{ T} \\ 7.5016 \text{ T} \\ -6.50 \text{ Tm} \\ -1.5002 \text{ T} \\ 7.5016 \text{ T} \\ 6.50 \text{ Tm} \end{Bmatrix} = \begin{Bmatrix} F_{x5} \\ F_{y5} \\ M_{z5} \\ F_{x6} \\ F_{y6} \\ M_{z6} \end{Bmatrix}$$

Los vectores de fuerzas externas $\{F\}^{A/B}$ para las barras [1], [2] y [3] son cero, debido a que no presentan cargas intermedias.

La superposición de todos los vectores de fuerzas externas $\{F\}^{A/B}$ inducidas por las cargas intermedias en las barras de toda la estructura queda representado por:

$$\{F\}^{A/B} = \begin{Bmatrix} F_{x1} \\ F_{y1} \\ M_{z1} \\ F_{x5} \\ F_{y5} \\ M_{z5} \\ F_{x6} \\ F_{y6} \\ M_{z6} \end{Bmatrix} = \begin{Bmatrix} -1.5002 \text{ T} \\ 7.5016 \text{ T} \\ -6.50 \text{ Tm} \\ -0.0015 \text{ T} \\ 13.4968 \text{ T} \\ 2.26 \text{ Tm} \\ 1.4987 \text{ T} \\ 5.9752 \text{ T} \\ 4.24 \text{ Tm} \end{Bmatrix}$$

La superposición de cargas nodales con cargas inducidas por cargas intermedias en las barras se obtiene de la siguiente forma:

$$\{F\} - \{F\}^{A/B} = \begin{Bmatrix} 5 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} - \begin{Bmatrix} -1.5002 \\ 7.5016 \\ -6.50 \\ -0.0015 \\ 13.4968 \\ 2.26 \\ 1.4987 \\ 5.9752 \\ 4.24 \end{Bmatrix} = \begin{Bmatrix} 6.5002 \text{ T} \\ -7.5016 \text{ T} \\ 6.50 \text{ Tm} \\ 0.0015 \text{ T} \\ -13.4968 \text{ T} \\ -2.26 \text{ Tm} \\ -1.4987 \text{ T} \\ -5.9752 \text{ T} \\ -4.24 \text{ Tm} \end{Bmatrix}$$

Con el resultado de esta última superposición de fuerzas externas se calculan los desplazamientos globales de la estructura:

$$\{F\} - \{F\}^{A/B} = [K] \{d\}$$

CALCULO DE LA MATRIZ DE RIGIDEZ GLOBAL DE LA ESTRUCTURA :

Para escribir la matriz de rigidez global $[K]$ se requiere conocer las matrices de rigidez $[MRA]_m$ de cada barra que conforma a la estructura, estas matrices se obtienen de la siguiente forma :

Para una barra de sección constante se asignan a los extremos por cordante, las siguientes rigideces se calculan de las siguientes relaciones :

$$\begin{aligned} \gamma_{AA} &= 4EI/L \\ \gamma_{AB} &= \gamma_{BA} = 2EI/L \\ \gamma_{BB} &= 4EI/L \\ \gamma_N &= \frac{AE}{L} \end{aligned}$$

Para el ejemplo tratado se tiene:

barra III : (columna).

$$\begin{aligned} A &= bh = 30 \times 30 = 900 \text{ cm}^2 = 0.09 \text{ m}^2 \\ I &= bh^3/12 = 30^4/12 = 67.500 \text{ cm}^4 = 0.000675 \text{ m}^4 \\ E &= 1'300,000 \text{ T/m}^2 \\ L &= 3.0 \text{ m.} \\ \theta &= 90^\circ \end{aligned}$$

$$\begin{aligned} \gamma_{AA} &= 4EI/L = 1170 \text{ Tm.} \\ \gamma_{AB} &= 2EI/L = 585 \text{ Tm} \\ \gamma_{BB} &= 4EI/L = 1170 \text{ Tm} \end{aligned}$$

Empleando de la ecuación (3-12) o la ecuación (3-22) se calculan las rigideces auxiliares :

$$\begin{aligned} \gamma_1 &= 585 & \gamma_5 &= 390 \\ \gamma_2 &= 585 & \gamma_6 &= 37.000 \\ \gamma_3 &= 0 & \gamma_7 &= 0 \\ \gamma_4 &= 0 & & \end{aligned}$$

Ensamblando la matriz $[MRA]_m$ de acuerdo a la ecuación (3-11) se tiene:

$$[MRA]_{II} = \begin{bmatrix} 270 & 0 & 0 & 0 & 0 & 0 \\ -270 & 0 & 0 & 0 & 0 & 0 \\ 0 & 270 & 0 & 0 & 0 & 0 \\ 0 & 0 & 270 & 0 & 0 & 0 \\ -270 & 0 & 0 & 0 & 0 & 0 \\ 0 & -270 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1170 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1170 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1170 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{matrix} d_{1x} \\ d_{1y} \\ d_{1z} \\ d_{2x} \\ d_{2y} \\ d_{2z} \\ d_{3x} \\ d_{3y} \\ d_{3z} \\ d_{4x} \\ d_{4y} \\ d_{4z} \end{matrix}$$

Debido a que no hay desplazamiento en $d_{1x} = d_{1y} = d_{1z} = 0$ (no hay apoyo en posición) se eliminan las respectivas filas y columnas por lo que la matriz resultante es de tamaño 9×9 en el nodo ①.

$$[MRA]_{II} = \begin{bmatrix} \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \end{bmatrix} [K] = \begin{bmatrix} \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \\ \text{de } d_2 & d_3 & d_4 & d_5 & d_6 & d_7 & d_8 & d_9 & d_{10} \end{bmatrix} \begin{matrix} (6 \times 6) \\ (9 \times 9) \end{matrix}$$

barra ②: (columna).

$$\begin{aligned} A &= 0.0714 \text{ m} \\ I &= 0.000433 \text{ m}^4 \\ E &= 17000 \text{ MPa} \\ L &= 4.0 \text{ m} \\ \theta &= 20^\circ \end{aligned} \quad \begin{aligned} \delta_{1A} &= 877.5 \text{ Tm} \\ \delta_{2A} &= 438.75 \text{ Tm} \\ \delta_{2B} &= 877.5 \text{ Tm} \\ \delta_{1N} &= 27250.0 \text{ T/m} \end{aligned}$$

Ensamblando la matriz $[MRA]_{II}$ en forma similar a la matriz $[MRA]_{Ia}$, se tiene:

$$[MRA]_{II} = \begin{bmatrix} d_{2z} & d_{2y} & d_{2x} & d_{3z} & d_{3y} & d_{3x} \\ 166.5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 27250 & 0 & 0 & 0 & 0 \\ 0 & 0 & 27250 & 0 & 0 & 0 \\ -166.5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 27250 & 0 & 0 \\ 0 & 0 & 0 & 0 & 27250 & 0 \\ 0 & 0 & 0 & 0 & 0 & 27250 \\ 0 & 0 & 0 & 438.75 & 877.5 & 0 \\ 0 & 0 & 0 & 877.5 & 438.75 & 0 \end{bmatrix} \begin{matrix} d_{2z} \\ d_{2y} \\ d_{2x} \\ d_{3z} \\ d_{3y} \\ d_{3x} \\ d_{4z} \\ d_{4y} \\ d_{4x} \end{matrix}$$

barra ③: (columna).

Esta barra tiene las mismas propiedades geométricas y materiales de la barra ②, por lo que:

$$[MRA]_{II} = [MRA]_{II}$$

barra [4] : (trabe).

$$\begin{aligned}
 A &= bh = 30 \times 50 = 1500 \text{ cm}^2 = 0.15 \text{ m}^2 \\
 I &= bh^3/12 = 30(50)^3/12 = 312500 \text{ cm}^4 = 0.003125 \text{ m}^4 \\
 E &= 1'300,000 \text{ T/m}^2 \\
 L &= 5.10 \text{ m} \\
 \theta &= 11.31^\circ
 \end{aligned}$$

$$[MRA]_{[4]} = \begin{bmatrix} d_{x4} & d_{y4} & \psi_4 & d_{x5} & d_{y5} & \psi_5 \\ 3678.82 & 7282.31 & 183.77 & -3678.82 & -7282.31 & 183.77 \\ 7282.31 & 183.77 & -718.74 & -7282.31 & -183.77 & 718.74 \\ 183.77 & -718.74 & 3186.27 & -183.77 & 718.74 & -3186.27 \\ -3678.82 & -7282.31 & -183.77 & 3678.82 & 7282.31 & -183.77 \\ -7282.31 & -183.77 & 718.74 & 7282.31 & 183.77 & -718.74 \\ 183.77 & -718.74 & -3186.27 & -183.77 & 718.74 & 3186.27 \end{bmatrix} \begin{matrix} d_{x4} \\ d_{y4} \\ \psi_4 \\ d_{x5} \\ d_{y5} \\ \psi_5 \end{matrix}$$

barra [5] : (trabe).

$$\begin{aligned}
 A &= 0.15 \text{ m}^2 \\
 I &= 0.003125 \text{ m}^4 \\
 E &= 1'300,000 \text{ T/m}^2 \\
 L &= 4.12 \text{ m} \\
 \theta &= 345.9638^\circ
 \end{aligned}$$

$$[HRA]_{[5]} = \begin{bmatrix} d_{x5} & d_{y5} & \psi_5 & d_{x6} & d_{y6} & \psi_6 \\ 44587.00 & -10772.44 & -348.28 & -1393.11 & 10772.44 & -348.28 \\ -10772.44 & 3440.18 & -1393.11 & 10772.44 & -3440.18 & -1393.11 \\ -348.28 & -1393.11 & 3744.17 & 348.28 & 1393.11 & 1772.07 \\ -1393.11 & 10772.44 & 348.28 & 44587.00 & -10772.44 & 348.28 \\ 10772.44 & -3440.18 & 1393.11 & -10772.44 & 3440.18 & -1393.11 \\ -348.28 & -1393.11 & 1772.07 & 348.28 & -1393.11 & 3744.17 \end{bmatrix} \begin{matrix} d_{x5} \\ d_{y5} \\ \psi_5 \\ d_{x6} \\ d_{y6} \\ \psi_6 \end{matrix}$$

Al superponer las matrices de rigidez $[MRA]$ de cada barra se obtiene la siguiente matriz de rigidez global $[K]$ de la estructura:

$$[K] = \begin{bmatrix} dx_4 & dy_4 & \varphi_4 & dx_5 & dy_5 & \varphi_5 & dx_6 & dy_6 & \varphi_6 \\ 37168.84 & 7282.24 & -401.18 & -36778.84 & -7282.24 & 183.82 & 0.00 & 0.00 & 0.00 \\ 7282.24 & 40824.09 & -919.12 & -7282.24 & -1824.09 & -919.12 & 0.00 & 0.00 & 0.00 \\ -401.18 & -919.12 & 4356.27 & -183.82 & 919.12 & 1593.14 & 0.00 & 0.00 & 0.00 \\ -36778.84 & -7282.24 & -183.82 & 81530.29 & -3690.48 & -860.90 & -44586.92 & 10972.72 & -348.02 \\ -7282.24 & -1824.09 & 919.12 & -3690.48 & 34513.31 & -472.94 & 10972.72 & -3439.21 & -1392.06 \\ 183.82 & -919.12 & 1593.14 & -860.90 & -472.94 & 8007.94 & 348.02 & 1392.06 & 1972.09 \\ 0.00 & 0.00 & 0.00 & -44586.92 & 10972.72 & 348.02 & 44976.92 & -10972.72 & -236.98 \\ 0.00 & 0.00 & 0.00 & 10972.72 & -3439.21 & 1392.06 & -10972.72 & 42439.21 & 1392.06 \\ 0.00 & 0.00 & 0.00 & -348.02 & -1392.06 & 1972.09 & -236.98 & 1392.06 & 5114.17 \end{bmatrix} \begin{matrix} dx_4 \\ dy_4 \\ \varphi_4 \\ dx_5 \\ dy_5 \\ \varphi_5 \\ dx_6 \\ dy_6 \\ \varphi_6 \end{matrix}$$

Del sistema que plantea el método de las rigideces se obtienen los desplazamientos globales de la estructura de la siguiente forma:

$$[K] \{d\} = \{F\} - \{F\}^{AB}$$

Resolviendo éste sistema se obtienen los desplazamientos $\{d\}$:

$$\{d\} = \begin{Bmatrix} dx_4 \\ dy_4 \\ \varphi_4 \\ dx_5 \\ dy_5 \\ \varphi_5 \\ dx_6 \\ dy_6 \\ \varphi_6 \end{Bmatrix} = \begin{Bmatrix} 0.006862 \\ -0.000171 \\ 0.002686 \\ 0.006786 \\ -0.000473 \\ -0.000579 \\ 0.006773 \\ -0.000167 \\ 0.000086 \end{Bmatrix}$$

Con la obtención de éstos desplazamientos se calculan las fuerzas internas correspondientes a cada elemento barra de la estructura.

CALCULO DE FUERZAS INTERNAS:

Para obtener las fuerzas internas de cada elemento barra q.c. conforman a la estructura, se emplean las ecuaciones simplificadas (4-13) a (4-17).

barra II: (columna).

Anteriormente se calcularon las siguientes variables de la barra II:

$$\begin{array}{ll} \delta_N = 39.000 \text{ T/m} & c = 0 \\ \delta_{AA} = 1.170 \text{ Tm} & s = 1 \\ \delta_{AB} = 585 \text{ Tm} & L = 3.0 \text{ m} \\ \delta_{BB} = 1.170 \text{ Tm} & \end{array}$$

Aislando los desplazamientos que existen en los extremos de la barra II, se tiene:

$$\{ \delta \}_{II} = \begin{Bmatrix} dx_1 \\ dy_1 \\ \phi_1 \\ dx_2 \\ dy_2 \\ \phi_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0.006862 \text{ m} \\ -0.000171 \text{ m} \\ 0.002686 \text{ rad} \end{Bmatrix}$$

Utilizando las ecuaciones simplificadas se tiene:

$$w = \frac{dy_B - dy_A}{0.006862} = \frac{dy_2 - dy_1}{0.006862} = \frac{0.006862 - 0}{0.006862}$$

$$x = \frac{dy_B - dy_A}{-0.000171} = \frac{dy_2 - dy_1}{-0.000171} = \frac{-0.000171 - 0}{-0.000171}$$

$$y = \frac{(x_C - w_s)/L + \phi_B}{[-0.000171(0) - 0.006862(1)]/3 + 0.002686} = \frac{0.000171}{0.000171}$$

$$z = \frac{(x_C - w_s)/L + \phi_B}{[-0.000171(0) - 0.006862(1)]/3 + 0} = \frac{-0.000171}{-0.000171}$$

$$N = \frac{(x_C + w_s) \gamma_A}{[-0.000171(1) + 0.006862(0)]/39000} = \frac{-6.84 \text{ T}}{(-0.000171)}$$

$$\begin{aligned}
 M_{ZA} &= z'_{AA} + y'_{AB} \\
 &= -0.002287 (1170) + 0.002227 (585) \\
 &= -2.44 \text{ Tm}
 \end{aligned}$$

$$\begin{aligned}
 M_{ZB} &= z'_{AB} + y'_{BB} \\
 &= -0.002287 (585) + 0.002279 (1170) \\
 &= -0.87 \text{ Tm}
 \end{aligned}$$

barra ② : (columna).

Se sigue el mismo procedimiento anterior, resumido se tiene:

$$\begin{aligned}
 N &= -13.82 \text{ T (compresión).} \\
 M_{ZA} &= -2.47 \text{ Tm} \\
 M_{ZB} &= -2.74 \text{ Tm}
 \end{aligned}$$

barra ③ : (columna).

$$\begin{aligned}
 N &= -6.50 \text{ T (compresión).} \\
 M_{ZA} &= -3.71 \text{ Tm} \\
 M_{ZB} &= -3.86 \text{ Tm}
 \end{aligned}$$

barra ④ : (trabe).

$$\begin{aligned}
 N &= -5.13 \text{ T (compresión)} \\
 M_{ZA} &= 7.37 \text{ Tm} \\
 M_{ZB} &= 2.17 \text{ Tm}
 \end{aligned}$$

barra ⑤ : (trabe).

$$\begin{aligned}
 N &= -4.09 \text{ T (compresión)} \\
 M_{ZA} &= -1.67 \text{ Tm} \\
 M_{ZB} &= -0.38 \text{ Tm}
 \end{aligned}$$

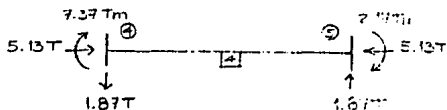
a) Diagramas de fuerzas cortantes, normales y momentos flexionantes.

Para dibujar como se indican los datos las cargas (1, 2, 3, 4) se colocará únicamente la barra 1, las otras 3 barras se dibujarán en una forma similar.

El eje de cada una de las barras 1, 2 y 3 se denominará de elementos longitudinales con dirección, es decir, a que no estén en cargas laterales.

Barra 1:

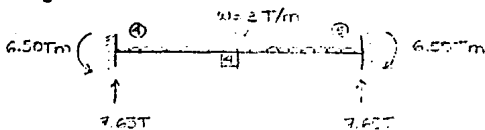
$$\begin{aligned} H_1 &= -5.13 \text{ T (compresión)} \\ H_{21} &= 7.37 \text{ Tm} \\ H_{25} &= 2.17 \text{ Tm} \end{aligned}$$



Estáticamente se calculan las fuerzas cortantes, mediante (por equilibrio de):

$$V = \frac{H_{2A} + H_{2B}}{L} = \frac{7.37 + 2.17}{5.10} = 1.87 \text{ T}$$

Carga estática inicial: $3P_e \frac{A}{B}$:



Superposición a otras cargas se tiene:

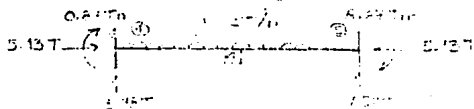


Diagrama de fuerzas normales:

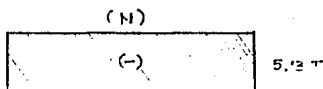


Diagrama de fuerzas cortantes:

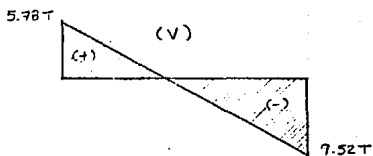
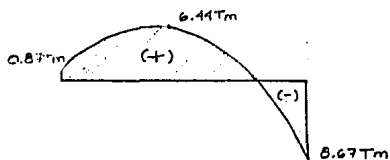
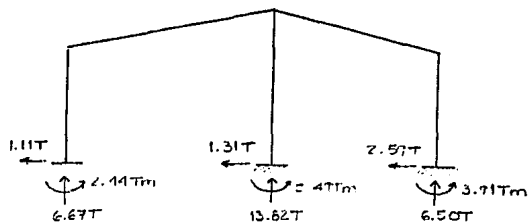


Diagrama de momentos flexionantes:



b) Reacciones en los apoyos:



CAPITULO V : DESARROLLO DE UN SUBPROGRAMA PARA EL ANALISIS SISMICO ESTATICO

CAPITULO V

DESARROLLO DE UN SUBPROGRAMA PARA EL ANALISIS SISMICO ESTATICO

5.1 DESCRIPCION GENERAL :

Las estructuras deben de ser analizadas y diseñadas para que puedan resistir los efectos de las fuerzas sísmicas. Para este fin en el análisis sísmico utilizado en la presente tesis, se idealiza al sismo como fuerzas que actúan en dos direcciones ortogonales, que coinciden con los ejes de los marcos principales del edificio analizado y que están concentradas en cada piso de la estructura.

Una vez realizado el análisis sísmico, la estructura se debe diseñar para resistir las fuerzas por sismo, en cada una de las direcciones analizadas por separado.

Para realizar el análisis de las solicitaciones que el sismo provoca en la estructura, existen diversos métodos, de los cuales solo algunos son recomendados por los Reglamentos Mexicanos de Construcción. Estos métodos tienen distintos grados de incertidumbre, en forma general se pueden dividir en dos grandes grupos:

- **Métodos Estáticos:** utilizan un modelo analítico, en el cuál se aplica a la estructura un sistema de cargas laterales cuyo efecto estático se idealiza equivalente al de la acción sísmica.
- **Métodos Dinámicos:** sus modelos analíticos realizan un análisis de la respuesta dinámica de un modelo generalmente simplificado.

El método de análisis sísmico utilizado, ya sea estático o dinámico, está en función de las características de la estructura, dependiendo de ellos se pueden utilizar los siguientes métodos:

- Método simplificado de análisis estático.
- Método de análisis estático.
- Método de análisis dinámico modal.
- Cálculo paso a paso de respuesta a temblores específicos.

Antes de iniciar el análisis sísmico, ya sea estático o dinámico, se deben definir algunas características inherentes a la estructura, debido a que las fuerzas sísmicas están en función directa de dichos aspectos. Estas características son obtenidas en este capítulo, pero también son utilizadas en el capítulo de análisis sísmico dinámico:

- Modelo analítico de la estructura.
- Masas de piso.
- Rigideces de entrepiso.
- Espectro de diseño.
- Factor de ductilidad.

Cada una de las anteriores variantes son manejadas para fines del desarrollo del programa principal (ANESAR), como una serie de subprogramas o subrutinas, todas ellas independientes de las demás. Una vez calculados y analizados los anteriores aspectos se continuará con el análisis sísmico Estático.

5.2 MODELO ANALITICO DE LA ESTRUCTURA :

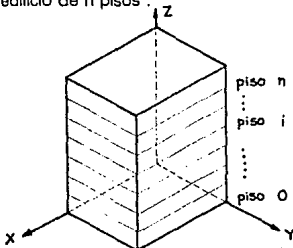
El análisis sísmico de un edificio puede ser planteado con distintos grados de refinamiento, hay que recordar que se trata de un problema probabilístico y no determinístico, por consecuencia cada persona puede plantear su propio modelo analítico de una manera cada vez más sofisticada, pero en general los planteamientos se reducen a modelar o idealizar la estructura real mediante un modelo teórico que pueda ser analizado con los procedimientos de cálculo estructural disponibles en la actualidad. Para este fin, un edificio compuesto por columnas, trabes y losas puede ser modelado como un sistema de marcos planos formados por barras con propiedades equivalentes, este tipo de modelación es la que se utilizará en los análisis sísmicos del presente trabajo.

La idealización de la estructura real es en sí una simplificación realizada mediante un modelo analítico que hace más eficiente el uso de la computadora en la solución de problemas dinámicos, debido a que se reduce el tamaño de las matrices y el número de operaciones generadas, cuyo manejo y ejecución estarían fuera del alcance de pequeños sistemas computacionales. Aún en computadoras de mayor capacidad (main frame), el análisis dinámico sofisticado es sumamente costoso y en muchas ocasiones, da como resultado respuestas equívocas debido a errores de redondeo numérico principalmente. Por consiguiente, en algunos casos es indispensable y necesario introducir una serie de simplificaciones que conduzcan a resultados con una exactitud aceptable, característica que debe tener cualquier obra de la Ingeniería Civil.

5.2.1 Idealización de un edificio real como un sistema plano de n grados de libertad :

Para obtener nuestro modelo analítico simplificado se toma en cuenta en forma general, que una estructura en el espacio (marco espacial o tridimensional) tiene seis grados de libertad por cada nodo: tres traslaciones y tres rotaciones sobre un sistema cartesiano.

Considerando un edificio de n pisos :



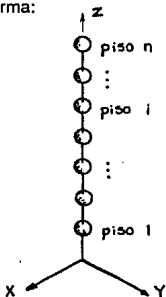
En el piso i se tienen los siguientes grados de libertad:

- Traslación paralela al eje x .
- Traslación paralela al eje y .
- Traslación paralela al eje z .
- Giro en torno a z (torsión).
- Giro en torno a x (balanceo).
- Giro en torno a y (balanceo).

Las simplificaciones realizadas al edificio para obtener el planteamiento del modelo analítico serían las siguientes:

- Los últimos dos giros en el piso i producen alargamientos por tensión o acortamientos por compresión en las columnas. Estas deformaciones generalmente son despreciables al plantear el modelo analítico del análisis sísmico.
- También se desprecia los efectos causados por la traslación paralela al eje z , causada por un movimiento trepidatorio del sismo. Quedando por lo tanto, tres grados de libertad por piso. Un edificio de n pisos, tendrá $3n$ grados de libertad.
- Se considera que el sismo actúa paralelamente a uno de los ejes horizontales (por ejemplo eje y) y que no gira, por lo que tendría en el piso i , únicamente un solo grado de libertad y_i . Bajo esta hipótesis, el edificio tendrá n grados de libertad. Posteriormente se analizará en la otra dirección horizontal (por ejemplo eje x) de igual forma.

Utilizando los anteriores criterios, nuestro Modelo Analítico para el análisis sísmico queda representado de la siguiente forma:



- En este modelo las masas m_i de los entrepisos son consideradas como si estuvieran concentradas al nivel de los pisos del edificio.
- La sumatoria de rigideces de los elementos resistentes por entrepiso que corren en dirección paralela a la considerada por el sismo son tomados como rigideces de entrepiso k_i y su masa es despreciable.

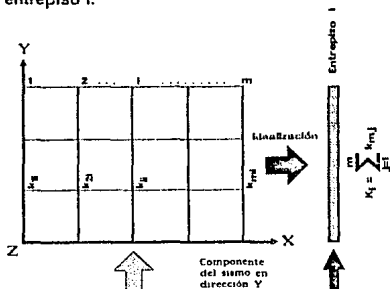
En términos generales lo que buscamos en este proceso de idealización es plantear un modelo analítico que represente de una forma muy aproximada a la respuesta global de la estructura ante las acciones de un sismo, pero además es recomendable que posteriormente se realice una serie de análisis de modelos locales para el estudio de las fuerzas internas que resulten en distintas partes críticas de la estructura.

5.2.2 Rigideces de Entrepiso :

La rigidez de entrepiso será en este caso, la suma de las rigideces de los elementos resistentes que corren en dirección paralela a la considerada por el sismo. Los elementos resistentes que contribuyen a la rigidez de entrepiso están formados principalmente por:

- Marcos planos.
- Muros de cortante.
- Contraventeos o diagonales.

Planta típica de entrepiso i:

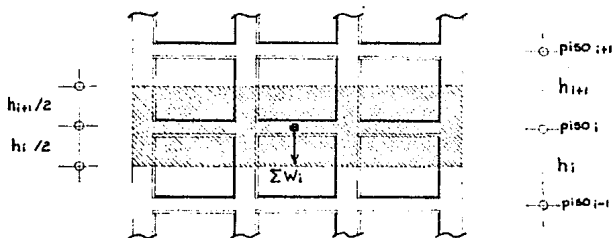


En donde:

k_{mi} : rigidez del entrepiso i, en el marco m.

5.2.3 Masas de piso :

Las masas se consideran concentradas al nivel de los pisos:



donde:

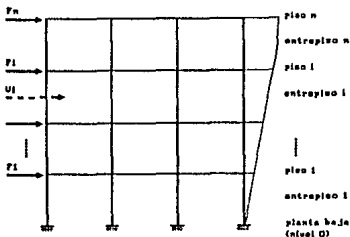
ΣW_i : suma de los pesos que gravitan sobre el piso i.

Sobre el piso i se consideran todos los pesos que sobre él gravitan (peso propio de la estructura y la carga viva), de medio entrepiso para arriba y medio entrepiso para abajo.

Las masas de piso de un edificio deben de ser calculadas externas al programa ANESAR y posteriormente deben entrar al programa como datos iniciales (INPUT). Por consiguiente, no es necesario desarrollar un subprograma que genere las masas de piso ya que su obtención es relativamente sencilla.

5.3 RIGIDECES DE ENTREPISO EN MARCOS PLANOS:

La rigidez de entrepiso en marcos planos la podemos definir como la fuerza cortante que ocasiona un desplazamiento relativo unitario (traslación) entre los dos pisos que limitan a dicho entrepiso.



$$V_i = \sum_{j=1}^n F_j \quad (5-1)$$

$$\Delta_i = y_i - y_{i-1} \quad (5-2)$$

$$K_i = \frac{V_i}{\Delta_i} \quad (5-3)$$

Las rigideces de entrepiso pueden ser obtenidas mediante los siguientes métodos:

- Ecuaciones de Wilbur.
- Utilizando un programa de análisis estructural.

Cada método tiene sus ventajas y desventajas, sin embargo se utiliza el segundo método en éste trabajo, debido a que cuenta con las siguientes características:

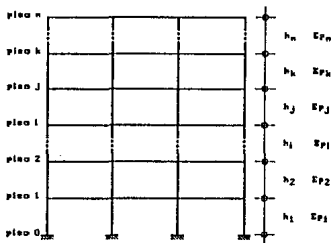
- Mayor aproximación a la rigidez que puede tener un entrepiso.
- Es aplicable tanto a estructuras regulares como a estructuras irregulares en su alzado.
- El programa principal (ANESAR) cuenta con el subprograma de análisis estructural de marcos planos (basado en el método general de las rigideces).

5.3.1 Ecuaciones de Wilbur :

Para sistemas estructurales regulares, la variación de la rigidez de entrepiso es pequeña, esta rigidez está en función directa del sistema de fuerzas laterales que se aplique al marco plano.

Wilbur desarrolló una serie de ecuaciones que permiten obtener de una forma muy aproximada la rigidez de entrepiso para un marco plano. Calculó éstas ecuaciones para dos tipos distintos de condiciones de frontera (apoyos en la base de la estructura), y son los siguientes:

a. Columnas empotradas en la base:



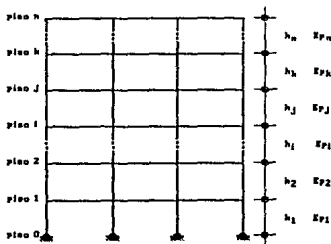
Las rigideces de entrepiso de este tipo de marco quedan definidas por las siguientes ecuaciones:

$$k_1 = \frac{48E}{h_1 \left(\frac{4h_1}{\sum K_{C_i}} + \frac{h_1 + h_2}{\sum k_{t_i} + \sum k_{C_i}/12} \right)} \quad (5-4)$$

$$k_2 = \frac{48E}{h_2 \left(\frac{4h_2}{\sum k_{c_2}} + \frac{h_1+h_2}{\sum k_{t_1}} + \frac{h_2+h_3}{\sum k_{t_2}} \right)} \quad (5-5)$$

$$k_j = \frac{48E}{h_j \left(\frac{4h_j}{\sum k_{c_j}} + \frac{h_j+h_l}{\sum k_{t_l}} + \frac{h_j+h_k}{\sum k_{t_k}} \right)} \quad (5-6)$$

b. Columnas articuladas en la base:



Las rigideces de entrepiso de este tipo de marco quedan definidas por las siguientes ecuaciones:

$$k_1 = \frac{24E}{h_1 \left(\frac{8h_1}{\sum k_{c_1}} + \frac{2h_1+h_2}{\sum k_{t_1}} \right)} \quad (5-7)$$

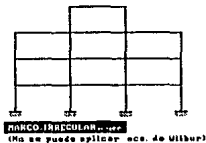
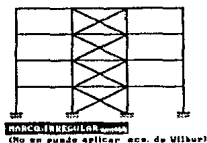
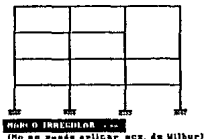
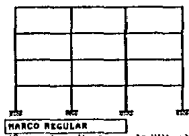
$$k_2 = \frac{48E}{h_2 \left(\frac{4h_2}{\sum k_{c_2}} + \frac{h_2+h_3}{\sum k_{t_2}} + \frac{2h_1+h_2}{\sum k_{t_1}} \right)} \quad (5-8)$$

k_j = Misma fórmula que para el caso a.

En las anteriores expresiones se emplean las siguientes variables:

k_i	Rigidez del entrepiso i.
k_c	Rigidez relativa de la columna: l_c / h_c
k_t	Rigidez relativa de una trabe : l_t / L_t
h_i	Altura de entrepiso i (entrepiso que llega por abajo al piso i).
$\sum k_{c_i}$	Suma de las rigideces relativas de las columnas del entrepiso i.
$\sum k_{t_i}$	Suma de las rigideces relativas de las trabes del piso i.
E	Módulo de Elasticidad del material que forma a la estructura.

Las ecuaciones de Wilbur son únicamente válidas para marcos planos regulares (sin cambios bruscos de rigidez) y con geometría regular en su alzado, como los marcos que a continuación se describen:



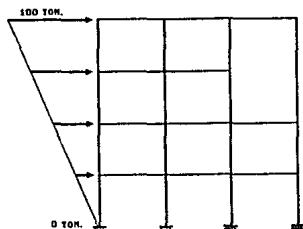
Estas Ecuaciones de Wilbur son fáciles de aplicar y tienen una aproximación muy buena a las rigideces de entrepiso, por lo que tienen la ventaja de ser un método rápido y sencillo, es recomendable si se realizan los cálculos de rigideces de entrepiso a mano.

5.3.2 Obtención de Rigideces de Entrepiso con programa de Análisis Estructural Estático :

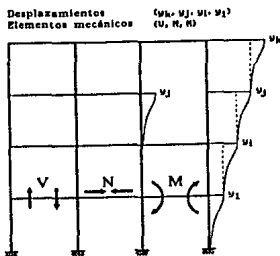
Cuando se dispone de un programa de computadora para el análisis estructural de marcos planos, las rigideces de entrepiso se pueden calcular de la siguiente forma:

Datos de Entrada (INPUT):

Se dan como datos de entrada a la geometría del marco y un sistema cualquiera de cargas laterales (se propone un sistema de variación lineal con valor máximo de 100 Ton. en el piso n y valor de 0 Ton. en el piso 0).



Datos de Salida (OUTPUT):



Se calculan las rigideces de entrepiso aplicando las siguientes definiciones:

$$\Delta_i = y_i - y_{i-1} \quad (5-9)$$

$$k_i = \frac{V_i}{\Delta_i} \quad (5-10)$$

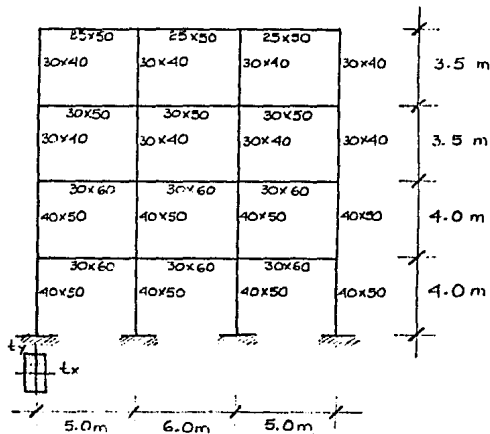
$$V_i = \sum_{j=1}^n F_j \quad (5-11)$$

5.3.3

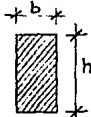
Comparación de métodos para la obtención de Rigideces de Entrepiso :

Se obtienen las rigideces de entrepiso del siguiente marco plano mediante las ecuaciones de Willbur y empleando el programa de análisis de marcos planos.

