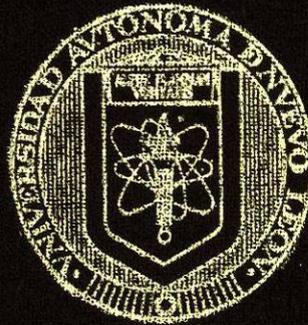


UNIVERSIDAD AUTONOMA DE NUEVO LEON
FACULTAD DE CIENCIAS FISICO-MATEMATICAS



GENERACION AUTOMATICA DE CURVAS
DE CONTORNO EN UNA SUPERFICIE
TRIANGULARIZADA

T E S I S

QUE PARA OBTENER EL TITULO DE:
LICENCIADO EN CIENCIAS COMPUTACIONALES

PRESENTA:

ARTURO ORTEGA MORAN

MONTERREY, N. L.

MAYO DE 1987

TL
QA643
.O78
1987
c.1



1080171529

JUN 87
~
?

Hora de inicio :: 7:30 PM:

" " FINAL :: 8:50 PM:

Presidente: Ing. Aurelio Ramírez Graa de

Vicepresidente: Lic. Fidencio San R.

Secretario: Ing. Adolfo Ferrández B.

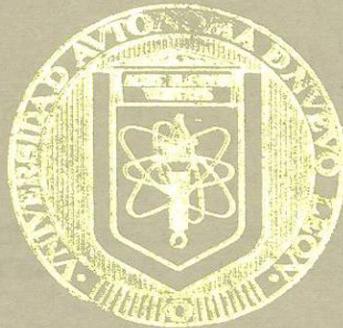
Evaluación

Presentación

Conocimiento

xp icon

UNIVERSIDAD AUTONOMA DE NUEVO LEON
FACULTAD DE CIENCIAS FISICO-MATEMATICAS



GENERACION AUTOMATICA DE CURVAS
DE CONTORNO EN UNA SUPERFICIE
TRIANGULARIZADA

T E S I S

QUE PARA OBTENER EL TITULO DE:
LICENCIADO EN CIENCIAS COMPUTACIONALES

PRESENTA:

ARTURO ORTEGA MORAN

MONTERREY, N. L.

MAYO DE 1987

UNIVERSIDAD AUTONOMA DE NUEVO LEON
FACULTAD DE CIENCIAS FISICO MATEMATICAS

GENERACION AUTOMATICA DE CURVAS DE CONTORNO
EN UNA SUPERFICIE TRIANGULARIZADA

T E S I S

TEMA QUE PARA OBTENER EL TITULO DE
LICENCIADO EN CIENCIAS COMPUTACIONALES
PRESENTA
ARTURO ORTEGA MORAN

MONTERREY N.L.

MAYO DE 1987

INDICE

----- 0 -----

	pagina
I.- RESUMEN	1
II.- INTRODUCCION	3
III.- BASE CONCEPTUAL PARA OBTENER LAS CURVAS DE CONTORNO	7
IV.- ESPECIFICACION DE ENTRADA	9
V.- ESPECIFICACION DE SALIDA	10
VI.- ESTRATEGIA GENERAL	13
VII.- ANALISIS DE CASOS EN EL PROCESO DE ELEMENTOS TRIANGULARES	15
VII-A TABLA DE ANALISIS DE CASOS	17
VII-B TABLA DE CODIGO DE ACCIONES	18
VII-C TABLA DE DECISION	19
VIII.- TECNICA DE DISENO DEL SOFTWARE	20
IX.- DISENO DE ESTRUCTURAS DE DATOS	21
X.- DISENO DE MODULARIDAD	28
XI.- DISENO DE MODULOS	29
XII.- IMPLEMENTACION DEL SOFTWARE	46
XIII.- CORRIDAS DE PRUEBA	48
XIV.- COMENTARIOS Y CONCLUSIONES	57

XV.- MEJORAS FUTURAS RECOMENDADAS	58
APENDICE A (una metodologia para generar una superficie trianqu- larizada)	60
APENDICE B (interseccion de un plano horizontal con un segmento recto)	63
BIBLIOGRAFIA	67

LISTA DE FIGURAS

FIG.	TITULO	PAGINA
1	CURVAS DE CONTORNO EN CARTOGRAFIA	4
2	CURVAS DE CONTORNO EN INGENIERIA	5
3	SUPERFICIE DISCRETIZADA EN TRIANGULOS	6
4	CURVAS DE CONTORNO	7
5	INTERSECCION DE UN PLANO CON UN TRIANGULO	7
6	GENERACION DE CURVAS DE CONTORNO	8
7a	CURVA DE CONTORNO COMO UN CONJUNTO DE SEGMENTOS	10
7b	CURVA DE CONTORNO COMO VECTOR	11
8	ESTRATEGIA GENERAL	14
9	INTERSECCION DE UN PLANO CON UN SEGMENTO RECTO	64
10	DIAGRAMA DE MODULARIDAD	28
11	GRAFICA DE $Z=Y^2$	49
12	GRAFICA DE $Z=X^2 + Y^2$	50
13	SECUENCIA EN LAS CORRIDAS DE PRUEBA	52
14	CURVAS DE CONTORNO EN $Z=X^2+Y^2$ (VISTA ISOMETRICA)	53
15	CURVAS DE CONTORNO EN $Z=X^2+Y^2$ (PROYECCION EN PLANO "XY")	54
16	CURVAS DE CONTORNO DE $Z=Y^2$	

	(VISTA ISOMETRICA)	55
17	CURVAS DE CONTORNO EN $Z=Y^2$	
	(PROYECCION EN PLANO "XY")	56

GENERACION DE CURVAS DE CONTORNO
EN UNA SUPERFICIE TRIANGULARIZADA

----- o -----

I.- RESUMEN

ESTE TRABAJO ES EL DISEÑO DE UN SOFTWARE PARA OBTENER CURVAS DE CONTORNO EN UNA SUPERFICIE DISCRETIZADA EN TRIANGULOS PLANOS.

LAS ENTRADAS REQUERIDAS SON:

- a).- Z_{max} .
- b).- Z_{min} .
- d).- Archivo de elementos triangulares definidos por las coordenadas de sus vértices.

LA SALIDA PRESENTA DOS OPCIONES:

- a).- CASO SIMPLE: Genera una estructura de datos en donde las curvas de contorno se definen como un conjunto de segmentos rectos independientes, que pueden ser enviados al dispositivo gráfico disponible.
- b).- CASO COMPLEJO: En esta opción, las curvas de contorno se definen como cadenas de puntos (abiertas o cerradas) ,que permiten mayor flexibilidad en procesos de etiquetación y de suavización.

LA IMPLEMENTACION DEL SOFTWARE SE HIZO EN FORTRAN 77 USANDO LOS RECURSOS GRAFICOS DE UN MEDIO AMBIENTE INTERGRAPH. SINEMBARGO LA MODULARIDAD DEL SOFTWARE PERMITE QUE FACILMENTE PUEDA SER ADAPTADO A OTROS DISPOSITIVOS GRAFICOS, ESTO MODIFICANDO ADECUADAMENTE LAS SUBROUTINAS DE INTERFASE GRAFICA, QUE ESTAN CLARAMENTE IDENTIFICADAS.

II.-INTRODUCCION

EL TRAZO DE CURVAS DE CONTORNO EN UNA SUPERFICIE, ES UNA NECESIDAD QUE SE PRESENTA EN DIVERSOS CAMPOS. ASI, EN CARTOGRAFIA EL TRAZAR CURVAS DE CONTORNO (CURVAS DE NIVEL), ES IMPORTANTE PARA CARACTERIZAR LA TOPOGRAFIA DE UNA REGION (FIG. 1). POR OTRO LADO, EN INGENIERIA ES COMUN MANEJAR EL CONCEPTO DE SUPERFICIE DE RESPUESTA, CUANDO ES DE INTERES LA DISTRIBUCION DE UNA PROPIEDAD (TEMPERATURA, CONCENTRACION, PRESION, ETC.) SOBRE UNA GEOMETRIA BIDIMENSIONAL, POR EJEMPLO PUEDE INTERESAR LA DISTRIBUCION DE LA TEMPERATURA EN UNA PIEZA PLANA, EN ESTE CASO ES DESEABLE CONOCER ALGUNAS ISOTERMAS (CURVAS DE CONTORNO) COMO UNA MANERA DE DEFINIR DICHA DISTRIBUCION. (FIG. 2).

AHORA BIEN, PARA LOGRAR EL TRAZADO AUTOMATICO DE LAS CURVAS DE CONTORNO, ES NECESARIO DISCRETIZAR LA SUPERFICIE EN CUESTION. UNA DE LAS ALTERNATIVAS QUE EXISTEN PARA ESTO ES LA DISCRETIZACION EN TRIANGULOS PLANOS GENERANDOSE ASI LO QUE LLAMAMOS UNA "SUPERFICIE TRIANGULARIZADA". (FIG. 3).

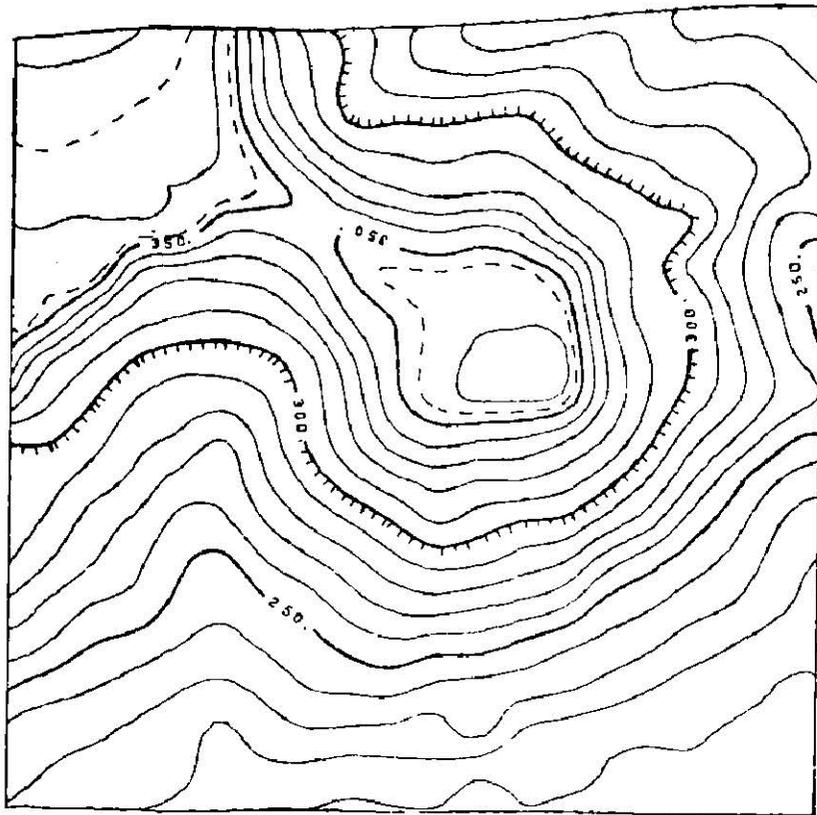
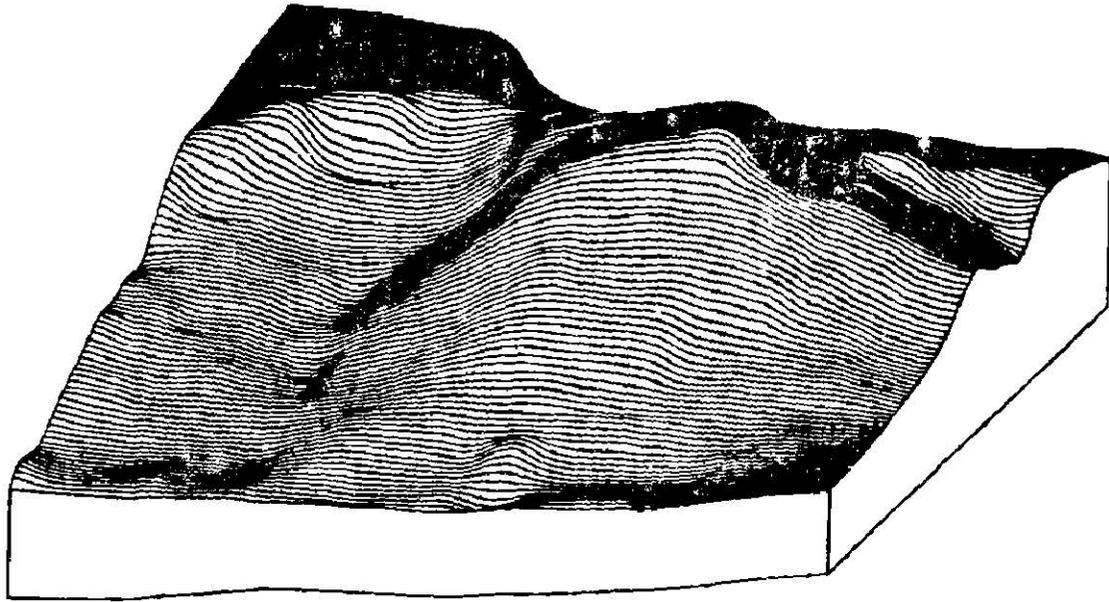


FIG. 1
CURVAS DE CONTORNO EN CARTOGRAFIA.