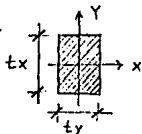
ALZADO :



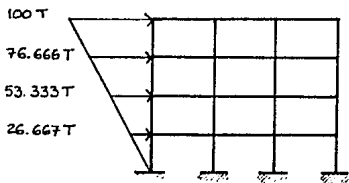
En traveses: $b \times h$



En columnas: $t_y \times t_x$



Se analiza el marco plano con la condición de carga lineal:

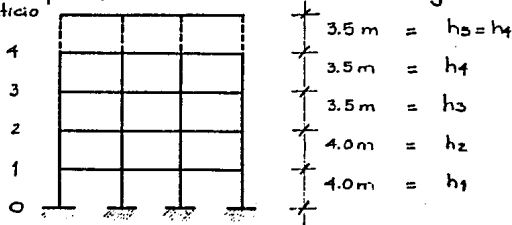


mediante el programa de análisis de marcos planos se obtienen los siguientes desplazamientos absolutos y_i :

Entrepiso	Piso	y_i [m]	Δ_i [m]	F_i [Ton]	V_i [Ton]	K_i [Ton/m]
	4	0.3936		100.0		
	3	0.3143	0.0793	76.6667	100.0	1260.97
	2	0.1897	0.1246	53.3333	176.6667	1417.56
	1	0.0866	0.1031	26.6667	230.0	2230.97
	0		0.0866		256.6667	2763.20

Empleando ecuaciones (5-1), (5-2) y (5-3) se obtienen: Δ_i , K_i y V_i

Para la obtención de las rigideces de entrepiso empleando las ecuaciones de Wilbur se requiere utilizar un piso ficticio en el último piso del marco plano analizado este piso ficticio debe de tener la misma altura que el último piso del marco plano, como se muestra en la figura:



Rigideces relativas:

$$K = I/L$$

en donde:

I Momento de inercia de la sección transversal de la barra.
L Longitud de la barra.

Cálculo de rigideces relativas:

En traves:

$$\begin{aligned} I_{t4} &= 25 \times 50^3 / 12 = 260416.67 \text{ cm}^4 \\ L &= 500 \text{ cm}; \quad K_{t4} = 260416.67 / 500 = 520.83 \text{ cm}^3 \\ L &= 600 \text{ cm}; \quad K_{t4} = 260416.67 / 600 = 434.03 \text{ cm}^3 \end{aligned}$$

$$\begin{aligned} I_{t3} &= 30 \times 50^3 / 12 = 312500 \text{ cm}^4 \\ L &= 500 \text{ cm}; \quad K_{t3} = 312500 / 500 = 625 \text{ cm}^3 \\ L &= 600 \text{ cm}; \quad K_{t3} = 312500 / 600 = 520.83 \text{ cm}^3 \end{aligned}$$

$$\begin{aligned} I_{t2} &= 30 \times 60^3 / 12 = 540000 \text{ cm}^4 \\ L &= 500 \text{ cm}; \quad K_{t2} = 540000 / 500 = 1080 \text{ cm}^3 \\ L &= 600 \text{ cm}; \quad K_{t2} = 540000 / 600 = 900 \text{ cm}^3 \end{aligned}$$

$$\begin{aligned} I_{t1} &= 30 \times 60^3 / 12 = 540000 \text{ cm}^4 \\ L &= 500 \text{ cm}; \quad K_{t1} = 540000 / 500 = 1080 \text{ cm}^3 \\ L &= 600 \text{ cm}; \quad K_{t1} = 540000 / 600 = 900 \text{ cm}^3 \end{aligned}$$

En columnas:

$$\begin{aligned} I_{c4} &= 40 \times 30^3 / 12 = 90000 \text{ cm}^4 \\ L &= 350 \text{ cm}; \quad K_{c4} = 90000 / 350 = 257.14 \text{ cm}^3 \end{aligned}$$

$$\begin{aligned} I_{c3} &= 40 \times 30^3 / 12 = 90000 \text{ cm}^4 \\ L &= 350 \text{ cm}; \quad K_{c3} = 90000 / 350 = 257.14 \text{ cm}^3 \end{aligned}$$

$$\begin{aligned} I_{c2} &= 50 \times 40^3 / 12 = 266666.67 \text{ cm}^4 \\ L &= 400 \text{ cm}; \quad K_{c2} = 266666.67 / 400 = 666.67 \text{ cm}^3 \end{aligned}$$

$$\begin{aligned} I_{c1} &= 50 \times 40^3 / 12 = 266666.67 \text{ cm}^4 \\ L &= 400 \text{ cm}; \quad K_{c1} = 266666.67 / 400 = 666.67 \text{ cm}^3 \end{aligned}$$

Suma de rigideces:

$$\begin{aligned}\sum Kt_4 &= 520.03 \times 2 & 434.03 &= 1474.07 \text{ cm}^3 \\ \sum Kt_3 &= 625.00 \times 2 & 520.83 &= 1770.83 \text{ cm}^3 \\ \sum Kt_2 &= 1080.00 \times 2 & 700.00 &= 3060.00 \text{ cm}^3 \\ \sum Kt_1 &= 1080.00 \times 2 & 700.00 &= 3060.00 \text{ cm}^3 \\ \sum Kc_4 &= 257.14 \times 4 & &= 1028.56 \text{ cm}^3 \\ \sum Kc_3 &= 257.14 \times 4 & &= 1028.56 \text{ cm}^3 \\ \sum Kc_2 &= 666.67 \times 4 & &= 2666.68 \text{ cm}^3 \\ \sum Kc_1 &= 666.67 \times 4 & &= 2666.68 \text{ cm}^3\end{aligned}$$

Empleando ecuaciones de Wilbur:

Entrepiso 1:

$$\begin{aligned}E &= 210,000 \text{ Kg/cm}^2 \\ h_1 &= 400 \text{ cm} \\ h_2 &= 400 \text{ cm} \\ \sum Kc_1 &= 2666.68 \text{ cm}^3 \\ \sum Kt_1 &= 3060.00 \text{ cm}^3 \\ \text{Empleando ecuación (5-4) se tiene:} \\ K_1 &= 29867.23 \text{ Kg/cm.}\end{aligned}$$

Entrepiso 2:

$$\begin{aligned}E &= 210,000 \text{ Kg/cm}^2 \\ h_1 &= 400 \text{ cm} \\ h_2 &= 400 \text{ cm} \\ h_3 &= 350 \text{ cm} \\ \sum Kt_1 &= 3060 \text{ cm}^3 \\ \sum Kc_1 &= 2666.68 \text{ cm}^3 \\ \sum Kt_2 &= 3060 \text{ cm}^3 \\ \sum Kc_2 &= 2666.68 \text{ cm}^3 \\ \text{Empleando ecuación (5-5) se tiene:} \\ K_2 &= 23144.06 \text{ Kg/cm.}\end{aligned}$$

Entrepiso 3:

$$\begin{aligned}E &= 210,000 \text{ Kg/cm}^2 \\ h_1 &= h_2 = 400 \text{ cm} \\ h_j &= h_3 = 350 \text{ cm} \\ h_k &= h_4 = 350 \text{ cm} \\ \sum Kt_1 &= \sum Kt_2 = 3060 \text{ cm}^3 \\ \sum Kt_j &= \sum Kt_3 = 1770.83 \text{ cm}^3 \\ \sum Kc_j &= \sum Kc_3 = 1028.56 \text{ cm}^3\end{aligned}$$

Empleando ecuación (5-6) se tiene:
 $K_j = K_3 = 14337.07 \text{ Kg/cm.}$

Entrepiso 4:

$E = 210,000 \text{ Kg/cm}^2$
 $h_i = h_3 = 350 \text{ cm}$
 $h_j = h_4 = 350 \text{ cm}$
 $h_k = h_5 = 350 \text{ cm}$ (piso ficticio).
 $\sum Kt_i = \sum Kt_3 = 1770.83 \text{ cm}^3$
 $\sum Kt_j = \sum Kt_4 = 1474.07 \text{ cm}^3$
 $\sum Kc_j = \sum Kc_4 = 1028.56 \text{ cm}^3$
 Empleando ecuación (5-6) se tiene:
 $K_j = K_4 = 12707.33 \text{ Kg/cm.}$

Si no se emplea altura de piso ficticio:
 $K_j = K_4 = 14444.38 \text{ Kg/cm.}$

Comparación de resultados:

Empleando piso ficticio:

Entrepiso	Rigideces de entrepiso:		Error relativo [%]
	Programa [T/m]	Ecs. Wilbur [T/m]	
4	1260.97	1270.73	2.36
3	1417.56	1438.71	1.50
2	2230.97	2314.41	3.74
1	2763.20	2786.72	0.79
4	1260.97	1444.44	14.55 (sin piso fict.).

En la comparación se observa un error máximo del 3.74%, lo que indica que las ecuaciones de Wilbur son confiables. Se recomienda usar el piso ficticio cuando se utilizan las ecuaciones de Wilbur, de lo contrario se generan errores en la obtención de las rigideces de entrepiso del último nivel del marco plano, éstos errores generalmente son cercanos al 15% de diferencia.

5.3.4 Subprograma de Rigideces de Entrepiso :

El subprograma aquí desarrollado de Rigideces de Entrepiso es auxiliado por el subprograma de análisis de marcos planos tratado en el Capítulo IV, en forma general calcula las rigideces de entrepiso en cada marco plano, posteriormente obtiene las rigideces totales de entrepiso en las direcciones X y Y.

Esta subrutina fue diseñada con el criterio de guardar parcialmente parte de la información a una unidad de respaldo (disco flexible o disco rígido) con el fin de aprovechar al máximo la capacidad de memoria de la computadora y además poder analizar edificios relativamente grandes. Esto quiere decir que conforme el proceso de análisis va requiriendo datos, éstos son obtenidos de la unidad de respaldo y después son desechados para liberar memoria utilizada en procesos posteriores e inmediatos.

En el planteamiento del modelo analítico utilizado en éste trabajo se simplifica un edificio de tres dimensiones (marco espacial) en un conjunto de marcos planos en dirección X y Y, debido a la naturaleza de éstos marcos planos generalmente se repiten algunos (marcos con igual geometría y propiedades mecánicas), por lo que se utiliza para fines del subprograma un concepto llamado marco tipo (marco plano), el cuál es un marco plano que forma parte del edificio en más de una vez, por lo que puede ser utilizado tanto en dirección X como en Y con el fin de ahorrar memoria y acelerar el análisis de rigideces de entrepiso.

El desarrollo del subprograma aquí tratado presentó algunas dificultades adicionales debido a que se planteó para resolver edificios que tuvieran una geometría Irregular, debido a esto se desarrollaron algoritmos para prever los distintos casos posibles de un edificio con distintas geometrías.

5.3.4.1 Diagrama de Bloques :

Se presenta el Diagrama de Bloques para explicar el proceso de éste Subprograma, en él se indican los distintos pasos a realizar, cada paso a su vez es tomado como una unidad de bloque, los cuales fueron desarrollados y analizados por separado. Este tipo de enfoque de programación facilita la solución de los problemas y es el más comúnmente utilizado en el planteamiento de sistemas grandes.

En éste diagrama de bloques se establece lo que realiza el subprograma pero no se especifica cómo se obtiene el resultado, debido a que nuevamente el explicar detalladamente paso por paso puede llevar a confusiones no relacionadas directamente con la Ingeniería Estructural. Si se requiere adentrar a una comprensión más detallada se recomienda seguir la metodología planteada en el programa fuente en Lenguaje C.

El siguiente diagrama de bloques está formado por módulos o bloques que a su vez representan una serie de procedimientos ó algoritmos para resolver un problema en específico, el conjunto de bloques a su vez resuelven el problema (encontrar las rigideces totales de entrepiso) del subprograma:

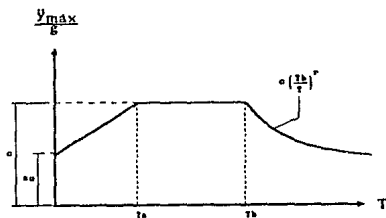
- Inicio del Subprograma.
- Cargar archivo que contiene las propiedades geométricas y mecánicas de los marcos tipo.
- Para cada marco se llevan a cabo los siguientes pasos para obtener sus respectivas rigideces de entrepiso:
 - Encontrar que nodos del marco tipo están cargados por masas de piso.
 - Calcular el sistema de cargas laterales auxiliares (con variación de 0 a 100 Ton.).
 - Cálculo de desplazamientos absolutos laterales empleando subprograma de análisis de marcos planos (subprograma de método de las rigideces).
 - Cálculo de desplazamientos relativos laterales.
 - Cálculo de rigideces de entrepiso para marco tipo analizado.
 - Guardar en un archivo de datos las rigideces de entrepiso de cada marco tipo.
- Después de calcular las rigideces de entrepiso para cada marco tipo, se siguen los siguientes pasos:
 - Cálculo de Rigideces Totales de Entrepiso en cada dirección X y Y.
 - Guarda en archivo de datos las rigideces totales de entrepiso para cada dirección X y Y.
 - Fin del subprograma.

5.4 ESPECTRO DE DISEÑO :

El espectro de diseño es una gráfica que en función del período proporciona la respuesta (pseudo aceleración) que debe ser considerada para el diseño de una estructura.

El espectro de diseño se obtiene como la envolvente de espectros de respuesta de sismos que pueden ocurrir durante la vida económica de la estructura que se diseña. Estos sismos se seleccionan con base en estudios geológicos, históricos, estadísticos y probabilísticos del sitio en que se ubica la estructura.

Los Espectros de diseño sísmico que se aplican hoy en día en México, han sido propuestos por el Instituto de Ingeniería de la U.N.A.M. para las diferentes Instituciones de Gobierno (D.D.F., C.F.E., S.C.T., Gobiernos de los Estados). Este tipo de espectros de aceleraciones propuestas toman la siguiente forma general:



donde:

- ao: Aceleraciones máximas del terreno, como fracción de g.
- c: Aceleración máxima maximorum de las estructuras, como una fracción de g.
- Ta,Tb: Períodos que limitan el intervalo de resonancia.
- r: Exponente adimensional.

El espectro de diseño sísmico está en función de algunas características de la estructura, las cuales determinan cada una de las anteriores variantes, y son las siguientes:

- Zona sísmica en la que se ubica la estructura (A, B, C, D y Cd. de México y zona conurbada).
- Tipo de terreno en que la estructura se desplanta.
- Clasificación de la estructura según su importancia.

5.4.1 Zona sísmica en la que se ubica la estructura :

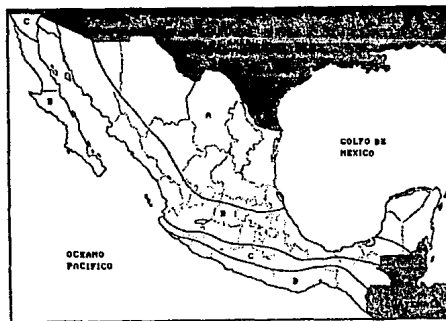
La zonificación sísmica de la República Mexicana esta basada en estudios de riesgo sísmico de lugares específicos. El riesgo sísmico de un edificio se puede determinar a partir de los siguientes factores:

- Factores geográficos, como el patrón, el tipo y el movimiento de una falla activa cercana, o las fallas y la distancia de fallas cercanas.
- Los datos de sismicidad, como la distribución de los epicentros de sismos pasados.
- Los mapas isosísmicos (Utilizando escala MM), lo cuales indican la intensidad, el grado del movimiento del terreno y la distribución de sismos futuros.
- Los rasgos geológicos, como la densidad del perfil del suelo, las velocidades de onda de cortante y los módulos de cortante.

Basándose en las técnicas anteriormente descritas, se determina el riesgo sísmico de un número de sitios suficientes para distinguir zonas de riesgo sísmico semejantes y poder asociar a ellas un valor de la aceleración y velocidad máxima del terreno.

Estudios de riesgo sísmico para el sitio específico se justifican sólo para obras de excepcional importancia, como lo son: presas y centrales núcleo eléctricas. Para las obras comunes se suele recurrir a estudios de riesgo sísmico de carácter más general, este tipo de estudios llevan a la regionalización o zonificación sísmica de un país.

El siguiente mapa de la República Mexicana muestra la regionalización sísmica en México, en dicho mapa se aprecia que el país se divide en cuatro zonas designadas con las letras de A a D en el mismo orden creciente de riesgo sísmico. Este riesgo sísmico esta en función predominantemente de la distancia a la zona de mayor generación de sismos de gran magnitud que es la que se ubica en la costa del Pacífico, frente a los estados de Guerrero, Oaxaca y Chiapas.



5.4.2 Tipo de terreno en que se desplanta la estructura :

La naturaleza del terreno sobre la cual se cimienta la estructura modifica la respuesta sísmica de la misma debido principalmente a las siguientes causas:

- Amplificación local, consiste en la modificación de las ondas sísmicas al transmitirse de la roca subyacente a los estratos del suelo que se encuentran sobre ésta y la cimentación.
- Alteración del movimiento del terreno por la presencia de la estructura, considerada ésta como cuerpo rígido.
- Interacción suelo-estructura, es la interacción entre la vibración de la estructura y la del suelo que ocasiona que una fracción significativa de la energía cinética introducida por el sismo se vuelva a transmitir al suelo y se disipe por el amortiguamiento proporcionado por éste.

En general se ha encontrado que la amplitud del suelo abajo de la estructura es menor que la que se tiene fuera de la estructura. Los efectos de las propiedades locales del suelo pueden notarse al examinar los espectros de diseño, los cuales muestran que a medida que el suelo se hace más compresible aumentan los períodos para los cuales se presentan las aceleraciones máximas (respuestas máximas).

Para efectos de la elección del Espectro de Diseño según los reglamentos mexicanos de construcción, el terreno en que se desplanta la estructura lo podemos clasificar de una forma general en:

- TIPO I: Suelo firme (roca, conglomerados compactos).
- TIPO II: Suelo intermedio (gravas, arenas compactas).
- TIPO III: Suelo blando (arcillas, arenas y limos sueltos).

5.4.3 Clasificación de las Estructuras según su importancia :

Para efectos de la seguridad estructural en las edificaciones desde el punto de vista de Reglamentos Nacionales, las construcciones pueden ser clasificadas o agrupadas en los siguientes grupos:

- A. Construcciones cuya falla estructural podría causar la pérdida de un número elevado de vidas humanas o pérdidas económicas o culturales excepcionalmente altas, o que constituyan un peligro significativo por contener sustancias tóxicas o explosivas, así como construcciones cuyo funcionamiento es esencial a raíz de una emergencia urbana, como hospitales y escuelas, estadios, templos, salas de espectáculos y hoteles que tengan salas de reunión que puedan alojar más de doscientas personas; gasolineras, depósito de sustancias inflamables o tóxicas, terminales de transporte, estaciones de bomberos, subestaciones eléctricas y centrales telefónicas y de telecomunicaciones, archivos y registros públicos de particular importancia, museos, monumentos y locales que alojen equipo especialmente costoso.

- B. Construcciones comunes destinadas a vivienda, oficinas y locales comerciales, hoteles y construcciones comerciales e industriales no incluidas en el grupo A. En general estructuras no importantes.

5.4.4 Espectros de Diseño para la República Mexicana :

Para fines prácticos de análisis sísmico, la siguiente tabla nos define el espectro de diseño a considerar, de acuerdo con las características específicas de la estructura a analizar, se tiene:

ZONA	tipo de terreno	ao	c	Ta	Tb	r
A	I	0.030	0.08	0.30	0.80	1/2
	II	0.045	0.12	0.50	2.00	2/3
	III	0.060	0.16	0.75	3.30	1
B	I	0.030	0.16	0.30	0.80	1/2
	II	0.045	0.20	0.50	2.00	2/3
	III	0.060	0.24	0.80	3.00	1
C	I	0.050	0.24	0.25	0.67	1/2
	II	0.080	0.30	0.45	1.60	2/3
	III	0.100	0.36	0.60	2.90	1
D	I	0.090	0.48	0.15	0.55	1/2
	II	0.140	0.56	0.30	1.40	2/3
	III	0.180	0.64	0.45	2.70	1
Cd. de México y Zona Conurb.	I	0.040	0.16	0.20	0.60	1/2
	II	0.080	0.32	0.30	1.50	2/3
	III	0.100	0.40	0.60	3.90	1

Los valores de la tabla anterior corresponden a construcciones del Grupo B. Para las estructuras del Grupo A, los valores de c y ao se incrementan multiplicándolos por un factor de importancia igual a 1.3 para las Zonas A a D y un factor de importancia igual a 1.5 para la Zona Metropolitana de la Cd. de México. Los valores constantes de Ta, Tb y r no cambian.

Esta disposición de los Reglamentos se maneja debido a que para las estructuras del Grupo B la intensidad de diseño corresponde a un sismo con período de retorno de 50 años, en tanto que para las estructuras del Grupo A, este período es de 200 años.

Tomando en cuenta que los reglamentos de diseño se actualizan constantemente, se ha considerado la posibilidad de proponer valores del espectro de diseño, ésta proposición se realiza en el programa de captura de datos, por lo que el espectro de diseño serán introducidos como datos de entrada (INPUT).

5.4.5 Subprograma de Espectro de Diseño :

El objetivo de éste subprograma es el de obtener el Espectro de Diseño. Para éste fin al Espectro de Diseño se le representará mediante las siguientes variables: **ao**, **c**, **Ta**, **Tb** y **r**. Para fines del subprograma aquí tratado, éstas variables a su vez van a estar en función de las variables: **zn**, **tt** y **gp**, las cuales tienen la siguiente representación:

zn: Zona sísmica en la que se ubica la estructura.

zn = 0	:	Zona A
zn = 1	:	Zona B
zn = 2	:	Zona C
zn = 3	:	Zona D
zn = 4	:	Ciudad de México y Zona Conurbada.

tt: Tipo de terreno en la que la estructura se desplanta.

tt = 0	:	Terreno Tipo I
tt = 1	:	Terreno Tipo II
tt = 2	:	Terreno Tipo III

gp: Clasificación de la estructura según su importancia.

gp = 0	:	Grupo A
gp = 1	:	Grupo B

El espectro de diseño también va a tener una representación en el subprograma, lo cual facilita el manejo interno de datos, el cambio consiste en guardar el espectro de diseño en un vector lineal llamado **es[]** de la forma siguiente:

es[0] = ao	es[1] = c	es[2] = Ta
es[3] = Tb	es[4] = r	

Se realiza éste proceso para facilitar y acelerar el paso de parámetros a los subprogramas, debido a que el espectro de diseño es requerido en los subprogramas de análisis sísmico estático y dinámico.

La tabla que define al espectro de diseño también es introducida al subprograma mediante un vector lineal llamado **ED[]** de 76 elementos, esto facilita el algoritmo del subprograma y lo simplifica.

En el caso específico del subprograma aquí tratado es más explícito incluir el diagrama de flujo que el diagrama de bloques.

5.4.5.1 Variables del subprograma :

Datos de entrada (INPUT):

zn	Zona sísmica en la que se ubica la estructura.
tt	Tipo de terreno en la que la estructura se desplanta.
gp	Clasificación de la estructura según su importancia.

Datos de salida (OUTPUT):

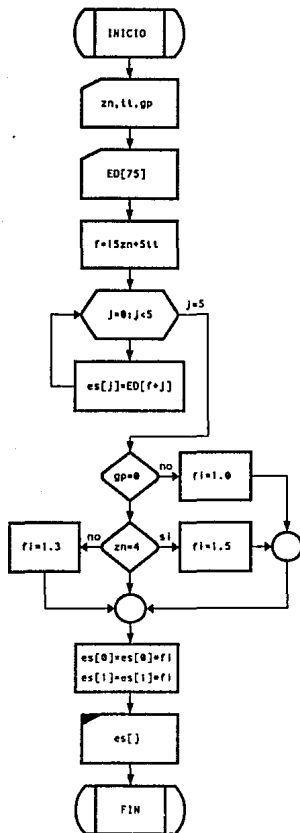
es[]	Espectro de Diseño.
------	---------------------

5.4.5.2 Diagrama de flujo :

En éste diagrama de flujo se muestra claramente la metodología para encontrar el espectro de diseño, para el subprograma analizado se puede seguir paso a paso todo el procedimiento. El diagrama de flujo está basado en los criterios de los Reglamentos Mexicanos, la teoría fue descrita en las secciones anteriores.

Los datos de entrada son datos capturados por el usuario, por lo que es responsabilidad del usuario determinar los valores de las variables de entrada, debido a que son datos que se apoyan en gráficas, criterios estructurales y criterios de mecánica de suelos.

Diagrama de flujo:



5.4.5.3 Programa fuente en Lenguaje C :

El programa fuente esta construido en base al diagrama de flujo anterior, para el desarrollo de este programa fuente se buscó el algoritmo más sencillo, ésta subrutina sólo es llamada una sola vez y además no consume mucho tiempo de ejecución, aún así se buscó optimizar todo el proceso pero sin afectar la claridad y legibilidad del programa.

```
/* INICIO DEL SUBPROGRAMA */  
  
void espectro(zn,tt,gp,es)  
int zn,tt,gp;  
double *es;  
  
{  
    double fi;  
    int f,j;  
    double ED[75] =  
    ( 0.030,0.08,0.30,0.80,0.500000, 0.045,0.12,0.50,2.00,0.666667,  
      0.060,0.16,0.75,3.30,1.000000, 0.030,0.16,0.30,0.80,0.500000,  
      0.045,0.20,0.50,2.00,0.666667, 0.060,0.24,0.80,3.00,1.000000,  
      0.050,0.24,0.25,0.67,0.500000, 0.080,0.30,0.45,1.60,0.666667,  
      0.100,0.36,0.60,2.90,1.000000, 0.090,0.48,0.15,0.35,0.500000,  
      0.140,0.56,0.30,1.40,0.666667, 0.180,0.64,0.45,2.70,1.000000,  
      0.040,0.16,0.20,0.60,0.500000, 0.080,0.32,0.30,1.50,0.666667,  
      0.100,0.40,0.60,3.90,1.000000  
    );  
  
    fi=15*zn+5*tt;  
    for(j=0;j<5;j++) es[j]=ED[f+j];  
  
    if(gp==0)  
        if(zn==4) fi=1.50; else fi=1.30;  
    else  
        fi=1.00;  
  
    es[0]=*fi;  
    es[1]=*fi;  
  
}  
  
/* FIN DEL SUBPROGRAMA */
```

5.5 FACTOR DE DUCTILIDAD :

Ductilidad: Propiedad que tienen las estructuras de experimentar grandes deformaciones antes de la falla o colapso.

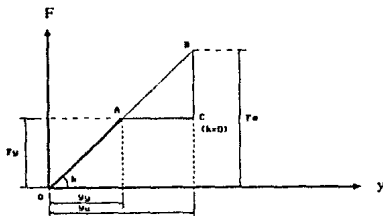
El comportamiento de las estructuras dúctiles a la falla no es elástico, debido a que la estructura **no** recupera su forma original si es retirada la carga. Tampoco es lineal porque las deformaciones no son proporcionales a las cargas.

La elasticidad de las estructuras obedece a 2 razones principalmente:

- Los materiales de construcción no son elástico lineales.
- La falla de una estructura se presenta por la acumulación de fallas locales que van degradando su rigidez, hasta crear una serie de nodos plásticos que llegan a formar un mecanismo local que puede colapsar total o parcialmente a la estructura.

5.5.1 Método práctico para la consideración de la ductilidad en los Espectros de Diseño :

En la práctica, la ductilidad es considerada de la siguiente forma: Se supone un comportamiento elastoplástico de la estructura, con la siguiente relación (F - y) :



donde:

- OA: Rama elástica, en la que la estructura tiene una rigidez k .
- AC: Rama plástica, en que la rigidez de la estructura es igual a 0.
- Yy: Deformación a la cual se inicia la fluencia. Al alcanzar esta deformación la fuerza ya no aumenta, conserva el valor F_y (yield = fluencia).
- Yu: Deformación última correspondiente a la falla de la estructura.

De los conceptos anteriores se define como factor de ductilidad Q a la relación dada por:

$$Q = \frac{y_u}{y_y} = \frac{\text{def. última}}{\text{def. al inicio de la fluencia}} = \frac{OC'}{OA'} \quad (5-12)$$

Por semejanza de triángulos:

$$\Delta OAA' \approx \Delta OBC' \quad (5-13)$$

$$\frac{OC'}{OA'} = \frac{BC'}{AA'} \quad (5-14)$$

Pero OB nos representa el comportamiento que tendría la estructura si fuese elástica hasta la falla y BC' nos representa la función elástica que se desarrollaría en tal caso (F_e). Como $AA' = F_y$, podemos escribir:

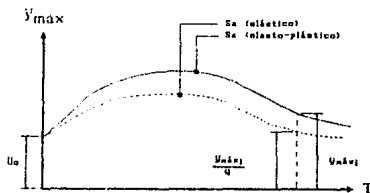
$$\frac{BC'}{AA'} = \frac{F_e}{F_y} \quad (5-15)$$

$$\frac{OC'}{OA'} = \frac{F_e}{F_y} \quad (5-16)$$

$$Q = \frac{F_e}{F_y} \quad (5-17)$$

$$F_y = \frac{F_e}{Q} \quad (5-18)$$

Estos resultados definido por la ecuación (5-18) nos indican que para el cálculo de fuerzas, se podrá usar el espectro elástico, dividiendo las ordenadas entre Q para obtener el espectro elastoplástico como lo muestra la siguiente figura:



Se debe notar que para $T = 0$ la aceleración de la estructura **no** puede ser menor que la del terreno, por lo que en este caso particular $Q = 1$.

Los valores de Q se definen experimentalmente y se expresan en los Reglamentos de Construcción en función del tipo de estructuración y de requisitos de diseño.

El factor de ductilidad Q puede variar en las dos direcciones ortogonales en que se analice la estructura, según sean las propiedades estructurales de ésta en dichas direcciones.

5.5.2 Factores de Ductilidad en Reglamentos Mexicanos :

Para el caso particular de las N.T.C. al factor de ductilidad se le conoce también como Factor de Comportamiento Sísmico. El factor de ductilidad Q se define experimentalmente y está dado en los reglamentos de construcción en función del tipo de estructura y de requisitos de diseño, adopta en términos generales los siguientes valores:

I. $Q = 4$

Se usará $Q = 4$ cuando se cumplan los requisitos siguientes:

1. La resistencia en todos los entrepisos es suministrada exclusivamente por marcos no contraventeados de acero o concreto reforzado en los que en cada entrepiso los marcos sean capaces de resistir, sin contar con muros ni contravientos, cuando menos 50 por ciento de las fuerzas sísmicas actuantes.
2. Si hay muros ligados a la estructura de tal forma que contribuyan a resistir las fuerzas laterales, su rigidez se tomará en cuenta en el análisis sísmico y se verificará su resistencia, pero su contribución a la capacidad ante fuerzas laterales sólo se tomará en cuenta si estos muros son de piezas macizas, y los marcos, sean o no contraventeados, y los muros de concreto reforzado son capaces de resistir al menos el 80 por ciento de las fuerzas laterales totales sin la contribución de los muros de mampostería.
3. El mínimo cociente de la capacidad resistente de un entrepiso entre la acción de diseño no difiere más de 35 por ciento del promedio de dichos cocientes para todos los entrepisos. Para verificar el cumplimiento de estos requisitos, se calculará la capacidad resistente de cada entrepiso teniendo en cuenta todos los elementos que puedan contribuir a la resistencia.
4. Los marcos y muros de concreto reforzado deben cumplir con los requisitos fijados para marcos y muros dúctiles.
5. Los marcos rígidos de acero deben satisfacer los requisitos para marcos dúctiles.

II. $Q = 3$

Se adoptará $Q = 3$ cuando se satisfacen las condiciones 2, 4 y 5 del caso I y en cualquier entrepiso dejan de satisfacer las condiciones 1 o 3 especificadas para el caso I pero la resistencia en todos los entrepisos es suministrada por columnas de acero o de concreto reforzado con losas planas, por marcos rígidos de acero, por marcos de concreto reforzado, por muros de este material, por combinaciones de éstos y marcos o por diafragmas de madera contrachapada. Las estructuras

con losas planas deberán además satisfacer los requisitos que sobre el particular marcan las normas técnicas complementarias para estructuras de concreto.

III. $Q = 2$

Se usará $Q = 2$ cuando la resistencia a fuerzas laterales es suministrada por losas planas con columnas de acero o de concreto reforzado, por marcos de acero o de concreto reforzado, contraventeados o no, o muros o columnas de concreto reforzado, que no cumplen en algún entrepiso lo especificado por los casos I y II de esta sección, o por muros de mampostería de piezas macizas confinadas por castillos, dadas, columnas o traveses de concreto reforzado o de acero que satisfacen los requisitos de las normas complementarias respectivas, o diafragmas contraventeados o por sistemas de muros formados por duelas de madera horizontales o verticales combinados con elementos diagonales de madera maciza. También se usará $Q = 2$ cuando la resistencia es suministrada por elementos de concreto prefabricado o preforsado, con las excepciones que sobre el particular marcan las normas técnicas complementarias para estructuras de concreto.

IV. $Q = 1.5$

Se usará $Q = 1.5$ cuando la resistencia a fuerzas laterales es suministrada en todos los entrepisos por muros de mampostería de piezas huecas, confinados o con refuerzo interno, que satisfacen los requisitos de las normas complementarias respectivas, o por combinaciones de dichos muros con elementos como los descritos para los casos II y III o por marcos y armaduras de madera.

V. $Q = 1$

Se usará $Q = 1$ en estructuras cuya resistencia a fuerzas laterales es suministrada al menos parcialmente por elementos o materiales diferentes de los arriba especificados, a menos que se haga un estudio que demuestre, a satisfacción del Departamento, que se puede emplear un valor más alto que el que aquí se especifica.

En todos los casos se usará para toda la estructura en la dirección de análisis el valor mínimo de Q que corresponde a los diversos entrepisos de la estructura en dicha dirección.

En resumen, la determinación del Factor de Ductilidad es responsabilidad del calculista pero basándose predominantemente en los Reglamentos, por consecuencia no es necesario realizar un subprograma para obtenerlo, el factor es leído como dato de entrada (INPUT).

5.5.3 Reducción de Fuerzas Sísmicas :

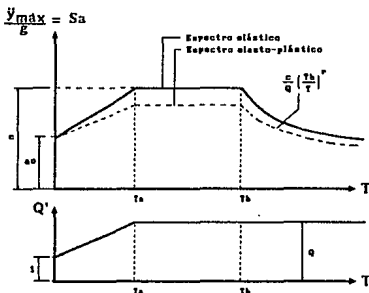
Con fines de diseño, las fuerzas sísmicas para análisis estático y las obtenidas del análisis dinámico, se podrán reducir dividiéndolas entre el factor reductor Q' .

Los valores de la ductilidad de una estructura son aplicables a las ramas horizontal y descendente del espectro. Para la rama ascendente Q se substituye por Q' mediante:

$$Q' = 1 + (Q - 1) / T_a * T \quad (5-19)$$

Si $T = 0$, $Q' = 1$, por que en este caso la aceleración de la estructura no puede ser mayor que la del terreno.

Gráficamente la descripción anterior se resume así:



Se debe de tomar en cuenta que la reducción de fuerzas sísmicas por ductilidad, es válida para el cálculo de fuerzas, pero no para el cálculo de desplazamientos. Los desplazamientos elasto-plásticos quedarán iguales a los elásticos.

5.5.4 Subprograma de reducción de fuerzas sísmicas (empleando el factor de ductilidad) :

Este subprograma forma parte del subprograma de análisis sísmico dinámico, sin embargo se da una explicación más amplia en ésta sección.

Las fuerzas sísmicas pueden ser reducidas empleando el método descrito por los Reglamentos Mexicanos, ésta reducción se realiza disminuyendo las respuestas de la estructura mediante la modificación del espectro de diseño, dicho espectro modificado es conocido como espectro de diseño plástico.

La reducción puede ser aplicada al método sísmico estático y al método sísmico dinámico, debido a que ésta reducción de fuerzas sísmicas está en función de los períodos de la estructura por lo regular no se le aplica al análisis estático. Si se requiere aplicar la reducción al análisis estático se tiene que conocer el período fundamental mediante un método aproximado (Método de Raleigh), además de que la estructura debe de cumplir con las condiciones de regularidad fijadas por los Reglamentos Mexicanos.

5.5.4.1 Variables del subprograma :

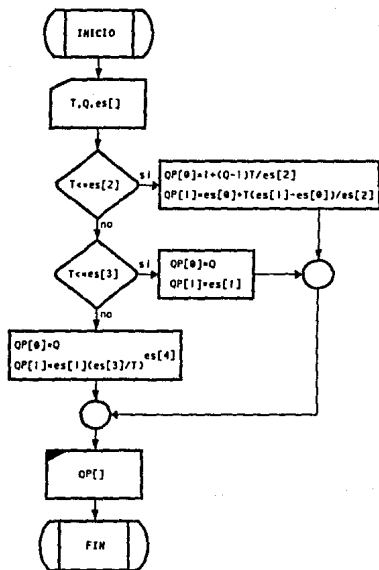
Variables de entrada (INPUT):

Q	Factor de ductilidad sin reducción.
T	Período a analizar de la estructura.
es[]	Espectro de diseño.
es[0]	: ao
es[1]	: c
es[2]	: Ta
es[3]	: Tb
es[4]	: r

Variables de salida (OUTPUT):

QP[]	Vector lineal que contiene a Q' y c'
QP[0]	: Q'
QP[1]	: c'

5.5.4.2 Diagrama de flujo :



5.5.4.3 Programa fuente en Lenguaje C :

```
/* INICIO DEL SUBPROGRAMA */  
void reduce (T, Q, es, QP)  
double T, Q, *es, *QP;  
{  
    if(T<=es[2]) {  
        QP[0]=1+(Q-1)*T/es[2];  
        QP[1]=es[0]+T*(es[1]-es[0])/es[2];  
    } else  
    if(T<=es[3]) {  
        QP[0]=Q;  
        QP[1]=es[1];  
    } else {  
        QP[0]=Q;  
        QP[1]=es[1]*pow(es[3]/T, es[4]);  
    }  
}  
/* FIN DEL SUBPROGRAMA */
```


5.6 ANALISIS SISMICO ESTATICO :

Ante las grandes dificultades que presentan los métodos de análisis sísmico dinámico, se han desarrollado, para el caso específico de edificios (y algunas otras estructuras más simples), otros métodos aproximados ó menos sofisticados, los cuales emplean cargas laterales para reproducir los efectos dinámicos en el sistema oscilatorio, dichos métodos también llegan a recibir el nombre de métodos cuasiestáticos.

Utilizando estos métodos estáticos o cuasiestáticos, se simplifica la elección de las cargas estáticas apropiadas para obtener un determinado tipo de respuesta (elementos mecánicos, esfuerzos y desplazamientos), y también se simplifica su aplicación al edificio por medio de cualquiera de los métodos tradicionales de análisis estructural estático.

La metodología de este análisis es más sencilla que la utilizada por cualquier método dinámico. Se trata de un criterio de análisis que parte de estudios experimentales de resistencias requeridas por distintas estructuras, ante la acción de fuerzas horizontales, con el fin de que tengan un comportamiento adecuado en relación con sismos de diversas magnitudes. Como resultado de estos estudios, se ha aceptado proponer una distribución de fuerzas laterales tal que su efecto en cada piso sea aproximado a los distintos efectos dinámicos que tendría un sismo sobre una estructura.

No existe una relación directa entre el comportamiento de la estructura ante un sismo y las fuerzas equivalentes consideradas en este método. En realidad lo que se busca es obtener una estructura con una determinada resistencia lateral, que sea capaz de resistir un movimiento sísmico sin sufrir daños estructurales de falla.

El Análisis Sísmico Estático es un análisis simplificado y conservador, se le debe de considerar como un método alternativo a los métodos dinámicos, aplicable a edificios regulares sin cambios bruscos en la distribución de sus masas, rigideces, etc.. Otra restricción que se debe cumplir, es que se trate de un edificio de menos de 60 metros de altura.

Los métodos estáticos tienen sus bases en la determinación de una fuerza equivalente lateral total (cortante basal) en función de las fuerzas de inercia que se generan en un sistema equivalente de un grado de libertad, para posteriormente redistribuir este cortante en fuerzas concentradas a las diferentes alturas de los pisos de la estructura analizada, las cuales son obtenidas suponiendo que ésta va a vibrar predominantemente en su primer modo natural. Este modelo de análisis estático es exclusivo de los Reglamentos Mexicanos, en otros países éste mismo análisis estático tiene otras variantes y fundamentos, aunque esencialmente es el mismo.

5.6.1 Método del Análisis Sísmico Estático :

El método estático está basado en la predominancia del primer modo (o modo fundamental) y además establece las siguientes simplificaciones:

- El período de la estructura cae en la rama horizontal del espectro, es decir :

$$T_a \leq T \leq T_b$$

- En consecuencia la respuesta espectral es la máxima posible. Para el caso de aceleraciones, esta respuesta es **c**.

- El cortante en la base es :

$$V_b = (c / Q) * W \quad (5-20)$$

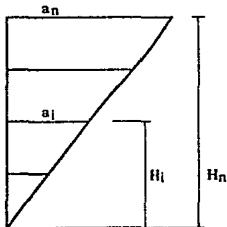
En donde :

c	(aceleración espectral máxima) / (aceleración de la gravedad)
Q	factor de ductilidad.
W	Peso total del edificio.

- Las fuerzas actuantes en cada piso son el producto de un coeficiente por el peso de ese piso. Por lo que para el piso *i*, se tiene :

$$F_i = a_i (W_i) \quad (5-21)$$

- Los coeficientes *a_i* se distribuyen linealmente desde un valor 0 en la base del edificio hasta un valor *a_n* en el último piso *n*, del edificio :



$$a_i = (a_n / H) * H_i \quad (5-22)$$

En donde :

a_i	Coficiente en piso i.
a_n	Coficiente en piso n.
H	Altura total del edificio.
H_i	Altura del piso i.

- Por lo que F_i , queda representado por :

$$F_i = (a_n / H) * H_i * W_i \quad (5-23)$$

- La suma de fuerzas laterales en los pisos, da el cortante en la base, de la siguiente forma:

$$V_b = \sum_{i=1}^n F_i = \sum_{i=1}^n \frac{a_n H_i W_i}{H} \quad (5-24)$$

$$V_b = \frac{a_n}{H} \sum_{i=1}^n W H_i = \frac{c}{Q} W \quad (5-25)$$

- De la ecuación (5-25) podemos obtener a_n :

$$a_n = \frac{c H}{Q} \frac{W}{\sum_{i=1}^n W H_i} \quad (5-26)$$

- Se debe verificar que c/Q no sea menor que a_o . En caso de que c/Q sea menor que a_o , entonces $c/Q = a_o$.
- Después de haber obtenido las fuerzas sísmicas estáticas es importante encontrar los desplazamientos relativos (traslaciones horizontales del modelo analítico), para posteriormente revisar estados límites de servicio. Estos desplazamientos se pueden obtener de la siguiente forma :

$$V_i = \sum_{k=1}^n F_k \quad (5-27)$$

$$\Delta_i = \frac{V_i}{K_i} \quad (5-28)$$

$$y_i = \Delta_i + y_{i-1} \quad (5-29)$$

En donde:

V_i	Cortantes sísmicos del entrepiso i.
Δ_i	Desplazamiento relativo del entrepiso i.
y_i	Desplazamiento total del piso i.
K_i	Rígidez del entrepiso i.
F_i	Fuerza sísmica del piso i.

El método de análisis estático tiene las variantes de poder analizar péndulos invertidos y apéndices, con algunas consideraciones que cambian al análisis para edificios, pero debido a que este trabajo está dedicado exclusivamente al análisis de edificios, no son incluidas estas opciones.

5.6.2 Subprograma de Análisis Sísmico Estático :

En los temas anteriores se obtuvieron las siguientes características estructurales del edificio analizado:

- Modelo analítico de la estructura.
- Rigideces de entrepiso.
- Factor de ductilidad.
- Masas de piso.
- Espectro de diseño.

Cada una de estas variantes fue calculada externamente a este subprograma, con el fin de que cada unidad de subprograma realice una función muy específica y a su vez sea lo más clara posible.

Los subprogramas auxiliares del subprograma de análisis sísmico estático generan como resultado unos datos de salida (OUTPUT) los cuales son parámetros leídos por éste subprograma como datos de entrada (INPUT).

El algoritmo está basado en la anterior metodología explicada en el análisis sísmico estático, funciona con un conjunto de datos básicos (parámetros), genera la salida de fuerzas sísmicas estáticas y desplazamientos totales de piso.

5.6.2.1 Variables del Subprograma :

Los siguientes datos de entrada, son datos calculados en otras subrutinas o son datos básicos capturados por el usuario del programa (ANESAR).

Entrada de datos (INPUT) :

hi	Altura del piso i.
Wi	Peso del entrepiso i, concentrada en el piso i.
Ki	Rigidez total del entrepiso i (en la dirección analizada X o Y).
es[]	Espectro de diseño.
Q	Factor de ductilidad (en la dirección analizada X o Y).
n	Número de pisos del edificio (modelo analítico).

Salida de datos (OUTPUT) :

Vi	cortante del entrepiso i.
yi	desplazamiento absoluto de cada piso i.

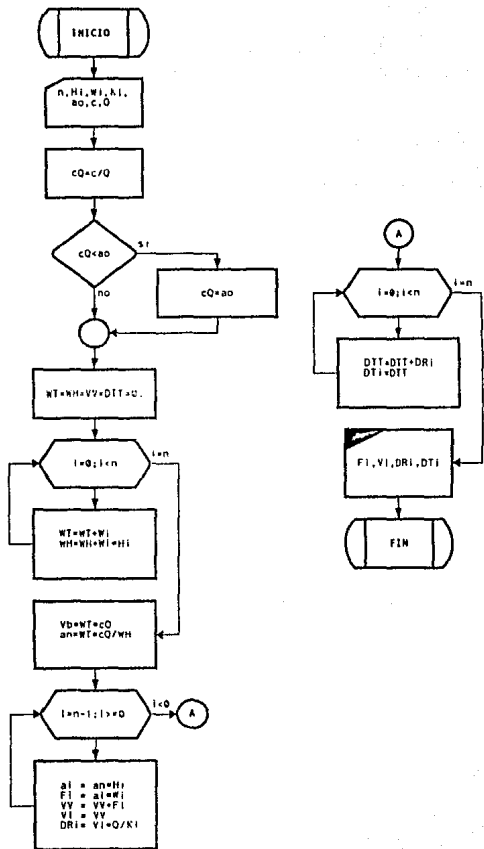
Todas las variables utilizadas por el subprograma que no hayan sido definidas en los datos de entrada o salida, las podemos considerar como auxiliares, las cuales realizan distintas funciones en el subprograma, ya sea como contadores, sumatorias, variables propias de algún algoritmo, etc.

5.6.2.2 Diagrama de flujo :

Si se quiere comprender la metodología del análisis sísmico estático se puede emplear el diagrama de flujo para tal fin, este diagrama muestra más claramente los distintos procesos que se deben de seguir para obtener las fuerzas sísmicas estáticas, para entenderlo de una forma más general se utiliza el digrama de bloques.

Al terminar de procesar este subprograma , se regresa el control del flujo al programa principal (ANESAR), se borran todas las variables auxiliares (locales) y solo se mantienen las variables de salida (OUTPUT), las cuales son necesarias en otros subprogramas posteriores al subprograma aquí planteado.

Se incluyen únicamente los diagramas de flujo en subprogramas relacionados directamente con la Ingeniería Civil Estructural, en el caso de subprogramas auxiliares sólo es incluido su diagrama de bloques y programa fuente, debido principalmente a que solo son métodos comunes de programación general y presenta las dificultades de que sus programas fuentes y diagramas de flujo (en algunos casos) resultan ser más complejos que los relacionados a la Ingeniería Civil, por lo que se podrían crear confusiones innecesarias.



5.6.2.3 Programa fuente en Lenguaje C :

El subprograma de análisis sísmico estático esta basado completamente en su diagrama de flujo, y es llamado por el programa principal como una subrutina, el código completo es presentado en el Apéndice A, junto con los demás subprogramas.

El diseño de éste subprograma esta basado en la concepción de seguir fielmente el diagrama de flujo del Análisis Sísmico Estático sin tratar de optimizar algún proceso, debido a que éste subprograma no consume demasiado tiempo de ejecución.

```
void an_esta(n, fse, Q, K, W, es, H)
    int n;
    double *fse, Q, *W, *K, *es, *H;

{
    int i;
    double *V, *DT, *DR, *a;
    double c0, WT, WH, DTI, VW, an;

/*  VARIABLES UTILIZADAS EN SUBPROGRAMA

n      :      Número de pisos de la estructura.
Q      :      Factor de ductilidad en función del tipo de estructura y de los requisitos de diseño
            por cada dirección analizada.
W(n)   :      Peso de los entrepisos concentrados en cada uno de los n pisos.
K(n)   :      Rigideces totales del entrepiso i, en la dirección del análisis.
es(0)=a0:  Aceleración máxima del terreno, como fracción de g.
es(i)=c :  Aceleración máxima máxima de las estructuras, como fracción de g.
H(n)   :      Altura de los n piso.
V(n)   :      Cortantes de entrepiso.
DT(n)  :      Desplazamientos absolutos de cada piso.
DR(n)  :      Desplazamientos relativos de cada piso.
fse(n) :      Fuerzas sísmicas actuantes en cada pisos.
a(n)   :      Coeficientes de cada piso.
an     :      Coeficiente en piso n.
WT     :      Peso total del edificio.
c0     :      Factor c/Q, siempre mayor que a0.

NOTA   :      Todas las demás variables que no están indicadas en esta lista son variables auxiliares
            que no tienen significado en el análisis sísmico estático.

*/

V = calloc(n+1, B); DT = calloc(n+1, B);
DR = calloc(n+1, B); a = calloc(n+1, B);
c0 = es[1]/Q; if(c0 < es[0]) c0 = es[0];
WT = WH + VW + DTI = 0;

for(i=1; i <= n; i++) {
    WT += W[i];
    WH += W[i] * H[i];
}

an = WT * c0 / WH;

for(i=n; i > 0; i--) {
    a [i] = an * H[i];
    fse[i] = a[i] * W[i];
    V [i] = VW + fse[i];
    DR [i] = V[i] * Q / K[i];
}

for(i=1; i <= n; i++) DT[i] = DTI + DR[i];
}
```


5.7 EJEMPLO :

Empleando el análisis sísmico estático calcular los cortantes de entrepiso y los desplazamientos de cada piso en la dirección X del edificio que a continuación se describe:

El edificio a analizar es un hospital ubicado en la Deleg. Gustavo A. Madero, D.F. La estructuración del edificio la conforman marcos dúctiles de concreto reforzado.

Hospital: Grupo A, factor de importancia 1.5 para zona Metropolitana.

Ubicación: Deleg. Gustavo A. Madero, zona sísmica especial para la Cd. de México y áreas conurbadas.

Terreno: Terreno blando, tipo III.

Estructuración: Edificio estructurado con marcos dúctiles de concreto reforzado en ambos sentidos de sus ejes principales, $Q_x = Q_y = 4$.

Para ejemplificar el análisis sísmico estático únicamente se analizará la dirección X, el análisis sísmico en la dirección Y se realiza de una forma similar

Espectro de diseño:

De acuerdo a las características del edificio su espectro de diseño correspondiente es el siguiente:

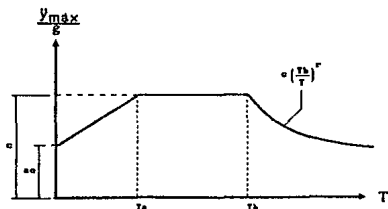
$$a_0 = 0.10 \times 1.5 = 0.15 \text{ g}$$

$$c = 0.40 \times 1.5 = 0.60 \text{ g}$$

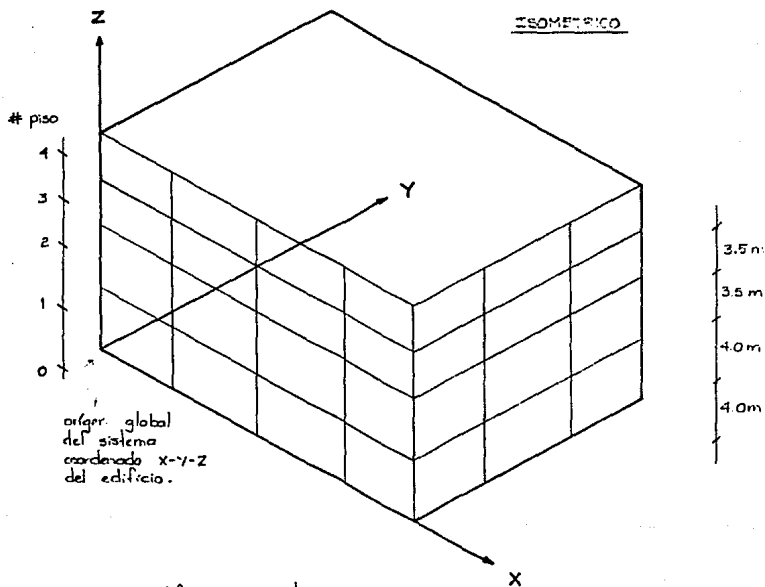
$$T_a = 0.60 \text{ seg.}$$

$$T_b = 3.90 \text{ seg.}$$

$$r = 1.00 \text{ []}$$



MODELO ANALITICO :



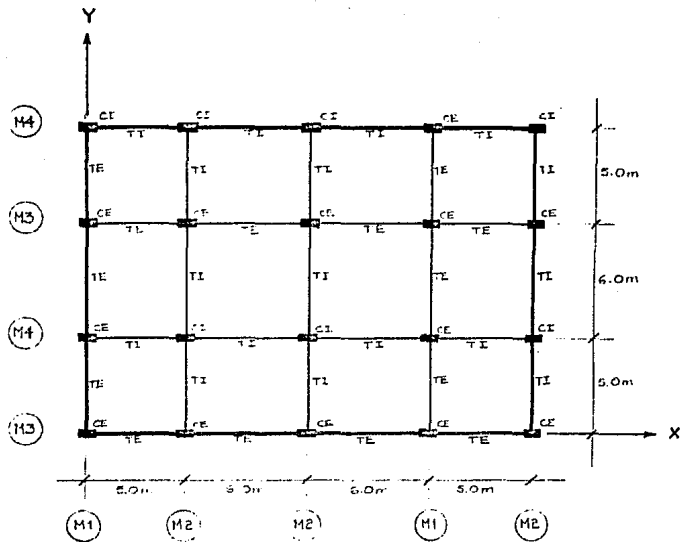
Edificio compuesto por marcos dúctiles de concreto.

$$E = 2'100,000 \text{ T/m}^2$$

$$Q_x = 4$$

$$Q_y = 4$$

PLANTA



M1: Marco tipo 1

M2: Marco tipo 2

M3: Marco tipo 3

M4: Marco tipo 4

Marcos tipos en X : 2 (M3, M4)

Marcos tipos en Y : 2 (M1, M2)

Marcos en X : 4 (M3, M4, M3, M4)

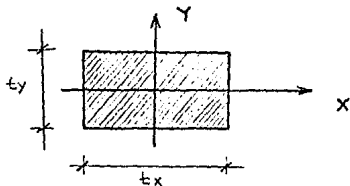
Marcos en Y : 5 (M1, M2, M2, M1, M2)

Número de marcos tipo : 4 (M1, M2, M3, M4).

SECCIONES EMPLEADAS :

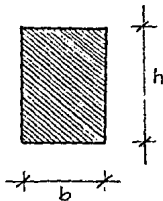
Columnas :

plano	CE t_x [cm]	t_y [cm]	CI t_x [cm]	t_y [cm]
4	50	30	40	30
3	50	30	40	30
2	40	40	40	40
1	40	40	40	40



Trabes :

plano	TE b [cm]	h [cm]	TI b [cm]	h [cm]
4	25	50	25	50
3	30	50	30	40
2	30	60	30	50
1	30	60	30	50



Peso total del edificio:

$$W = 350\text{Ton} + 380\text{Ton} + 450\text{Ton} + 470\text{Ton} = 1650 \text{ Ton.}$$

Cortante basal:

$$V_b = 1650\text{Ton} (0.15) = 247.50 \text{ Ton.}$$

Análisis sísmico estático en la dirección X:

piso	Wi [Ton]	Hi [m]	Wi(Hi) [Ton.m]
4	350	15.0	5250.0
3	380	11.5	4370.0
2	450	8.0	3600.0
1	470	4.0	1880.0
$\Sigma Wi(Hi) =$			15100.0

Empleando ecuación (5-26) se tiene :

$$\begin{aligned} a_4 &= 0.15(15.0)(1650)/15100 = 0.2459 \\ a_3 &= 0.15(11.5)(1650)/15100 = 0.1885 \\ a_2 &= 0.15(8.0)(1650)/15100 = 0.1311 \\ a_1 &= 0.15(4.0)(1650)/15100 = 0.0656 \end{aligned}$$

Fuerzas sísmicas estáticas :

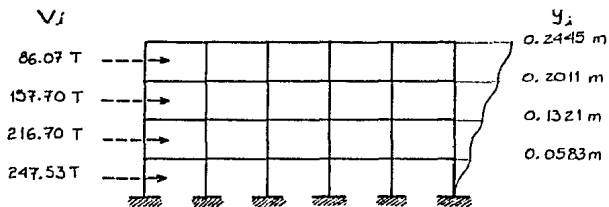
$$\begin{aligned} F_4 &= 0.2459 (350\text{Ton.}) = 86.07\text{Ton.} \\ F_3 &= 0.1885 (380\text{Ton.}) = 71.63\text{Ton.} \\ F_2 &= 0.1311 (450\text{Ton.}) = 59.00\text{Ton.} \\ F_1 &= 0.0656 (470\text{Ton.}) = 30.83\text{Ton.} \end{aligned}$$

$$V_b = 247.53\text{Ton.}$$

Emplando ecuaciones (5-27), (5-28) y (5-29) se tiene :

PISO	ENTREPISO	F_x [Ton]	V_x [Ton]	K [Ton/m]	Δ [m]	$y=\Sigma\Delta$ [m]
4		86.07				0.2445
3	4	71.63	86.07	7939.8885	0.0434	0.2011
2	3	59.00	157.70	9148.1710	0.0690	0.1321
1	2	30.83	216.70	11740.4294	0.0738	0.0583
	1		247.53	16990.5027	0.0583	

CORTANTES DE ENTREPISO :



CAPITULO VI: DESARROLLO DE UN SUBPROGRAMA PARA EL ANALISIS SISMICO DINAMICO

CAPÍTULO VI

DESARROLLO DE UN SUBPROGRAMA PARA EL ANÁLISIS SISMICO DINAMICO

6.1 DESCRIPCION GENERAL :

Es importante tener en cuenta la actividad vibratoria o dinámica de una estructura además de su comportamiento estático. Las cargas dinámicas como las que ocasionan los sismos, vientos, explosiones, mareas, etc. son las que suelen provocar un gran número de estados de fallas estructurales. Esto se debe principalmente a que las cargas dinámicas generalmente actúan en dirección lateral, para lo cual la resistencia o rigidez de las estructuras son las más desfavorables.

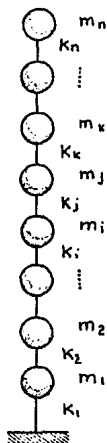
El análisis de los efectos dinámicos en una estructura está a cargo de la dinámica estructural, esta área de la Ingeniería Civil ha tomado un gran auge actualmente debido al fácil acceso que se puede tener a computadoras digitales de relativa gran capacidad, sin dichas herramientas sería imposible (en algunos casos) efectuar estudios y análisis para los complejos sistemas estructurales que se construyen hoy en día.

En el análisis sísmico dinámico aquí empleado se utilizó el análisis modal. Este análisis incluye los efectos de todos los modos naturales de vibración con períodos mayores o iguales a 0.4 seg., incluyendo como mínimo a los tres primeros modos de traslación para cada dirección analizada (ejes principales X y Y). El efecto de las excentricidades estáticas adicionales es calculado en el capítulo de Distribución de Cortantes.

El método dinámico que se emplea es uno de los métodos más utilizados en la práctica profesional, éste tipo de análisis modal está basado en técnicas de espectro de respuesta. Los Reglamentos Mexicanos describen éste método, y proponen que se analice un edificio en forma independiente para cada dirección ortogonal sin tomar en cuenta otros importantes efectos secundarios como el de torsión, aunque posteriormente al análisis modal sí se toma en cuenta dicho efecto. En el análisis sísmico de cada dirección indepen-

diente se recomienda que se superpongan los resultados de los análisis sísmico estático y análisis sísmico dinámico, dicha superposición es obtenida en el capítulo posterior.

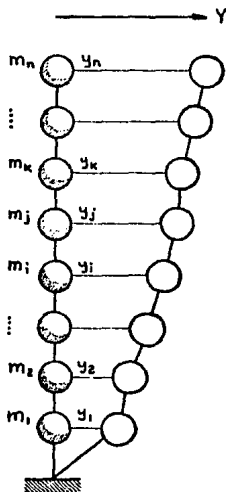
Para llevar al cabo el análisis de los efectos dinámicos de un sismo sobre una estructura se emplea el Análisis Dinámico. El método sísmico presentado al igual que en el estático toma en cuenta el modelo analítico planteado anteriormente (estructura sencillamente acoplada), en dicho modelo se idealizaba a un edificio simplificando sus grados de libertad por nodo a un solo grado, dicho grado restante sólo permite traslaciones horizontales en cada dirección analizada (X o Y), ésta representación transforma al edificio de un modelo continuo a otro modelo discreto, debido a que las masas de entrepiso son concentradas a la altura de las losas, quedando las masas ligadas entre sí mediante elementos con resistencia lateral, los cuales son representados por su rigidez y cuya masa es despreciable. El modelo analítico anteriormente utilizado queda representado de la siguiente forma :



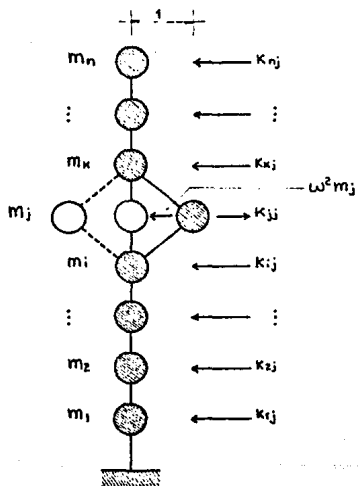
6.2 ANALISIS MODAL :

Si se somete al modelo analítico de la estructura a la acción externa de un impulso y suponiendo que *no existe amortiguamiento* en su respuesta, se tendrá como resultado que la estructura vibrará libremente.

Considerando un sistema de n masas $m_1, m_2, \dots, m_i, m_j, \dots, m_k, \dots, m_n$ enlazadas entre sí y cada una con libertad de desplazarse en una sola dirección horizontal "y", como se muestra en la siguiente figura:



La deformada del sistema en un instante cualquiera de su oscilación queda definida por las abscisas $y_1, y_2, \dots, y_i, y_j, \dots, y_k, \dots, y_n$. Se trata entonces de un sistema de n grados de libertad. Para estudiar la oscilación de éste sistema, se considera primero que todas las masas permanecen fijas, excepto la masa m_j la cual oscila con un desplazamiento máximo $y_j \text{ máx} = 1$:



Cuando m_j tiene el desplazamiento igual a 1, estará sujeta a una fuerza k_{jj} que causa ese desplazamiento y a una fuerza de inercia de valor igual a : masa x aceleración: $m_j \times \omega^2$ x desplazamiento.

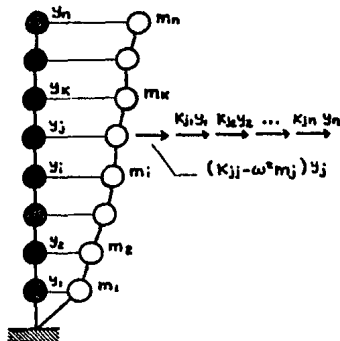
Estas dos fuerzas están en equilibrio:



Aparecen en las otras dos masas, reacciones que denominamos de la siguiente forma:

- k_{ij} Reacción que actúa en la masa i cuando la masa j tiene un desplazamiento unitario y las demás masas del sistema están fijas.
- k_{jj} Fuerza que produce un desplazamiento unitario en la masa j estando las demás masas del sistema fijas.

Consideremos ahora que soltamos las masas y que el sistema oscila libremente hasta alcanzar una configuración extrema definida por los desplazamientos: $y_1, y_2, \dots, y_p, y_p, \dots, y_k, \dots, y_n$



En las masas m_j aparecen las siguientes fuerzas:

$(k_{jj} - \omega^2 m_j) y_j$	debida al desplazamiento y_j
$k_{j1} y_1$	debida al desplazamiento y_1
$k_{j2} y_2$	debida al desplazamiento y_2
.....	
$k_{ji} y_i$	debida al desplazamiento y_i
.....	
$k_{jn} y_n$	debida al desplazamiento y_n

Estas fuerzas están en equilibrio, por lo que se puede escribir de la siguiente forma:

$$(k_{jj} - \omega^2 m_j) y_j + k_{j1} y_1 + k_{j2} y_2 + \dots + k_{ji} y_i + \dots + k_{jn} y_n = 0 \quad (6-1)$$

Ecuaciones similares pueden escribirse para las otras masas, por lo que reordenando se tiene:

$$\begin{aligned} (k_{11} - \omega^2 m_1) y_1 + k_{12} y_2 + \dots + k_{1j} y_j + \dots + k_{1n} y_n &= 0 \\ k_{21} y_1 + (k_{22} - \omega^2 m_2) y_2 + \dots + k_{2j} y_j + \dots + k_{2n} y_n &= 0 \\ \dots &\dots \\ k_{j1} y_1 + k_{j2} y_2 + \dots + (k_{jj} - \omega^2 m_j) y_j + \dots + k_{jn} y_n &= 0 \\ \dots &\dots \\ k_{n1} y_1 + k_{n2} y_2 + \dots + k_{nj} y_j + \dots + (k_{nn} - \omega^2 m_n) y_n &= 0 \end{aligned}$$

Estas ecuaciones se pueden escribir abreviadamente en forma matricial de la siguiente manera:

$$([K] - \omega^2 [M]) \{y_i\} = 0 \quad (6-2)$$

en donde:

$[K]$: matriz simétrica de rigideces.

$$[K] = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1j} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2j} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ k_{j1} & k_{j2} & \dots & k_{jj} & \dots & k_{jn} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nj} & \dots & k_{nn} \end{bmatrix} \quad (6-3)$$

Esta matriz es simétrica porque los elementos situados simétricamente respecto a la diagonal son iguales: $k_{ij} = k_{ji}$.

$$[M] = \begin{bmatrix} m_1 & 0 & \dots & 0 & \dots & 0 \\ 0 & m_2 & \dots & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & m_j & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 0 & m_n \end{bmatrix} \quad (6-4)$$

Una matriz diagonal es aquella matriz en la que todos los elementos son iguales a cero, excepto los ubicados en la diagonal, $i \neq j \rightarrow m_{ij} = 0$.

Vector de desplazamientos: $\{ y_i \}$.

$$\{ y_j \} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \\ \vdots \\ y_n \end{bmatrix} \quad (6-5)$$

Un vector o matriz columna es una matriz con una sola columna.

A la ecuación (6-2) se le llama: *Ecuación general del equilibrio dinámico* de un sistema de n grados de libertad.

Observando la ecuación matricial (6-2) observamos que para que el producto de los dos términos del lado izquierdo sea igual a cero es preciso que $\{ y_i \}$ sea igual a cero o bien que el determinante:

$$| [K] - \omega^2 [M] | = 0 \quad (6-6)$$

- La primera posibilidad $\{ y_i \} = 0$ nos conduce a una *solución trivial* en la que $y_1 = y_2 = \dots = y_i = \dots = y_n = 0$ ya que corresponde a la posición de equilibrio inicial antes de la oscilación.
- La segunda posibilidad es la siguiente:

$$| [K] - \omega^2 [M] | = 0 \quad (6-7)$$

La ecuación (6-7) en forma ampliada se puede escribir así:

$$\left[\begin{array}{cccc} k_{11} - \omega^2 m_1 & k_{12} & k_{1j} & k_{1n} \\ k_{21} & k_{22} - \omega^2 m_2 & k_{2j} & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{j1} & k_{j2} & k_{jj} - \omega^2 m_j & k_{jn} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & k_{nj} & k_{nn} - \omega^2 m_n \end{array} \right] \quad (6-8)$$

La ecuación (6-7) se denomina "ecuación de los valores característicos" de un sistema de n grados de libertad y es una ecuación de n grados en ω^2 , del tipo:

$$C + C_1 \omega^2 + C_2 (\omega^2)^2 + \dots + C_j (\omega^2)^j + \dots + C_n (\omega^2)^n = 0 \quad (6-9)$$

Esta ecuación (6-9) tiene raíces $\omega_1, \omega_2, \dots, \omega_j, \dots, \omega_n$ que son las frecuencias características del sistema.