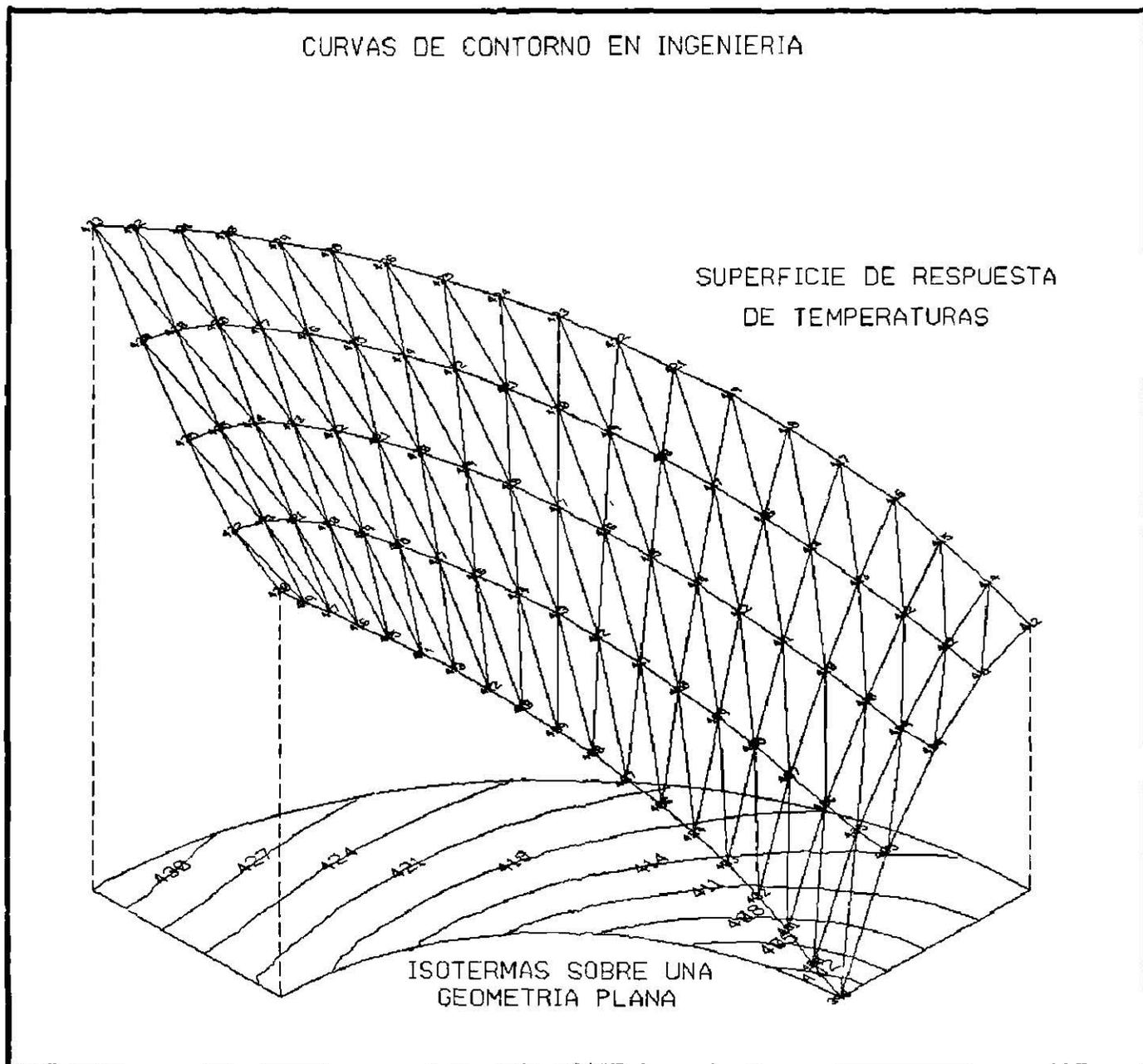
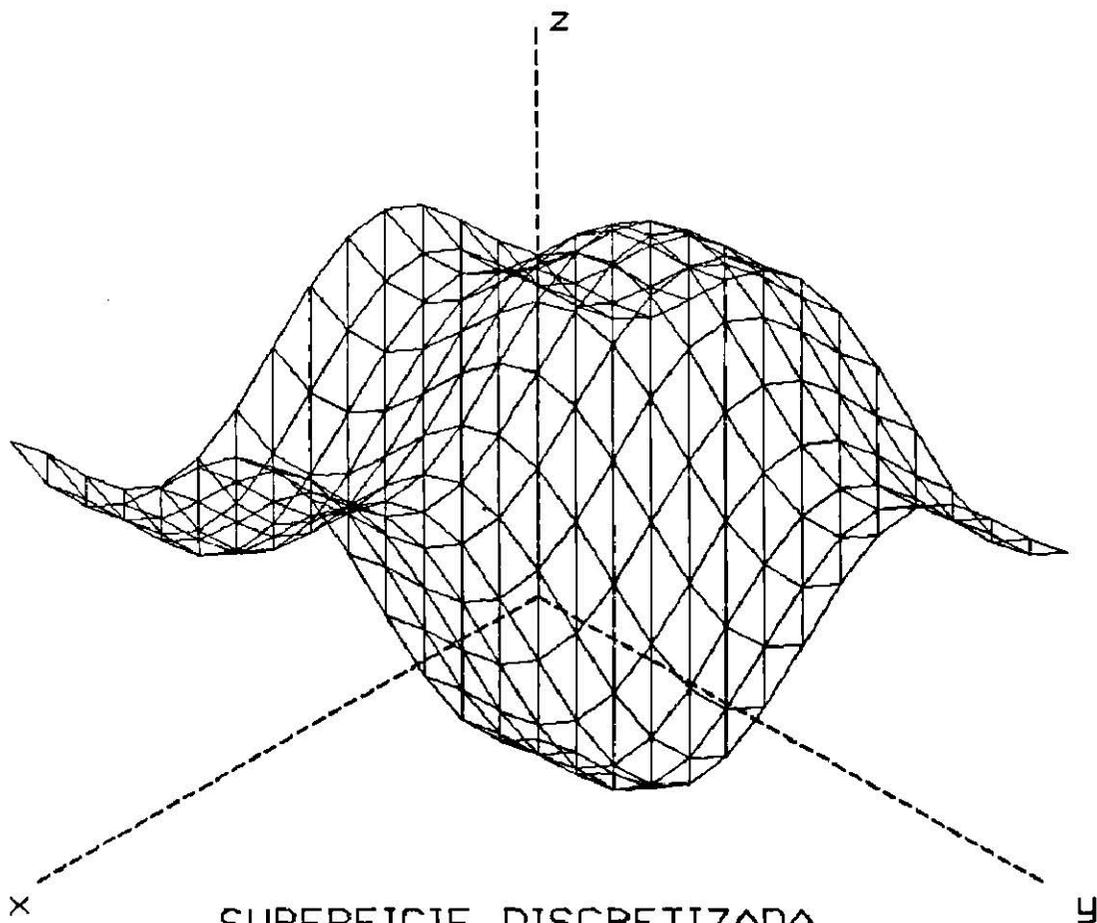


FIG. 2

$$Z = \text{SIN}(X) + \text{COS}(Y)$$



SUPERFICIE DISCRETIZADA
EN TRIANGULOS

FIG. 3

III.- BASE CONCEPTUAL PARA LA OBTENCION DE CURVAS DE CONTORNO

EMPEZAREMOS POR DECIR QUE UNA CURVA DE CONTORNO ES LA PROYECCION EN EL PLANO "XY" DE LA INTERSECCION DE LA SUPERFICIE CON UN PLANO HORIZONTAL $Z=Z_0$.

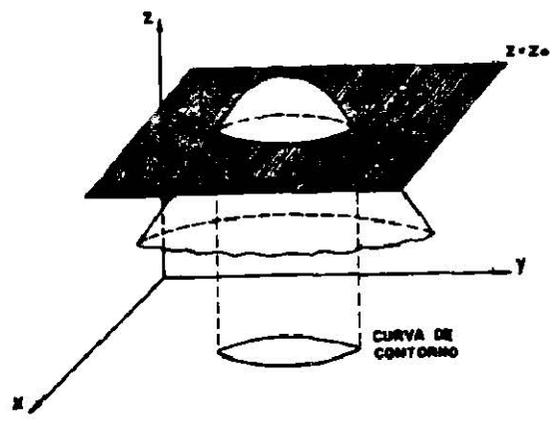


FIG. 4

AHORA, DADO QUE PARTIMOS DEL HECHO DE QUE LA SUPERFICIE ESTA DISCRETIZADA TRIANGULARMENTE, COMO SE MOSTRO EN LA FIG. 3, ENTONCES AL INTERSECTAR CON UN PLANO HORIZONTAL $Z=Z_0$ LOS TRIANGULOS QUE SON CORTADOS POR EL PLANO GENERAN UN SEGMENTO RECTO COMO INTERSECCION.

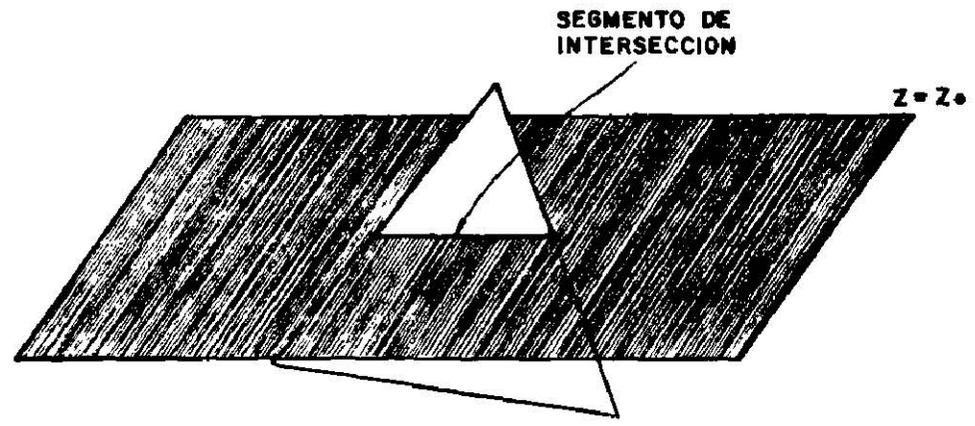


FIG. 5

DE MODO QUE UNA CURVA DE CONTORNO PUEDE OBTENERSE CON LA UNION DE TODOS LOS SEGMENTOS GENERADOS AL INTERSECTAR UN PLANO $Z=Z_0$ CON CADA UNO DE LOS ELEMENTOS TRIANGULARES QUE FORMAN LA SUPERFICIE.

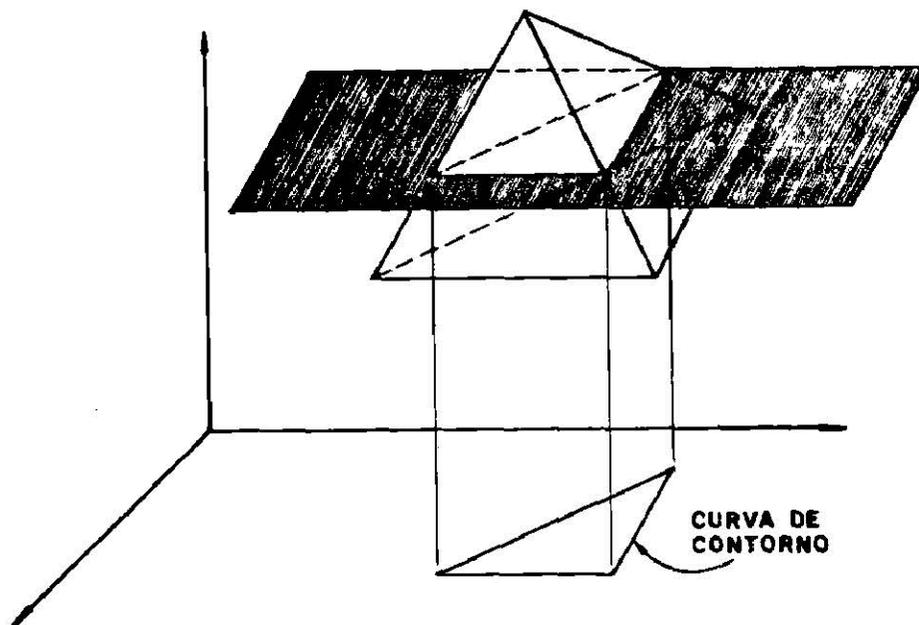


FIG. 6

ESTE HECHO SUGIERE UNA METODOLOGIA QUE SERA LA BASE DEL DISEÑO DEL SOFTWARE, PROPOSITO DE ESTE TRABAJO. LOS DETALLES SE DAN EN EL PUNTO VI.

IV.- ESPECIFICACION DE ENTRADA.

EL SOFTWARE PARA TRAZAR AUTOMATICAMENTE CURVAS DE CONTORNO , AL QUE DE AQUI EN ADELANTE LLAMAREMOS "CONTORNO.SYS", REQUIERE COMO ENTRADA:

- a).- Zmax : Máxima elevación de la superficie.
- b).- Zmin : Punto mínimo de la superficie.
- c).- NCS : Numero de curvas de contorno equiespaciadas que se desean.
- d).- Archivo de elementos triangulares: Un conjunto de registros de la forma
(X1, Y1, Z1, X2, Y2, Z2, X3, Y3, Z3)

ESTA INFORMACION SE DEBE PROOVER EN UN ARCHIVO SECUENCIAL LLAMADO "CONTORNO.DAT" CON LA SIGUIENTE ORGANIZACION:

NO. ELEM

1

ZMAX, ZMIN, NCS

X1, Y1, Z1, X2, Y2, Z2, X3, Y3, Z3

2

X1, Y1, Z1, X2, Y2, Z2, X3, Y3, Z3

3

X1, Y1, Z1, X2, Y2, Z2, X3, Y3, Z3

.

.

.

.

K

X1, Y1, Z1, X2, Y2, Z2, X3, Y3, Z3

V.- ESPECIFICACION DE SALIDA.

EL OBJETIVO DE "CONTORNO.SYS" ES LA GENERACION DE LAS CURVAS DE CONTORNO, DE MODO QUE LA SALIDA ES BASICAMENTE UNA ESTRUCTURA DE DATOS QUE PERMITAN DEFINIR A DICHAS CURVAS.