Tomando una de éstas frecuencias, por ejemplo ω_j , y llevándola a la ecuación (6-2) se tiene:

$$([K] - \omega_j^2 [M]) \{y_j\} = 0 \quad (6-10)$$

De la ecuación (6-10) podemos despejar $\{y_j\}$, el vector de desplazamientos y_j asociados a la frecuencia ω_j :

$$\{y_j\}_i = \begin{bmatrix} Y_{1i} \\ Y_{2i} \\ \dots \\ Y_{ji} \\ \dots \\ Y_{ni} \end{bmatrix} \quad (6-11)$$

Sin embargo la ecuación (6-10) no es determinada y existen un sinnúmero de vectores $\{y_j\}$, que la resuelven, pero todos los vectores solución son proporcionales entre sí.

Para transformar a la ecuación (6-10) en una ecuación determinada, se dividen todos los elementos del vector entre el último, teniendo entonces:

$$\{\phi_j\}_i = \begin{bmatrix} \phi_{1j} \\ \phi_{2j} \\ \dots \\ \phi_{ji} \\ \dots \\ \phi_{ni} \end{bmatrix} = \begin{bmatrix} Y_{1i} / Y_{ni} \\ Y_{2i} / Y_{ni} \\ \dots \\ Y_{ji} / Y_{ni} \\ \dots \\ Y_{ni} / Y_{ni} \end{bmatrix} \quad (6-12)$$

Al anterior vector se le llama "*vector de desplazamientos normalizados asociado a una frecuencia i*", ó también se le llama *ley de la frecuencia i*. Se le llama *ley* porque no da los desplazamientos modales en unidades de longitud (en cm.) sino da la relación relativa que esos desplazamientos guardan entre sí.

Los criterios anteriores nos indican que la oscilación de un sistema de n grados de libertad se puede teóricamente descomponer en n oscilaciones simples, cada una con su frecuencia y su ley de desplazamientos. A cada una de éstas oscilaciones simples se les llama **modo**.

Al modo i corresponde la frecuencia ω_i , el período $T_i = 2\pi/\omega_i$ y la ley $\{\phi_i\}_i$. Multiplicando los desplazamientos normalizados por el llamado "*coeficiente de participación*" del modo i , C_i , se tienen los desplazamientos reales de ese modo.

$$\{y\}_i = C_i \{\phi\}_i \quad (6-13)$$

La expresión total del sistema, en un instante dado queda dada por la suma de las respuestas de cada modo, que en ese instante no necesariamente es la máxima (los desplazamientos no son los máximos, no son simultáneos porque las frecuencias son diferentes).

Como la respuesta total depende preponderantemente del primer modo, a este modo se le llama "*modo fundamental*" y a su período "*período fundamental*", éste modo tiene como característica que no presenta ningún punto de inflexión.

Al aumentar los modos de vibrar, el período decrece y el número de puntos de inflexión aumenta consecutivamente.

El análisis modal mencionado nos proporciona un conjunto de valores, los cuales denominamos como *valores característicos* del modelo analizado, estos valores quedan representados por:

ω_i	Frecuencia circular del modo i .
ϕ_{ij}	Ley normalizada del modo i .
T_i	Período del modo i .

Los anteriores valores característicos van a ser calculados mediante el método de Hozler descrito a continuación.

6.2.1 Método de Hozler :

El método de Hozler es un proceso numérico iterativo con el cual se realizará el análisis modal, es de los métodos mas empleados en la obtención de las características de los modos superiores de estructuras sencillamente acopladas.

Este método consiste en resolver las ecuaciones características tomando como punto de partida una frecuencia circular congruente con las condiciones de soporte en los extremos de los sistemas constituidos por masas y rigideces discretas colocadas en serie, procediendo sucesivamente hacia el otro extremo. Si los resultados obtenidos, concuerdan con las condiciones de soporte de ese extremo, el valor supuesto inicialmente es correcto al igual que la frecuencia circular correspondiente, de lo contrario, se repiten los ciclos con un valor de prueba diferente, hasta que se satisfagan las condiciones de soporte en el extremo final. Una vez obtenida la frecuencia verdadera, se sustituye dentro de las ecuaciones características, con el objeto de encontrar su respectiva forma modal.

Al método de Hozler se le ha llegado a considerar como un método aproximado, lo que significaría que tiene un cierto margen de error, en comparación con algunos otros métodos de análisis modal considerados como "exactos". Esta aseveración es errónea, debido a que el método de Hozler puede generar resultados tan exactos como los métodos así considerados, la precisión de los resultados de éste método están en función del número de iteraciones que se emplee para su obtención, considerando un número grande de ciclos se puede llegar a una solución muy aceptable. De acuerdo con éstas características, el proceso también es conocido como un "método iterativo".

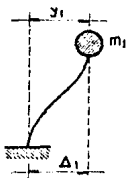
Otra ventaja que representa el método de Hozler es la siguiente, a diferencia de otros métodos de análisis modal en el método de Hozler sólo es necesario determinar las respuestas estructurales debidas a unas cuantas de las primeras formas modales, debido a que en general, parte de la respuesta total de los edificios que provocan los modos superiores es muy pequeña.

METODOLOGIA:

El método de Hozler permite calcular cualquier modo natural y su ley, la metodología en que se basa es la siguiente:

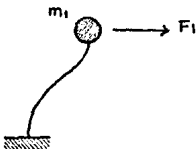
1. Supongase un valor de ω^2 .

2. Supongase un desplazamiento total, y_1 , para la primera masa. Este desplazamiento será igual al desplazamiento relativo del primer entrepiso $y_1 = \Delta_1$. Por comodidad la suposición puede ser $y_1 = \Delta_1 = 1$.



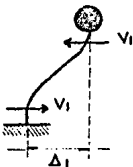
3. Calcúlese la fuerza de inercia en la primera masa:

$$\begin{aligned}
 F_1 &= m_1 \ddot{y}_1 && (6-14) \\
 &= m_1 \omega^2 y_1 && (6-15) \\
 &= m_1 \omega^2 && (6-16)
 \end{aligned}
 \quad , (y_1 = 1)$$



4. Calcúlese la fuerza cortante de restitución en el primer entrepiso:

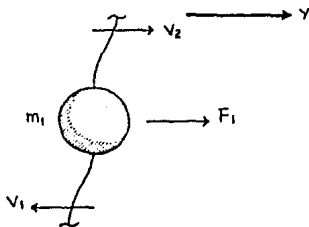
$$\begin{aligned}
 V_1 &= \Delta_1 k_1 && (6-17) \\
 &= k_1 && (6-18)
 \end{aligned}
 \quad , (\Delta_1 = 1)$$



5. Por equilibrio de fuerzas en la primera masa, obténgase el cortante en el segundo entrepiso:

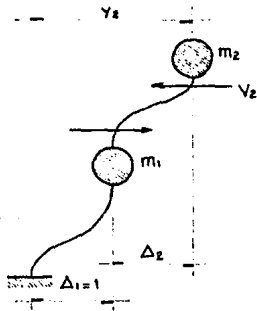
$$\Sigma F_y = F_1 - V_1 - V_2 = 0 \quad (6-19)$$

$$V_2 = F_1 - V_1 \quad (6-20)$$



6. Apartir de V_2 , obténgase el desplazamiento relativo Δ_2 del segundo entrepiso:

$$\Delta_2 = V_2 / k_2 \quad (6-21)$$



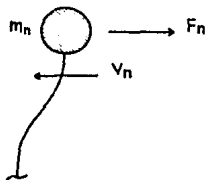
7. El desplazamiento total y_2 será:

$$y_2 = \Delta_1 + \Delta_2 = 1 + \Delta_2 \quad (6-22)$$

8. La fuerza de inercia F_2 en la masa m_2 será:

$$F_2 = m_2 \ddot{y}_2 = m_2 \omega^2 y_2 \quad (6-23)$$

9. Por equilibrio de fuerzas, en la segunda masa se obtiene el cortante del tercer entrepiso, de éste cortante se obtiene el desplazamiento total de la tercera masa, etc, ...
10. Se continúa con el mismo procedimiento hasta llegar a la masa m_n de la siguiente forma:



En ésta masa m_n el cortante V_n y la fuerza de inercia F_n deberían estar en equilibrio. Pero como el valor supuesto de la frecuencia ω^2 no corresponde por lo general a una frecuencia característica, esa condición de equilibrio no se cumple.

$$V_n + F_n = R \quad (6-24)$$

$$R \neq 0 \quad (6-25)$$

Donde R es un residuo.

Se repite todo el procedimiento ensayando otros valores de ω^2 hasta lograr que el residuo sea aproximadamente igual a cero ($R = 0$). Cuando esto se logra, la ω^2 propuesta corresponde a una frecuencia característica, los valores de $y_1, y_2, \dots, y_1, \dots, y_n$ corresponden a la ley del modo. El número de nodos de esa ley nos indica a qué modo corresponde esa frecuencia y a que ley calculada.

6.2.1.1 Subprograma del Método de Hozler :

El proceso esencial del método de Hozler se simplificó y optimizó lo más posible, debido a que este método es requerido en bastantes ocasiones para encontrar los valores característicos, de hecho es la parte que consume más tiempo de ejecución en éste subprograma. Además de que es utilizado tanto por el análisis modal como por el subprograma auxiliar de bisección.

Variables del subprograma:

Variables de entrada (INPUT):

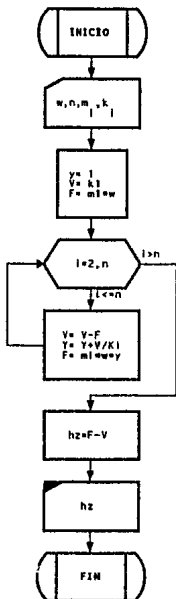
w	Frecuencia circular w^2 .
n	Número de pisos de la estructura o número de modos del modelo analítico.
mi	Vector columna que contiene masas concentradas en los pisos I.
ki	Vector columna de rigideces de entrepiso.

Variables de salida (OUTPUT):

hz	Residuo del método de hozler (R).
----	-----------------------------------

6.2.1.2 Diagrama de flujo :

METODO DE HOZLER



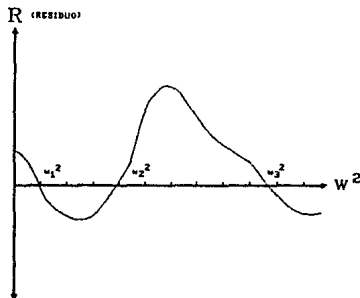
6.2.1.3 Programa fuente en Lenguaje C :

```
/* INICIO DE SUBPROGRAMA DE HOZLER */  
  
double hz(w,n,m,k)  
double w,*m,*k;  
int n;  
  
/*     VARIABLES DE ENTRADA:  
  
    n      :   número de pisos de la estructura  
    w      :   frecuencia angular  $\omega$   
    m[n]   :   vector de masas por cada piso  
    k[n]   :   vector de rigideces de entrepiso  
  
    VARIABLES DE SALIDA:  
  
    hz()   :   residuo del metodo de Hozler  
*/  
  
{  
    double y=1,V,F;  
    int i;  
  
    V=k[1]*F+m[1]*w;  
    for(i=2;i<=n;i++){  
        V=F;  
        y=V/k[i];  
        F =m[i]*w*y;  
    }  
    return F-V;  
}  
  
/* FIN DE SUBPROGRAMA DE HOZLER */
```

6.2.2 Subprograma de Análisis Modal :

Los fundamentos para la construcción de éste subprograma son presentados a continuación, en dichos fundamentos se describe de una forma general que es lo que se realiza en el subprograma aquí mencionado.

Utilizando en el Análisis Modal el método de Hozler se hace evidentemente que en éste método se requiere estar proponiendo distintos valores para ω^2 que cumplan con la condición de que su residuo R sea aproximadamente igual a cero (equilibrio dinámico). Una forma de plantear la proposición de ω^2 es la de graficar el residuo R en función de ω^2 . De ésta forma se va teniendo una mejor idea de como se comporta ésta relación (ω^2 - R) y se llega a acelerar el proceso del método mencionado.



Para efectos del presente trabajo se utilizó el planteamiento gráfico para encontrar las distintas frecuencias circulares del modelo analítico. Esto no quiere decir que se grafique físicamente ésta relación para cada edificio analizado, lo que quiere decir es que el subprograma de análisis modal inicia proponiendo valores de ω^2 pero de una forma progresiva e iniciando desde cero como si se tratara de realizar una tabulación para posteriormente graficar.

Una vez iniciada la iteración del progresivo aumento de la frecuencia ω^2 , el subprograma detecta si hay un cambio de signo en el residuo R , lo que significaría que se ha encontrado un intervalo en donde se debe de encontrar una frecuencia ω^2 cuyo residuo sea aproximadamente igual a cero.

Quando se detecta el intervalo de cambio de signo en el residuo, se procede a encontrar la ω^2 más próxima a cero utilizando otro subprograma de métodos numéricos llamado método de aproximaciones sucesivas de bisección.

Este método auxiliar de bisección acelera enormemente la convergencia del método de Hozler. El método de bisección aquí utilizado es un método modificado de bisección, debido a que combina una regresión lineal con el método para acelerar aún más la condición de $R = 0$.

Después de encontrar una frecuencia ω^2 que tenga un residuo R lo suficientemente aproximado a cero, se realiza una verificación, basada en las características siguientes:

Debido a que las frecuencias circulares presentan la siguiente particularidad:

$$(\omega_1)^2 < (\omega_2)^2 < \dots < (\omega_n)^2 < \dots < (\omega_n)^2 \quad (6-26)$$

Se realiza por tanto una comprobación, para que cada frecuencia circular $(\omega_i)^2$ tenga su correspondiente modo i . Esto significa que la primera frecuencia encontrada debe corresponder al primer modo natural, la segunda frecuencia al segundo modo, etc..

La anterior verificación es realizada debido a que por las características propias del método de análisis modal propuesto, se puede dar el caso en que se llegara a omitir alguna frecuencia circular al iterar progresivamente ω^2 , si esto sucediera entonces el subprograma se regresaría a la anterior frecuencia y reduciría el intervalo del incremento de la frecuencia para tratar de buscar nuevamente a la frecuencia que no haya sido calculada.

La terminación del subprograma de análisis modal se puede llevar al cabo cuando se cumplen las siguientes consideraciones:

- Cuando se hayan encontrado todas las frecuencias de los modos correspondientes al modelo analítico de la estructura.
- Cuando sean encontrados aquellos modos naturales que tengan un período mayor o igual a 0.4 seg., pero en ningún caso se consideran menos de los tres primeros modos de traslación, por cada dirección analizada.

Para fines del desarrollo del subprograma de Análisis Modal, podemos decir que la obtención de los valores característicos está en función de los siguientes subprogramas:

- Análisis Modal.
- Método de Hozler.
- Método de Bisección con Interpolación Lineal.

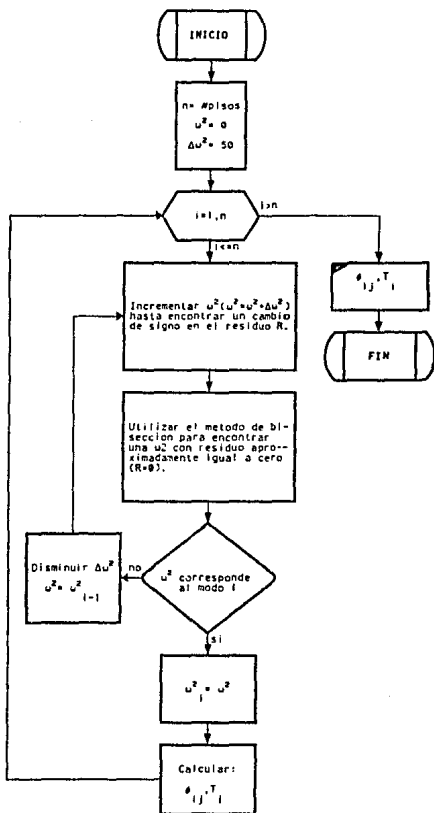
6.2.2.1 Diagramas de flujo :

Cabe aclarar que los diagramas de flujo aquí presentados están escritos en un lenguaje de pseudo-código, lo que significa que formalmente son diagramas de flujo que representan procesos generales y en algunas partes son descritos textualmente dichos procesos. Para comprender en detalle como trabajan éstos algoritmos se recomienda nuevamente estudiar el programa fuente de éste subprograma.

El programa fuente del Análisis Modal y del método de Bisección están incorporados al programa fuente del subprograma general del análisis sísmico dinámico, son incluidos al final del presente capítulo.

A continuación son presentados sus diagramas de flujo de cada uno de los anteriores subprogramas, pero hay que recordar que éstos a su vez son herramientas auxiliares y básicas del objetivo principal de éste capítulo: encontrar las fuerzas sísmicas mediante el análisis sísmico dinámico.

OBTENCION DE VALORES CARACTERISTICOS



6.3 ANALISIS SISMICO DINAMICO :

Para obtener las fuerzas sísmicas mediante un análisis sísmico dinámico se siguen los siguientes aspectos:

- Cálculo de valores característicos (Utilizando análisis modal).
- Selección del espectro de diseño.
- Cálculo de las aceleraciones máximas elasto-plásticas para el cálculo de las fuerzas.
- Cálculo de los factores de participación.
- Desplazamientos máximos de los pisos en cada modo i .
- Desplazamiento relativos de los entrepisos en cada modo.
- Cortantes de entrepiso para cada modo.
- Cortantes totales de entrepiso.
- Fuerzas en cada piso, por sismo.
- Desplazamientos relativos.
- Desplazamientos totales relativos.

Nuevamente, para el desarrollo del subprograma de Análisis Sísmico Dinámico se dividió el problema en los once aspectos anteriores, los cuales son tratados cada uno de ellos en forma separada de tal forma de poder desarrollar un subprograma específico e independiente de los demás.

6.3.1 Cálculo de valores característicos :

Los valores característicos son calculados en la sección de análisis modal vista anteriormente, en ese tema se desarrolló el subprograma que define a dichos valores.

6.3.2 Selección del espectro de diseño :

Este aspecto es tratado ampliamente en el capítulo V (Análisis sísmico estático) debido a que el espectro de diseño es requerido en los dos tipos de análisis (estático y dinámico), en dicha sección es desarrollado su subprograma correspondiente.

6.3.3 Cálculo de las aceleraciones máximas elasto-plásticas para el cálculo de las fuerzas:

La explicación del cálculo de las fuerzas elasto-plásticas está descrito en el capítulo anterior, en el tema correspondiente a la Reducción de fuerzas sísmicas en función del factor de ductilidad, aquí es nuevamente presentado pero con la diferencia de que además de obtener el factor de ductilidad reducido también se obtiene las aceleraciones sísmicas por cada modo.

6.3.4 Cálculo de los factores de participación :

Una vez encontrados los distintos modos de vibrar que tiene una estructura, surge el problema de calcular la forma en que se pueden llegar a combinar éstos para conocer la respuesta probable de la estructura ante una acción sísmica específica.

Al someter a un edificio a una excitación externa, cada modo responderá como un sistema independiente de un grado de libertad. La respuesta total del edificio será la combinación de las respuestas independientes de cada uno de éstos modos, multiplicadas por un factor adecuado. A dicho factor se le conoce como **factor de participación**.

La forma en que son calculados los factores de participación es la siguiente:

$$C_j = \frac{\sum_{i=1}^n W_j \phi_i}{\sum_{i=1}^n W_j \phi_i^2} \quad (6-27)$$

En donde:

n	Número de pisos de la estructura.
W_i	Pesos de todos los piso de la estructura.
ϕ_{ij}	Ley normaliza para cada modo.
C_j	Factor de participación en cada modo j .

Una interpretación que se le puede dar al *factor de participación* es la siguiente: define la magnitud relativa en la que interviene el modo " j " en el movimiento dinámico de una estructura en general.

6.3.5 Desplazamiento máximos de los pisos en cada modo l :

El desplazamiento máximo lo define la siguiente expresión:

$$y_{ij \text{ máx}} = \frac{A_j}{\omega_j^2} C_j \phi_{ij} \quad (6-28)$$

En donde:

ω_j^2	Frecuencia circular en cada modo.
$y_{ij \text{ máx}}$	Desplazamiento máximo.
A_j	Aceleración máxima por cada modo.
C_j	Factor de participación por cada modo.
ϕ_{ij}	Ley normalizada por cada modo.

6.3.6 Desplazamientos relativos de los entrepisos en cada modo:

El desplazamiento relativo queda definido por la expresión:

$$\Delta_{ij} = y_{ij} - y_{(i-1)j} \quad (6-29)$$

En donde:

Δ_{ij}	Desplazamiento relativo por entrepiso en cada modo.
y_{ij}	Desplazamiento máximo por piso en cada modo.

6.3.7 Cortantes de entrepiso para cada modo :

$$V_{ij} = k_i \Delta_{ij} \quad (6-30)$$

En donde:

V_{ij}	Cortantes de entrepiso por cada modo.
K_i	Rigideces de entrepiso.
Δ_{ij}	Desplazamiento relativo por entrepiso en cada modo.

6.3.8 Cortantes totales de entrepiso :

Basándose en estudios probabilísticos se ha demostrado que la respuesta total de un sistema para un instante dado, es representada por la raíz de la suma de los cuadrados de las respuestas estructurales de cada modo, esto se debe a que las respuestas (cortantes o desplazamientos para nuestro caso) de todos los modos no son simultáneas.

Utilizando el anterior criterio, los cortantes totales de entrepiso son calculados con la siguiente expresión:

$$V_i = \sqrt{\sum_{j=1}^n V_{ij}^2} \quad (6-31)$$

En donde:

V_{ij}	Cortantes de entrepiso por cada modo.
V_i	Cortantes totales por entrepiso.

6.3.9 Fuerzas en cada piso, por sismo :

Las fuerzas en cada piso son definidas por la ecuación:

$$F_i = V_i - V_{i+1} \quad (6-32)$$

En donde:

V_i	Cortantes totales por entrepiso.
F_i	Fuerzas por piso.

6.3.10 Desplazamientos relativos :

En el cálculo de los desplazamientos máximos relativos del sistema, se utiliza el mismo criterio empleado en la obtención de los cortantes totales de entrepiso (6.3.8).

Adicionalmente se debe tener en cuenta que la reducción por ductilidad es válida para el cálculo de fuerzas pero no para el de desplazamientos, por lo que debe considerarse que los desplazamientos elasto-plásticos son iguales a los elásticos. De aquí que se multiplica en la siguiente relación los desplazamientos relativos máximos modales por el factor de ductilidad reducido para cada modo.

$$\Delta_i = \sqrt{\sum_{j=1}^n (Q_j \Delta_j)^2} \quad (6-33)$$

En donde:

Δ_i	Desplazamiento relativo por entrepiso.
Q_j	Factor de ductilidad reducido por cada modo.
Δ_j	Desplazamiento relativo por entrepiso en cada modo.

6.3.11 Desplazamientos totales relativos :

$$y_i = y_{i-1} + \Delta_i \quad (6-34)$$

En donde:

y_i	Desplazamiento total relativo por piso.
Δ_i	Desplazamiento relativo por entrepiso.

6.3.12 Diagramas de flujo :

Los siguientes diagramas de flujo representan la metodología necesaria para la obtención de las fuerzas sísmicas dinámicas, son tratadas aisladamente para simplificar su análisis, comparten sus datos de entrada y salida (ver programa fuente).

La representación que tienen estos diagramas de flujo es de codificación directa, lo que significa que se pueden seguir los distintos procesos empleados y a su vez programarlos en algún lenguaje de cómputo. Esto se verifica si se compara el código fuente contra los diagramas de flujo.

Las variables que a continuación se describen son utilizadas por los diagramas de flujo (6.3.3 - 6.3.11) y por el programa fuente. Las variables de entrada y salida se refieren al subprograma en general, no se refieren a las variables de entrada y salida de cada diagrama de flujo en particular.

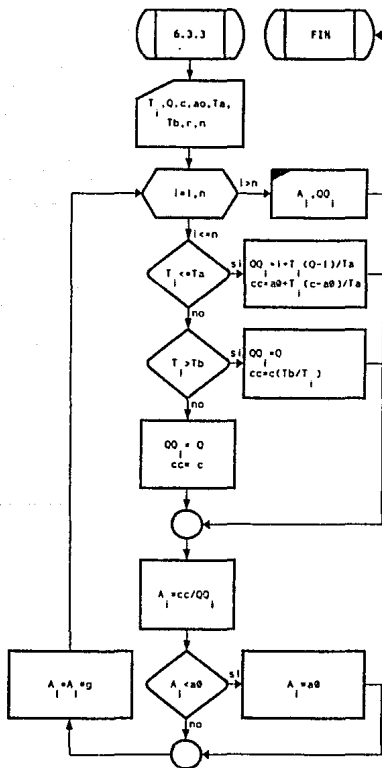
Variables de Entrada (INPUT):

n	Número de pisos de la estructura.
Q	Factor de ductilidad en función del tipo de estructura y de los requisitos de diseño por cada dirección analizada.
W_i	Peso de los entrepisos concentrados en cada uno de los n pisos.
k_i	Rigideces totales del entrepiso i, en la dirección del análisis.
ao	Aceleración máxima del terreno, como fracción de g.
c	Aceleración máxima maximorum de las estructuras, como fracción de g.
Ta	Período que limita el límite inferior del intervalo de resonancia.
Tb	Período que limita el límite superior del intervalo de resonancia.
r	Exponente adimensional.

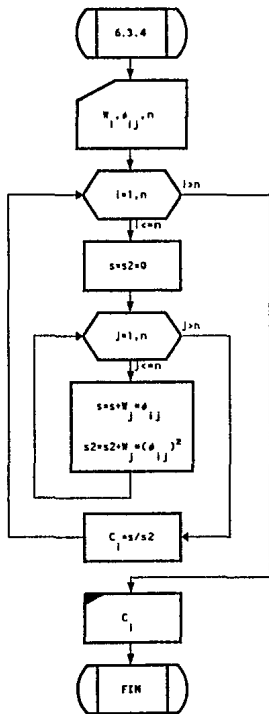
Variables de Salida (OUTPUT):

C_i	Factores de participación para los n modos de vibrar de la estructura.
A_i	Aceleración máxima elasto-plástica para los n modos.
QQ _i	Factores de ductilidad reducido por cada modo.
VT _i	Cortantes totales de entrepiso.
F _i	Fuerzas en cada piso, por sismo.
D _i	Desplazamientos relativos por entrepiso.
DT _i	Desplazamientos totales relativos por entrepiso.
w2 _i	Frecuencia característica del modo n.
T _i	Períodos de vibración de cada modo.
ym _{ij}	Desplazamientos máximos de los entrepisos en cada modo.
Dr _{ij}	Desplazamientos relativos de los entrepisos en cada modo.
V _{ij}	Cortantes de entrepiso para cada modo.
ϕ_{ij}	LEY normalizada de los n modos.

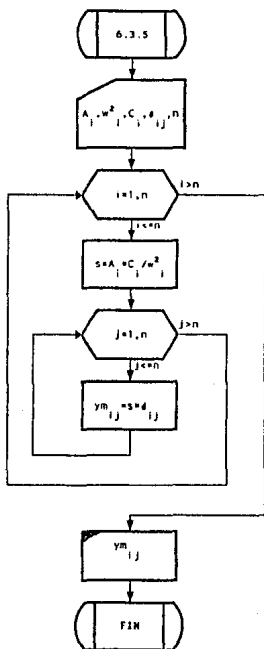
CALCULO DE ACELERACIONES MAXIMAS ELASTOPLASTICAS PARA EL CALCULO DE FUERZAS



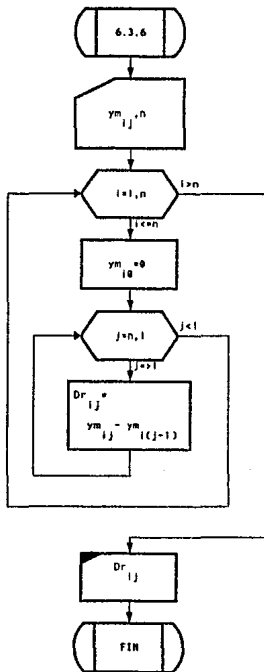
CALCULO DE LOS FACTORES DE PARTICIPACION



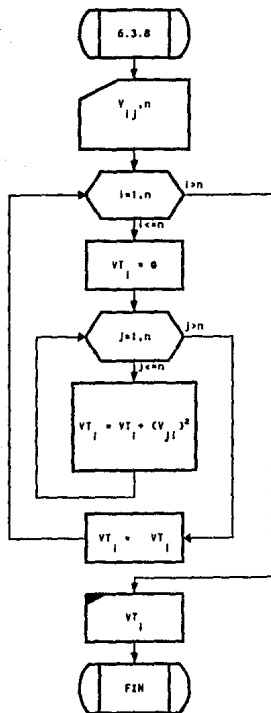
DESPLAZAMIENTOS MAXIMOS DE LOS PISOS EN CADA MODO



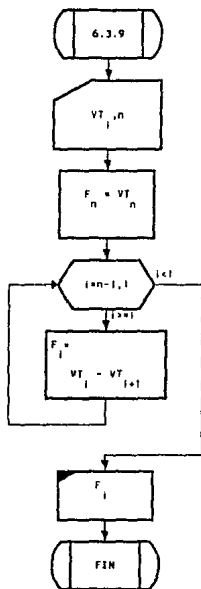
DESPLAZAMIENTOS RELATIVOS DE LOS ENTREPISOS EN CADA MODO



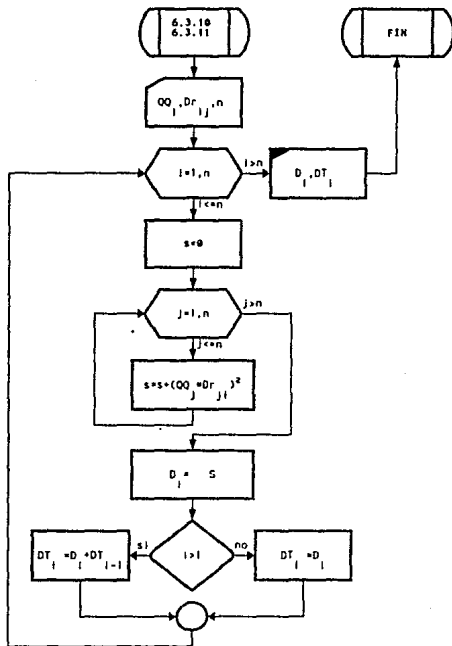
CORTANTES TOTALES DE ENTREPISO



FUERZAS EN CALA PISO. POR SISMO



DESPLAZAMIENTOS RELATIVOS Y DESPLAZAMIENTOS TOTALES RELATIVOS



6.3.13 Programa fuente en Lenguaje C :

En el programa fuente del subprograma de análisis sísmico dinámico aquí presentado se integran todos los subprogramas anteriores en un solo programa fuente, dicho programa esta basado en los diagramas de flujo anteriores y además es tratado como un solo subprograma general.

```
/*
*****
***** UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO *****
***** ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES ARAGON *****
*****

TESIS      :      [ ANESAR ] PROGRAMA PARA EL ANALISIS SISMICO DE EDIFICIOS .

CAPITULO   :      <6> ANALISIS SISMICO DINAMICO .

Autor      :      Francisco Reyes Téllez .
No.Cta.    :      8428671-6 .
Carrera    :      INGENIERIA CIVIL <87-91> .

Fecha      :      20 de septiembre de 1991 .
Asesor     :      Ing. Amílcar Galindo Solorzano .

*/

#include <conio.h>
#include <stdio.h>
#include <alloc.h>
#include <math.h>

#define g 981
#define pl 3.141592654

void an_dina(n,q,w,k,es)
    int n;
    double q,*w,*k,*es; /* w[1-n],k[1-n],es[0-4] */
{
    int i,j;
    int verifica();

    double *f,*v,*ym,*Dr,*t,*w2,*m,*QD,*A,*C,*VT,*F,*D,*DT;
    double hz(),bisec();
    double s,s2,cc;

/* VARIABLES UTILIZADAS EN SUBPROGRAMA

n :      Número de pisos de la estructura .
q :      Factor de ductilidad en función del tipo de estructura y de los requisitos de diseño por cada
         dirección analizada.

W[n] :   Peso de los entrepisos concentrados en cada uno de los n pisos .
k[n] :   Rigideces totales del entrepiso i, en la dirección del análisis .
es[0] :  Aceleración máxima del terreno, como fracción de g .
es[1] :  Aceleración máxima maximumum de las estructuras, como fracción de g .
es[2] :  Periodo que limita el límite inferior del intervalo de resonancia .
es[3] :  Periodo que limita el límite superior del intervalo de resonancia .
es[4] :  Exponente adimensional .
```

C [n] : Factores de participación .
 A [n] : Aceleración máxima elasto-plástica .
 QQ [n] : Factores de ductilidad para los n períodos .
 VT [n] : Cortantes totales de entrepiso .
 F [n] : Fuerzas en cada piso, por sismo .
 D [n] : Desplazamientos relativos .
 DT [n] : Desplazamientos totales relativos .
 w2 [n] : Frecuencia característica del modo n .
 T [n] : Períodos de vibración de cada modo .
 W [n] : Peso total del piso l .
 ym [n] [n] : Desplazamientos máximos de los entrepisos en cada modo .
 Dr [n] [n] : Desplazamientos relativos de los entrepisos en cada modo .
 V [n] [n] : Cortantes de entrepiso para cada modo .
 f [n] [n] : Ley normalizada de los n modos .

NOTA: Todos los demás variables que no están indicadas en esta lista son variables auxiliares, que no tienen significado en el Análisis Estructural.

*/

```

V=malloc(n*(n+2)+1,8);ym=malloc(n*(n+2)+1,8);
Dr=malloc(n*(n+2)+1,8);f=malloc(n*(n+2)+1,8);

```

```

QQ=malloc(n+1,8);A=malloc(n+1,8);
C=malloc(n+1,8);VT=malloc(n+1,8);
F=malloc(n+1,8);D=malloc(n+1,8);
DT=malloc(n+1,8);m=malloc(n+1,8);
T=malloc(n+1,8);w2=malloc(n+1,8);

```

```
for(i=1;i<=n;i++) m[i]=W[i]/g;
```

/* 1) Cálculo de valores característicos (Análisis Modal) */

```
( double dw2,dwa2,dwb2,Ra,Rb,delta,V,F; int dd,ee;
```

```

ee=1;
delta=50;
dw2=delta;
w2[0]=dwa2=0.1;
Ra=h2(dwa2,n,m,k);
do {
  dwb2=dwa2+dw2;
  Rb=h2(dwb2,n,m,k);
  if(Ra>Rb<0) {
    w2[ee]=bisecc(dwa2,dwb2,n,m,k);
    if(verifica(w2[ee],ee,n,m,k)) {
      dw2=delta;
      dd=n*ee+ee;
      T[ee]=2*pi/sqrt(w2[ee]);
      f[dd+1]=1;
      V=k[1];
      Fm[1]=w2[ee];
      for(i=2;i<=n;i++) {
        V=F;
        f[dd+i]=V/k[i]+f[dd+i-1];
        Fm[i]=w2[ee]*f[dd+i];
      }
      for(i=1;i<=n;i++) f[dd+i]/=f[dd+n];
      dwb2=w2[ee+1]+dw2;
    } else {
      dw2/=2;dwb2=w2[ee-1]+dw2;
    }
    Rb=h2(dwb2,n,m,k);
  }
  dwa2=dwb2;Ra=Rb;
} while (n>=ee);
)

```

```
/* 3) Cálculo de las aceleraciones máximas elasto-plásticas para el cálculo de las fuerzas. */
```

```
for(i=1;i<=n;i++) {  
    if ( T(i) <= es[2] ) {  
        QQ[i]=1+T(i)/es[2]*(Q-1);  
        cc=es[0]+(es[1]-es[0])/es[2]*T(i);  
    }  
    if ( T(i) > es[2] && T(i) <= es[3] ) { QQ[i]=Q;cc=es[1]; }  
    if ( T(i) > es[3] ) {  
        QQ[i]=Q;  
        cc=es[1]*pow(es[3]/T(i),es[4]);  
    }  
    A[i]=cc/QQ[i]; /* Aceleración como fracción de g */  
    if ( A(i) < es[0] ) A[i]=es[0];  
    A[i]*=g; /* Aceleración en el espectro elasto-plástico */  
}
```

```
/* 4) Cálculo de los factores de participación. */
```

```
for(i=1;i<=n;i++) {  
    s=s2+0;  
    for(j=1;j<=n;j++) {  
        s +=W[j]*f[1*n+i+j];  
        s2+=W[j]*pow(f[1*n+i+j],2);  
    }  
    C[i]=s/s2;  
}
```

```
/* 5) Desplazamientos máximos de los pisos en cada modo. */
```

```
for(i=1;i<=n;i++) {  
    s=A[i]*u2[i]*C[i];  
    for(j=1;j<=n;j++) ym[(n+1)*i+j]=s*f[1*n+i+j];  
}
```

```
/* 6) Desplazamientos relativos de entrepisos en cada modo. */
```

```
for(i=1;i<=n;i++) {  
    ym[(n+1)*i+0]=0;  
    for(j=n;j>=1;j--)  
        Dr[(n+1)*i+j]=ym[(n+1)*i+j]-ym[(n+1)*i+j-1];  
}
```

```
/* 7) Cortantes de entrepiso para cada modo. */
```

```
for(i=1;i<=n;i++)  
    for(j=1;j<=n;j++) V[(n+1)*i+j]=k[j]*Dr[(n+1)*i+j];
```

```

/* 8) Cortantes totales de entrepiso. */
for(i=1;i<=n;i++) {
    VT[i]=0;
    for(j=1;j<=n;j++) VT[i]+=pow(V[(n+1)*j+1],2);
    VT[i]=sqrt(VT[i]);
}

/* 9) Fuerzas en cada piso, por alamo. */
F[n]=VT[n];
for(i=n-1;i>=1;i--) F[i]=VT[i]-VT[i+1];

/* 10) Desplazamientos relativos y totales relativos. */
for(i=1;i<=n;i++) {
    s=0;
    for(j=1;j<=n;j++) s+=pow(OQ[j]*D[(n+1)*j+1],2);
    D[i]=sqrt(s);
    if(i>1) DT[i]=D[i]+DT[i-1]; else DT[i]=D[i];
}
}

```

```

int verifica(w,m,n,m,k)
double w,*m,*k;
int mm,n;
{
    int i,rm=1;
    double y,ya=1,V,F;
    V=k[1];F=m[1]*w;
    for(i=2;i<=n;i++) {
        V=wF;
        y =V/k[i]+ya;
        F =m[i]*w*y;
        if(y*ya<0) rm++;
        ya=y;
    } return (rm==mm);
}
}

```

```

double bisec(a,b,n,m,k)
    double a,b,*m,*k;
    int n;
(
    double c,fa,fb,fc,hz();
    fa=hZ(a,n,m,k);
    fb=hZ(b,n,m,k);
    do(
        c=(a+b)/2;fc=hZ(c,n,m,k);
        if(fa*fc<0) (b=c;fb=fc;) else (a=c;fa=fc;)
        c=fa*(b-a)/(fa-fb)+a;fc=hZ(c,n,m,k);
        if(fa*fc<0) (b=c;fb=fc;) else (a=c;fa=fc;)
    ) while( (fc>0?fc:-fc) > 0.000001 );
    return fa*(b-a)/(fa-fb)+a;
)

```

```

double hz(w,n,m,k)
    double w,*m,*k;
    int n;
(
    double y=1,V,F;
    int i;
    V=k[1];F=m[1]*w;
    for(i=2;i<=n;i++) (
        V=V-F;
        y=y+V/k[i];
        F =m[i]*w*y;
    )
    return F-V;
)

```

```

/* FIN DE SUBPROGRAMA DE ANALISIS SISMICO DINAMICO */

```

6.4 EJEMPLO :

Empleando el análisis sísmico dinámico calcular los cortantes de entrepiso y los desplazamientos totales de cada piso en la dirección X del edificio que a continuación se describe:

El edificio a analizar es un hospital ubicado en la Deleg. Gustavo A. Madero, D.F. La estructuración del edificio la conforman marcos dúctiles de concreto reforzado.

Hospital: Grupo A, factor de importancia 1.5 para zona Metropolitana.

Ubicación: Deleg. Gustavo A. Madero, zona sísmica especial para la Cd. de México y áreas conurbadas.

Terreno: Terreno blando, tipo III.

Estructuración: Edificio estructurado con marcos dúctiles de concreto reforzado en ambos sentidos de sus ejes principales, $Q_x = Q_y = 4$.

Para ejemplificar el análisis sísmico dinámico únicamente se analizará la dirección X, el análisis sísmico en la dirección Y se realiza de una forma similar.

Espectro de diseño:

De acuerdo a las características del edificio su espectro de diseño correspondiente es el siguiente:

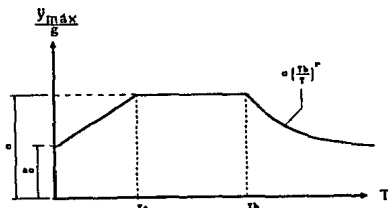
$$a_0 = 0.10 \times 1.5 = 0.15 \text{ g}$$

$$c = 0.40 \times 1.5 = 0.60 \text{ g}$$

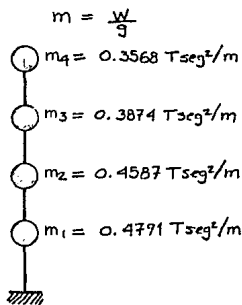
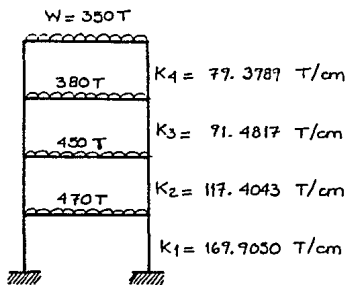
$$T_a = 0.60 \text{ seg.}$$

$$T_b = 3.90 \text{ seg.}$$

$$r = 1.00 \square$$



1) CALCULO DE VALORES CARACTERISTICOS :



Empleando el método de Hozler se obtienen los siguientes valores característicos :

PRIMER MODO :

$$\begin{aligned}
 \omega_1^2 &= 38.64 \text{ (rad/seg)}^2 \\
 T_1 &= 1.0108 \text{ seg} \\
 \{\phi_j\}_1 &= \left\{ \begin{array}{l} 0.2360 \\ 0.5404 \\ 0.8263 \\ 1.0 \end{array} \right\}
 \end{aligned}$$

SEGUNDO MODO :

$$\begin{aligned}
 \omega_2^2 &= 250.2498 \text{ (rad/seg)}^2 \\
 T_2 &= 0.3972 \text{ seg} \\
 \{\phi_j\}_2 &= \left\{ \begin{array}{l} -0.6793 \\ -0.9686 \\ -0.1246 \\ 1.0 \end{array} \right\}
 \end{aligned}$$

TERCER MODO :

$$\begin{aligned}
 \omega_3^2 &= 581.4479 \text{ (rad/seg)}^2 \\
 T_3 &= 0.2606 \text{ seg}
 \end{aligned}$$

$$\{\phi_i\}_3 = \begin{Bmatrix} 1.2117 \\ 0.0101 \\ -1.6120 \\ 1.0 \end{Bmatrix}$$

CUARTO MODO:

$$\begin{aligned} \omega_4^2 &= 848.3983 \text{ (rad/seg)}^2 \\ T_4 &= 0.2157 \text{ seg.} \\ \{\phi_i\}_4 &= \begin{Bmatrix} -3.7231 \\ 3.7816 \\ -2.8117 \\ 1.0 \end{Bmatrix} \end{aligned}$$

2) SELECCION DEL ESPECTRO DE DISEÑO:

En el capítulo V se definió el espectro de diseño de la siguiente forma:

$$\begin{aligned} a_0 &= 0.15 \text{ g} \\ c &= 0.60 \text{ g} \\ T_a &= 0.60 \text{ seg.} \\ T_b &= 3.70 \text{ seg.} \\ r &= 1.00 \end{aligned}$$

3) ACELERACIONES PARA EL CALCULO DE FUERZAS:

PRIMER MODO:

$$T_1 = 1.0108 \text{ seg.}$$

$T_a < T_1 < T_b \Rightarrow$ cae en rama horizontal de espectro, no hay reducción por ductilidad.

$$\ddot{y}_{máx e} = c = 0.60 \text{ g}$$

$$A_1 = \frac{\ddot{y}_{máx e}}{Q} = \frac{0.60}{4} = 0.15 \text{ g} \geq a_0$$

$$A_1 = 0.15 (781 \text{ cm/seg}^2) = 117.15 \text{ cm/seg}^2 \text{ (aceleración elástica)}.$$

SEGUNDO MODO:

$$T_2 = 0.3772 \text{ seg.}$$

$$T_2 < T_a \Rightarrow \text{cae en rama ascendente, por lo que hay reducción por ductilidad.}$$

$$\begin{aligned} \ddot{u}_{\text{máx. } z} &= \dot{d}_0 + \frac{I_z}{T_a} (g - \dot{d}_0) \\ &= 0.15 + \frac{0.2772}{0.60} (0.60 - 0.15) \\ &= 0.4479 \text{ g (aceleración elástica)}. \end{aligned}$$

$$\begin{aligned} Q' &= 1 + \frac{(Q-1)I_z}{T_a} = 1 + \frac{(4-1) \cdot 0.2772}{0.6} \\ &= 2.9860 \end{aligned}$$

$$\begin{aligned} A_z &= \frac{\ddot{u}_{\text{máx. } z}}{Q'} = \frac{0.4479 \text{ g}}{2.9860} = 0.15 \text{ g} > \dot{d}_0 \\ &= 0.15 \text{ g} = 0.15 (981 \text{ cm/seg}^2) \\ &= 147.15 \text{ cm/seg}^2 \end{aligned}$$

De igual forma se calcula el TERCER Y CUARTO MODO :

$$\begin{aligned} Q_3 &= 2.3030 \\ Q_4 &= 2.0785 \end{aligned}$$

$$\begin{aligned} A_3 &= 147.15 \text{ cm/seg}^2 \\ A_4 &= 147.15 \text{ cm/seg}^2 \end{aligned}$$

4) CALCULO DE FACTORES DE PARTICIPACION :

Empleando la ecuación (6-27) se tiene:

$$\begin{aligned} C_1 &= \frac{470(0.2360) + 450(0.5404) + 380(0.8263) + 350(1)}{470(0.2360)^2 + 450(0.5404)^2 + 380(0.8263)^2 + 350(1)^2} \\ &= 1.3273 \end{aligned}$$

$$\begin{aligned} C_2 &= \frac{470(-0.6773) + 450(-0.7686) + 380(-0.1246) + 350(1)}{470(0.6773)^2 + 450(0.7686)^2 + 380(-0.1246)^2 + 350(1)^2} \\ &= -0.4548 \end{aligned}$$

$$\begin{aligned} C_3 &= \frac{470(1.2117) + 450(0.0701) + 380(-1.6120) + 350(1)}{470(1.2117)^2 + 450(0.0701)^2 + 380(-1.6120)^2 + 350(1)^2} \\ &= 0.1711 \end{aligned}$$

$$\begin{aligned} C_4 &= \frac{470(-3.9231) + 450(3.7816) + 380(-2.8117) + 350(1)}{470(-3.9231)^2 + 450(3.7816)^2 + 380(-2.8117)^2 + 350(1)^2} \\ &= -0.0435 \end{aligned}$$

5) DESPLAZAMIENTOS MAXIMOS DE LOS PISOS EN CADA MODO :

Empleando la ecuación (6-28) se tiene:

$$y_{ij} \max = \frac{A_j}{\omega_j^2} C_j \phi_{ij}$$

Primer modo:

$$y_{i1} \max = \frac{147.15}{38.64} (1.3273) \begin{Bmatrix} 0.2360 \\ 0.5404 \\ 0.8263 \\ 1.0 \end{Bmatrix} = \begin{Bmatrix} 1.1929 \\ 2.7315 \\ 4.1767 \\ 5.0547 \end{Bmatrix}$$

Segundo modo:

$$y_{i2} \max = \frac{147.15}{250.2498} (-0.4548) \begin{Bmatrix} -0.6773 \\ -0.9686 \\ -0.1246 \\ 1.0 \end{Bmatrix} = \begin{Bmatrix} 0.1817 \\ 0.2570 \\ 0.0333 \\ -0.2674 \end{Bmatrix}$$

Tercer modo:

$$y_{i3} \max = \frac{147.15}{581.4479} (0.1711) \begin{Bmatrix} 1.2117 \\ 0.0701 \\ -1.6120 \\ 1.0 \end{Bmatrix} = \begin{Bmatrix} 0.0525 \\ 0.0370 \\ -0.0678 \\ 0.0433 \end{Bmatrix}$$

Cuarto modo:

$$y_{i4} \max = \frac{147.15}{848.3783} (-0.0435) \begin{Bmatrix} -3.9231 \\ 3.9816 \\ -2.8117 \\ 1.0 \end{Bmatrix} = \begin{Bmatrix} 0.0276 \\ -0.0300 \\ 0.0212 \\ -0.0075 \end{Bmatrix}$$

6) DESPLAZAMIENTOS RELATIVOS DE LOS ENTREPISOS EN CADA MODO :

Empleando la ecuación (6-29) se tiene:

$$\Delta_{ij} = y_{ij} - y_{(i-1)j}$$

Primer modo:

$$\Delta_{ij} = \begin{Bmatrix} 1.1929 \\ 1.5386 \\ 1.4452 \\ 0.8780 \end{Bmatrix}$$

Segundo modo:

$$\Delta_{ij} = \begin{Bmatrix} 0.1817 \\ 0.0773 \\ -0.2257 \\ -0.3007 \end{Bmatrix}$$

$$\text{Tercer modo:} \\ \Delta_{3j} = \begin{Bmatrix} 0.0545 \\ -0.0135 \\ -0.1088 \\ 0.1131 \end{Bmatrix}$$

$$\text{Cuarto modo:} \\ \Delta_{4j} = \begin{Bmatrix} 0.0276 \\ -0.0579 \\ 0.0512 \\ -0.0287 \end{Bmatrix}$$

7) CONSTANTES DE ENTREPISO PARA CADA MODO:

Empleando la ecuación (6-30) se tiene:

$$V_{ij} = K_j \Delta_{ij}$$

$$\text{Primer modo:} \\ V_{j1} = \begin{Bmatrix} 1.1927 (169.7050) \\ 1.5386 (117.4043) \\ 1.4452 (91.4817) \\ 0.8760 (77.3989) \end{Bmatrix} = \begin{Bmatrix} 202.6777 \\ 180.6363 \\ 132.2074 \\ 69.7122 \end{Bmatrix}$$

$$\text{Segundo modo:} \\ V_{j2} = \begin{Bmatrix} 0.1817 (169.7050) \\ 0.0773 (117.4043) \\ -0.2257 (91.4817) \\ -0.3007 (77.3989) \end{Bmatrix} = \begin{Bmatrix} 30.8717 \\ 9.0754 \\ -20.6174 \\ -23.8452 \end{Bmatrix}$$

$$\text{Tercer modo:} \\ V_{j3} = \begin{Bmatrix} 0.0525 (169.7050) \\ -0.0135 (117.4043) \\ -0.1088 (91.4817) \\ 0.1131 (77.3989) \end{Bmatrix} = \begin{Bmatrix} 8.9200 \\ -1.5850 \\ -7.7532 \\ 8.78 \end{Bmatrix}$$

$$\text{Cuarto modo:} \\ V_{j4} = \begin{Bmatrix} 0.0276 (169.7050) \\ -0.0579 (117.4043) \\ 0.0512 (91.4817) \\ -0.0287 (77.3989) \end{Bmatrix} = \begin{Bmatrix} 5.0272 \\ -6.7773 \\ 4.6839 \\ -2.2787 \end{Bmatrix}$$

B) CONSTANTES TOTALES DE ENTREPISO :

Empleando la ecuación (6-31) se tiene:

$$V_i = \sqrt{\sum_{j=1}^n V_{ij}^2}$$

Primer entrepiso :

$$V_1 = \frac{\sqrt{(202.6777)^2 + (30.8717)^2 + (8.92)^2 + (5.0292)^2}}{205.27 T}$$

Segundo entrepiso :

$$V_2 = \frac{\sqrt{(180.6383)^2 + (9.0754)^2 + (-1.585)^2 + (-6.7773)^2}}{181.01 T}$$

Tercer entrepiso :

$$V_3 = \frac{\sqrt{(132.2094)^2 + (-20.6474)^2 + (-9.9532)^2 + (4.6839)^2}}{134.26 T}$$

Cuarto entrepiso :

$$V_4 = \frac{\sqrt{(69.7122)^2 + (-23.8752)^2 + (8.98)^2 + (-2.2787)^2}}{74.27 T}$$

9) FUERZAS EN CADA PISO, POR SISMO :

Empleando la ecuación (6-32) se tiene :

$$F_i = V_i - V_{i+1}$$

$$\begin{aligned} \text{Cuarto piso: } F_4 &= V_4 - 0 \\ &= 205.27 T \end{aligned}$$

$$\begin{aligned} \text{Tercer piso: } F_3 &= V_3 - V_4 \\ &= 134.26 T - 74.27 T \\ &= 59.99 T \end{aligned}$$

$$\begin{aligned} \text{Segundo piso: } F_2 &= V_2 - V_3 \\ &= 181.01 T - 134.26 T \\ &= 46.75 T \end{aligned}$$

$$\begin{aligned} \text{Primer piso: } F_1 &= V_1 - V_2 \\ &= 205.27 T - 181.01 T \\ &= 24.26 T \end{aligned}$$

10) DESPLAZAMIENTOS RELATIVOS Y TOTALES RELATIVOS:

Empleando la ecuación (6-33) se tiene:

$$\Delta_i = \sqrt{\sum (Q_j \Delta_{ij})^2}$$

Primer entrepiso:

$$\Delta_1^2 = (4 \times 1.1929)^2 + (2.986 \times 0.1817)^2 + (2.303 \times 0.0525)^2 + (2.0785 \times 0.0276)^2$$

$$\Delta_1 = 4.87 \text{ cm} = 0.0480 \text{ m}$$

Segundo entrepiso:

$$\Delta_2^2 = (4 \times 1.5386)^2 + (2.986 \times 0.0773)^2 + (2.303 \times 0.0135)^2 + (2.0785 \times 0.0276)^2$$

$$\Delta_2 = 6.16 \text{ cm} = 0.0616 \text{ m}$$

Tercer entrepiso:

$$\Delta_3^2 = (4 \times 1.4452)^2 + (2.986 \times 0.2257)^2 + (2.303 \times 0.1082)^2 + (2.0785 \times 0.0512)^2$$

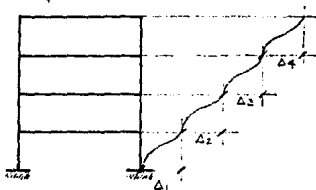
$$\Delta_3 = 5.83 \text{ cm} = 0.0583 \text{ m}$$

Cuarto entrepiso:

$$\Delta_4^2 = (4 \times 0.8783)^2 + (2.986 \times 0.3007)^2 + (2.303 \times 0.1131)^2 + (2.0785 \times 0.0287)^2$$

$$\Delta_4 = 3.63 \text{ cm} = 0.0363 \text{ m}$$

Desplazamientos relativos:

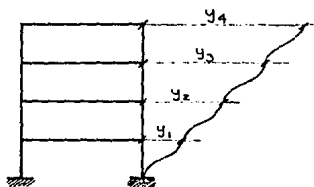


Empleando ecuación (6-24) se tiene:

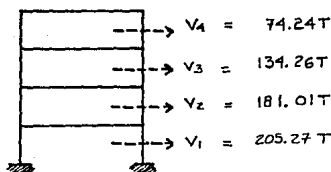
$$y_j = y_{j-1} + \Delta_j$$

Primer piso:	$y_1 = 0 + \Delta_1 = 4.80 \text{ cm}$
Segundo piso:	$y_2 = y_1 + \Delta_2 = 4.80 + 6.16 = 10.76 \text{ cm}$
Tercer piso:	$y_3 = y_2 + \Delta_3 = 10.76 + 5.82 = 16.77 \text{ cm}$
Cuarto piso:	$y_4 = y_3 + \Delta_4 = 16.77 + 3.63 = 20.42 \text{ cm}$

Desplazamientos totales:



Cortantes de entrepiso:



**CAPITULO VII :
DESARROLLO DE UN
SUBPROGRAMA PARA LA
DISTRIBUCION DE CORTANTES
DE ENTREPISO**

CAPITULO VII

DESARROLLO DE UN SUBPROGRAMA PARA LA DISTRIBUCION DE CORTANTES DE ENTREPISO

7.1 DESCRIPCION GENERAL :

Después de haber obtenido las fuerzas cortantes sísmicas en cada entrepiso, se requiere distribuirías entre los elementos resistentes que conforman la estructura en el sentido en que actúa la fuerza sísmica. Hay que recordar que las fuerzas sísmicas anteriormente calculadas mediante los análisis estático y dinámico, son fuerzas cortantes aplicadas a los entrepisos del modelo analítico simplificado del edificio , por lo que se requiere distribuir dichas fuerzas a cada elemento resistente que conforman al edificio real.

Se debe de señalar que el procedimiento descrito en éste capítulo, realiza una distribución de las fuerzas cortantes sísmicas que actúan en los pisos de un edificio entre los diferentes elementos resistentes, basándose en la hipótesis de que los pisos son diafragmas rígidos en su plano, ésta hipótesis puede llegar a ser inadmisibles en edificios que cuenten con alguna de las siguientes características:

- En edificios cuya longitud en planta sea varias veces su ancho.
- En edificios cuya rigidez ante cargas laterales no está distribuida de una manera sensiblemente uniforme en todo su largo.
- En edificios con sistema de losas precoladas.

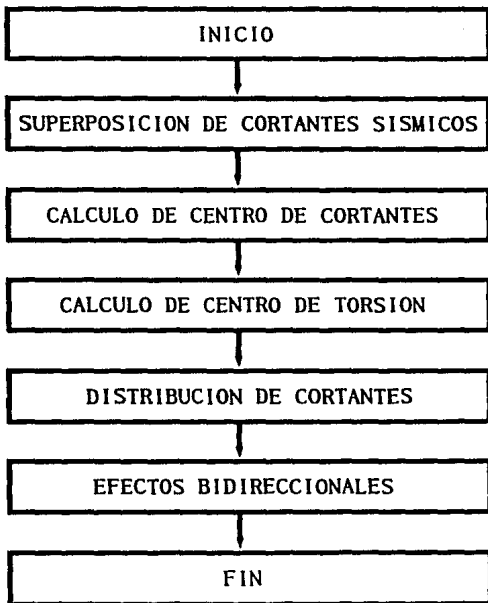
Si una estructura en el espacio se sujeta a un movimiento horizontal del terreno y el centro de masa no coincide con el centro de la reacción de la estructura, se presenta una vibración torsional. Un análisis dinámico que incluya estos efectos de torsión resulta ser muy elaborado para las estructuras comunes, en el análisis aquí utilizado se toman los efectos de torsión en forma estática, superponiendo sus resultados a los obtenidos

de un análisis sísmico estático o dinámico, que tomen en cuenta los efectos de traslación calculados de forma independiente. Son considerados éstos tres desplazamientos (traslación X-Y y giro en Z) debido a que son los que más efectos tienen en las respuestas dinámicas de un edificio.

Los distintos procesos que integran el subprograma de distribución de cortantes de entrepiso, son tratados en forma aislada, se presentan únicamente su respectivos diagramas de flujo y al final del capítulo se presentan ensamblados todos en el código fuente del subprograma.

En forma general, el proceso de la distribución de cortantes de entrepiso consta de los siguientes pasos, vistos en forma esquemática en el siguiente diagrama de bloques:

Diagrama de bloques:



7.2 SUPERPOSICION DE CORTANTES SISMICOS :

Es recomendable realizar una superposición de fuerzas cortantes sísmicas cuando se realice el análisis sísmico estático y análisis sísmico dinámico, principalmente en edificios que requieren un análisis más refinado, como lo son los edificios de más de 60 mts. de altura (aproximadamente más de 20 pisos) y que cuenten con una estructura preferentemente regular.

La superposición consiste en verificar que en ningún entrepiso la fuerza cortante calculada por algún método dinámico sea menor que el 80% del cortante determinado estáticamente, sino se cumple la anterior condición se toma el 80% del cortante calculado estáticamente. De ésta superposición se obtienen los cortantes de entrepiso que se utilizarán en el análisis de distribución de cortantes.

Una vez realizada dicha superposición se obtienen los cortantes de entrepiso, con éstos se calculan las fuerzas sísmicas de diseño por cada piso y para cada dirección de la estructura analizada.

7.2.1 Subprograma :

El subprograma aquí presentado realiza la superposición y al mismo tiempo encuentra las fuerzas sísmicas de diseño. Los cortantes estáticos y dinámicos que utiliza ésta subrutina fueron calculados en los capítulos anteriores. La superposición queda mejor indicada en el siguiente diagrama de flujo, en donde son utilizadas las siguientes variables:

Variables Auxiliares:

- i Índice que determina la dirección que se analiza (i=0: dirección X, i=1: dirección Y).
- j Índice que determina que piso es analizado (j=1,2,...,np).

Variables de Entrada (INPUT):

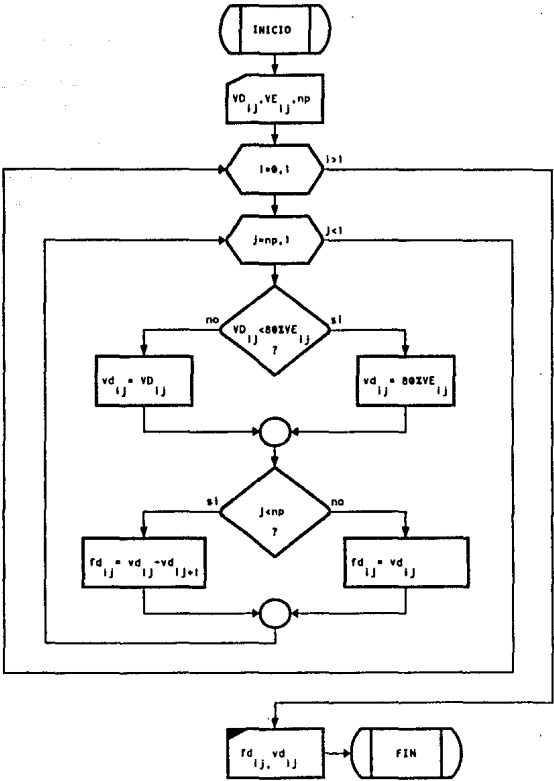
- np Número de pisos del edificio.
- VD_{ij} Cortantes de entrepiso sísmicos obtenidos por el método dinámico.
- VE_{ij} Cortantes de entrepiso sísmicos obtenidos por el método estático.

Variables de Salida (OUTPUT):

- vd_{ij} Cortantes de entrepiso obtenidos de la superposición.
- fd_{ij} Fuerzas de piso sísmicas obtenidas de la superposición.

Diagrama de flujo:

SUPERPOSICION DE CORTANTES SISMICOS



7.3 CENTRO DE CORTANTES :

Centro de cortante:

El centro de cortantes es el punto donde se idealiza que está aplicado el cortante de entrepiso (fuerza sísmica). Se obtiene calculando la posición de la resultante de las fuerzas laterales que integran el cortante.

Para el entrepiso i:

$$V_i = \sum_{j=1}^{np} F_j \quad (7-1)$$

Para cada piso, la fuerza lateral inducida por el sismo está aplicada en el centro de masas, por lo que:

$$xV_i = \frac{\sum_{j=1}^{np} F_j x m_j}{\sum_{j=1}^{np} F_j} \quad (7-2)$$

$$yV_i = \frac{\sum_{j=1}^{np} F_j y m_j}{\sum_{j=1}^{np} F_j} \quad (7-3)$$

En donde:

xV_i, yV_i
 F_j
 xm_j, ym_j
 np

Coordenadas del centro de cortantes en el entrepiso i.
Es la fuerza lateral por sismo en el piso j.
Coordenadas del centro de masas del piso j.
Número de pisos del edificio.

7.3.1 Subprograma :

El cálculo del centro de cortantes se realiza en las dos direcciones de análisis (dirección X y Y), para fines del desarrollo de ésta parte del subprograma, se utilizan las siguientes variables:

Variables Auxiliares:

- i Índice que determina la dirección que se analiza ($i=0$, dirección X; $i=1$, dirección Y).
- j Índice que determina que piso es analizado ($j=1,2,\dots,np$).

Variables de Entrada (INPUT):

- np Número de pisos del edificio.
- fd_{ij} Fuerzas laterales por sismo en la dirección i , en el piso j .
- xm_{ij} Coordenadas del centro de masa en la dirección i del piso j .

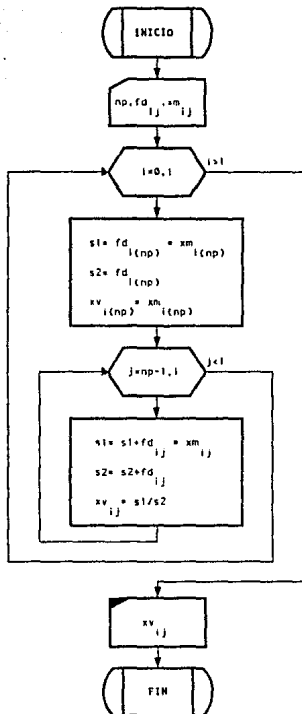
Variables de salida (OUTPUT):

- xv_{ij} Coordenadas del centro de cortantes en la dirección i del piso j .

El listado de ésta parte del subprograma fuente en Lenguaje C, se presenta al final del presente capítulo, se encuentra acoplado a los demás temas aquí tratados.

Diagrama de flujo:

CENTRO DE CORTANTES



7.4 CENTRO DE TORSION :

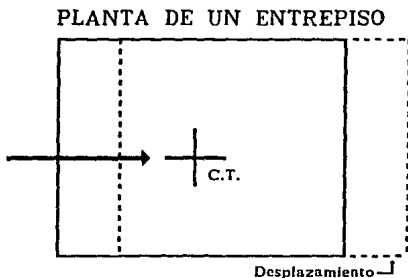
Los análisis sísmicos aquí utilizados para calcular las respuestas sísmicas, consideran que la estructura únicamente tenía desplazamientos de traslación. Un modelo más real debería de incluir como mínimo una rotación en el eje Z por cada masa, la importancia de las rotaciones y la magnitud de las solicitaciones que por éste se inducen en la estructura, están en función de la distribución en planta de las masas y de las rigideces laterales.

Por equilibrio, la acción de la fuerza sísmica en cada piso está situada en el centro de cortante, mientras que la fuerza resistente lo está en el centro de torsión, o sea donde se ubica la resultante de las fuerzas laterales que toma cada elemento resistente.

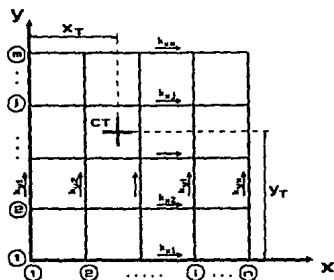
Si entre la localización de estos dos centros existe una excentricidad, la acción en cada entrepiso estará constituida por una fuerza cortante más un momento torsionante cuyo efecto es calculado en ésta sección y debe ser considerado para el análisis y diseño sismorresistente de un edificio.

Centro de torsión:

El centro de torsión es el punto en el que la aplicación de una fuerza cortante solo provoca la traslación del entrepiso en una dirección paralela a la de la fuerza y no induce ninguna rotación.



La localización de este centro de torsión se define encontrando la resultante de los vectores que representan la rigidez de cada elemento resistente:



$$x_{T_k} = \frac{\sum_{i=1}^n ky_i x_i}{\sum_{i=1}^n ky_i} \quad (7-4)$$

$$y_{T_k} = \frac{\sum_{j=1}^m kx_j y_j}{\sum_{j=1}^m kx_j} \quad (7-5)$$

En donde:

x_{T_k}, y_{T_k}	Son las coordenadas del centro de torsión para un determinado entrepiso.
kx_j	Rigidez de entrepiso del elemento resistente j paralelo al eje principal x .
ky_i	Rigidez de entrepiso del elemento resistente i paralelo al eje principal y .
y_j	Distancia del elemento j al eje x .
x_i	Distancia del elemento i al eje y .
n	Número de elementos resistentes en la dirección x .
m	Número de elementos resistentes en la dirección y .

7.4.1 Subprograma :

El diseño de éste subprograma está orientado a obtener el centro de torsión de un edificio con planta regular (ortogonal) y con la opción de tener en alzado una forma irregular. En el desarrollo de éste subprograma se utiliza el concepto de marcos tipo, con el objetivo de minimizar el tiempo de proceso y reducir la cantidad de datos necesaria para obtener el centro de torsión, con éste fin se ahorra información sobre las rigideces de entrepiso para distintos marcos con geometría y elementos resistentes iguales.

La obtención del centro de torsión utilizando el subprograma aquí presentado, se fundamenta en las bases anteriormente mencionadas y utiliza las siguientes variables:

Variables Auxiliares:

i	Índice que indica la dirección que es analizada (i=0, dirección X; i=1, dirección Y).
j	Índice que indica que piso es analizado, j=1,...,np
k	Índice que señala que marco es analizado en una determinada dirección principal (X o Y).