"CONTORNO.SYS" CONTEMPLA DOS TIPOS DE SALIDAS PARA DEFINIR LAS CURVAS DE CONTORNO

a).- SALIDA SIMPLE: En este estilo se genera un conjunto de segmentos independientes disponibles en una estructura del tipo:

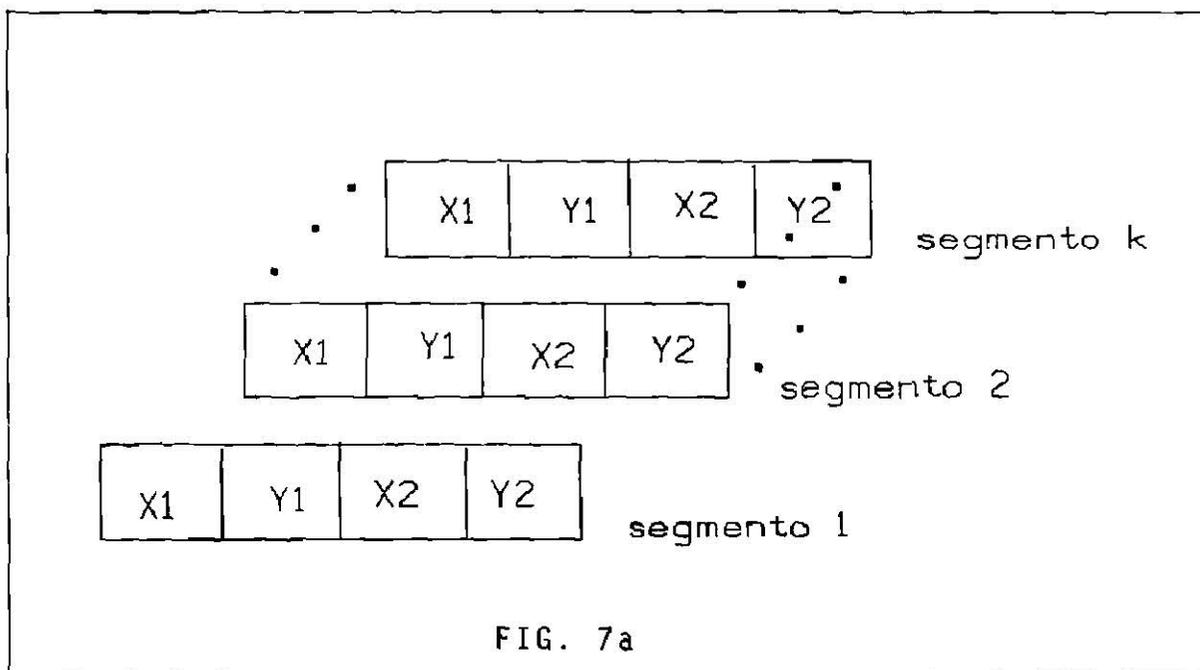


FIG. 7a

Que pueden ser enviados al dispositivo gráfico disponible para obtener una imagen de las curvas de contorno.

b).- SALIDA COMPLEJA: En esta opción se generan cadenas de puntos que pueden ser abiertas o cerradas, para definir las curvas de contorno. Las cadenas quedan disponibles en un vector de la forma:

X1	Y1	X2	Y2	Xn	Yn
----	----	----	----	---	---	---	---	----	----

FIG. 7b

LAS VENTAJAS DE ESTE ESTILO SON:

- 1.- Pueden implementarse procedimientos para suavizar las curvas.
- 2.- Se pueden etiquetar con mayor facilidad.
- 3.- En sistemas gráficos sofisticados se facilita la manipulación, ya que las cadenas pueden manejarse como elementos.

EL TIPO DE SALIDA SE ELIGE INTERACTIVAMENTE Y SOLO SE REQUIERE CONTAR CON UNA SUBROUTINA DE INTERFASE GRAFICA QUE DEBE DISEÑARSE DE ACUERDO CON EL DISPOSITIVO GRAFICO DISPONIBLE. EN ESTE TRABAJO "CONTORNO.SYS" INCLUYE UNA SUBROUTINA QUE PERMITE OBTENER LA IMAGEN DE LAS CURVAS DE CONTORNO EN UNA TERMINAL DE TRABAJO INTERACT , AUXILIANDOSE DE "IGDS" (Interactive Graphics Design Systems), SOFTWARE DE INTERGRAPH.

DADO UN NIVEL $Z=Z_0$, PARA DEFINIR LA CURVA DE CONTORNO CORRESPONDIENTE, SE PUEDE PROCEDER DE LA SIGUIENTE MANERA:

- 1.- DEFINIR Z_0

- 2.- /PARA CADA ELEMENTO TRIANGULAR/
 - 2.1.- Determinar si existe la intersección.
 - 2.2.- Si la respuesta es afirmativa continuar, de lo contrario analizar el siguiente elemento.
 - 2.3.- Obtener una representación matemática del segmento de intersección.

- 3.- USAR UN DISPOSITIVO GRAFICO PARA TENER UNA IMAGEN DE LA CURVA DE CONTORNO PARTIENDO DE LA REPRESENTACION MATEMATICA DE LOS SEGMENTOS DE INTERSECCION.

ESTRATEGIA GENERAL

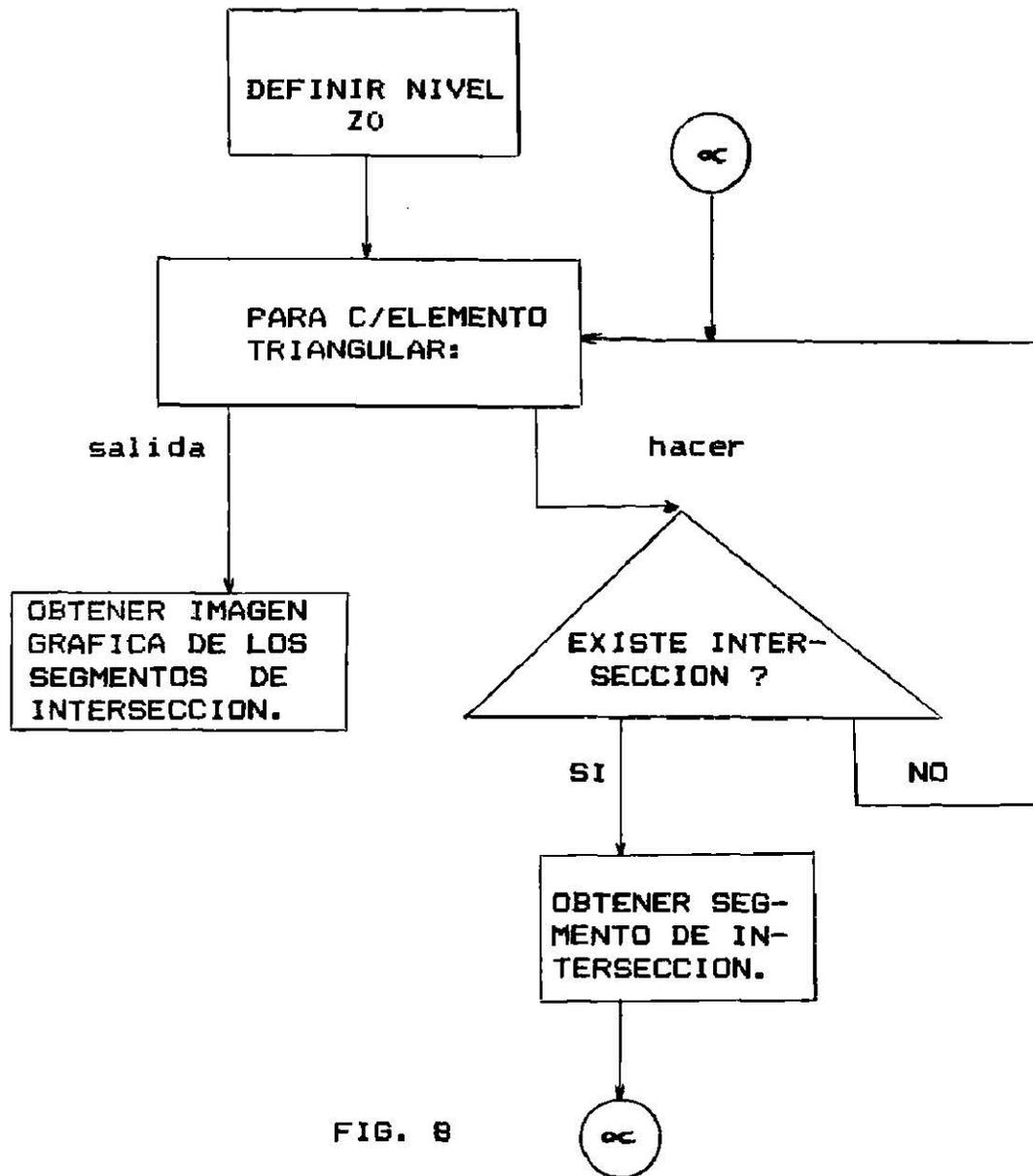


FIG. 8

VII.- ANALISIS DE CASOS EN EL PROCESO DE ELEMENTOS TRIANGULARES.

CON OBJETO DE SIMPLIFICAR EL DISEÑO DEL PROCESO DE LOS ELEMENTOS TRIANGULARES, SE SIGUIO EL SIGUIENTE ESQUEMA:

- 1.- Efectuar un análisis de las diferentes situaciones que pueden presentarse en un elemento con respecto a un nivel definido por un plano $Z=Z_0$. Este análisis se presenta en VII-A, en donde además se define mediante un código, la acción que debe ejecutarse. La codificación de las acciones se muestra en VII-B.
- 2.- A partir del análisis anterior se diseño una tabla de decisión mostrada en VII-C, en donde se define que acción tomar de acuerdo a la situación del elemento.

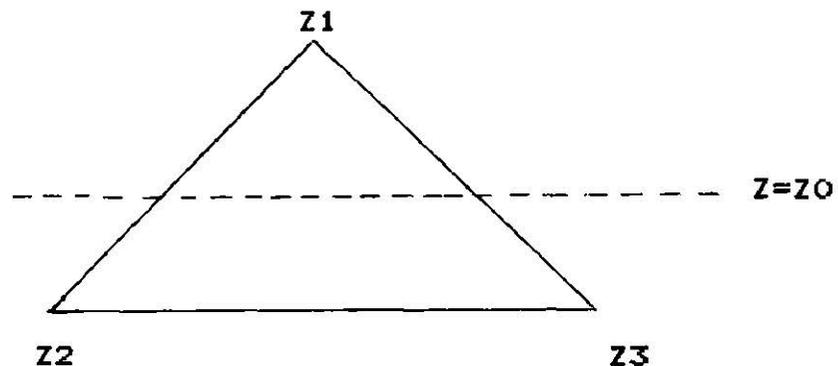
PARA DEFINIR EN QUE SITUACION SE ENCUENTRA UN ELEMENTO DADO,
SE DETERMINA EL NUMERO DE:

$$1).- \#Z's > Z0$$

$$2).- \#Z's < Z0$$

$$3).- \#Z's = Z0$$

ejemplo:



En este elemento:

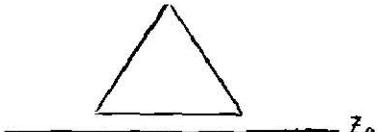
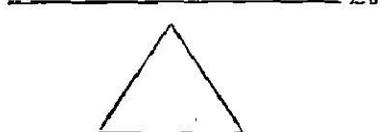
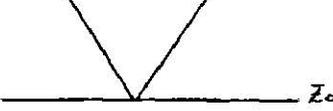
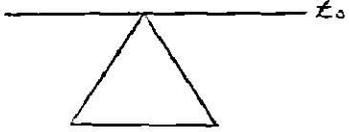
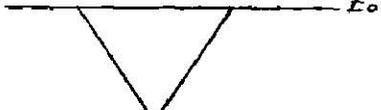
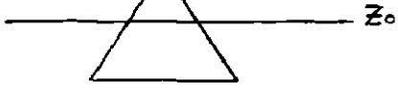
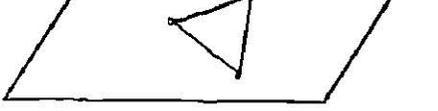
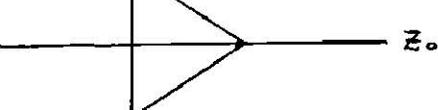
$$\#Z's > Z0 = 1$$

$$\#Z's < Z0 = 2$$

$$\#Z's = Z0 = 0$$

De modo que de acuerdo a la tabla VII-C debe de ejecutarse la acción #4, que significa de acuerdo a la tabla VII-B, " ENCONTRAR LAS INTERSECCIONES DE $Z=Z0$ CON SEGMENTO (ZMAY-ZMEN1) Y CON SEGMENTO (ZMAY-ZMEN2)".

VII-A.- TABLA DE ANALISIS DE CASOS.

No.	SITUACION GEOMETRICA	CODIGO DE ACCION
1		1
2		1
3		1
4		1
5		2
6		2
7		3
8		4
9		5
10		6

VII-B.- TABLA DE CODIGO DE ACCIONES.

No.	DESCRIPCION DE LA ACCION.
1	NO ACCION (NO HAY INTERSECCION.)
2	SEGMENTO DE INTERSECCION ES UN LADO DEL TRIANGULO. HALLAR 2 VERTICES TALES QUE $Z=Z_0$.
3	EL TRIANGULO ES CORTADO EN DOS DE SUS LADOS Y UN VERTICE ESTA POR ABAJO DEL PLANO. HALLAR PUNTOS DE CORTE.
4	EL TRIANGULO ES CORTADO EN DOS DE SUS LADOS Y UN VERTICE ESTA POR ARRIBA DEL PLANO. HALLAR PUNTOS DE CORTE.
5	TODO EL TRIANGULO ESTA EN EL NIVEL $Z=Z_0$. LOS TRES LADOS DEBEN CONSIDERARSE COMO PARTE DE LA CURVA DE CONTORNO.
6	EL TRIANGULO ES CORTADO EN UNO DE SUS LADOS Y EN UN VERTICE. ENCONTRAR PUNTO DE CORTE Y VERTICE TAL QUE $Z=Z_0$.

VII-C.- TABLA DE DECISION.

#Z's > Z0	#Z's < Z0	#Z's = Z0	# ACCION
3	0	0	1
0	3	0	1
2	0	1	1
0	2	1	1
1	0	2	2
0	1	2	2
2	1	0	3
1	2	0	4
0	0	3	5
1	1	1	6

VIII.- TECNICA DE DISEÑO DE SOFTWARE»

PARA LOGRAR UN DISEÑO ESTRUCTURADO DEL SOFTWARE SE SIGUIÓ LA SIGUIENTE METODOLOGÍA:

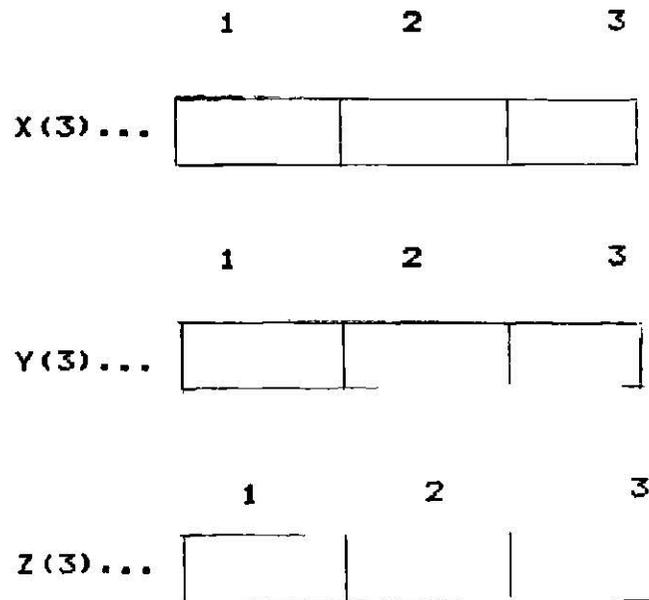
- A).- SE DISEÑARON LAS ESTRUCTURAS DE DATOS NECESARIAS PARA LOGRAR EL OBJETIVO.»
- B)» SE DISEÑÓ UN DIAGRAMA DE MODULARIDAD, A PARTIR DEL CUAL SE INICIÓ EL DESARROLLO DE MÓDULOS.»
- C). SE LOS MÓDULOS SE DESARROLLARON UTILIZANDO LA TÉCNICA DE PSEUDOCÓDIGO.