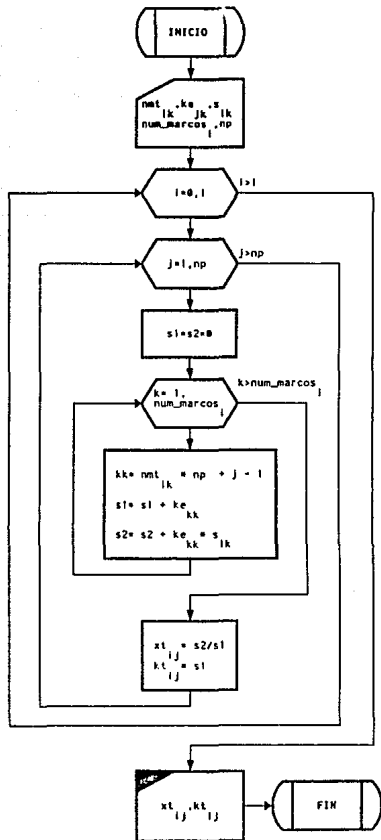
Variables de Entrada (INPUT):

np	Número de pisos de la estructura.
num_marcos _i	Número de marcos del edificio para cada dirección analizada.
s _{ik}	Coordenada X(Y) de cada marco en la dirección X(Y).
ke _{jk}	Rigidez de entrepiso, para el piso j del marco k.

Variables de Salida (OUTPUT):

xt _{ij}	Coordenada X(Y) del centro de torsión en la dirección i del piso j.
kt _{ij}	Producto de la rigidez de entrepiso j por la distancia en X(Y) del marco i al origen del sistema general. Esta variable auxiliar se emplea posteriormente para obtener el momento polar de inercia y la distribución de cortantes de entrepiso.

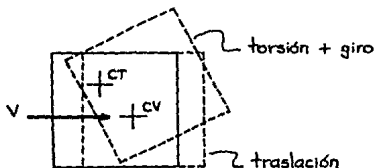
Diagrama de flujo:
CENTRO DE TORSION



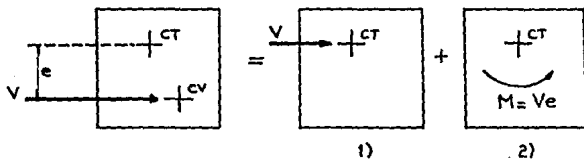
7.5 DISTRIBUCION DE CORTANTES :

7.5.1 Distribución de cortantes :

Como en general el centro de torsión y el centro de cortante no coinciden, la fuerza cortante provoca además de una traslación, un giro:



Para obtener la distribución de las fuerzas cortantes, se supone que en la planta actúan:



- 1) Una fuerza cortante aplicada en el centro de torsión que sólo produce efectos de traslación.
- 2) Un momento torsionante en torno al centro de torsión que sólo produce efectos de giro.

En cada elemento resistente el cortante produce un cortante directo debido a la traslación y un cortante de torsión debido al giro.

Considerando primero que en el entrepiso que se está analizando el cortante es paralelo al eje y , se tiene:

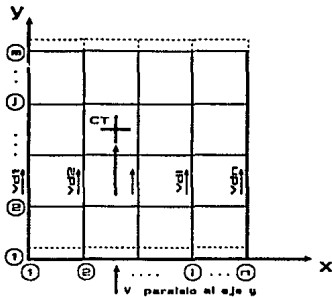
■ **Primer Etapa:**

El cortante aplicado en el centro de torsión produce en los marcos paralelos al eje y cortantes directos que pueden evaluarse mediante la siguiente expresión:

$$Vd_i = \frac{V}{\sum_{j=1}^n ky_j} ky_i \quad (7-6)$$

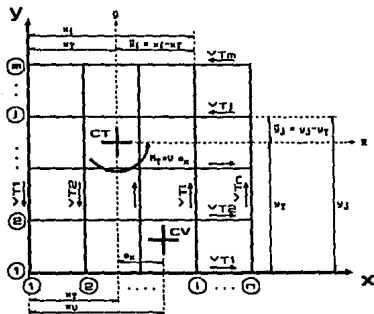
En donde:

- ky_i Rigidez del elemento i, paralelo al eje y.
- Vd_i Cortante directo en el elemento i.
- V Cortante directo total de entrepiso.
- n Número de elementos en la dirección x.



■ **Segunda Etapa:**

El momento torsionante produce en todos los marcos cortantes de torsión que pueden calcularse de la siguiente forma:



$$V_T \frac{M_T}{J} k_y \bar{x}_i \quad (7-7)$$

$$V_T \frac{M_T}{J} k_x \bar{y}_j \quad (7-8)$$

En donde:

J Momento polar de inercia:

$$J = \sum_{i=1}^m k_x \bar{y}_i^2 + \sum_{j=1}^n k_y \bar{x}_j^2 \quad (7-9)$$

x_i Distancia al eje y del elemento i:
 $\bar{x}_i = x_i - x_T \quad (7-10)$

y_j Distancia al eje x del elemento j:
 $\bar{y}_j = y_j - y_T \quad (7-11)$

El momento torsionante M_T es el producto $V(e_x)$, en donde:

V Cortante de entrepiso.
 e_x Excentricidad paralela al eje x.

Subprograma:

A continuación son presentados los diagramas de flujo que representan a la distribución de cortantes, también es presentado un diagrama auxiliar para el cálculo del momento polar de inercia, lo cual se realiza para ejemplificar mas claramente éste desarrollo. El código fuente de éstos diagramas se presenta al final del capítulo en conjunto con los demás temas aquí tratados en el capítulo VII, formando parte del subprograma completo: "DISTRIBUCION DE CORTANTES DE ENTREPISO".

MOMENTO POLAR DE INERCIA

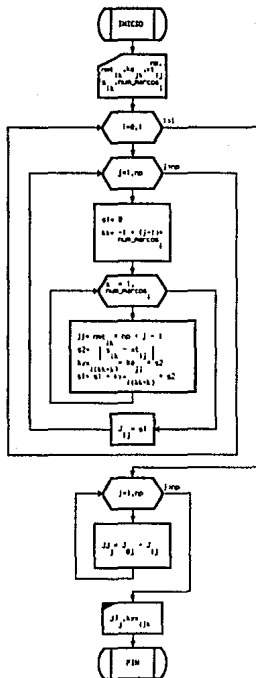
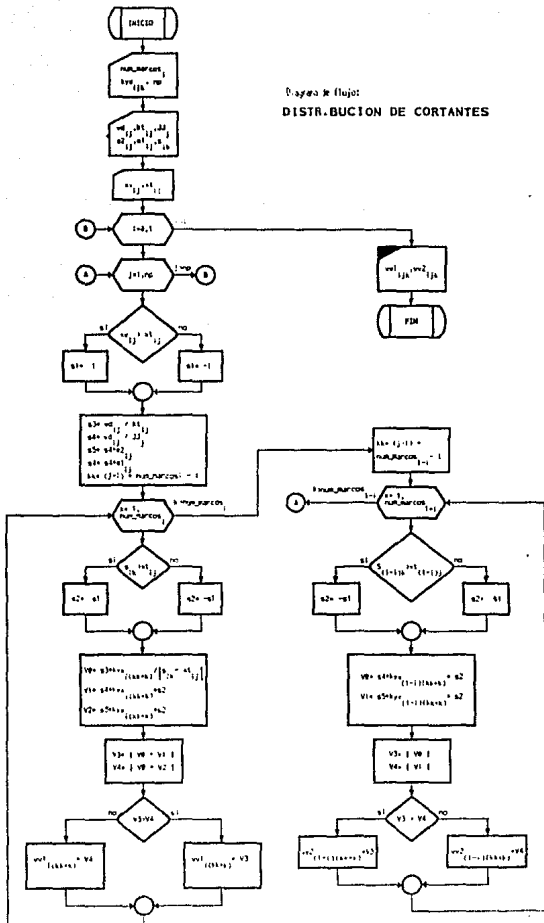


Diagrama de flujo:

DISTRIBUCION DE CORTANTES



7.5.2 Excentricidades reglamentarias :

La excentricidad torsional de rigideces se debe de calcular en cada entrepiso, corresponde a la distancia entre el centro de torsión en el nivel correspondiente y la fuerza cortante en dicho nivel, teóricamente esta excentricidad vale:

$$e_{xo} = x_v - x_T \quad (7-12)$$

El subíndice c corresponde a la excentricidad calculada. Sin embargo, para fines de diseño, los reglamentos de construcciones toman en cuenta:

- Incertidumbre en el cálculo de las rigideces.
- Variaciones que puede tener la distribución de masa.

Para tomar en cuenta los efectos dinámicos de la torsión, se estipula que la excentricidad calculada se modifique conforme a la posibilidad más desfavorable de las siguientes dos:

$$e_{x1} = 1.50 e_{xo} + 0.10 bx \quad (7-13)$$

$$e_{x2} = 1.00 e_{xo} - 0.10 bx \quad (7-14)$$

En donde:

bx Es el ancho de la planta en la dirección x.
 e_{xo} Excentricidad torsional de rigideces.

Con la excentricidad más desfavorable se calcula para cada marco dos cortantes de torsión:

$$M_{T1} = V e_{x1} \quad (7-15)$$

$$M_{T2} = V e_{x2} \quad (7-16)$$

El cortante total en cada marco, para la fuerza V paralela al eje y, será el más desfavorable de las siguientes combinaciones:

$$V_i = V_{di} + V_{T1i} \quad (7-17)$$

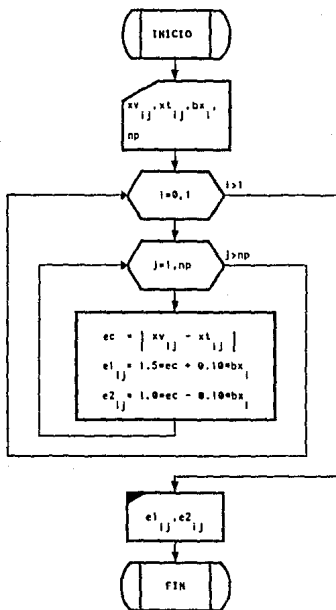
$$V_i = V_{di} + V_{T2i} \quad (7-18)$$

Con lo que queda definido V_{iy} como el cortante total en el marco i cuando el sismo se supone paralelo al eje y.

- Otro análisis similar se hace suponiendo que el sismo actúa paralelo al eje x, con lo que se obtendrá para el marco i un cortante total V_{ix} .

Diagrama de flujo:

EXCENRICIDADES REGLAMENTARIAS



7.6 EFECTOS BIDIRECCIONALES :

Se debe de considerar ambas componentes horizontales del movimiento del terreno en una misma dirección, debido a que el sismo actúa en cualquier dirección aleatoria. En cada dirección en que se analice la estructura se debe de tomar en cuenta el 100% de los componentes que obran en esa dirección y el 30% de los efectos de las componentes que obran perpendicularmente a ella, teniendo en cuenta los signos que para cada combinación resulten más desfavorables se tiene:

$$V_{1i} = V_{i(x)} + 0.30 V_{i(y)} \quad (7-19)$$

$$V_{2i} = V_{i(y)} + 0.30 V_{i(x)} \quad (7-20)$$

7.6.1 Subprograma :

La consideración de los efectos bidireccionales se realiza para cada elemento resistente y para cada piso analizado, las variables utilizadas en el diagrama de flujo son las mismas que se utilizan en el código fuente, y son las que se mencionan a continuación:

Variables Auxillares:

- i Índice que indica la dirección que es analizada (i=0, dirección X; i=1, dirección Y).
- j Índice que indica que piso es analizado, j=1,...,np
- k Índice que señala que marco es analizado en una determinada dirección principal (X o Y).

Variables de Entrada (INPUT):

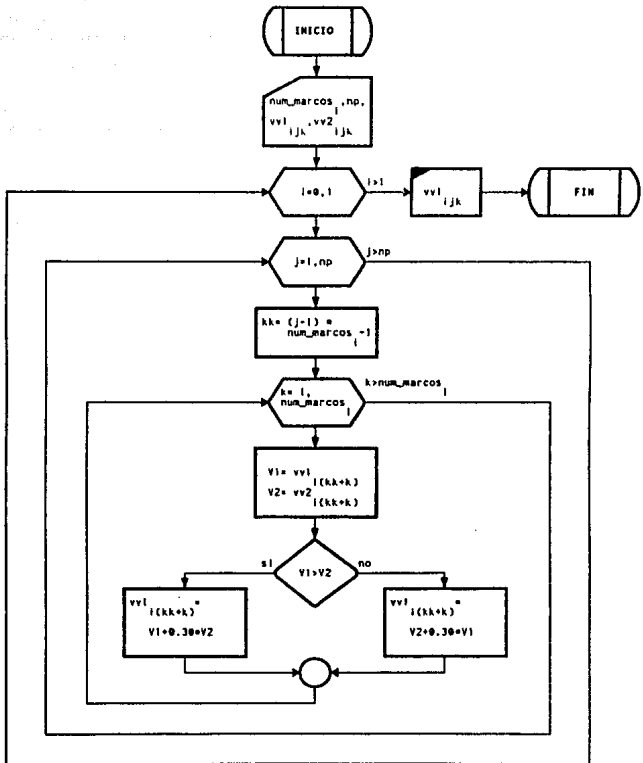
- np Número de pisos del edificio.
- num_marcos, Número de marcos planos del edificio para cada dirección i analizada.
- wv1_{ijk} Primer combinación de cortantes en cada marco k, para la fuerza cortante (V) paralela a la dirección i (X ó Y) analizada, en el piso j.
- wv2_{ijk} Segunda combinación de cortantes en cada marco k, para la fuerza cortante (V) paralela a la dirección i (X ó Y) analizada, en el piso j.

Variables de Salida (OUTPUT):

- wv1_{ijk} Cortante total en cada marco k, para la fuerza V paralela a la dirección i (X ó Y) analizada, en el piso j. (Se emplea gradualmente las mismas posiciones de memoria de la primer combinación de cortantes, para ahorrar memoria).

Diagrama de flujo:

EFECTOS BIDIRECCIONALES



7.7 SUBPROGRAMA :

7.7.1 Variables del subprograma :

En ésta parte del capítulo se resumen los temas anteriormente tratados, formando parte del subprograma general del capítulo, las variables que a continuación son descritas, corresponden a los diagramas de flujo anteriormente mencionados, estas variables también son aplicables al subprograma general del capítulo VII.

Variables Auxiliares:

i	Índice que indica la dirección que es analizada (i=0, dirección X; i=1, dirección Y).
j	Índice que indica que piso es analizado, j=1,...,np
k	Índice que señala que marco es analizado en una determinada dirección principal (X o Y).

Variables de Entrada (INPUT):

np	Número de pisos del edificio.
VD _{ij}	Cortantes de entrepiso sísmicos obtenidos por el método dinámico.
VE _{ij}	Cortantes de entrepiso sísmicos obtenidos por el método estático.
xm _{ij}	Coordenadas del centro de masa en la dirección i del piso j.
num_marcos _i	Número de marcos planos del edificio para cada dirección i analizada.
s _{ik}	Coordenada X(Y) de cada marco en la dirección X(Y), es equivalente a la distancia del elemento k al eje principal X.
ke _{jk}	Rigidez de entrepiso, para el piso j del marco k.
nmt _{ik}	Número de marco tipo que representa al marco k en la dirección i.
num_marcot	Número de marcos tipo con los que cuenta el edificio en ambos ejes principales.

Variables de Salida (OUTPUT):

vv1 _{ijk}	Cortante total de diseño en cada marco k, para la fuerza cortante (V) paralela a la dirección i (X ó Y) analizada, en el piso j. (Se emplea gradualmente las mismas posiciones de memoria de la primer combinación de cortantes, para ahorrar memoria).
vv1 _{ijk}	Primer combinación de cortantes en cada marco k, para la fuerza cortante (V) paralela a la dirección i (X ó Y) analizada, en el piso j.
vv2 _{ijk}	Segunda combinación de cortantes en cada marco k, para la fuerza cortante (V) paralela a la dirección i (X ó Y) analizada, en el piso j.
xt _{ij}	Coordenadas (X,Y) del centro de torsión en la dirección i del piso j.
xv _{ij}	Coordenadas (X,Y) del centro de cortantes en la dirección analizada i del piso j.

vd_{ij}	Cortantes de entrepiso obtenidos de la superposición de fuerzas sísmicas estáticas y dinámicas, para el entrepiso j en la dirección analizada i .
fd_{ij}	Fuerzas de piso obtenidas de la superposición (fuerzas sísmicas estáticas y dinámicas), para el piso j en la dirección analizada i .
J_j	Momento polar de inercia para el piso j .
$e1_{ij}$	Primer condición de excentricidad reglamentaria para la dirección i del piso j .
$e2_{ij}$	Segunda condición de excentricidad reglamentaria para la dirección i del piso j .
bx_i	Máxima dimensión longitudinal en la planta del edificio, medida perpendicularmente a la dirección i del movimiento del terreno que se está analizando.

7.7.2 Programa fuente en Lenguaje C :

Es el programa fuente que representa a los diagramas de flujo utilizados en temas anteriores, formando parte del subprograma para la distribución de cortantes de entrepiso. La filosofía de su programación es el de tratar de realizar un programa eficiente en el uso de memoria y con un tiempo de ejecución mínimo. En algunas partes del código, se podría pensar que no concuerda con la teoría descrita a lo largo de éste capítulo, esto se debe a que se hace variar un poco para utilizar más eficientemente el Lenguaje C pero sin alterar las bases y resultados de los métodos empleados.

Se presenta al subprograma conteniendo a todas las subrutinas empleadas en el capítulo, los únicos datos que se requieren son los datos de entrada (INPUT), los datos de salida (OUTPUT) son guardados en un archivo secuencial que contiene a todos los resultados que genera el programa general ANESAR. Dicho archivo secuencial puede ser visto o impreso al terminar de ejecutar el programa, utilizando comandos del sistema operativo empleado en la computadora.

Se muestra a continuación el bloque de código que fue desarrollado a lo largo de este capítulo :

/*

***** UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO *****
***** ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES ARAGON *****

TESIS : ANESAR, PROGRAMA PARA EL ANALISIS SISMICO DE EDIFICIOS .

CAPITULO : VII.
DESARROLLO DE UN SUBPROGRAMA PARA EL ANALISIS SISMICO DINAMICO .

Autor : Francisco Reyes Téllez .
No.Cta. : 8428671-6 .
Carrera : INGENIERIA CIVIL <87-91> .

Asesor : Ing. Amílcar Galindo Solorzano .

NOTA: Todas las unidades deben de estar en : [Ton,m,seg]
*/

```
void dis_cort(rp,num_marcos,rnt,s,xm,VE,VD,ke)
double *s1[2],*s2[2],*v1[2],*v2[2],*kx[2],*kx[2];
int rp,*num_marcos,*rnt[2];

(

double *xv[2],*j[2],*xt[2],*j[2],*kt[2],*fd[2],*vd[2],
      *e1[2],*e2[2],*v1[2],*v2[2],*kx[2],*kx[2];

/* NOTA: bx, son datos externos que son definidos en esta subrutina */

double ec;
int i,j,k,kk,jj;
double s1,s2,s3;
double vvd,vt1,vt2;

/* La separación entre marcos debe estar en orden ascendente. */
bx[0]=s[1][num_marcos[1]];
bx[1]=s[0][num_marcos[0]];

kxy[0]=calloc(rp*num_marcos[0],8);kxy[1]=calloc(rp*num_marcos[1],8);
v1[0]=calloc(rp*num_marcos[0],8);v1[1]=calloc(rp*num_marcos[1],8);
v2[0]=calloc(rp*num_marcos[0],8);v2[1]=calloc(rp*num_marcos[1],8);

j1=calloc(rp+1,8);
xt[0]=calloc(rp+1,8);xt[1]=calloc(rp+1,8);
xv[0]=calloc(rp+1,8);xv[1]=calloc(rp+1,8);
j[0]=calloc(rp+1,8);j[1]=calloc(rp+1,8);
kt[0]=calloc(rp+1,8);kt[1]=calloc(rp+1,8);
e1[0]=calloc(rp+1,8);e1[1]=calloc(rp+1,8);
e2[0]=calloc(rp+1,8);e2[1]=calloc(rp+1,8);
vd[0]=calloc(rp+1,8);vd[1]=calloc(rp+1,8);
fd[0]=calloc(rp+1,8);fd[1]=calloc(rp+1,8);

/* Cálculo de la superposición de fuerzas sísmicas estáticas y fuerzas dinámicas */
for(i=0;i<2;i++)
for(j=np;j-->0;j--) {
    v1[i][j]=VD[i][j]<0.80*VE[i][j] ? 0.80*VE[i][j] : VD[i][j];
    fd[i][j]=(-rp) * vd[i][j]-vd[i][j+1] : vd[i][j];
}

/* Cálculo del centro de cortante */
for(i=0;i<2;i++) {
    s1=fd[i][rp]*xm[i][rp];
    s2=fd[i][rp];
    xv[i][rp]=xm[i][rp];
    for(j=np-1;j>0;j--) {
        s1=fd[i][j]*xm[i][j];
        s2=fd[i][j];
        xv[i][j]=s1/s2;
    }
}

/* Cálculo del centro de torsión */
for(i=0;i<2;i++)
for(j=1; j<=rp; j++) {
    s1=s2=0;
    for(k=0;k<=num_marcos[i];k++) {
        kk=(rnt[i][k]-1)*rp-1;
        s1+=kekkk;
        s2+=kekkk*s1[k+1];
    }
    kt[i][j]= s1;
    xt[i-1][j]= s2/s1;
}
)
```

```

/* Cálculo de Excentricidades Reglamentarias */
for(i=0;i<2;i++)
  for(j=1;j<=np;j++) {
    ec=sv1(i,j)-xt(1,j);
    s1=(ec>0)?1:-1;
    ec=fabs(ec);
    e1(1,j)=(1.5*ec+0.10*bx(i))*s1;
    e2(1,j)=(1.0*ec-0.10*bx(i))*s1;
  }

/* Cálculo del momento polar de inercia */
for(i=0;i<2;i++)
  for(j=1;j<=np;j++) {
    s1=0;
    for(k=0;k<=num_marcos(i);k++) {
      kk = (nm1(i)[k]-1)*np+1;
      jj = k*np+1;
      s1 = s1+(k-1)*xt(1-i,j);
      kxy(1,j)=ke(kk)*s1;
      s2 += ke(kk)*s1*s1;
    }
    J(1,j)=s2;
  }
  JJ(j)=JJ(0,j)+J(1,j);

/* Distribución de cortantes */
for(i=0;i<2;i++)
  for(j=1;j<=np;j++) {
    s1=vd(1,j)/kt(1,j);
    s2=vd(1,j)/JJ(j)*s1(1-i,j);
    s3=vd(1,j)/JJ(j)*s2(1-i,j);
    for(k=0;k<=num_marcos(i);k++) {
      kk = (nm1(i)[k]-1)*np+1;
      jj = k*np+1;
      vvd = s1*ke(kk);
      vt1 = s2*kxy(1,j);
      vt2 = s3*kxy(1,j);
      if (vt1>vt2) vvd+=vt1; else vvd+=vt2;
      vv1(1,j) = vvd;
    }
    for(k=0;k<=num_marcos(i-1);k++) {
      jj = k*np+1;
      vt1 = fabs(s2*kxy(1-i,j));
      vt2 = fabs(s3*kxy(1-i,j));
      vvd = (vt1>vt2) ? vt1 : vt2;
      vv2(1-i,j) = vvd;
    }
  }

/* Combinación de direcciones */
for(i=0;i<2;i++)
  for(j=1;j<=np;j++) {
    for(k=0;k<=num_marcos(i);k++) {
      jj = k*np+1;
      vt1 = vv1(1,j);
      vt2 = vv2(1-i,j);
      vv1(1,j) = (vt1>vt2) ? vt1+0.30*vt2 : vt2+0.30*vt1;
    }
  }

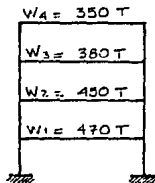
// IMPRESION DE RESULTADOS (OUTPUT)
fprintf(out,"\n\n\nSUPERPOSICION DE FUERZAS SISMICAS ESTATICAS Y DINAMICAS >>>");
linea(59);
fprintf(out,"\nEntrepiso Vx[Ton] Vy[Ton] Piso Fx[Ton] Fy[Ton]");
linea(59);
for(j=np;j>0;j--)
  fprintf(out,"\n X3d X8.2f X8.2f X3d X8.2f X8.2f",
    J,vd(0,j),vd(1,j),fd(0,j),fd(1,j));
fprintf(out,"\n\n\nCENTROS DE >>>");
linea(66);
fprintf(out,"\n [ MASA ] [ CORTANTE ] [ TORSION ]");
fprintf(out,"\nPiso xm[m] ym[m] xv[m] yv[m] xt[m] yt[m]");
linea(66);
for(j=np;j>0;j--)
  fprintf(out,"\nX3d X7.3f X7.3f X7.3f X7.3f X7.3f X7.3f",
    J,xm(0,j),xm(1,j),xv(0,j),xv(1,j),xt(0,j),xt(1,j));
fprintf(out,"\n\n\nCORTANTES SISMICAS DE DISERO >>>");
for(i=0;i<2;i++) {
  fprintf(out,"\n\nDIRECCION ANALIZADA [ % ] :",i+88);
  for(k=0;k<=num_marcos(i);k++) {
    fprintf(out,"\n");
    linea(32);
    fprintf(out,"\nMARCO [ %c-X-2d]\nEntrepiso Vxc[Ton] Fxc[Ton]",
      i+88,j+1,i+120,i+120);
    linea(32);vvd=0;
    for(knp=k+1;k-->0) {
      jj = j*np-k+1;
      vvd = vv1(1,j)-vvd;
      fprintf(out,"\n X-2d X8.2f X8.2f",k,vv1(1,j),vvd);
      vvd = vv1(1,j);
    }
  }
}

```

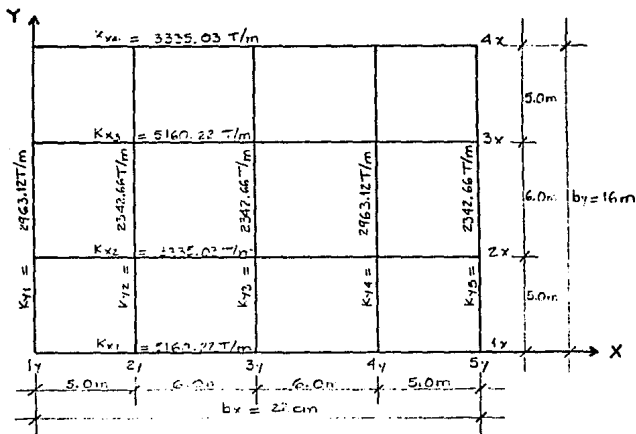
```
void lineaf(int num) {
    static char cad1[90]={"-----"};
    char cad2[90];
    strncpy(cad2, cad1, num); cad2[num]='\x0';
    fprintf(fout, "\n%a", cad2);
}
```

7.8 EJEMPLO :

Para el edificio analizado a lo largo de los capítulos anteriores, calcular los cortantes de diseño (V_{DX}, V_{DY}) para cada marco plano (elemento resistente) en el primer entrepiso.



DATOS ADICIONALES:



Distribución de fuerzas cortantes en los marcos planos del edificio analizado en el ENTREPISO 1.

Dirección	V	C.V.	C.T.	ec	ec1	ec2	Nt1	Nt2
	(7-2)	(7-4)	(7-12)	(7-13)	(7-16)	(7-15)	(7-16)	

x	205.29	11.0600	10.7605	0.2555	2.5833	-1.9445	530.33	-399.19
y	202.83	8.1860	7.4629	0.7231	2.6847	-0.8769	544.54	-177.86

$$J = \sum x_i \bar{y}_i^2 + \sum y_i \bar{x}_i^2 = 615251.55 + 832264.48 = 1447516.09$$

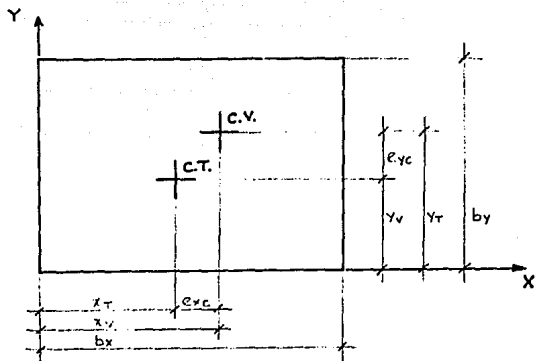
Eje	Kx	y	Kx.y	$\bar{y} \cdot y_i$	Kx. \bar{y}	Kx. \bar{y}^2	Efectos de cortante directo				Efectos de torsión			Combinación de direcciones				
							Vd	Vt ₁	Vt ₂	Vtx	Eje	Vt ₁	Vt ₂	Vtx	Eje	V ₁₁	V ₁₂	V0X
							(7-6)	(7-7)	(7-7)	(7-17)	(7-8)	(7-8)	(7-17)	(7-19)	(7-20)			
1x	5160.22	0	0.00	-7.4629	-30510.21	287397.85	62.35	-14.66	4.79	67.14	1y	12.14	-3.97	12.14	1x	71.32	34.08	71.32
2x	3335.03	5	16675.15	-2.4629	-8213.85	20229.88	40.30	-3.13	1.02	41.32	2y	5.13	-1.68	5.13	2x	42.21	15.36	42.21
3x	5160.22	11	56762.43	3.5371	18252.21	64559.91	62.35	6.95	-2.27	69.30	3y	-0.21	0.07	0.21	3x	71.28	27.40	71.28
4x	3335.03	16	53360.49	8.5371	28471.48	243063.91	40.30	10.84	-3.54	51.14	4y	-7.04	2.30	7.04	4x	54.23	25.85	54.23
Σ	16990.50	Σ	126798.07			Σ	615251.55				5y	-10.03	3.27	10.03				

$$y_T = 126798.07 / 16990.50 = 7.4629 \text{ m.}$$

Eje	Ky	x	Ky.x	$\bar{x} \cdot x_i$	Ky. \bar{x}	Ky. \bar{x}^2	Efectos de cortante directo				Efectos de torsión			Combinación de direcciones				
							Vd	Vt ₁	Vt ₂	Vty	Eje	Vt ₁	Vt ₂	Vty	Eje	V ₁₁	V ₁₂	V0T
							(7-6)	(7-7)	(7-7)	(7-17)	(7-8)	(7-8)	(7-17)	(7-19)	(7-20)			
1y	2963.12	0	0.00	-10.7605	-31884.65	343094.81	46.39	-11.54	8.69	55.08	1x	13.94	-10.49	13.94	1y	58.72	28.66	58.72
2y	2342.66	5	11713.31	-5.7605	-13494.89	77737.33	36.68	-4.88	3.68	40.36	2x	2.97	-2.24	2.97	2y	41.90	17.24	41.90
3y	2342.66	11	25769.29	0.2395	561.07	134.38	36.68	0.20	-0.15	36.88	3x	-6.61	4.97	6.61	3y	36.94	11.27	36.94
4y	2963.12	17	50372.99	6.2395	18488.39	115358.29	46.39	6.69	-5.04	53.08	4x	-10.31	7.58	10.31	4y	55.19	22.96	55.19
5y	2342.66	22	51538.58	11.2395	26330.33	295939.71	36.68	9.53	-7.17	46.21					5y	49.22	23.89	49.22
Σ	12954.22	Σ	139394.17			Σ	832264.52											

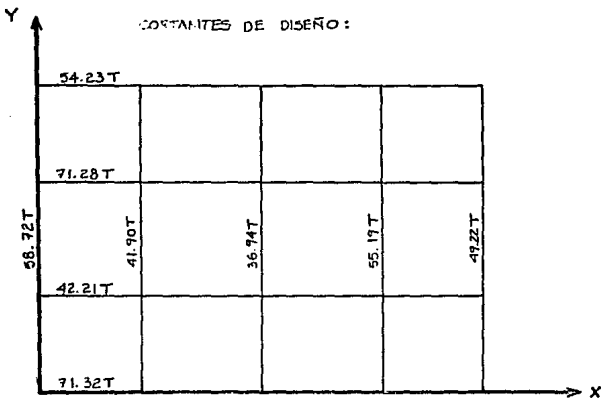
$$x_T = 139394.17 / 12954.22 = 10.7605 \text{ m.}$$

INTERPRETACION :



PLANTA : (PRIMER ENTREPISO) :

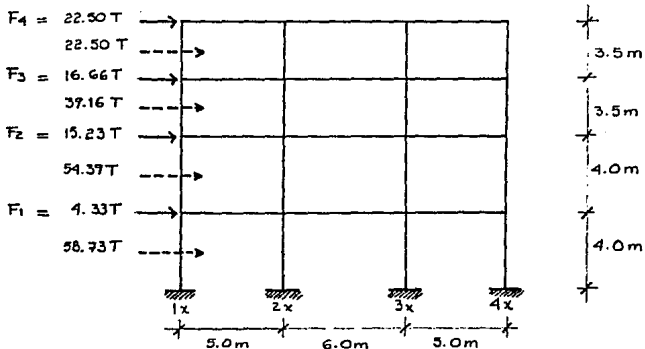
CONSTANTES DE DISEÑO :



Del análisis anteriormente realizado se obtuvieron las fuerzas cortantes de diseño del primer entrepiso, para los entrepisos siguientes se realiza el mismo análisis con el fin de obtener los cortantes de diseño en cada uno de los entrepisos del edificio. La obtención de los cortantes de diseño es el último proceso contemplado en los alcances de la presente tesis, posteriormente a este análisis se requieren los siguientes pasos con el fin de encontrar los elementos mecánicos de diseño sísmico.

Posteriormente a la obtención de los cortantes de diseño se deben de obtener las fuerzas sísmicas de diseño y con ellas analizar estructuralmente los marcos planos del edificio para la obtención de los elementos mecánicos de los miembros que conforman a la estructura.

Ejemplo: Para obtener los elementos mecánicos de diseño del marco plano 1y, se analiza estructuralmente al marco 1y con las siguientes fuerzas de diseño (F_1, F_2, F_3 y F_4):



Los cortantes de entrepiso para los niveles 2, 3 y 4 se obtuvieron mediante el programa ANESAR.

CAPTULO VIII : CONCLUSIONES



CAPITULO VIII

CONCLUSIONES

En México actualmente se cuenta con programas alternativos relativamente sofisticados para el análisis sísmico de edificios, el objetivo de éste trabajo no es el de competir con ellos, lo que se persigue es el desarrollar un programa que sirva en labores docentes y profesionales actuales, además de ser la base para la elaboración a futuro de un programa con mayores características que lo hagan más competitivo y conforme transcurra el tiempo, más actualizado. La posible actualización es muy factible debido a que el programa ANESAR fue desarrollado bajo un Lenguaje que permite la programación estructurada, lo que facilita una posterior modificación, actualización ó corrección.

Hay que tener en cuenta que los programas comerciales de análisis sísmico no son universales, esto quiere decir que no contemplan a todas las posibles necesidades de análisis, ésta desventaja es más significativa cuando se llega a presentar un problema estructural especial que no pueda ser resuelto por algún programa comercial de análisis sísmico, en este caso el problema podría no tener solución. De aquí surge una posible aplicación del programa ANESAR, debido a que sería posible modificarlo de acuerdo a las necesidades especiales requeridas para resolver el problema en cuestión, mediante la modificación del programa fuente.

En el desarrollo del programa ANESAR es evidente el conocimiento necesario de las ciencias computacionales con las que actualmente se cuenta, no se trata de ser un programador profesional o un analista de sistemas, lo que se plantea es tener los conocimientos suficientes y necesarios para poder emplear herramientas basadas en la computación que hagan posible la solución de distintos problemas estructurales, debido a que en algunos casos es imposible analizar estructuras sin contar con un procedimiento automatizado de cálculo.

Con la llegada de eficientes programas de análisis estructural se deberían de eliminar gradualmente las actitudes de algunos ingenieros estructuralistas que dedican gran parte de su tiempo a realizar cálculos repetitivos y metódicos, deberían de enfocar sus esfuerzos a la comprensión de las bases teóricas del análisis estructural, a la solución de sus aspectos fundamentales y a la revisión de los resultados que generen los cálculos efectuados con programas de análisis estructural. No obstante, en los distintos despachos estructurales

que emplean programas comerciales de análisis estructural se observa el surgimiento de profesionistas que se dedican a la codificación y generación de los datos de entrada (INPUT) requeridos por los programas, de cierta mane: a se ha vuelto a los aspectos rutinarios que se presentaban anteriormente a la llegada masiva de las computadoras.

APENDICE A

PROGRAMA FUENTE EN LENGUAJE C

■ PROGRAMA ANESAR :

En el siguiente listado del programa fuente ANESAR se acoplan todos los subprogramas desarrollados a lo largo de la tesis. El programa es coordinado por la subrutina main() en la cual se puede conocer a grandes rasgos lo que realiza el programa, debido a que contiene a todas los subprogramas principales y es la subrutina que controla el tráfico de parámetros entre ellas.

/*

```
*****  
***** UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO *****  
***** ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES ARAGON *****  
*****
```

TESIS : ANESAR, PROGRAMA PARA EL ANALISIS SISMICO DE EDIFICIOS .

Autor : Francisco Reyes Téllez .
No.Cta. : 8428671-6 .
Carrera : INGENIERIA CIVIL <87-91> .

Asesor : Ing. Amílcar Galindo Solorzano .

NOTA : Todas las unidades deben de estar en : [Ton,m,seg] .

*/

```
#include <conio.h>  
#include <stdio.h>  
#include <math.h>  
#include <alloc.h>
```

```
#define g 9.81
#define pi 3.141592654
```

```
/* Inicialización de archivos globales */
FILE *out,*inp;
void lines();
```

```
/*
```

```
.....
SUBROUTINA PRINCIPAL A.M.E.S.A.R.
.....
```

```
*/
```

```
int main() {
```

```
void entrepiso(),an_esta(),an_dina(),dta_cort();
int np,num_marcos[2],num_marcot,*rnt[2];
double Q[2],ea[5];
double *a[2],*xm[2],*K[2],*VE[2],*VD[2],*U,*H,*ke;
FILE *f_marcos;
char nom[200];
```

```
int i,;}
```

```
/*
```

```
np : Número de pisos en toda la estructura .
H[np] : Altura de pisos en Z .
K[0][np]: Rigidez total de entrepiso en X .
K[1][np]: Rigidez total de entrepiso en Y .
VE[2] : Cortantes sísmicos estáticos en direcciones X y Y .
VD[2] : Cortantes sísmicos dinámicos en direcciones X y Y .
Q[2] : Factor de ductilidad en direcciones principales X y Y .
num_marcot : Número de marcos tipo
num_marcos[2] : Número total de marcos en las direcciones X y Y .
rnt[0][num_marcos[0]] : Número de marco tipo correspondiente a cada marco en X .
rnt[1][num_marcos[1]] : Número de marco tipo correspondiente a cada marco en Y .
*/
```

```
// Abrir archivo de datos generales de la estructura.
```

```
clear();
printf(" < AMESAR > PROGRAMA PARA EL ANALISIS SISMICO DE EDIFICIOS (Compilador)\n");
printf("\n [ Versión 1.00 ] [ Autor : Reyes Téllez Francisco ] [ Junio 1992 ]");
printf("\n [ Ingeniería Civil, E.N.E.P. Aragón, U.N.A.M. ]");
printf("\n\n Archivo de datos (INPUT) : ");scanf("%s",nom);
if( (inp = fopen(nom,"r")) == NULL ) {
printf("\n\n Error, AMESAR no puede leer archivo (INPUT) !!! \n\n");
return 1;
}
printf(" Archivo de resultados (OUTPUT) : ");scanf("%s",nom);
if( (out = fopen(nom,"w")) == NULL ) {
printf("\n\n Error, AMESAR no puede escribir archivo (OUTPUT) !!!\n\n");
return 1;
}
```

```
// Lectura de descripción general de la estructura (Comentarios).
```

```
i=0;while((nom[i]=getc(inp))!=EOF && nom[i++]!=10);
if (i>=80) nom[80]='\x0'; else nom[i-1]='\x0';
fprintf(out,"DESCRIPCION GENERAL DEL EDIFICIO A ANALIZAR >>>");lines(80);
fprintf(out,"\n%s",nom);
i=0;while((nom[i]=getc(inp))!=EOF && nom[i++]!=10);
if (i>=80) nom[80]='\x0'; else nom[i-1]='\x0';
fprintf(out,"\n%s",nom);lines(80);
```

```
// Lectura de datos generales de la estructura.
```

```
fscanf(inp,"%d%d%f%f%f",&np,&num_marcot,&num_marcos[0],&num_marcos[1],&Q[0],&Q[1]);
```

```

// Dimensionamiento de variables principales.
ke = calloc(np*num_marcot,8);
H = calloc(np+1,8);
W = calloc(np+1,8);
K [0] = calloc(np+1,8);
K [1] = calloc(np+1,8);
rmt[0] = calloc(num_marcos[0],2);
rmt[1] = calloc(num_marcos[1],2);
s [0] = calloc(num_marcos[0]+1,8);
s [1] = calloc(num_marcos[1]+1,8);
xm [0] = calloc(np+1,8);
xm [1] = calloc(np+1,8);

// Lectura de datos generales de la estructura.
for(i=0; i< 3 ; i++) fscanf(inp, "%lf", &s[i]);
for(i=0; i< 2 ; i++)
    for(j=0; j<num_marcos[i]; j++) fscanf(inp, "%d", &rmt[i][j]);
for(i=1; i<np; i++) fscanf(inp, "%lf", &H[i]);
for(i=1; i<np; i++) fscanf(inp, "%lf", &W[i]);
for(i=0; i< 2 ; i++)
    for(j=1; j<num_marcos[i]; j++) fscanf(inp, "%lf", &s[i][j]);
for(i=0; i< 2 ; i++)
    for(j=1; j<np; j++) fscanf(inp, "%lf", &xm[i][j]);

// Imprime datos generales de la estructura (comprobación).
fprintf(out, "\n\nDATOS GENERALES DEL EDIFICIO >>>");
lines(55);
fprintf(out, "\nNúmero de pisos ..... : %d", np);
fprintf(out, "\nNúmero de marcos tipo ..... : %d", num_marcot);
fprintf(out, "\nNúmero de marcos en X ..... : %d", num_marcos[0]);
fprintf(out, "\nNúmero de marcos en Y ..... : %d", num_marcos[1]);
lines(55);

fprintf(out, "\n\nFACTORES DE DUCTILIDAD >>>"); lines(55);
fprintf(out, "\nFactor de ductilidad en X (Gx) [1] ..... : %8.4f", G[0]);
fprintf(out, "\nFactor de ductilidad en Y (Gy) [1] ..... : %8.4f", G[1]);
fprintf(out, "\nESPECTRO DE DISEÑO >>>"); lines(55);
fprintf(out, "\nAceleración máxima del terreno (ao) [g] .... : %8.4f", es[0]);
fprintf(out, "\nAceleración máxima max(ormax) (c) [g] ..... : %8.4f", es[1]);
fprintf(out, "\nPeriodo inferior de resonancia (Ta) [seg] .. : %8.4f", es[2]);
fprintf(out, "\nPeriodo superior de resonancia (Tb) [seg] .. : %8.4f", es[3]);
fprintf(out, "\nExponente adimensional (r) [ ] ..... : %8.4f", es[4]);

fprintf(out, "\n\nDEFINICION DE MARCOS TIPO >>>");
lines(55);
for(i=0; i<2; i++) {
    for(j=0; j<num_marcos[i]; j++)
        fprintf(out, "\nMARCO %c-%d => MARCO TIPO %d, Xci=%8.3f m.",
            'S'+i, j+1, rmt[i][j], 121-i, s[i][j+1]);
    fprintf(out, "\n");
}

fprintf(out, "\n\n");
lines(55);
fprintf(out, "\n piso W[Ton] H[m] xm[m] ym[m]");
lines(55);
for(i=np; i>=1; i--)
    fprintf(out, "\n %2d %8.2f %7.3f %7.3f %7.3f", i, W[i], H[i], xm[0][i], xm[1][i]);

// Rigideces de entrepiso para cada dirección analizada.
printf("\n\n 1. Cálculo de rigideces de entrepiso ..... ");
entrepiso(np, H, K, num_marcos, num_marcot, rmt, inp, ke);
fflush(out);

// Cierra archivo de datos generales de la estructura.
fclose(inp);

// Dimensión de vectores de fuerzas sísmicas estáticas y dinámicas.
VE[0] = calloc(np+1,8); VE[1] = calloc(np+1,8);
VD[0] = calloc(np+1,8); VD[1] = calloc(np+1,8);

```

```

// Fuerzas sísmicas estáticas en direcciones principales X y Y.
printf("ok\n 2. Cálculo de fuerzas sísmicas estáticas .... ");
fprintf(out, "\n\nANALISIS SISMICO ESTATICO >>>");
for(i=0; i<2; i++) {
    fprintf(out, "\n\nDIRECCION ANALIZADA [ %c ] : ", 88+i);
    if(num_marcos[i]>0) an_esta (np, VE[i], Q[i], K[i], W, e, H);
}
fflush(out);

// Fuerzas sísmicas dinámicas en direcciones principales X y Y.
printf("ok\n 3. Cálculo de fuerzas sísmicas dinámicas .... ");
fprintf(out, "\n\nANALISIS SISMICO DINAMICO >>>");
for(i=0; i<2; i++) {
    fprintf(out, "\n\nDIRECCION ANALIZADA [ %c ] : ", 88+i);
    if(num_marcos[i]>0) an_dina (np, VD[i], Q[i], K[i], W, ea );
}
fflush(out);

// Distribución de cortantes de entrepiso.
printf("ok\n 4. Distribución de cortantes de entrepiso .... ");
if(num_marcos[0]>0 && num_marcos[1]>0)
    dis_cort(np, num_marcos, rmt, s, xm, VE, VD, ke);

// Cierra archivo de resultados generales (OUTPUT)
fclose(out);
printf("ok\n\n Análisis finalizado .... "); getch();
printf("\n\n");

return 0;
}

```

```

/*
.....
RIGIDEZES DE ENTREPISO
.....
*/

void entrepiso (np, H, K, num_marcos, num_marcot, rmt, f_marcos, ke)
int np, num_marcos[2], num_marcot, *rmt[2];
double *H, *K[2], *ke;
FILE *f_marcos;

{

void rig();
FILE *f_carga, *f_gen;
int i, j, k, kk, nbt;
double *kke;
float *BB[3];

kke = calloc(np-1, 8);

fscanf(f_marcos, "%d", &nbt);
for(i=0; i<3; i++) BB[i] = calloc(nbt, 4);
fprintf(out, "\n\n\nELEMENTOS BARRA TIPO >>>");
línea(55);
fprintf(out, "\nELEM-TIPO E[Ton/m2] AREA[m2] Iz[m6]");
línea(55);

/* Lectura de propiedades mecánicas y geométricas */
for(i=0; i<nbt; i++) {
    fscanf(f_marcos, "%d", &k);
    fscanf(f_marcos, "%f%f%f", &BB[0][i], &BB[1][i], &BB[2][i]);
    fprintf(out, "\n %3d %12.2f %7.4f %9.6f", k, BB[0][i], BB[1][i], BB[2][i]);
}
}

```

```

/* Rigideces de entrepiso para cada marco tipo */
fprintf(out, "\n\n\nRIGIDECES DE ENTREPISO DE MARCOS TIPO >>>");

for(i=1; i<=num_marcot; i++) {

    /* Rigideces de entrepiso */
    rig(f_marcos, f_carga, f_gen, 2, H, np, ke, BB);
    k = np*(i-1);
    for(j=k; j<=np; j++) ke[j]=kke[j-k+1];

    /* Rigideces totales de entrepiso */
    for(j=0; j<2; j++)
        for(k=0; k<=num_marcos[j]; k++)
            if(i==num[j][k]) for(kk=1; kk<=np; kk++)
                K[j][kk]+=kke[kk];

    /* Impresión de resultados (OUTPUT) de rigideces de entrepiso de marco tipo */
    fprintf(out, "\n\nMARCO TIPO      [ Xd ]", i);
    líneas(25);
    fprintf(out, "\nEntrepiso      RIG(Ton/m)");
    líneas(25);
    for(j=np; j>=1; j--) fprintf(out, "\n      X2d      X12.4f      ", j, kke[j]);
}

/* Impresión de resultados (OUTPUT) de rigideces totales de entrepiso */
fprintf(out, "\n\n\nRIGIDECES TOTALES DE ENTREPISO >>>");
líneas(41);
fprintf(out, "\nEntrepiso      RIG-X(Ton/m)      RIG-Y(Ton/m)");
líneas(41);
for(j=np; j>=1; j--)
    fprintf(out, "\n      X2d      X12.4f      X12.4f", j, K(0)[j], K(1)[j]);
}

```

```

/*
*****
MÉTODO GENERAL DE LAS RIGIDECES PARA EL ANÁLISIS DE MARCOS PLANOS
*****
*/

```

```

void rig(a_proy, a_carg, a_gen, opcion, H, np, ke, BB)
FILE *a_proy, *a_carg, *a_gen;
int opcion, np;
double *H, *ke;
float *BB[3];

```

```

/*
fproy.est (a_proy), archivo secuencial de solo lectura ASCII .
Define las características geométricas y mecánicas de la estructura :

nr      : Número total de nodos en la estructura .
nnp     : Número total de nodos restringidos en la estructura .
nb      : Número total de elementos barras .
X[1..nr] : Coordenadas en el eje X del nodo i .
Y[1..nr] : Coordenadas en el eje Y del nodo i .
NMR[1..nnp] : Número de nodo restringido .
GNR[1..nnp] : Grado de libertad de nodos restringidos .
A[1..nb] : Nodo inicial A al cual se conecta la barra j .
B[1..nb] : Nodo final B al cual se conecta la barra j .
G[0..nb*4] : Rigideces angulares y rigidez axial del elemento barra j .

```

```

fproy.car (a_carg), archivo secuencial de solo lectura ASCII .
Contiene todas las condiciones de carga (Fuerzas Externas) :

nc      : Número de condiciones de carga .

```


Los siguientes datos se requieren para cada condición de carga (nc):

nnc : Número de nodos cargados .
DMC[1..nnc] : Direcciones de los nodos cargados .
FEN[3][1..nnc]: Fuerzas externas concentradas en los nodos .

fproy.gen (a_gen), archivo secuencial de solo escritura ASCII .
Almacena los resultados generados por las distintas fuerzas externas :

opcion : Variable de decisión, para almacenar distintos resultados .
nn : Número de nodos .
nb : Número de barras .
ds : Número de desplazamientos externos por nodo .
ab : Ancho de banda, para reducir la Matriz [K] .
K[ab*ds*]: Matriz de Rigidez Global de la Estructura .
nc : Número de condiciones de carga .
D[1..3*nn]: Desplazamientos externos por nodo .
P[1..6*nb]: Elementos mecánicos por cada miembro .

*/

C

```
int nn,nnr,nb,*NHR,*GMR,*A,*B,nc,*pose;
float *X,*Y,*G,*p,*YI,HR,FF,VV,sum,E,AA,I;
void shell();

double *K,*ap,*ccp,*ok,*q,t[11],dd,ee,l,c,s,w,x,y,z;

int *L,l[1,1,1,1],k,a,b,ab=1,ds=1,
    ll,kk,cc,e,d,*nd,*nk,*N,N[7],
    M[22]={0,8,10,1,8,10,2,9,3,10,9,4,7,1,3,6,8,10,2,9,4,5},
    S[22]={0,1,1,1,-1,-1,1,1,-1,-1,-1,-1,1,1,-1,1,1,1,1,1,1,1};

/* Lectura de información general del marco plano analizado */
fscanf(a_proy,"%d%d",&nn,&nnr,&nb);

/* Lectura de datos de nodos en la estructura */
X = calloc(nn+1,4);
Y = calloc(nn+1,4);
for(i=1;i<=nn;i++) fscanf(a_proy,"%f%f",&X[i],&Y[i]);

/* Lectura de datos de nodos restringidos */
NHR=calloc(nn+1,2);
GMR=calloc(nn+1,2);
for(i=1;i<=nnr;i++) fscanf(a_proy,"%d%d",&NHR[i],&GMR[i]);

/* Cálculo del número global de desplazamientos (ds) en la estructura */
L = calloc(nn+1,2);
nd = calloc(nn+2,2);
for(i=1;i<=nn;i++){L[i][1]=7;
for(l=1;l<=nnr;l++){L[NHR[l]]+=GMR[l];
free(NHR);free(GMR);
for(i=1;i<=nn;i++){
nd[i][ds]=L[i][1]&&L[i][2]&&L[i][3];
}

/* Lectura de información de elementos barra */
A=calloc(nb+1,2);
B=calloc(nb+1,2);
G=calloc(nb+1,4);
for(i=1;i<=nb;i++){
fscanf(a_proy,"%d%d",&A[i],&B[i],&G[i]);
a=A[i];b=B[i];c=X[b]-X[a];s=Y[b]-Y[a];l=sqrt(c*c+s*s);
E = BB[0][i-1];
AA = BB[1][i-1];
I = BB[2][i-1];
G[4*i-4]=AA*E/l;
G[4*i-3]=4*E*I/l;
G[4*i-2]=2*E*I/l;
G[4*i-1]=4*E*I/l;
}
)
```

```

/* Cálculo del ancho de banda (ab) */
nd[nn+1]=ds;
for(i=1;i<=nb; i++) {
    a=A[i];b=B[i];
    if(a>b) (d=a;a=b;b=d);
    a=nd[b+1]-nd[a];if(a>b)ab=a;
}

K=calloc(--ds*ab+1,B);
if(!K) { printf("\n memoria insuficiente !!!");exit(); }

/* Ensamble de la matriz de rigidez global (K) de la estructura */
ab--;
for(i=1;i<=nb; i++) {

    /* Calculo de la matriz de rigidez de barra (HRA) */
    p=EG*(4*-2);q=t;
    a=A[i];b=B[i];c=X[b]-X[a];s=Y[b]-Y[a];l=sqrt(c*c+s*s);a/=l;c/=l;
    w=p*p[l];
    x=p*p[l];
    y=(w+x)/l/l;
    **+q*w*s/l; **+q*w*s/l;
    **+q*w*c/l; **+q*w*c/l;
    **+q*w*p; **+q*w*-p; **+q*w*-p;
    z=c*c*(*-p-y);
    **+q* z+y;
    **+q* z+p;
    **+q* c*s*(p-y);
    e=d=0;
    for(j=0;j<3;j++) {
        H[j+1]=(l[a]#1<)<)>nd[a]+e++*0;
        H[j+4]=(l[b]#1<)<)>nd[b]+d++*0;
    }

    cc=0;Hj=N;
    for(j=1;j<7;j++) {
        if(*+Nj) {
            Nk=Nj-1;
            for(k=j;k<7;k++)if(*+Nk) {
                if((k+k=Nj)>{(l=Nk)} { kk=l;{l=Nj};
                    K[{l=ab+kk}]=S[d+cc+k-j+1]*c[M[d]];
                }
            }
            cc+=7-j;
        }
    }
    ab++;

    /* Prepara archivo de salida de datos (OUTPUT) */
    if(opcion&B) fprintf(a_gen," %d %d %d ",opcion,nn,nb);

    /* Imprime matriz de rigidez global (K) de la estructura */
    if(opcion&I || opcion&B) {
        fprintf(a_gen,"\n\nrd %d\n",ds,ab);
        for(i=0;i<=ab*ds+1;i++) fprintf(a_gen,"\nX10.2f",K[i]);
        fflush(a_gen);
    }

    /* Factoriza a la matriz de rigidez global (K) en la matriz (L) */
    for(i=1;i<=ds;i++) {
        kk=i-ab;cc=i*ab-ab;ll=(kk<0)?kk+1:1;
        for(j=ll;j<=ab;j++) {
            y=0;d=(j*kk)*ab;ccp=K+ll+cc;ok=K+d-1;
            for(ap=K+ll+d-j;ap<ok;y=*ccp++*ap++);
            x=K[cc+1]-y;
            K[cc+1]=(j=ab)?sqrt(x):x/K[d];
        }
    }
}

```

/* EVALUACION DE LAS DISTINTAS CONDICIONES DE CARGA */

C

```

int nc, nnc, DNC;
double FEN(3), *F, *D, G(3);

if(opcion&B) {
    facanf(a_carg, "%d ", &nc);
    fprintf(a_gen, "\n\nrd%d ", nc);
} else nc=1;

D = calloc(ds+1, 8);

for(i=1; i<=nc; i++) {

    F = calloc(ds+1, 8);

    if(opcion&B) {

        /* Condición de fuerzas externas extraídas de archivo (INPUT) a_carg */
        facanf(a_carg, "%d ", &nnc);

        /* Vector de fuerzas externas concentradas en los nodos */
        if(nnc>0)
            for(i=1; i<=nnc; i++) {
                facanf(a_carg, "%X%X|fX|f", &DNC, &FEN(0), &FEN(1), &FEN(2));
                d=0; for(j=0; j<3; j++) if(L[DNC]&1<<j) F[ind[DNC]+d+]=FEN[j];
            }

    } else {

        /* Condición de fuerzas externas para la obtención de rigideces de entrepiso */
        YY = calloc(nw+1, 4);
        posc = calloc(nw+1, 2);
        for(i=1; i<=nw; i++) { YY[i]=Y[i]; posc[i]=1; }

        shell (&YY[i], &posc[i], nw);

        /* Calcula la máxima altura del marco en donde se concentran masas de entrepiso */
        for(i=nw; i>=1; i--)
            for(j=nw; j>=1; j--)
                if(YY[i]==N[j]) { NR=N[j]; i=0; break; }

        /* Cálculo de la distribución de fuerzas para obtener rigideces de entrepiso */
        k=1;
        for(i=1; i<=nw; i++) {
            for(j=i+1; j<=nw; j++) if (YY[i]!=YY[j]) break;
            FF = 100*YY[i]/NR/(j-i);
            if (N[i]==YY[i])
                for(k=i; k<=j-1; k++) if (L[posc[k]]&1) F[ind[posc[k]]]=FF;
            else
                if (N[i]>YY[i]) i--; else {i=k; continue; }
            i=j;
        }

    }

}

/* METODO DE CHOLESKY PARA LA SOLUCION DE ECUACIONES SIMULTANEAS */

/* Solución complementaria (Z) mediante sustitución progresiva */
for(i=1; i<=ds; i++) {
    kk=i-ab; cc=i*ab-ab; ll=(kk<0)?-kk+1:1;
    ccp=cc+ll*cc; ok=0+kk*ab-1; y=0;
    for(ap=0+kk*ll; ap<=ok; y+=*ccp+*ap+);
    D[i]=(F[i]-y)/K[i*ab];
}

/* Solución general (D) mediante sustitución regresiva */
for(i=ds; i>0; i--) {
    kk=i-ab; cc=i*ab-ab; ll=(kk<0)?-kk+1:1;
    x=0; [j]=K[i*ab]; ccp=K+ll*cc; ok=0+kk*ab-1;
    for(ap=0+ll*kk; ap<=ok; x+=*ccp+*a);
}

```

```

/* Almacena el vector de desplazamientos (D) en archivo secuencial a_gen */
if(opcion&2 && opcion&8) {
  fprintf(a_gen, "\n\n\n");
  for(i=1; i<=n; i++)
    for(j=0; j<3; j++) {
      w=(L[i]&1<<j)*70/((+e)*i);
      fprintf(a_gen, "%r%l(%i, w);
    }
}

/* Rigideces de entrelaso */
if(!opcion&8) {
  /* Desplazamientos absolutos */
  for(i=1; i<=n; i++) ke[i]=0;
  i=k=1;
  for(i=1; i<=n; i++) {
    sum=0;
    for(j=i; j<=n; j++)
      if (Y[i][j]==Y[j][j]) break; else
        if(L[poic[j]][i]&1) sum+=D[nd[poic[j]]];
    sum/=(-1);
    if (H[i]==Y[i][i])
      ( ke[i]=sum; k++; )
    else
      if (H[i]>Y[i][i]) i--; else { i=k; continue; }
    i++;
  }

  /* Desplazamientos relativos y rigideces de entrelaso */
  VV=100; H[D]=ke[D]=0;
  for(i=1; i>=1; i--) if( ke[i]!=0 )
    for(j=i-1; j>=0; j--)
      if( ke[j]!=0 || j==0 ) {
        ke[j]=VV/(ke[i]-ke[j]);
        VV+=H[j]=100/HH;
        i=j+1;
        break;
      }
}

/* Calcula y almacena los elementos mecánicos de cada barra */
if(opcion&4 && opcion&5) {
  fprintf(a_gen, "\n\n\n");
  for(i=1; i<=nb; i++) {
    a=A[i]; b=B[i]; c=d=0;
    c=X[b]-X[a]; s=Y[b]-Y[a]; l=sqrt(c*c+s*s); s/=l; c/=l;
    for(j=0; j<3; j++) {
      M[j+1]=(L[a]&1<<j)*70/nd(a)+e+1;0;
      N[j+4]=(L[b]&1<<j)*70/nd(b)+d+1;0;
    }
    D[0]=0; p=&G[4*(i-1)];
    w=D[N[4]]-D[N[1]];
    x=D[N[5]]-D[N[2]];
    y=(x*c-u*s)/L+D[N[6]];
    z=y+D[N[3]]-D[N[6]];
    Q[0]=(x**a+w*c)**p;
    Q[1]=z**p[1]+y**p[2];
    Q[2]=z**p[2]+y**p[3];
    fprintf(a_gen, "\n%l(%i X%f X(%i, Q[0], Q[1], Q[2]);
  }
}
free(F);
}

```

```

/*
.....
ANALISIS SISMICO ESTADICO
.....
*/

void an_esta(n,V,Q,K,W,es,H)
int n;
double *V,Q,*W,*K,*es,*H; /* W[1-n],K[1-n],es[0-4],H[1-n] */

{
    int i;
    double *fse,*DT,*DR,*a;
    double cQ,Wt,WH,DTT,VV,an;

    /* VARIABLES UTILIZADAS EN SUBPROGRAMA

    n : Número de pisos de la estructura .
    Q : Factor de ductilidad en función del tipo de estructura y
        de los requisitos de diseño por cada dirección analizada.
    W[n] : Peso de los entrepisos concentrados en cada uno de los n pisos .
    K[n] : Rigideces totales del entrepiso i, en la dirección del análisis .
    es[0]=a0: Aceleración máxima del terreno, como fracción de g .
    es[1]=c : Aceleración máxima maximum de las estructuras, como fracción de g .
    H[n] : Altura de los n piso .

    V[n] : cortantes de entrepiso .
    DT[n] : desplazamientos absolutos de cada piso .
    DR[n] : desplazamientos relativos de cada piso .
    fse[n] : fuerzas sísmicas actuantes en cada pisos .
    a[n] : coeficientes de cada piso .
    an : coeficiente en piso n .
    Wt : peso total del edificio .
    cQ : factor c/Q, siempre mayor que a0 .

    NOTA: Todas las demás variables que no están indicadas en esta lista
    son variables auxiliares, que no tienen significado en el
    Análisis Estructural.

    */

    fse = calloc(n+1,B);
    DT = calloc(n+1,B);
    DR = calloc(n+1,B);
    a = calloc(n+1,B);

    cQ=es[1]/Q; if(cQ<es[0])cQ=es[0];
    Wt=WH=VV=DTT=0;

    for(i=1;i<=n;i++) {
        Wt+=W[i];
        WH+=W[i]*H[i];
    }
    an=Wt*cQ/WH;

    for(i=n;i>0;i--) {
        a [i] = an*H[i];
        fse[i] = a[i]*W[i];
        V [i] = VV += fse[i];
        DR [i] = V[i]*Q/K[i];
    }
    for(i=1;i<=n;i++) DT[i]=DTT+DR[i];

    /* Impresión de resultados (OUTPUT) de análisis sísmico estático */
    linea(76);
    fprintf(out,"\nEntrepiso DESPL.REL[m] FUERZA [Ton] Piso DESPL.ABS[m] CORTANTE[Ton]");
    linea(76);
    for(i=n;i>=1;i--)
        fprintf(out,"\n %2d %7.4f %5.2f %2d %7.4f %5.2f",
            i,DR[i],fse[i],i,DT[i],V[i]);
}

```

3

```

/*
*****
ANALISIS SISMICO DINAMICO
*****
*/

void an_dina(n,VT,Q,K,W,es)
int n;
double *VT,Q,*W,*K,*es; /* W[1-n],K[1-n],es[0-4] */

{
register int i,j;
int verifica();

double *f,*V,*ym,*Dr,*T,*w2,*m,*OO,*A,*C,*fsd,*D,*DT;
double hz(),bisec();
double s,s2,cc;

/* VARIABLES UTILIZADAS EN SUBPROGRAMA

n : Número de pisos de la estructura .
Q : Factor de ductilidad en función del tipo de estructura y
de los requisitos de diseño por cada dirección analizada.
W[n] : Peso de los entrepisos concentrados en cada uno de los n pisos .
K[n] : Rigideces totales del entrepiso i, en la dirección del análisis .
es[0]=a0 : Aceleración máxima del terreno, como fracción de g .
es[1]=ac : Aceleración máxima maximum de las estructuras, como fracción de g .
es[2]=Ta : Período que limita el límite inferior del intervalo de resonancia .
es[3]=Tb : Período que limita el límite superior del intervalo de resonancia .
es[4]=r : Exponente adimensional .

C [n] : Factores de participación .
A [n] : Aceleración máxima elasto-plástica .
OO [n] : Factores de ductilidad para los n períodos .
VT [n] : Cortantes totales de entrepiso .
fsd[n] : Fuerzas en cada piso, por sismo .
D [n] : Desplazamientos relativos .
DT [n] : Desplazamientos totales relativos .
w2 [n] : Frecuencia característica del modo n .
T [n] : Períodos de vibración de cada modo .
W [n] : Peso total del piso i .
ym[n][n] : Desplazamientos máximos de los entrepisos en cada modo .
Dr[n][n] : Desplazamientos relativos de los entrepisos en cada modo .
V [n][n] : Cortantes de entrepiso para cada modo .
f [n][n] : Ley normalizada de los n modos .

NOTA: Todas las demás variables que no estén indicadas en esta lista
son variables auxiliares, que no tienen significado en el
Análisis Estructural.

*/

V = calloc(n*(n+1),8);
ym = calloc(n*(n+1),8);
Dr = calloc(n*(n+1),8);
f = calloc(n*(n+1),8);

OO = calloc(n+1,8); A = calloc(n+1,8);
C = calloc(n+1,8); fsd = calloc(n+1,8);
D = calloc(n+1,8); DT = calloc(n+1,8);
m = calloc(n+1,8); T = calloc(n+1,8);
w2 = calloc(n+1,8);

for(i=1;i<=n;i++) m[i]=W[i]/g;

/* 1) Cálculo de valores característicos por el método de Hozier. */
{
double dw2,dm2,dwb2,Ra,Rb,delta,V,F;
int dd,ee;
ee=1;
delta=10;
dw2=delta;

```

```

w2(0)=dwa2*0.1;
Rb=hz(dwa2,n,m,K);
do (
  dwb2=dwa2*dw2;
  Rb=hz(dwb2,n,m,K);
  if(Ra>Rb<0) (
    w2(ee)=blsec(dwa2,dwb2,n,m,K);
    if(verifica(w2(ee),ee,n,m,K)) (
      dw2=delta;
      dd=n*ee+ee;
      T(ee)=2*pi/sqrt(w2(ee));
      f(dd+1)=1;
      V=K(1);
      Fm(1)=w2(ee);
      for(i=2;i<n;i++) (
        V=V*f;
        f(dd+i)=V/K(i)+f(dd+i-1);
        Fm(i)=w2(ee)*f(dd+i);
      )
      for(i=1;i<n;i++) f(dd+i)/=f(dd+n);
      dwb2=w2(ee+)*dwb2;
    ) else (
      dw2/=2;dwb2=w2(ee-1)+dwb2;
    )
    Rb=hz(dwb2,n,m,K);
  )
  dwa2=dwb2;Ra=Rb;
) while (n>=ee);
)

/* 3) Cálculo de las aceleraciones máximas elasto-plásticas para el cálculo
de las fuerzas. */
for(i=1;i<n;i++) (
  if ( T(i) <= es[2] ) ( Q0(i)=1+T(i)/es[2]*(0-1);cc=es[0]+(es[1]-es[0])/es[2]*T(i); )
  if ( T(i) > es[2] && T(i) <= es[3] ) ( Q0(i)=0;cc=es[1]; )
  if ( T(i) > es[3] ) ( Q0(i)=0;cc=es[1]*pow(es[3]/T(i),es[4]); )
  A(i)=cc/Q0(i); /* Aceleración como fracción de g */
  if ( A(i) < es[0] ) A(i)=es[0];
  A(i)*=g; /* Aceleración en el espectro elasto-plástico */
)

/* 4) Cálculo de los factores de participación. */
for(i=1;i<n;i++) (
  s=s2+0;
  for(j=1;j<n;j++) (
    s +=w(j)*f(i*n+i+j);
    s2+=w(j)*pow(f(i*n+i+j),2);
  )
  C(i)=s/s2;
)

/* 5) Desplazamientos máximos de los pisos en cada modo. */
for(i=1;i<n;i++) (
  s=A(i)/w2(i)*C(i);
  for(j=1;j<n;j++) ym((n+1)*i+j)=s*f(i*n+i+j);
)

/* 6) Desplazamientos relativos de los entrepisos en cada modo. */
for(i=1;i<n;i++) (
  ym((n+1)*i+0)=0;
  for(j=n;j>1;j--) Dr[(n+1)*i+j]=ym((n+1)*i+j)-ym((n+1)*i+j-1);
)

/* 7) Cortantes de entrepiso para cada modo. */
for(i=1;i<n;i++) (
  for(j=1;j<n;j++) V((n+1)*i+j)=K(j)*Dr[(n+1)*i+j];
)

/* 8) Cortantes totales de entrepiso. */
for(i=1;i<n;i++) (
  VT(i)=0;for(j=1;j<n;j++) VT(i)+=pow(V((n+1)*i+j),2);
  VT(i)=sqrt(VT(i));
)