EN LOS SIGUIENTES PUNTOS SE DETALLA CADA UNO DE LOS CASOS SEGUIDOS EN EL DISEÑO DE ESTE SOFTWARE.

IX.- DISEÑO DE ESTRUCTURAS DE DATOS.

A).- / ESTRUCTURA " ELEMENTO " /

PARA DEFINIR UN ELEMENTO TRIANGULAR HAY QUE ESPECIFICAR LAS COORDENADAS CORRESPONDIENTES A SUS VERTICES. ESTO SE HARA MEDIANTE TRES VECTORES COMO SE MUESTRA A CONTINUACION:



B).- / ESTRUCTURA " DEC " /

DEC ES UNA MARTRIZ DE 10 X 4 EN LA CUAL
SE IMPLEMENTA LA TABLA DE DECISION DISEÑADA
EN VII-C.

	1	2	3	4
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

DEC (10,4)

C).- / ESTRUCTURA " CASO "

ESTA ESTRUCTURA PERMITE CLASIFICAR A UN ELEMENTO TRIANGULAR PARA POSTERIORMENTE UBICARLO EN " DEC " Y DEFINIR ASI LA ACCION QUE DEBE TOMARSE.

# $Z > Z_0$	# $Z < Z_0$	# $Z = Z_0$
1	2	3



CASO (3)

D).- / ESTRUCTURA " SEG "

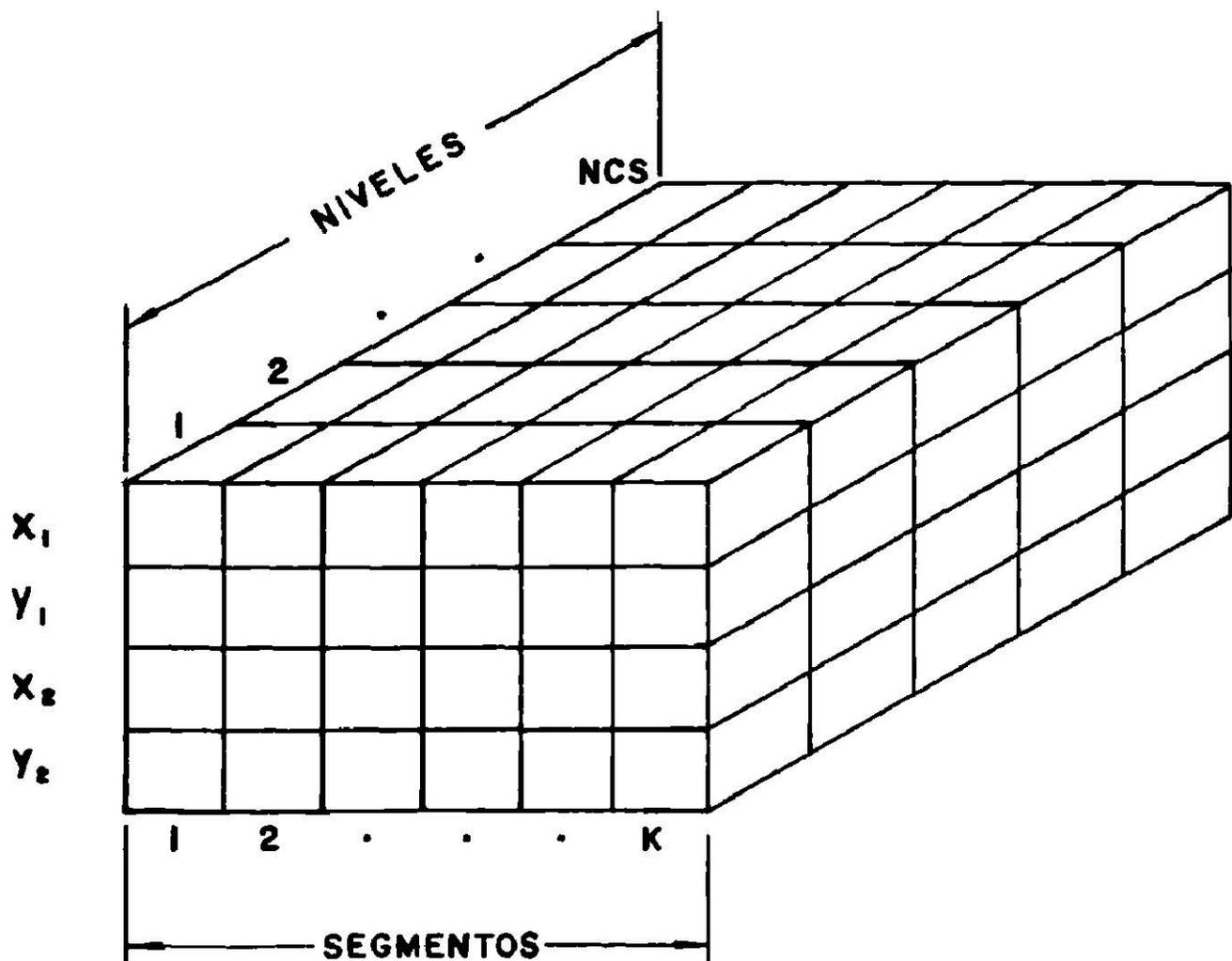
AQUI SE DEFINE UN SEGMENTO DE INTERSECCION COMO UNA ANTESALA PARA PASAR A LA ESTRUCTURA GENERAL:

	1	2
1	X1	X2
2	Y1	Y2

SEG(2,2)

E).- / ESTRUCTURA "CURV" /

ESTA ES LA ESTRUCTURA MAESTRA PARA LA DEFINICION DE LAS CURVAS DE CONTORNO. AQUI SE ALMACENAN LOS SEGMENTOS RECTOS COMO RESULTADO DEL PROCESO DE LOS ELEMENTOS TRIANGULARES.

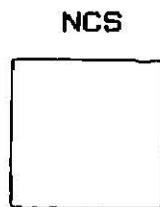


CURV (4,100,20)

*Se prevee un máximo de 20 niveles equiespaciados
y un máximo de 100 segmentos/curva.

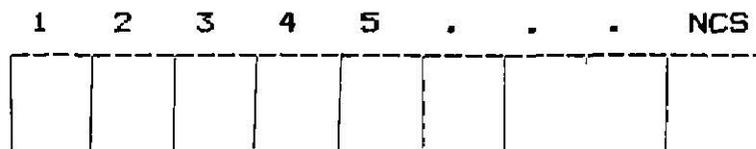
F).- / ESTRUCTURA " NCS "

ESTA ES UNA VARIABLE SIMPLE QUE ALMACENA
EL NUMERO DE CURVAS DE CONTORNO EQUIESPACIADAS
QUE DESEA EL USUARIO.



G).- / ESTRUCTURA " NSEG "

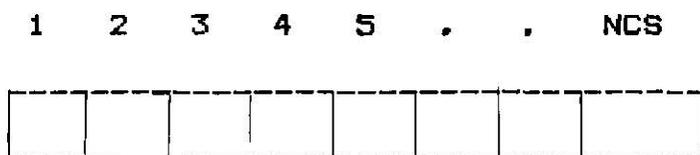
EN ESTA ESTRUCTURA SE ALMACENA EL NUMERO DE
SEGMENTOS QUE COMPONEN A CADA CURVA.



NSEG (20)

H).- /ESTRUCTURA " NIVEL " /

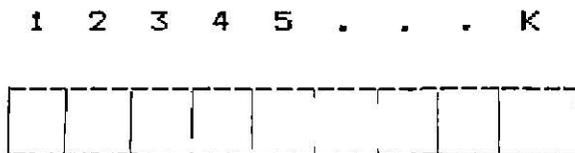
AQUI SE ALMACENA EL VALOR DE ZO QUE CORRESPONDE A CADA CURVA DE CONTORNO:



NIVEL (20)

I).- /ESTRUCTURA " VEC " /

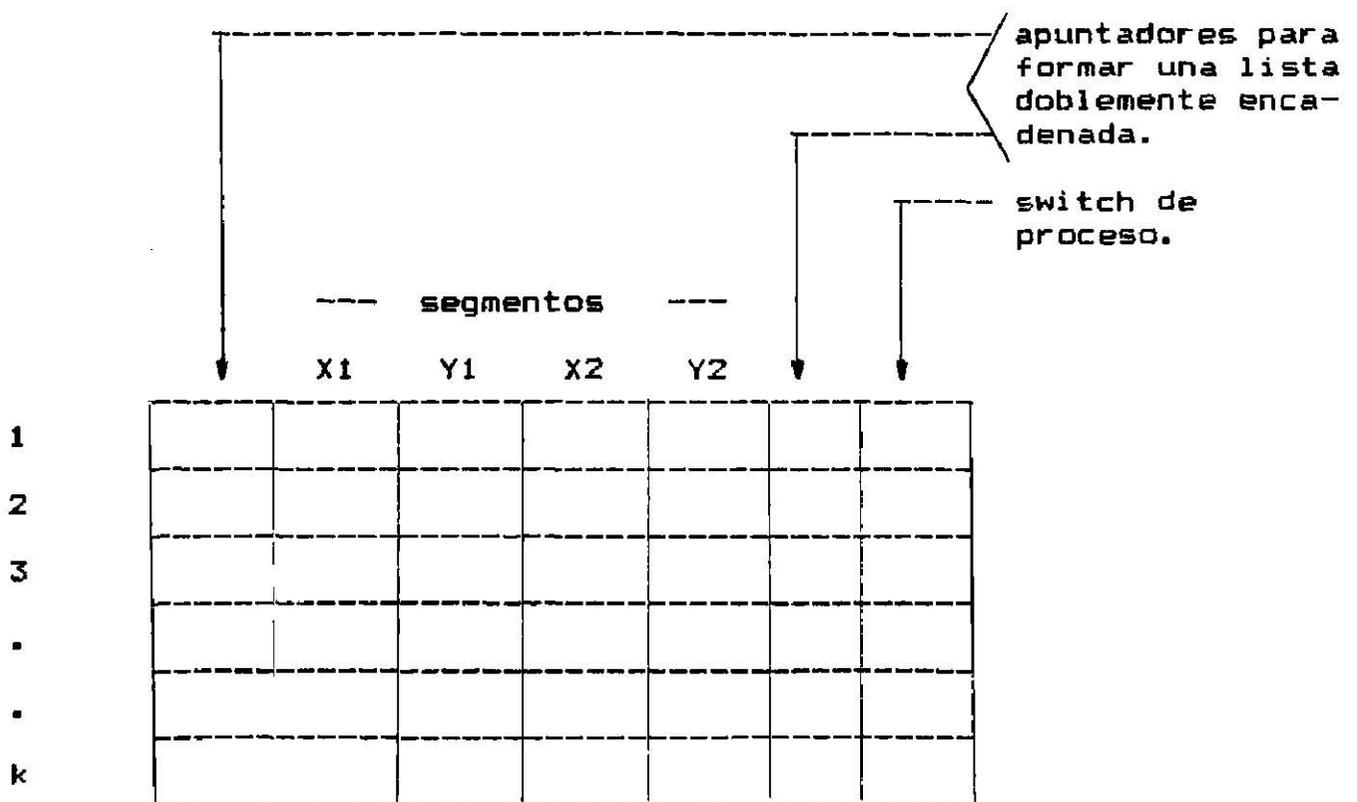
ES LA ESTRUCTURA DE COMUNICACION CON LA SUBROUTINA DE INTERFASE GRAFICA. PUEDE CONTENER UN SEGMENTO ,EN EL CASO DE LA SALIDA SIMPLE, O UNA CADENA EN EL CASO DE LA SALIDA COMPLEJA.



VEC (300)

J).- / ESTRUCTURA " SELEC " /

ESTA ESTRUCTURA ALMACENA UNA CURVA DE CONTORNO
DEFINIENDO UNA LISTA DOBLEMENTE ENCADENADA QUE PER-
MITE EXTRAER LAS CURVAS EN FORMA DE CADENAS Y ASI
SATISFACER LOS REQUERIMIENTOS DE LA SALIDA COMPLEJA.



SELEC (100,7)

X.- DISEÑO DE MODULARIDAD.

PARA FACILITAR LA IMPLEMENTACION DEL SOFTWARE SE DISEÑO UN ESQUEMA MODULAR QUE SE MUESTRA EN EL SIGUIENTE DIAGRAMA:

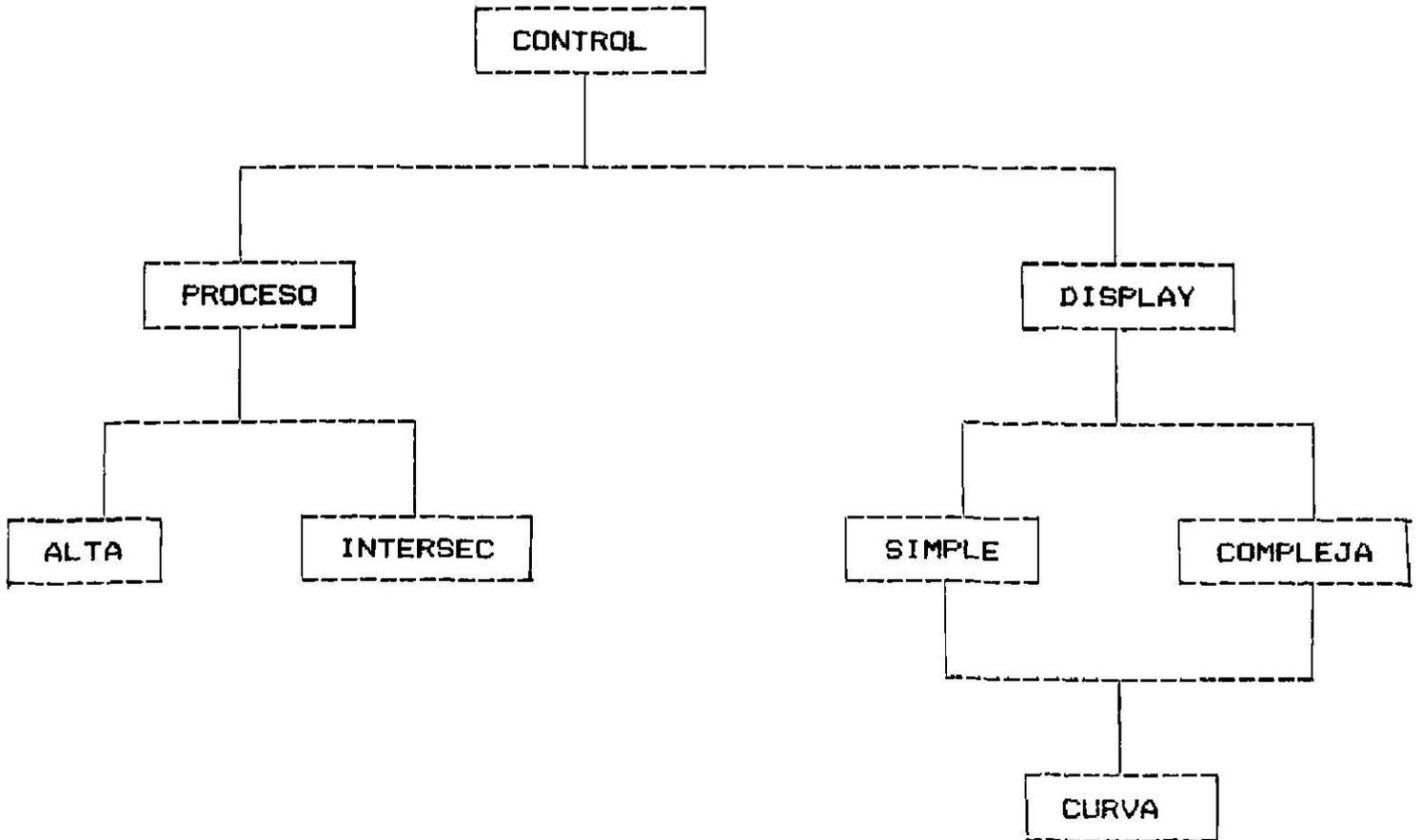


FIG. 10
DIAGRAMA DE MODULARIDAD
=====

X1.- DISEÑO DE MODULOS.