```

```

/* 9) Fuerzas en cada piso, por sismo. */
fsd[n]=VT[n];
for(i=n-1;i>=1;i--) fsd[i]=VT[i]-VT[i+1];

/* 10) Desplazamientos relativos y totales relativos. */
for(i=1;i<=n;i++) {
  a=0;
  for(j=1;j<=n;j++) s+=pow(OQ[j]*Dr[(n+1)*j+1],2);
  D[i]=sqrt(s);
  if(i>1) DT[i]=D[i]+DT[i-1]; else DT[i]=D[i];
}

/* Impresión de resultados (OUTPUT) de análisis sísmico dinámico */
l/wra(76);
fprinf(fout,"vEntrepiso DESPL.REL[m] FUERZA [ton] Piso DESPL.ABS[m] CORTANTE[ton]";
l/wra(76);
for(i=n;i>=1;i--)
  fprinf(fout,"un      x2d      x7.4f      x8.2f      x2d      x7.4f      x8.2f",
    ,D(i),fsd(i),DT(i),VT(i));

```

```

/*
.....
DISTRIBUCION DE CORTANTES DE ENTREPISO
.....
*/

```

```

void   dia_cort(np,num_marcos,rnt,e,xm,VE,vD,ke)
double *s[2],*xm[2],*VE[2],*vD[2],*ke;
int    np,*num_marcos,*rnt[2];

```

```

double *xv[2],*Jj,*xt[2],*J[2],*kt[2],*fd[2],*vd[2],
       *e1[2],*e2[2],*vv1[2],*vv2[2],*kxy[2],*bx[2];

```

```

/* NOTA: bx, son datos externos que son definidos en esta subrutina */

```

```

double ec;
int     i,j,k,kk,jj;
double a1,a2,a3;
double vvd,vt1,vt2;

```

```

/* La separación entre marcos debe estar en orden ascendente. */

```

```

bx[0]=s[1][num_marcos[1]];
bx[1]=s[0][num_marcos[0]];

```

```

kxy[0]=calloc(np*num_marcos[0],8);kxy[1]=calloc(np*num_marcos[1],8);
vv1[0]=calloc(np*num_marcos[0],8);vv1[1]=calloc(np*num_marcos[1],8);
vv2[0]=calloc(np*num_marcos[0],8);vv2[1]=calloc(np*num_marcos[1],8);

```

```

Jj =calloc(np+1,8);
xt[0]=calloc(np+1,8);xt[1]=calloc(np+1,8);
xv[0]=calloc(np+1,8);xv[1]=calloc(np+1,8);
J [0]=calloc(np+1,8);J [1]=calloc(np+1,8);
kt[0]=calloc(np+1,8);kt[1]=calloc(np+1,8);
e1[0]=calloc(np+1,8);e1[1]=calloc(np+1,8);
e2[0]=calloc(np+1,8);e2[1]=calloc(np+1,8);
vd[0]=calloc(np+1,8);vd[1]=calloc(np+1,8);
fd[0]=calloc(np+1,8);fd[1]=calloc(np+1,8);

```

```

/* Cálculo de la superposición de fuerzas sísmicas estáticas y fuerzas dinámicas */

```

```

for(i=0;i<2;i++) {
  for(j=np;j>=0;j--) {
    vd[i][j]=VD[i][j]<0.80*VE[i][j] ? 0.80*VE[i][j] : VD[i][j];
    fd[i][j]=Cj*np ? vd[i][j]-vd[i][j+1] : vd[i][j];
  }
}

```



```

/* Cálculo del centro de cortante */
for(i=0; i<2; i++) (
  s1=fd(i) [np]*xm(i) [np];
  s2=fd(i) [rp];
  xv(i) [np]=xm(i) [np];
  for(j=np-1; j>0; j--) (
    s1=fd(i) [j]*xm(i) [j];
    s2+=fd(i) [j];
    xv(i) [j]=s1/s2;
  )
)

/* Cálculo del centro de torsión */
for(i=0; i<2; i++)
  for(j=1; j<np; j++) (
    s1=s2=0;
    for(k=0; k<num_marcos(i); k++) (
      kk=(rnt(i) [k]-1)*rp-1;
      s1+=kk[k+1];
      s2+=kk[k+1]*s(i) [k+1];
    )
    kt(i) [j] = s1;
    xt(i-1) [j] = s2/s1;
  )
)

/* Cálculo de Excentricidades Reglamentarias */
for(i=0; i<2; i++)
  for(j=1; j<np; j++) (
    ec=xv(i) [j]-xt(i) [j];
    s1=(ec>0)?1:-1;
    ec=fabs(ec);
    e1(i) [j]=(1.5*ec+0.10*bx(i))*s1;
    e2(i) [j]=(1.0*ec-0.10*bx(i))*s1;
  )
)

/* Cálculo del momento polar de inercia */
for(i=0; i<2; i++)
  for(j=1; j<np; j++) (
    s1=s2=0;
    for(k=0; k<num_marcos(i); k++) (
      kk = (rnt(i) [k]-1)*rp-j-1;
      jj = k*rp-j-1;
      s1 = s(i) [k+1]-xt(i-1) [j];
      kxy(i) [j]=kk[kk]*s1;
      s2 += kk[kk]*s1*s1;
    )
    J(i) [j]=s2;
  )
  for(j=1; j<np; j++)
    JJ[j]=J(0) [j]+J(1) [j];
)

/* Distribución de cortantes */
for(i=0; i<2; i++)
  for(j=1; j<np; j++) (
    s1=vd(i) [j]/kt(i) [j];
    s2=vd(i) [j]/JJ[j]*e1(i-1) [j];
    s3=vd(i) [j]/JJ[j]*e2(i-1) [j];
    for(k=0; k<num_marcos(i); k++) (
      kk = (rnt(i) [k]-1)*rp-j-1;
      jj = k*rp-j-1;
      vvd = s1*ke [kk]; vt1 = s2*kxy(i) [j]; vt2 = s3*kxy(i) [j];
      if (vt1>vt2) vvd+=vt1; else vvd+=vt2;
      vt(i) [j] = vvd;
    )
    for(k=0; k<num_marcos(i-1); k++) (
      jj = k*rp-j-1;
      vt1 = fabs(s2*kxy(i-1) [j]);
      vt2 = fabs(s3*kxy(i-1) [j]);
      vvd = (vt1>vt2) ? vt1 : vt2;
      vvd(i-1) [j] = vvd;
    )
  )
)

```

```

/* Combinación de direcciones */
for(i=0;i<2;i++)
  for(j=1;j<np;j++) {
    for(k=0;k<num_marcos[i];k++) {
      jj = k*np+i-1;
      vt1 = vv1[i][j][j];
      vt2 = vv2[i][j][j];
      vv1[i][j][j] = (vt1>vt2) ? vt1+0.30*vt2 : vt2+0.30*vt1;
    }
  }

// IMPRESION DE RESULTADOS (OUTPUT)
fprintf(out, "\n\n\nSUPERPOSICION DE FUERZAS SISMICAS ESTATICAS Y DINAMICAS >>>");
linea(76);
fprintf(out, "\nEntrepiso      Vx[Ton]      Vy[Ton]      Piso      Fx[Ton]      Fy[Ton]");
linea(76);

for(j=np;j>0;j--)
  fprintf(out, "\n      X3d      X8.2f      X8.2f      X3d      X8.2f      X8.2f",
    j, vd[0][j], vd[1][j], j, fd[0][j], fd[1][j]);
fprintf(out, "\n\n\nCENTROS DE >>>");
linea(76);
fprintf(out, "\n      [ MASA      ]      [ CORTANTE ]      [ TORSION      ]");
fprintf(out, "\n      xm[m]      ym[m]      xv[m]      yv[m]      xt[m]      yt[m]");
linea(76);

for(j=np;j>0;j--)
  fprintf(out, "\nX3d      X7.3f      X7.3f      X7.3f      X7.3f      X7.3f      X7.3f",
    j, xm[0][j], xm[1][j], xv[0][j], xv[1][j], xt[0][j], xt[1][j]);

fprintf(out, "\n\n\nCORTANTES SISMICOS DE DISEÑO >>>");
for(i=0;i<2;i++) {
  fprintf(out, "\n\nDIRECCION ANALIZADA [ Xc ] :=", i+88);
  for(j=0;j<num_marcos[i];j++) {
    fprintf(out, "\n");
    linea(32);
    fprintf(out, "\nMARCO      [ Xc-X-2d] \nEntrepiso      Vxc[Ton]      Fxc[Ton]",
      i+88, i+1, i+120, i+120);
    linea(32); vvd=0;
    for(k=np;k>=1;k--) {
      jj = j*np+k-1;
      vvd = vv1[i][j][j]-vvd;
      fprintf(out, "\n      X-2d      X8.2f      X8.2f", k, vv1[i][j][j], vvd);
    }
  }
}

}

void linea(int num) {
  static char cad1[90]={"-----"};
  char cad2[90];
  strncpy(cad2, cad1, num); cad2[num]='\n0';
  fprintf(out, "\nXa", cad2);
}

```

```

void shell(item, posc, cont)
float *item;
int cont, *posc;

{
    register int i, j, k, s, w;
    float x, y;
    int a[5] = {0, 5, 3, 3, 1};

    for(w=0; w<5; w++) {
        k=a[w]; s=-k;

        for(i=k; i<cont; ++i) {
            x=item[i];
            y=posc[i];
            j=i-k;

            if(s==0) {
                s=-k;
                s++;
                item[s]=x;
                posc[s]=y;
            }

            while(j>=0 && x<item[j] && j<=cont) {
                item[j+k] = item[j];
                posc[j+k] = posc[j];
                j=j-k;
            }

            item[j+k]=x;
            posc[j+k]=y;
        }
    }
}

```

```

int verifica(w, mm, n, m, K)
double w, *m, *K;
int mm, n;

{
    register int i;
    int nm=1;
    double y, ya=1, V, F;

    V=K[1]; F=m[1]*w;

    for(i=2; i<=n; i++) {
        V=F;
        y = V/K[1]+ya;
        F = m[1]*w*y;
        if(y*ya<0) nm++;
        ya=y;
    }

    return (nm==mm);
}

```

```

double biseq(a,b,n,m,K)
double a,b,*m,*K;
int n;

{
    double c,fa,fb,fc,hz();
    fa=hz(a,n,m,K);
    fb=hz(b,n,m,K);

    do{
        c=(a+b)/2; fc=hz(c,n,m,K);
        if(fa*fc<0) (b=c;fb=fc); else (a=c;fa=fc);
        c=fa*(b-a)/(fa-fb)+a; fc=hz(c,n,m,K);
        if(fa*fc<0) (b=c;fb=fc); else (a=c;fa=fc);
    } while( (fc>0?fc:-fc) > 0.000001 );

    return fa*(b-a)/(fa-fb)+a;
}

```

```

double hz(w,n,m,K)
double w,*m,*K;
int n;

{
    double y=1,V,F;
    int i;

    V=K[1]; F=m[1]*w;
    for(i=2;i<=n;i++) {
        V=F;
        y=V/K[i];
        F = m[i]*w*y;
    }
    return F-V;
}

```

APENDICE B

MANUAL DEL USUARIO

■ INFORMACION DE ENTRADA DE DATOS :

El archivo de entrada de datos (INPUT) debe de estar escrito en formato ASCII, es el formato más general que se emplea en la plataforma basada en PC-IBM, XT, AT, PS/2 y computadoras 100% compatibles.

En el lenguaje de programación C la entrada de datos (INPUT) es realizada con formato de lectura libre, a diferencia de FORTRAN en donde generalmente se tiene formato definido. La lectura de datos con formato libre significa que no hay restricción en cuanto a formatos, campos y líneas o tarjetas. El único dato que toma en cuenta la entrada de datos con formato son las dos primeras líneas del archivo de datos, éstas primeras dos líneas deben contener una breve descripción del edificio a analizar.

Debido a que la entrada de datos (INPUT) es realizada con formato libre, el Lenguaje C distingue un dato de otro al detectar las siguientes condiciones:

- Espacio en blanco.
- Coma
- Retorno de carro.

La utilización del formato libre facilita la captura de información y disminuye la posibilidad de errores en la entrada de datos.

Se recomienda respetar las unidades [Ton,m,seg] que se presentan para cada dato en particular, debido a que los formatos de salida están centrados y formateados para las magnitudes que usualmente se generan en esas unidades. Esto es indispensable debido a que el programa ANESAR contiene constantes en éstas unidades, por lo que se generarían errores numéricos si se cambian las unidades.

No obstante el formato libre, el archivo de datos (INPUT) evidentemente debe de seguir una secuencia ordenada de entrada de datos como a continuación se indica:

DATOS GENERALES :

Datos que definen las características generales del edificio a analizar.

- [1] Primeras dos líneas:
Descripción de la estructura, con un máximo por línea de 80 caracteres.
(Único dato con formato)
- [2] Número de pisos.
- [3] Número de marcos tipo.
- [4] Número de marcos (elementos resistentes) en la dirección X [num_marcosX].
- [5] Número de marcos (elementos resistentes) en la dirección Y [num_marcosY].

FACTORES DE DUCTILIDAD :

Debido a que un edificio puede ser modelado con distintos tipos de estructuración en sus direcciones principales X y Y, se pueden dar distintos tipos de factores de ductilidad (factor de comportamiento sísmico) para cada dirección principal analizada.

- [1] Factor de ductilidad (Q_x) en la dirección X.
- [2] Factor de ductilidad (Q_y) en la dirección Y.

ESPECTRO DE DISEÑO :

Los coeficientes del espectro de diseño son obtenidos generalmente de reglamentos o normas de diseño sísmico para el sitio en donde se desplanta la estructura, son los parámetros que introducen la intensidad al edificio analizado.

- [1] Aceleración máxima del terreno (a_0) como fracción de la aceleración de la gravedad g ($9.81\text{m}/\text{seg}^2$).
- [2] Aceleración máxima maximorum (c) como fracción de la aceleración de la gravedad g ($9.81 \text{ m}/\text{seg}^2$).
- [3] Período inferior (T_a) que limita el intervalo de resonancia [seg].
- [4] Período superior (T_b) que limita el intervalo de resonancia [seg].
- [5] Exponente adimensional (r) [].

DEFINICION DE MARCOS TIPO :

Debido a la introducción del concepto de **MARCO TIPO** se requiere definir para cada dirección analizada el número de **MARCO TIPO** (marco plano) que corresponde al marco plano real en esa dirección.

Dirección X:

En orden progresivo se requiere introducir el MARCO TIPO que representa al marco plano en X [1..num_marcosX]

[1] Número i de MARCO TIPO [1..num_marcot].

Dirección Y:

En orden progresivo se requiere introducir el MARCO TIPO que representa al marco plano en Y [1..num_marcosY]

[1] Número i de MARCO TIPO [1..num_marcot].

ALTURAS DE PISOS :

Son las alturas en donde se considera que están concentradas las masas de entrepiso. Los datos se introducen progresivamente del piso 1 al piso n.

[1] Altura del piso i [m].

PESOS DE ENTREPISO :

Son los pesos de todos los entrepisos concentrados en los pisos del edificio, generalmente a la altura de las losas. Estos datos son introducidos progresivamente del entrepiso 1 al entrepiso n.

[1] Pesos del entrepiso i [Ton].

LOCALIZACION DE MARCOS PLANOS :

Es la localización de los marcos planos que conforman al edificio, la posición queda definida para cada marco plano en función de la distancia que existe del marco plano al origen del edificio. Esta localización se requiere para cada marco plano en dirección X y en dirección Y, en orden progresivo del marco mas cercano al origen al marco mas lejano al origen.

Marcos resistentes en X:

En orden progresivo se requiere introducir la distancia en Y del marco plano en X.

[1] Distancia en Y del marco plano resistente en X al origen [m].

Marcos resistentes en Y:

En orden progresivo se requiere introducir la distancia en X del marco plano en Y.

- [1] Distancia en X del marco plano resistente en Y al origen [m].

CENTROS DE MASA :

Son las coordenadas (xm,ym) que representan al centro de masa por entrepiso, en donde se considera que se encuentran concentrados los pesos de entrepiso del edificio.

Coordenadas en X:

Se introducen progresivamente del piso 1 al piso n.

- [1] Coordenada en X (xm) del entrepiso i [m].

Coordenadas en Y:

Se introducen progresivamente del piso 1 al piso n.

- [1] Coordenada en Y (ym) del entrepiso i [m].

ELEMENTOS BARRA TIPO :

Los elementos barra tipo se definen para el uso común de todos los MARCOS TIPO que conforman al edificio real, por ésta razón se introducen antes que los marcos.

- [1] Número total de elementos barra tipo (nbt).

Para cada elemento barra tipo (1..nbt) se requiere:

- [2] Número de elemento barra tipo (1..nbt).
[3] Módulo de elasticidad (E) [Ton/m²].
[4] Area geométrica de la sección transversal (A) [m²].
[5] Momento de inercia de la sección transversal (Iz) [m⁴].

MARCOS TIPO :

Para definir cada marco tipo (1..num_marcot) se requiere el siguiente conjunto de datos:

- [1] Número de nodos del marco tipo (nn) (incluyendo apoyos).
- [2] Número de nodos restringidos (nnr) (condiciones de continuidad y de frontera).
- [3] Número de elementos barra en el marco tipo (nb).

Localización de los nodos:

Esta localización se realiza mediante la referencia del nodo i al sistema coordenado local del marco tipo analizado (X-Y) éste sistema es diferente al sistema coordenado (X-Y-Z) del edificio. El siguiente conjunto de datos se requiere para cada nodo (1..nn).

- [1] Coordenada en X del nodo i [m].
- [2] Coordenada en Y del nodo i [m].

Restricciones en los nodos:

Son las condiciones de continuidad y de frontera de todos los nodos que conforman al marco plano. Los siguientes datos se requieren para todos los nodos restringidos (1..nnr).

- [1] Número de nodo a restringir (1..nnr).
- [2] Tipo de restricción en el nodo (1..7) (consultar capítulo IV).

Localización de elementos barra:

La localización de cada elemento barra i esta referenciada a sus nodos en los extremos (nodo A - nodo B), las propiedades geométricas y mecánicas de cada barra se obtiene de los elementos barra tipo introducidos anteriormente. El siguiente conjunto de datos se requiere por cada elemento barra (1..nb) del marco plano tipo.

- [1] Nodo origen A (1..nn).
- [2] Nodo final B (1..nn).
- [3] Elemento barra tipo que representa (1..nbt).

■ INFORMACION DE SALIDA DE RESULTADOS :

La información de salida (OUTPUT) se refiere a los resultados del análisis que genera el programa ANESAR, adicionalmente se genera un reporte de los datos de entrada, con el fin de que sean revisados al final de la ejecución del programa para corregir posibles errores en la entrada de datos (INPUT).

INFORMACION DE ENTRADA DE DATOS :

Los siguientes datos son leídos del archivo de entrada y son escritos en el archivo de resultados, para verificar que la introducción de datos sea correcta.

- DATOS GENERALES.
- FACTORES DE DUCTILIDAD.
- ESPECTRO DE DISEÑO.
- DEFINICION DE MARCOS TIPO.
- ALTURAS DE PISO.
- PESOS DE ENTREPISO (MASAS DE ENTREPISO).
- LOCALIZACION DE MARCOS PLANOS (SISTEMA GLOBAL DEL EDIFICIO).
- CENTROS DE MASA.
- ELEMENTOS BARRA TIPO.

INFORMACION DE RESULTADOS :

- RIGIDECES DE ENTREPISO DE MARCOS TIPO.
- RIGIDECES TOTALES DE ENTREPISO.
- ANALISIS SISMICO ESTATICO.
- ANALISIS SISMICO DINAMICO.
- SUPERPOSICION DE FUERZAS SISMICAS.
- CENTROS DE MASA, CORTANTE Y TORSION.
- CORTANTES SISMICOS DE DISEÑO.

▣ ARCHIVO DE DATOS (INPUT) :

El siguiente conjunto de datos representa al edificio analizado en los distintos capítulos de la tesis, este archivo (INPUT) tiene formato ASCII y contiene la secuencia de datos que requiere el programa ANESAR para realizar el análisis sísmico del edificio.

EDIFICIO : EJEMPLO DE TESIS (1a. línea de la descripción).
PROGRAMA : ANESAR V1.00 (2a. línea de la descripción).

4
4
4 5
4 4

0.15 0.60 0.60 3.90 1

3 4 3 4
1 2 2 1 2

4 8 11.5 15
470 450 380 350

0 5 11 16
0 5 11 17 22

12.0 11.5 11.0 10.4
8.2 7.9 8.0 8.5

8
1 2100000 0.20 0.004167
2 2100000 0.16 0.002133
3 2100000 0.18 0.005400
4 2100000 0.15 0.003125
5 2100000 0.12 0.001600
6 2100000 0.125 0.002604
7 2100000 0.20 0.002667
8 2100000 0.12 0.000900

20 4 28

0 0 5 0 11 0 16 0 0 4 5 4 11 4 16 4 0 8 5 8
11 8 16 8 0 11.5 5 11.5 11 11.5 16 11.5 0 15 5 15 11 15 16 15

1 0 2 0 3 0 4 0

1 5 7 2 6 2 3 7 7 4 8 7 5 6 3 6 7 3 7 8 3 5 9 7 6 10 7 7 11 7 8 12 7
9 10 3 10 11 3 11 12 3 9 13 8 10 14 8 11 15 8 12 16 8 13 14 4 14 15 4
15 16 4 13 17 8 14 16 8 15 19 8 16 20 8 17 18 6 18 19 6 19 20 6

20 4 28

0 0 5 0 11 0 16 0 0 4 5 4 11 4 16 4 0 8 5 8 11 8 16 8
0 11.5 5 11.5 11 11.5 16 11.5 0 15 5 15 11 15 16 15

1 0 2 0 3 0 4 0

1 5 7 2 6 2 3 7 7 4 8 2 5 6 4 6 7 4 7 8 4 5 9 7 6 10 2 7 11 7 8 12 2
9 10 4 10 11 4 11 12 4 9 13 8 10 14 8 11 15 8 12 16 8 13 14 5 14 15 5
15 16 5 13 17 8 14 16 8 15 19 8 16 20 8 17 18 6 18 19 6 19 20 6

25 5 36

0 0 5 0 11 0 17 0 22 0 0 4 5 4 11 4 17 4 22 4 0 8 5 8 11 8 17 8
22 8 0 11.5 5 11.5 11 11.5 17 11.5 22 11.5 0 15 5 15 11 15 17 15 22 15

1 0 2 0 3 0 4 0 5 0

1 6 1 2 7 1 3 8 1 4 9 1 5 10 1 6 7 3 7 8 3 8 9 3 9 10 3 6 11 1
7 12 1 8 13 1 9 14 1 10 15 1 11 12 3 12 13 3 13 14 3 14 15 3 11 16 5 12 17 5
13 18 5 14 19 5 15 20 5 16 17 4 17 18 4 18 19 4 19 20 4 16 21 5 17 22 5 18 23 5
19 24 5 20 25 5 21 22 6 22 23 6 23 24 6 24 25 6

25 5 36

0 0 5 0 11 0 17 0 22 0 0 4 5 4 11 4 17 4 22 4 0 8 5 8 11 8 17 8
22 8 0 11.5 5 11.5 11 11.5 17 11.5 22 11.5 0 15 5 15 11 15 17 15 22 15

1 0 2 0 3 0 4 0 5 0

1 6 1 2 7 2 3 8 2 4 9 1 5 10 2 6 7 4 7 8 4 8 9 4 9 10 4 6 11 1
7 12 2 8 13 2 9 14 1 10 15 2 11 12 4 12 13 4 13 14 4 14 15 4 11 16 5 12 17 5
13 18 5 14 19 5 15 20 5 16 17 5 17 18 5 18 19 5 19 20 5 16 21 5 17 22 5 18 23 5
19 24 5 20 25 5 21 22 6 22 23 6 23 24 6 24 25 6

■ ARCHIVO DE RESULTADOS (OUTPUT):

El siguiente conjunto de datos es generado por el programa ANESAR, corresponden al análisis sísmico del edificio analizado en los distintos capítulos de la presente tesis. Se pueden comprobar los resultados obtenidos por el programa con los cálculos realizados manualmente en los anteriores capítulos. Son el resultado del archivo de datos (INPUT) descrito anteriormente.

DESCRIPCION GENERAL DEL EDIFICIO A ANALIZAR >>>

EDIFICIO : EJEMPLO DE TESIS (1a. línea de la descripción).
PROGRAMA : ANESAR V1.00 (2a. línea de la descripción).

DATOS GENERALES DEL EDIFICIO >>>

Número de pisos : 4
Número de marcos tipo : 4
Número de marcos en X : 4
Número de marcos en Y : 5

FACTORES DE DUCTILIDAD >>>

Factor de ductilidad en X (α_x) [] : 4.0000
Factor de ductilidad en Y (α_y) [] : 4.0000

ESPECTRO DE DISEÑO >>>

Aceleración máxima del terreno (a_0) [g] : 0.1500
Aceleración máxima maximorum (c) [g] : 0.6000
Período inferior de resonancia (T_a) [seg] .. : 0.6000
Período superior de resonancia (T_b) [seg] .. : 3.9000
Exponente adimensional (r) [] : 1.0000

DEFINICION DE MARCOS TIPO >>>

MARCO X-1 => MARCO TIPO 3, y1 = 0.000 m.
MARCO X-2 => MARCO TIPO 4, y1 = 5.000 m.
MARCO X-3 => MARCO TIPO 3, y1 = 11.000 m.
MARCO X-4 => MARCO TIPO 4, y1 = 16.000 m.

MARCO Y-1 => MARCO TIPO 1, x1 = 0.000 m.
MARCO Y-2 => MARCO TIPO 2, x1 = 5.000 m.
MARCO Y-3 => MARCO TIPO 2, x1 = 11.000 m.
MARCO Y-4 => MARCO TIPO 1, x1 = 17.000 m.
MARCO Y-5 => MARCO TIPO 2, x1 = 22.000 m.

piso	W[ton]	H[m]	xm[m]	ym[m]
4	350.00	15.000	10.400	8.500
3	380.00	11.500	11.000	8.000
2	450.00	8.000	11.500	7.900
1	470.00	4.000	12.000	8.200

ELEMENTOS BARRA TIPO >>>

ELEM-TIPO	E [Ton/m ²]	AREA [m ²]	Iz [m ⁴]
1	2100000.00	0.2000	0.004167
2	2100000.00	0.1600	0.002133
3	2100000.00	0.1800	0.005400
4	2100000.00	0.1500	0.003125
5	2100000.00	0.1200	0.001500
6	2100000.00	0.1250	0.002604
7	2100000.00	0.2000	0.002667
8	2100000.00	0.1200	0.000900

RIGIDECES DE ENTREPISO DE MARCOS TIPO >>>

MARCO TIPO [1]

Entrepiso	RIG [Ton/m]
4	1260.8878
3	1417.5079
2	2230.8731
1	2963.1170

MARCO TIPO [2]

Entrepiso	RIG [Ton/m]
4	1035.3651
3	1144.3331
2	1631.8108
1	2342.6627

MARCO TIPO [3]

Entrepiso	RIG [Ton/m]
4	2232.3828
3	2606.8616
2	3624.9423
1	5160.2207

MARCO TIPO [4]

Entrepiso	RIG [Ton/m]
4	1737.5615
3	1967.2239
2	2245.2724
1	3335.0306

RIGIDECES TOTALES DE ENTREPISO >>>

Entrepiso	RIG-X [Ton/m]	RIG-Y [Ton/m]
4	7939.8885	5627.8709
3	9148.1710	6268.0150
2	11740.4294	9337.1786
1	16990.5027	12954.2220

ANALISIS SISMICO ESTATICO >>>

DIRECCION ANALIZADA [X] :

Entreplao	DESPL.REL [m]	FUERZA [Ton]	Piso	DESPL.ABS [m]	CORTANTE [Ton]
4	0.0434	86.05	4	0.2444	86.05
3	0.0689	71.63	3	0.2010	157.68
2	0.0738	59.01	2	0.1321	216.69
1	0.0583	30.81	1	0.0583	247.50

DIRECCION ANALIZADA [Y] :

Entreplao	DESPL.REL [m]	FUERZA [Ton]	Piso	DESPL.ABS [m]	CORTANTE [Ton]
4	0.0612	86.05	4	0.3308	86.05
3	0.1006	71.63	3	0.2697	157.68
2	0.0926	59.01	2	0.1691	216.69
1	0.0764	30.81	1	0.0764	247.50

ANALISIS SISMICO DINAMICO >>>

DIRECCION ANALIZADA [X] :

Entreplao	DESPL.REL [m]	FUERZA [Ton]	Piso	DESPL.ABS [m]	CORTANTE [Ton]
4	0.0363	74.24	4	0.2042	74.24
3	0.0582	59.52	3	0.1679	134.05
2	0.0616	47.02	2	0.1097	181.07
1	0.0480	24.22	1	0.0480	205.29

DIRECCION ANALIZADA [Y] :

Entreplao	DESPL.REL [m]	FUERZA [Ton]	Piso	DESPL.ABS [m]	CORTANTE [Ton]
4	0.0522	74.94	4	0.2763	74.94
3	0.0854	59.50	3	0.2241	134.44
2	0.0764	44.52	2	0.1387	178.95
1	0.0623	23.88	1	0.0623	202.83

SUPERPOSICION DE FUERZAS SISMICAS ESTATICAS Y DINAMICAS >>>

Entreplao	Vx [Ton]	Vy [Ton]	Piso	Fx [Ton]	Fy [Ton]
4	74.24	74.94	4	74.24	74.94
3	134.05	134.44	3	59.52	59.50
2	181.07	178.95	2	47.02	44.52
1	205.29	202.83	1	24.22	23.88

CENTROS DE >>>

Piso	[MASA]		[CORTANTE]		[TORSION]	
	xm [m]	ym [m]	xv [m]	yv [m]	xt [m]	yt [m]
4	10.400	8.500	10.400	8.500	10.800	7.688
3	11.000	8.000	10.668	8.279	10.782	7.650
2	11.500	7.900	10.884	8.184	10.680	7.412
1	12.000	8.200	11.016	8.186	10.761	7.463

CORTANTES SISMICOS DE DISEÑO >>>

DIRECCION ANALIZADA [X] :

MARCO		[X-1]	
Entrepiso	Vx(Ton)	Fx(Ton)	
4	24.08	24.08	
3	44.33	20.25	
2	63.33	19.00	
1	71.32	7.99	

MARCO		[X-2]	
Entrepiso	Vx(Ton)	Fx(Ton)	
4	17.12	17.12	
3	30.43	13.31	
2	36.12	5.69	
1	42.21	6.09	

MARCO		[X-3]	
Entrepiso	Vx(Ton)	Fx(Ton)	
4	23.97	23.97	
3	43.39	19.42	
2	63.91	20.52	
1	71.26	7.37	

MARCO		[X-4]	
Entrepiso	Vx(Ton)	Fx(Ton)	
4	22.30	22.30	
3	38.58	16.29	
2	46.50	7.91	
1	54.23	7.73	

DIRECCION ANALIZADA [Y] :

MARCO		[Y-1]	
Entrepiso	Vy(Ton)	Fy(Ton)	
4	22.50	22.50	
3	39.16	16.66	
2	54.39	15.23	
1	58.73	4.33	

MARCO		[Y-2]	
Entrepiso	Vy(Ton)	Fy(Ton)	
4	16.31	16.31	
3	28.34	12.03	
2	35.77	7.44	
1	41.90	6.13	

MARCO		[Y-3]	
Entrepiso	Vy(Ton)	Fy(Ton)	
4	13.85	13.85	
3	24.67	10.82	
2	31.51	6.84	
1	36.95	9.44	

MARCO		
Entrepiso	Vy(Ton)	[Y-4] Fy(Ton)
4	19.17	19.17
3	34.99	15.82
2	50.85	15.86
1	55.20	4.34

MARCO		
Entrepiso	Vy(Ton)	[Y-5] Fy(Ton)
4	17.32	17.32
3	31.23	13.91
2	41.94	10.71
1	49.22	7.28

BIBLIOGRAFIA:

- [1] **Manual de Diseño Sísmico de edificios.**
Autores: Enrique Bazán Zurita y Roberto Meli Piralla.
Editorial: Limusa, México 1985.

- [2] **Apuntes de la Materia de Ingeniería Sísmica.**
Impartida por el Ing. Amílcar Galindo Solorzano, en la E.N.E.P. Aragón.
México 1991.

- [3] **Diseño de Estructuras Sismorresistentes.**
Autores: Minoru Wakabayashi y Enrique Martínez Romero.
Editorial: Mc. Graw Hill, México 1990.

- [4] **Estructuras Antisísmicas.**
Autor: Gabriel Estrada Uribe.
Editorial: C.E.C.S.A., Colombia 1975.

- [5] **Normas Técnicas Complementarias para Diseño por Sismo.**
Diario Oficial de la Federación, México 1987.

- [6] **Reglamento de Construcciones para el Distrito Federal.**
Diario Oficial de la Federación, México 1988.

- [7] **Lenguaje C, Programación Avanzada.**
Autor: Herbert Schildt.
Editorial: Osborne/Mc. Graw Hill, México 1988.

- [8] **C, Manual de Referencia.**
Autor: Herbert Schildt.
Editorial: Osborne/Mc. Graw Hill, México 1991.

- [9] **Programación en TURBO C.**
Autor: Herbert Schildt.
Editorial: Borland, Osborne/Mc.Graw Hill, México 1989.

- [10] **Álgebra Lineal y sus Aplicaciones.**
Autor: Gilbert Strang.
Editorial: IBEROAMERICANA, México 1986.