PARA EL DISEÑO DE LOS MODULOS QUE CONFORMAN EL SISTEMA " CONTORNO.SYS ", SE SIGUIO UNA TECNICA QUE CONSISTE EN LO SIGUIENTE:

1.- Describir el o los objetivos del módulo.

2.- Diseñar el algoritmo en un lenguaje precomputacional.

LA VENTAJA DE ESTE ESTILO CON RESPECTO AL DISEÑO MEDIANTE DIAGRAMAS DE FLUJO, ES QUE LA ESCRITURA DEL CODIGO FUENTE SE HACE MAS DIRECTAMENTE.

SIGUIENDO ESTE PATRON ES COMO SE PRESENTAN LOS DIFERENTES MODULOS DE " CONTORNO.SYS ".

X1-A.- MODULO DE CONTROL.

OBJETIVO: CONDUCIR LA ESTRATEGIA GENERAL.

ALGORITMO:

1.- Alimentar Zmax , Zmin , NCS.

/ CALCULO DE "DZ" PARA DEFINIR NIVELES/

2.- $DZ = (Zmax - Zmin) / (NCS + 1)$

3.- $Z0 = Zmin + DZ$

/ ITERACION PARA PROCESO DE NIVELES/

4.- NC = 1 A NCS

4.1.- NIVEL (NC) = Z0

4.2.- ALIMENTAR (X(I), Y(I), Z(I), I=1,3)

4.3.- Si EOF ENTONCES IR A "4.6"

4.4.- CALL PROCESO

4.5.- IR A "4.2"

4.6.- $Z0 = Z0 + DZ$

5.- CALL DISPLAY

6.- FIN

X1-B.- MODULO DE PROCESO.

OBJETIVO: DETERMINAR QUE TIPO DE ACCION DEBE TOMARSE
CON EL ELEMENTO TRIANGULAR EN PROCESO.

ADMINISTRA TAMBIEN EL EL CALCULO DEL SEG-
MENTO DE INTERSECCION CUANDO ESTA EXISTE, ASI
COMO SU INCORPORACION A LA ESTRUCTURA MAESTRA
" CURV".

BASICAMENTE SE DESARROLLAN TRES SUBPROCE-
SOS:

1.- DEFINICION DE " CASO "

2.- UBICACION DE " CASO " EN LA TABLA
DE DECISION " DEC" , PARA DEFINIR
LA ACCION.

3.- EJECUCION DE LA ACCION DETERMINADA
EN EL PASO ANTERIOR.

ALGORITMO:

1.- / DEFINICION DE "CASO" /.

1.1.- $\left[\begin{array}{l} I = 1 \text{ A } 3 \\ 1.1.A.- \text{ CASO}(I) = 0 \end{array} \right.$

1.2.- $\left[\begin{array}{l} I = 1 \text{ A } 3 \\ 1.2.A.- \text{ Si } Z(I) > Z_0 \text{ ENTONCES CASO}(I) = \text{CASO}(I) + 1 \\ 1.2.B.- \text{ Si } Z(I) < Z_0 \text{ ENTONCES CASO}(I) = \text{CASO}(I) + 1 \\ 1.2.C.- \text{ Si } Z(I) = Z_0 \text{ ENTONCES CASO}(I) = \text{CASO}(I) + 1 \end{array} \right.$

2.-/ UBICACION EN TABLA DE DECISION /

2.1 $\left[\begin{array}{l} I = 1 \text{ A } 10 \\ 2.1.A.- \left[\begin{array}{l} J = 1 \text{ A } 3 \\ \text{Si CASO}(J) \neq \text{DEC}(I, J) \text{ ENTONCES SIG. } I \end{array} \right. \\ 2.1.B.- \text{ K} = \text{DEC}(I, 4) \\ 2.1.C.- \text{ IR A "3"} \end{array} \right.$

3.- IR A EJECUTAR ACCION K.

4.- / ACCIONES /

ACCION 1:

1.- RETURN

ACCION 2:

1.- L=0

2.- I=1 A 3

2.1.-	Si $Z(I) = Z_0$ ENTONCES:	2.1.A.- $L=L+1$
		2.1.B.- $SEG(1,L)=X(I)$
		2.1.C.- $SEG(2,L)=Y(I)$

3.- CALL ALTA

4.- RETURN

ACCION 3:

1.-	I=1 A 3
	Si $Z(I) < Z_0$ ENTONCES $K=I$

2.- L=0

3.- I=1 A 3

	Si $I = K$ ENTONCES SIGUIENTE I
	CALL INTERSEC($X(K), Y(K), Z(K), X(I), Y(I), Z(I), X_0, Y_0, Z_0$)
	$L=L+1$
	$SEG(1,L)=X_0$
	$SEG(2,L)=Y_0$

4.- CALL ALTA

5.- RETURN

```
1.- I=1 A 3
   SI Z(I) > ZO ENTONCES K=I

2.- L=0

3.- I=1 A 3
   SI I= K ENTONCES SIGUIENTE I
   CALL INTERSEC(X(K),Y(K),Z(K),X(I),Y(I),Z(I),XO,YO,ZO)
   L=L+1
   SEG(1,L)=XO
   SEG(2,L)=YO

4.- CALL ALTA

5.- RETURN
```

ACCION 5:

```
1.- I=1 A 3
   K=I
   SEG(1,1)=X(K)
   SEG(2,1)=Y(K)
   SI I=3 ENTONCES K=0
   SEG(1,2)=X(K+1)
   SEG(2,2)=Y(K+1)
   CALL ALTA

2.- RETURN
```

ACCION 6:

```
1.- I=1 A 3
   |
   | Si Z(I) = ZO ENTONCES: SEG(1,1)=X(I)
   |                               SEG(2,1)=Y(I)
   |
   | Si Z(I) > ZO ENTONCES K=I
   |
   | Si Z(I) < ZO ENTONCES L=I
   |
2.- CALL INTERSEC(X(K),Y(K),Z(K),X(L),Y(L),Z(L),XO,YO,ZO)
3.- SEG(1,2)=XO
4.- SEG(2,2)=YO
5.- CALL ALTA
6.- RETURN
```

XI-C.- MODULO ALTA.

OBJETIVO: INSERTAR UN SEGMENTO DEFINIDO EN LA ESTRUCTURA
"SEG" EN LA ESTRUCTURA MAESTRA " CURV " .

ALGORITMO:

1.- / INICIALIZACION DE ESTRUCTURA "NSEG", CONTADORA
DE SEGMENTOS POR NIVEL /.

Si SW=0 ENTONCES: $\left[\begin{array}{l} 1.1.- \quad \left[\begin{array}{l} I=1 \text{ A } NCS \\ NSEG(I)=0 \end{array} \right. \\ \\ 1.2.- \text{ RETURN} \end{array} \right.$

2.-/ INCORPORACION DE UN SEGMENTO /

2.1.- NSEG(NC)=NSEG(NC)+1

2.2.- L=NSEG(NC)

2.3.- K=0

2.4.- $\left[\begin{array}{l} J=1 \text{ A } 2 \\ \\ I=1 \text{ A } 2 \\ \\ K=K+1 \\ \\ CURV(K,L,NC)=SEG(I,J) \end{array} \right.$

2.5.- RETURN

XI-D.- MODULO "INTERSEC."

OBJETIVO: DADO UN SEGMENTO DEFINIDO POR (X_1, Y_1, Z_1) Y
POR (X_2, Y_2, Z_2) , CALCULAR EL PUNTO DEFINIDO
POR (X_0, Y_0, Z_0) PARA UN VALOR DADO DE Z_0 .

EL DISEÑO DE ESTE MODULO SE BASA EN EL
DESARROLLO HECHO EN VIII.

ALGORITMO:

- 1.- $T = (Z_0 - Z_1) / (Z_2 - Z_1)$
- 2.- $X_0 = X_1 + (X_2 - X_1) * T$
- 3.- $Y_0 = Y_1 + (Y_2 - Y_1) * T$
- 4.- RETURN

XI-E.- MODULO "DISPLAY"

OBJETIVO: PERMITIR AL USUARIO DECIDIR QUE TIPO
DE SALIDA PREFIERE:

- SIMPLE.
- COMPLEJA.

ESTA DECISION SE TOMA INTERACTIVA -
MENTE.

ALGORITMO:

- 1.- IMPRIMIR " Que tipo de salida:
 - 1.- simple.
 - 2.- compleja. "
- 2.- ALIMENTAR OPCION : K
- 3.- Si $K = 1$ ENTONCES CALL SIMPLE
- 4.- Si $K = 2$ ENTONCES CALL COMPLEJA
- 5.- Si $K \neq 1$ o $K \neq 2$ ENTONCES IR A "1"
- 6.- RETURN

XI-F .- MODULO " SIMPLE "

OBJETIVO: DE AQUI SE ENVIA AL MODULO " CURVA "
 UNO POR UNO, CADA SEGMENTO DE LA ES -
 TRUCTURA " CURV ", PARA QUE SEAN PIN -
 TADOS POR EL DISPOSITIVO GRAFICO EN
 USO.

SE CONSIDERA UN SISTEMA GRAFICO
 EN 3-D Y LOS SEGMENTOS SE ENVIAN POR
 MEDIO DEL VECTOR " VEC ".

ALGORITMO:

```

1.- [ NC = 1 A NCS
      [ J = 1 A NSEG (NC)
        VEC (1) = CURV (1, J, NC)
        VEC (2) = CURV (2, J, NC)
        VEC (3) = NIVEL (NC)
        VEC (4) = CURV (3, J, NC)
        VEC (5) = CURV (4, J, NC)
        VEC (6) = NIVEL (NC)
        CALL CURVA
      ]
    ]
2.- RETURN
  
```

XI-G.- MODULO " COMPLEJA "

OBJETIVO: SE ENVIAN LOS SEGMENTOS AL MODULO
" CURVA " , PERO NO COMO ELEMENTOS
AISLADOS, SINO COMO CADENAS YA SEA
ABIERTAS O CERRADAS. ESTO PRESENTA
VENTAJAS TALES COMO:

- MAYOR EFICIENCIA EN SISTEMAS
GRAFICOS SOFISTICADOS.
- MAYOR FACILIDAD PARA LA ETI-
QUETACION AUTOMATICA.
- FACILIDAD PARA APLICAR ALGORITMOS
DE SUAVIZACION.

LAS CADENAS SE DEFINEN EN EL VECTOR " VEC " , QUE
ALMACENA UN CONJUNTO DE PUNTOS EN 3-D.

ALGORITMO:

1.- / SELECCION DE CURVA "NC" /

1.1.- K= NSEG(NC)

1.2.- J= 1 A K

I=1 A 4

SELEC(J, I+1)=CURV(I, J)

2.- / INICIALIZACION DE APUNTAORES /

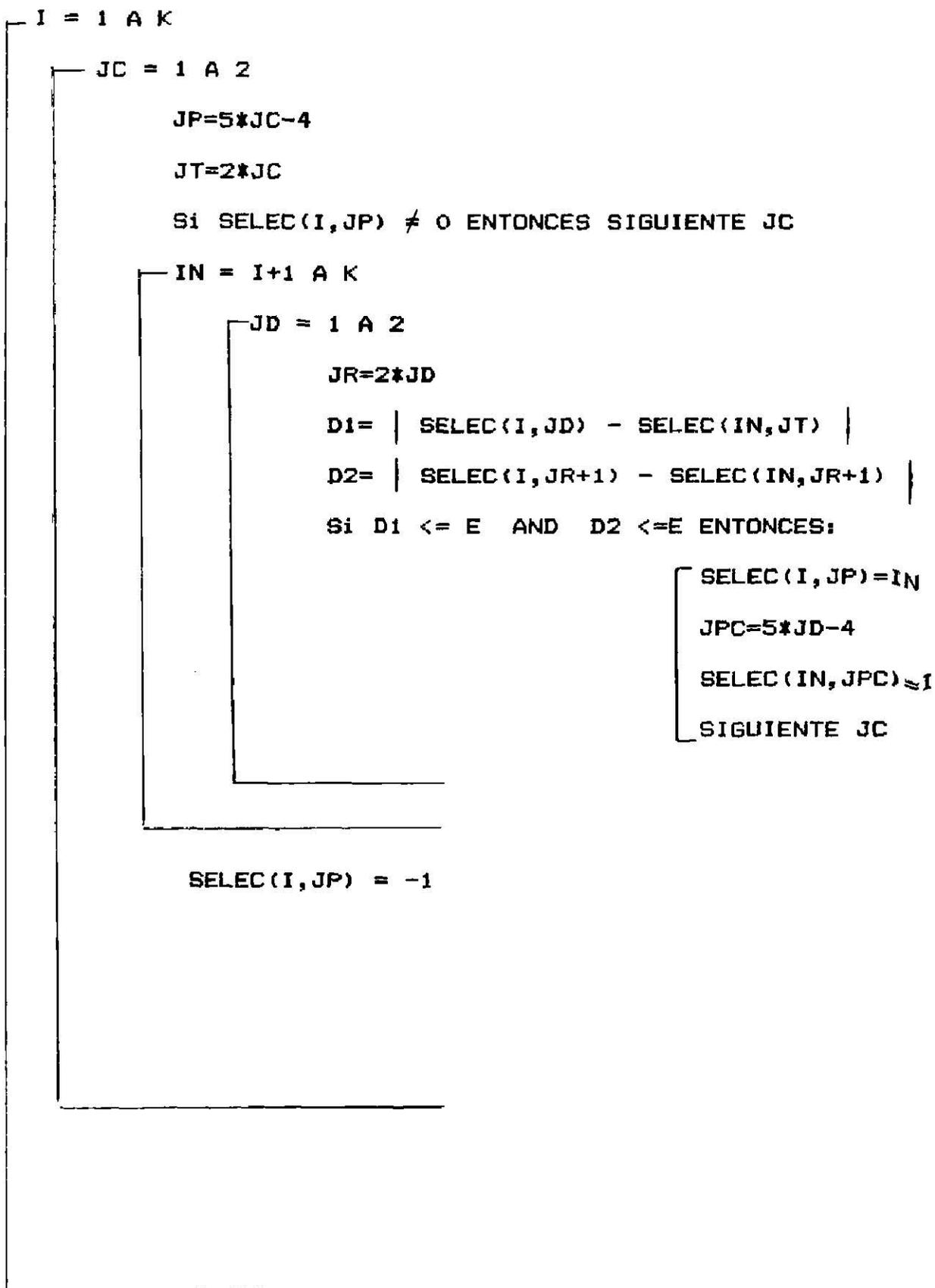
2.1.- I= 1 A K

SELEC(I, 1)=0

SELEC(I, 6)=0

SELEC(I, 7)=0

3.- / FORMACION DE LISTA DOBLEMENTE ENCADENADA /



4.- / EXTRACCION DE CADENAS ABIERTAS /

I = 1 A K

SI SELEC (I,7) < 0 ENTONCES SIGUIENTE I

JC = 1 A 2

4.1.- JP=5*JC-4

JA=11-5*JC

JD=2*JC

Si SELEC(I,JP) ≠ -1 ENTONCES SIGUIENTE JC

4.2.- L=0

L=L+1

VEC(L)=SELEC(I,JD)

L=L+1

VEC(L)=SELEC(I,JD+1)

L=L+1

VEC(L)=NIVEL(NC)

Si JC=2 ENTONCES JD=0

L=L+1

VEC(L)=SELEC(I,JD+2)

L=L+1

VEC(L)=SELEC(I,JD+3)

4.3.- SELEC(I,7)=-1

IR=I

IC=SELEC(IR,JA)

4.4.- SI IC= -1 ENTONCES IR A " 4.8"

I1=SELEC(IC,1)

I2=SELEC(IC,6)

4.5.- SI I1 = IR ENTONCES:

L=L+1

VEC(L)=SELEC(IC,4)

L=L+1

VEC(L)=SELEC(IC,5)

L=L+1

VEC(L)=NIVEL(NC)

JZ=6

4.6.- SI I2 = IR ENTONCES:

L=L+1

VEC(L)=SELEC(IC,2)

L=L+1

VEC(L)=SELEC(IC,3)

L=L+1

VEC(L)=NIVEL(NC)

JZ=1

4.7.- SELEC(IC,7) = -1

IR=IC

IC=SELEC(IC,JZ)

IR A " 4.4"

4.8.- CALL CURVA

5.- / EXTRACCION DE CADENAS CERRADAS /

I = 1 A K

5.1.- SI SELEC(I,7) < 0 ENTONCES SIGUIENTE I

L=0

L=L+1

VEC(L)=SELEC(I,2)

L=L+1

VEC(L)=SELEC(I,3)

L=L+1

VEC(L)=NIVEL(NC)

L=L+1

VEC(L)=SELEC(I,4)

L=L+1

VEC(L)=SELEC(I,5)

L=L+1

VEC(L)=NIVEL(NC)

5.2.- SELEC(I,7) = -1

IR = I

IC = SELEC(IR,6)

5.3.- Si IC = -1 ENTONCES IR A "5.7"

I1 = SELEC(IC,1)

I2 = SELEC(IC,6)

5.4.- Si I1 = IR ENTONCES :

L=L+1
VEC(L)=SELEC(IC,4)
L=L+1
VEC(L)=SELEC(IC,5)
L=L+1
VEC(L)=NIVEL(NC)
JZ=6

5.5.- Si I2 = IR ENTONCES:

L=L+1
VEC(L)=SELEC(IC,2)
L=L+1
VEC(L)=SELEC(IC,3)
L=L+1
VEC(L)=NIVEL(NC)
JZ=1

5.6.- SELEC(IC,7) = -1
 IR = IC
 IC = SELEC(IC,JZ)
 IR A " 5.3 "

5.7.- CALL CURVA

X1-6.- MODULO " CURVA "

OBJETIVO: OBTENER LA IMAGEN GRAFICA DE LAS
CURVAS DE CONTORNO POR MEDIO DE
EL DISPOSITIVO GRAFICO DISPONIBLE.

ESTE MODULO ES PROPIAMENTE LA INTERFASE GRAFICA QUE DEBE
DISEÑARSE DE ACUERDO A LOS RECURSOS GRAFICOS CON QUE SE
CUENTE.

EN ESTE TRABAJO SE UTILIZO EL SISTEMA GRAFICO INTERGRAPH
QUE ES UN SISTEMA SOFISTICADO QUE PERMITE GRAFICAR EN 3-D. Y
CUENTA CON RUTINAS LLAMABLES DESDE FORTRAN, QUE PERMITEN OBTENER
UN ARCHIVO GRAFICO QUE PUEDE VISUALIZARSE EN PANTALLA O BIEN
ENVIARSE A UN GRAFICADOR.

XII.- IMPLEMENTACION DEL SOFTWARE.

1.- DESCRIPCION DEL MEDIO AMBIENTE:

"CONTORNO.SYS" SE IMPLEMENTO EN UN MEDIO DONDE LOS RECURSOS DE HARDWARE SON:

- A).- VAX 11/750
- B).- 2 TERMINALES DE TRABAJO INTERACT
- C).- GRAFICADOR HP 7585-B.
- D).- 2 TERMINALES ALFANUMERICAS
TELEVIDEO.

2.- SOFTWARE DE APOYO : EN LAS CORRIDAS QUE SE HICIERON PARA PODER PROBAR "CONTORNO.SYS", SE UTILIZARON ALGUNOS PROGRAMAS AUXILIARES:

- A).- MALLA.FOR: EFECTUA UNA DISCRETIZACION TRIANGULAR DE UNA GEOMETRIA BIDIMENSIONAL.
- B).- CONVERT.FOR: GENERA UNA SUPERFICIE TRIANGULARIZADA EN BASE A LA DISCRETIZACION HECHA POR MALLA.FOR. Y UNA FUNCION $Z=f(x,y)$ ESPECIFICADA POR EL USUARIO.

LA SALIDA DE ESTE PROGRAMA PUEDE USARSE DIRECTAMENTE COMO ENTRADA DE CONTORNO.SYS.

C).- IGDS. (Interactive Graphics Design Systems): ES UN SISTEMA GRAFICO PARTICULAR DE UN MEDIO AMBIENTE INTERGRAPH, CUENTA ENTRE OTRAS COSAS, CON UN CONJUNTO DE SUBROUTINAS QUE SON LLAMABLES DESDE FORTRAN-77 Y QUE PERMITEN LA CREACION DE FORMAS EN UN ARCHIVO GRAFICO QUE DESPUES PUEDE VISUALIZARSE EN UNA TERMINAL INTERACT.

3.- LENGUAJE FUENTE: SE DETERMINO QUE EL LENGUAJE MAS ADECUADO PARA IMPLEMENTAR EL SOFTWARE, ES EL LENGUAJE FORTRAN-77 POR LAS SIGUIENTES RAZONES:

A).- EN LA ACTUALIDAD ESTE LENGUAJE ES EL DE MAYOR USO EN APLICACIONES DE INGENIERIA. ESTO PERMITE QUE ESTE SOFTWARE PUEDA SER USADO COMO MODULO POR OTROS SISTEMAS.

B).- LA ESTANDARIZACION DEL LENGUAJE PERMITE SU TRANSPORTACION A OTRAS COMPUTADORAS SIN MAYORES DIFICULTADES.

XIII.- CORRIDAS DE PRUEBA

PARA TENER UNA PRUEBA OBJETIVA DE QUE EL SOFTWARE " CONTCRNC.SYS " FUNCIONA CORRECTAMENTE, SE DISEÑO UNA SERIE DE CORRIDAS DE PRUEBA USANDO SUPERFICIES MATEMÁTICAS TÍPICAS , QUE PERMITIERAN JUZGAR LA CALIDAD DE LOS RESULTADOS

SE ELIGIERON LAS SIGUIENTES SUPERFICIES:

$$1.- z = x^2 + y^2 \quad (\text{paraboloide elíptico})$$

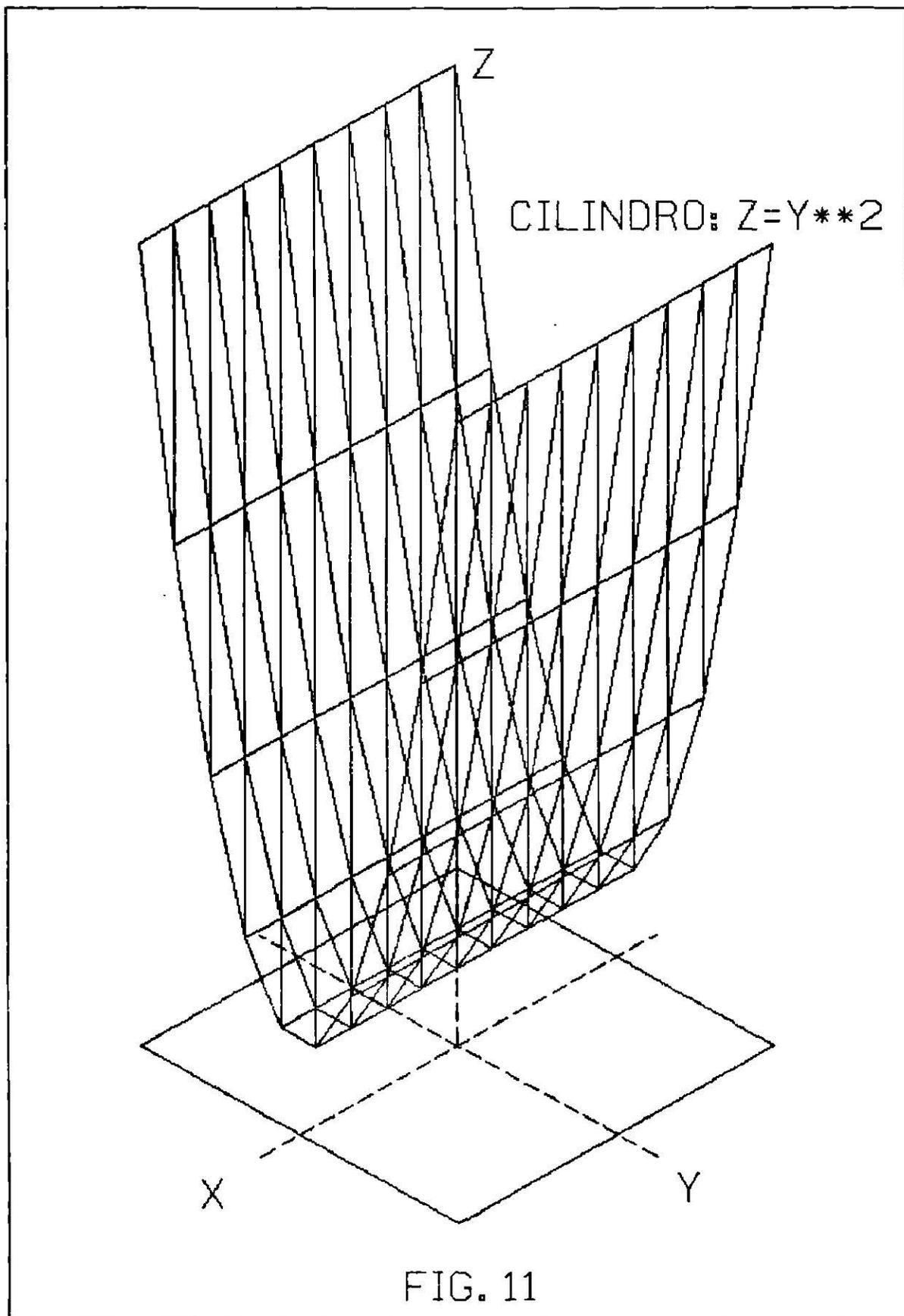
$$2.- z = y^2 \quad (\text{cilindro parabólico})$$

EN AMBOS CASOS SE ELIGIO COMO DOMINIO DE DEFINICION

LA REGION ESPECIFICADA POR:

$$R: \{ -4 < x < 4 \ ; \ -4 < y < 4 \}$$

EN LAS FIGURAS 11 Y 12 SE MUESTRAN LAS GRAFICAS DE ESTAS SUPERFICIES



PARABOLOIDE CIRCULAR

$$Z = X^2 + Y^2$$

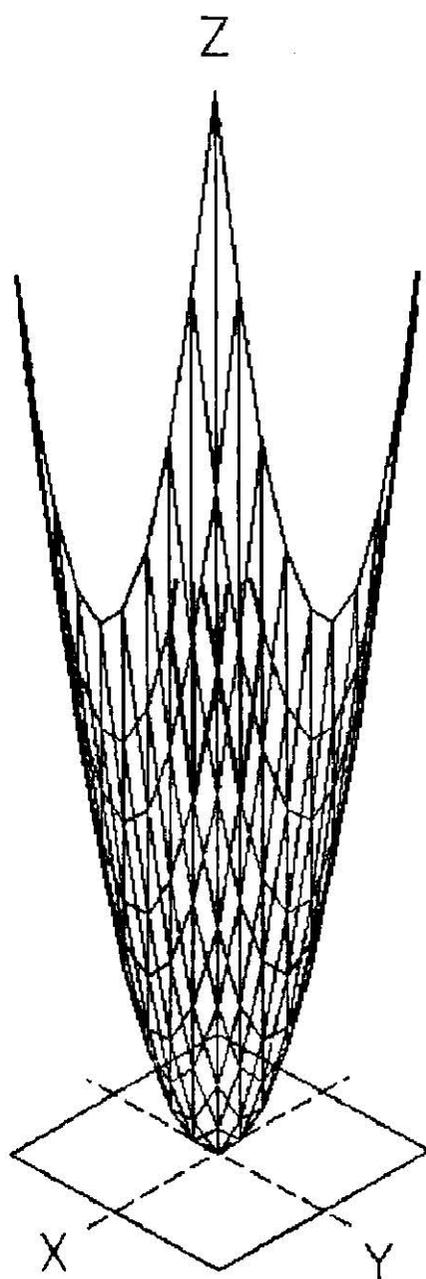


FIG. 12

LAS CORRIDAS SE EFECTUARON SIGUIENDO LA SECUENCIA MOSTRADA EN LA FIGURA 13

DE DECIDIÓ GENERAR 5 CURVAS DE CONITORNO PARA CADA SUPERFICIE, LA ELECCION DEL TIPO DE SALIDA (SIMPLE O COMPLEJA) NO AFECTA LA IMAGEN OBTENIDA, PERO SI EL TIEMPO DE EJECUCION YA QUE LA SALIDA SIMPLE ES SENSIBLEMENTE MAS LENTA.

LOS RESULTADOS DE LAS CORRIDAS SE MUESTRAN Y SE COMENTAN EN LAS FIGURAS DE LA 14 A LA 17.

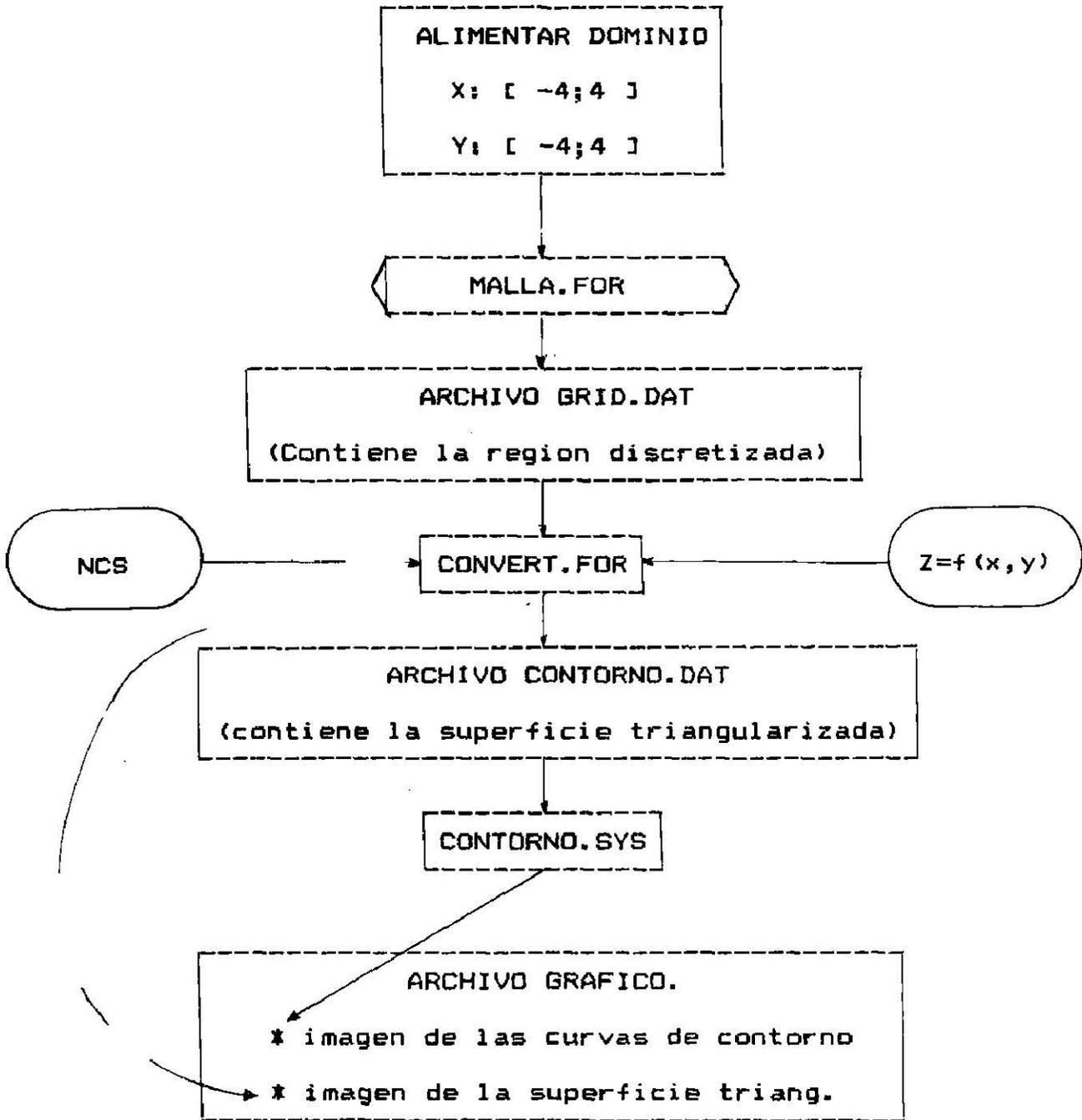
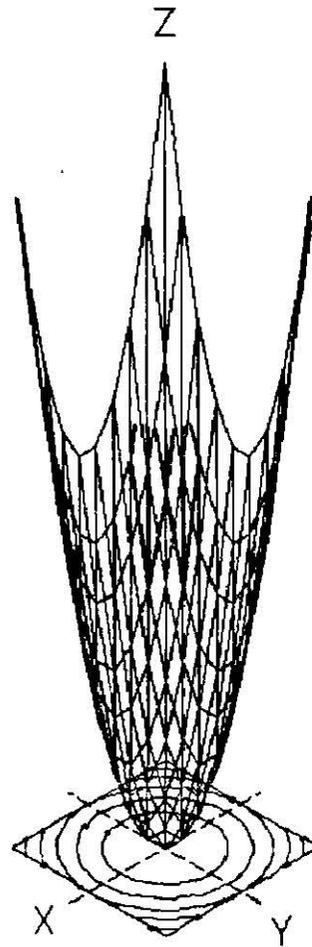


FIG. 13

SECUENCIA EN LAS CORRIDAS DE PRUEBA

PARABOLOIDE CIRCULAR

$$Z = X^2 + Y^2$$



CURVAS DE CONTORNO

FIG. 14

CURVAS DE CONTORNO
EN EL PLANO 'XY'
DEL PARABOLOIDE CIRCULAR
 $Z=X**2 + Y**2$

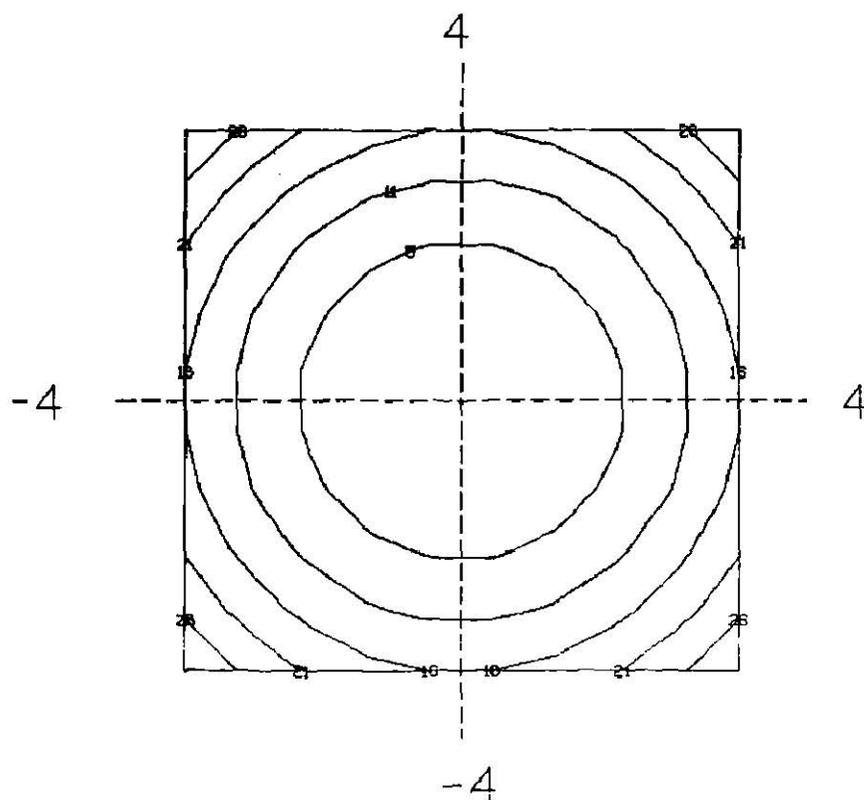


FIG. 15

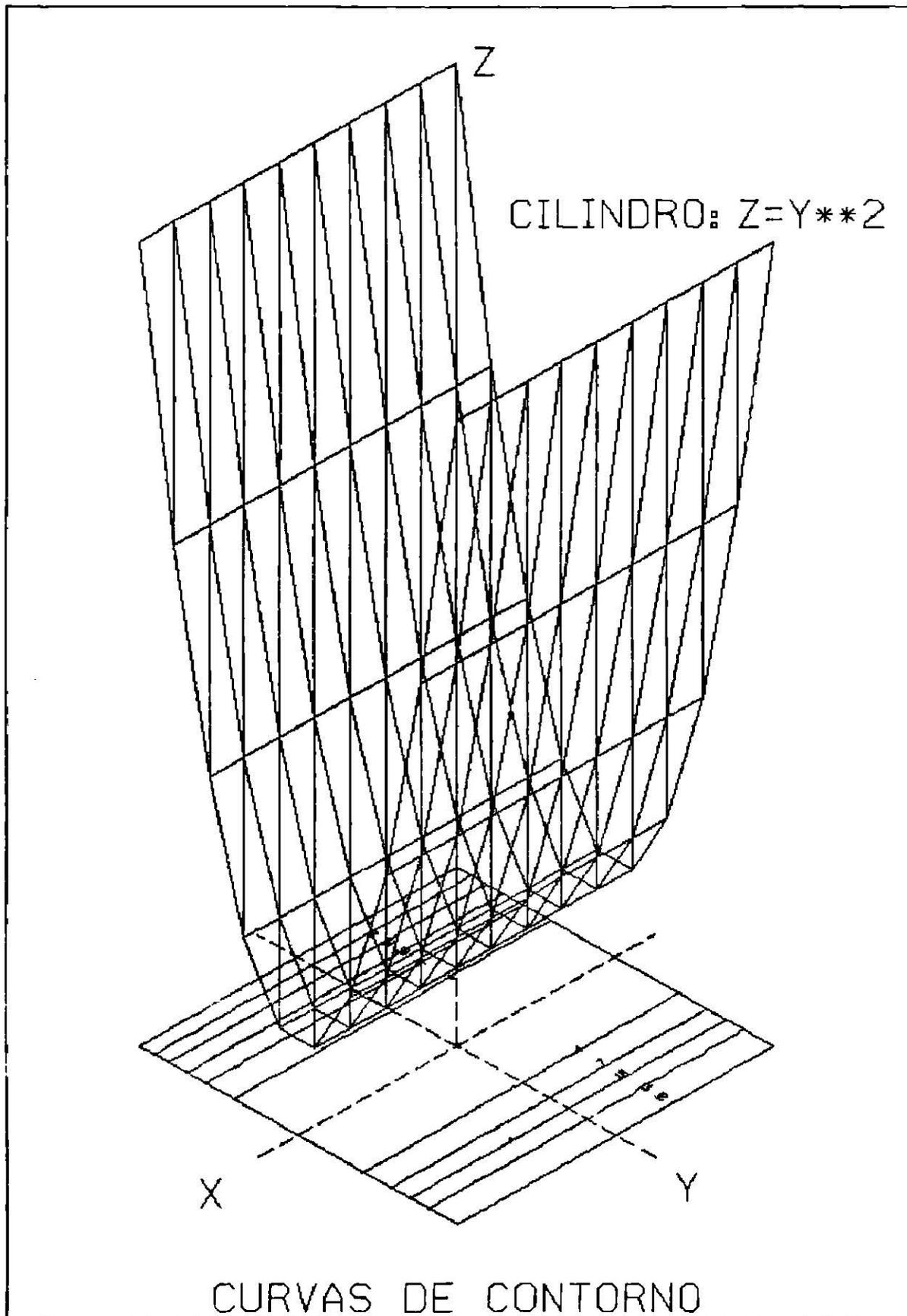


FIG. 16

CURVAS DE CONTORNO
DEL CILINDRO: $Z = Y^2 + 2$

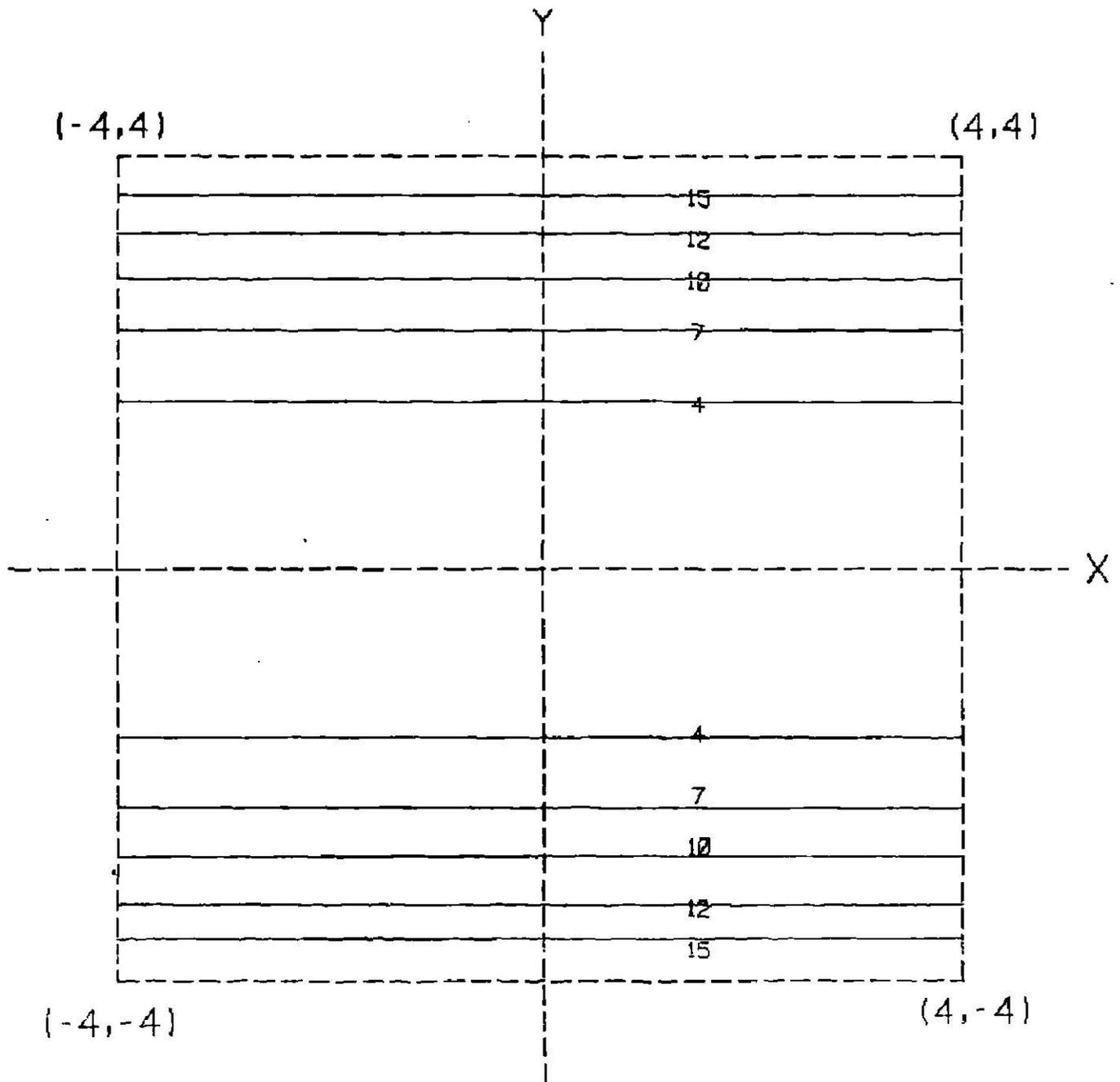


FIG. 17

XIII COMENTARIOS Y CONCLUSIONES

1. - LOS RESULTADOS DE LAS CORRIDAS MUESTRAN QUE
CONTORNO.SYS TRABAJA SATISFACTORIAMENTE.
2. - EN ESTE CASO, AL USAR UN SISTEMA GRAFICO SOFIS-
TICADO, LA SALIDA COMPLEJA ES SIGNIFICATIVAMENTE
MAS EFICIENTE QUE LA SALIDA SIMPLE.
3. - LA CALIDAD DE LAS CURVAS DEPENDE PRINCIPALMENTE
DEL GRADO DE DISCRETIZACION DE LA SUPERFICIE.
4. - EL TIEMPO DE PROCESO DEPENDE PRINCIPALMENTE DE
LA NATURALEZA DE LA SUPERFICIE, EL GRADO DE DIS-
CRETIZACION Y DEL TIPO DE SALIDA ELIGIDA.

XV.- MEJORAS FUTURAS RECOMENDADAS.

- 1.- CONSIDERAR SUPERFICIES DISCRETIZADAS EN ELEMENTOS DISTINTOS A LOS TRIANGULARES.
- 2.- DESARROLLAR INTERFASES GRAFICAS PARA OTROS SISTEMAS GRAFICOS (EJ: TERMINAL TEKTRONIX)
- 3.- DESARROLLAR UNA RUTINA DE SUAVIZACION PARA MEJORAR LA CALIDAD DE LAS CURVAS.
- 4.- DESARROLLAR UNA RUTINA DE ETIQUETACION AUTOMATICA.

APENDICE 3

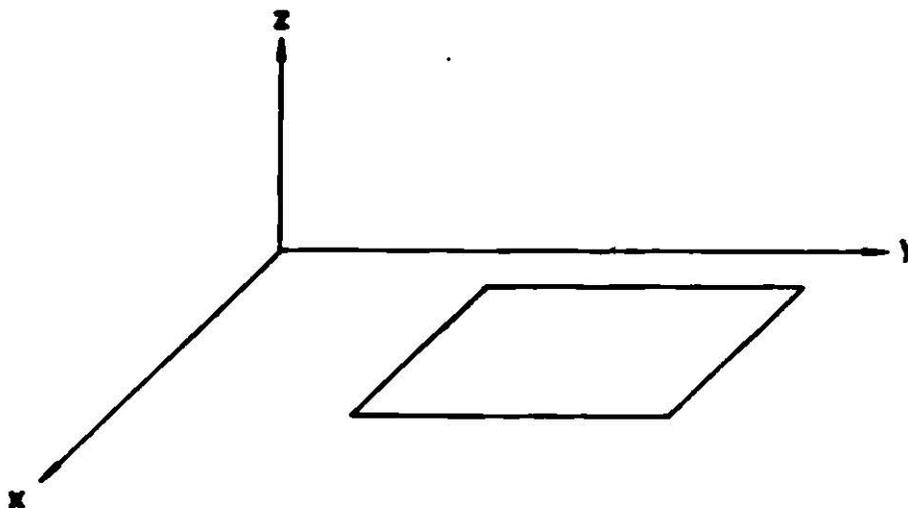
- - - - -

METODOLOGIA PARA GENERAR UNA SUBESTRUCTURA
TRIANGULARITADA

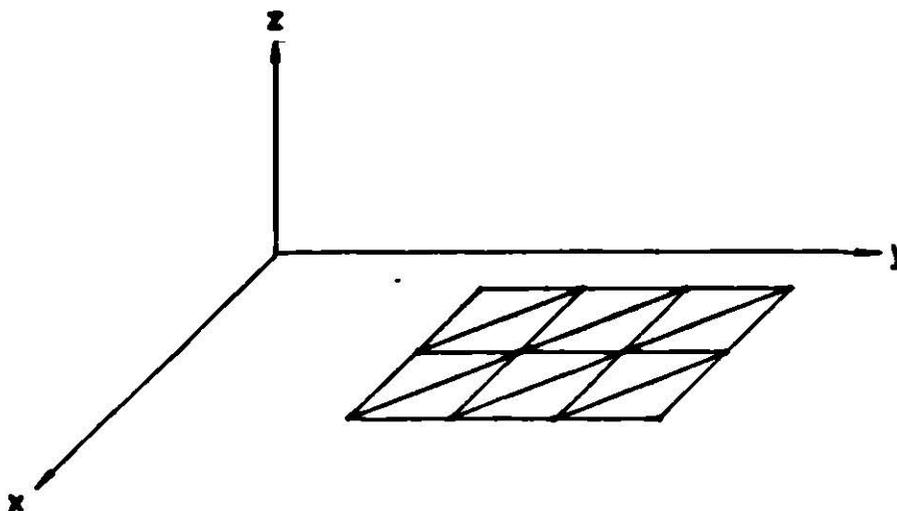
APENDICE A

METODOLOGIA PARA TRIANGULARIZAR UNA SUPERFICIE

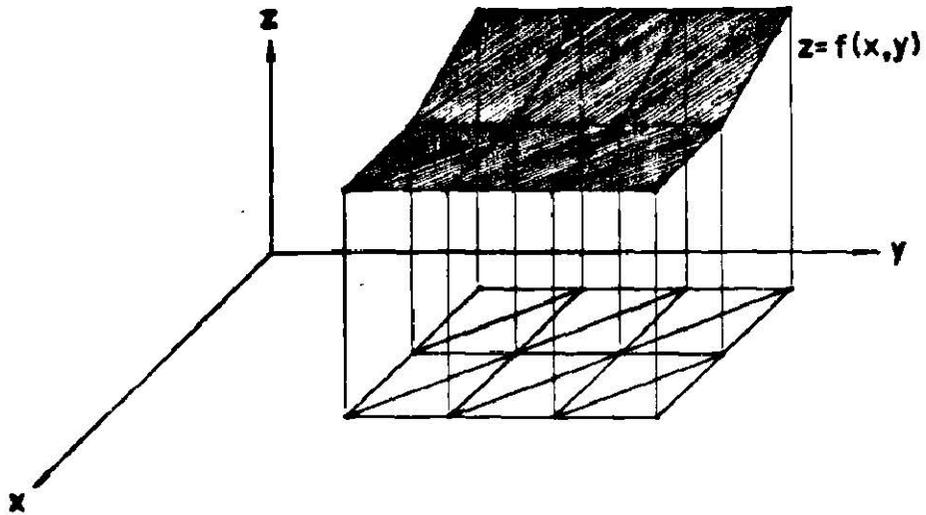
PASO 1. ESPECIFICACION DE UNA REGION EN EL PLANO "XY"



PASO 2: DISCRETIZACION DE LA REGION EN TRIANGULOS.



PASO 3: EVALUACION DE $Z = F(x,y)$ EN CADA NODO-



AFI 'DIDE B

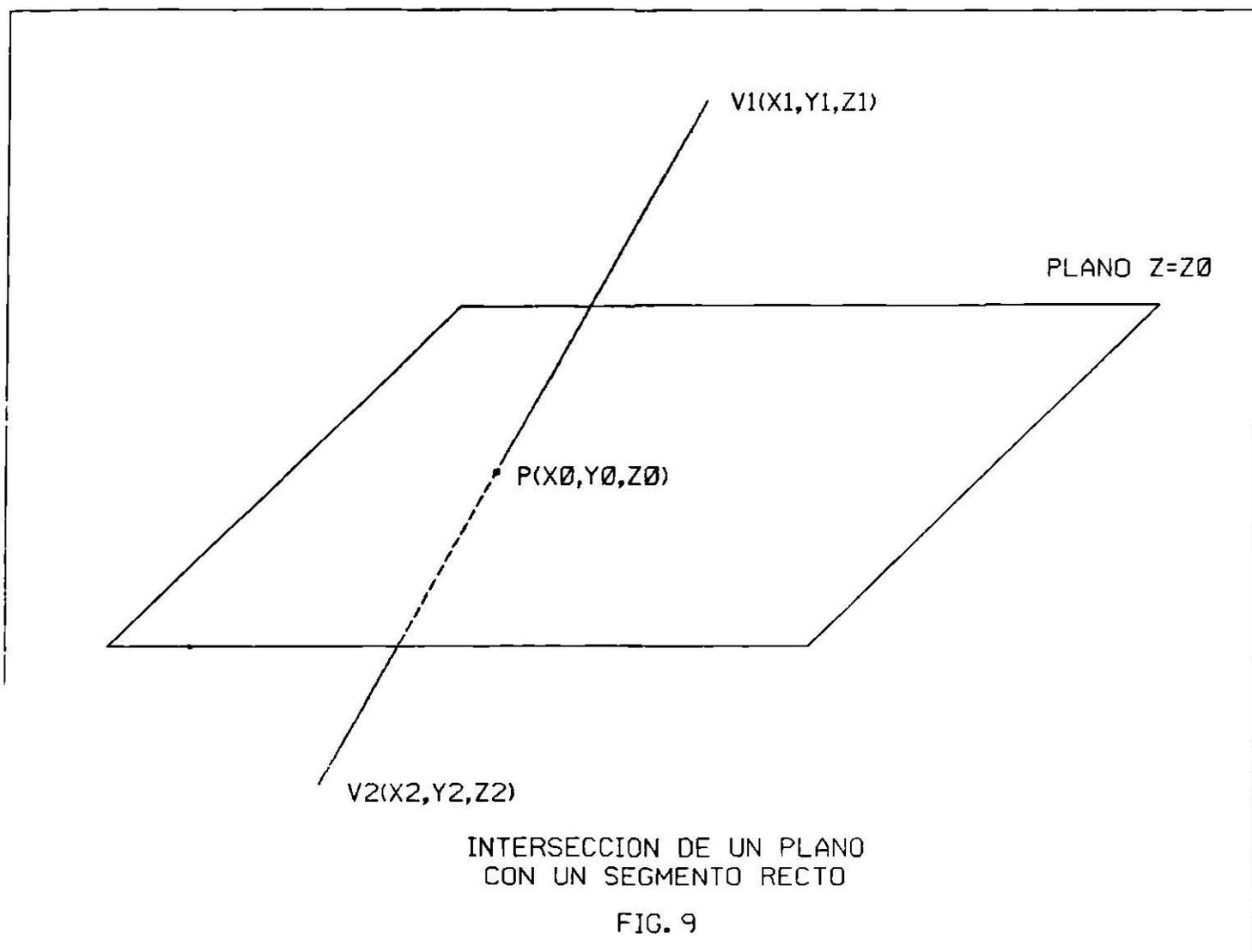
-

INTERSECCION DE UN PLANO CON UN SEGMENTO RECTO

- DESARROLLO MATEMATICO PARA EL CALCULO
DE LOS PUNTOS DE INTERSECCION.

EN VARIAS DE LAS ACCIONES DESCRITAS ANTERIORMENTE , SE PRE-
SENTA LA NECESIDAD DE CONOCER LOS PUNTOS DE INTERSECCION DE UN
PLANO HORIZONTAL $Z=Z_0$, AL QUE HEMOS LLAMADO NIVEL, CON UNO DE
LOS LADOS DEL ELEMENTO TRIANGULAR INTERSECTADO.

GEOMETRICAMENTE PODEMOS VISUALIZAR LA SITUACION COMO SE
MUESTRA EN LA FIG. 9.



DE MODO QUE SE DEBEN ENCONTRAR VALORES DE x_0 Y y_0 , TALES QUE EL PUNTO (x_0, y_0, z_0) PERTENEZCAN AL SEGMENTO $\overline{V_1V_2}$.

LA RECTA QUE CONTIENE AL SEGMENTO $\overline{V_1V_2}$. TIENE LA REPRESENTACION PARAMETRICA:

$$X = X_1 + a t$$

$$Y = Y_1 + b t$$

$$Z = Z_1 + c t$$

DONDE (x_1, y_1, z_1) ES UN PUNTO CONOCIDO DE LA RECTA, Y $[a , b , c]$ ES UN VECTOR CUYA DIRECCION ES LA MISMA QUE LA DE LA RECTA.

DADO QUE CONOCEMOS DOS PUNTOS $V_1(x_1, y_1, z_1)$ Y $V_2(x_2, y_2, z_2)$ PODEMOS ENCONTRAR UN VECTOR CUYA DIRECCION SEA LA MISMA QUE LA DEL SEGMENTO:

$$[a , b , c] = [(x_2 - x_1) , (y_2 - y_1) , (z_2 - z_1)]$$

SUBSTITUYENDO ESTA EQUIVALENCIA EN LA ECUACION PARAMETRICA DE LA RECTA, Y USANDO EL PUNTO $V_1(x_1, y_1, z_1)$. AHORA LA ECUACION PODEMOS EXPRESARLA ASI:

$$X = X_1 + (X_2 - X_1) t$$

$$Y = Y_1 + (Y_2 - Y_1) t$$

$$Z = Z_1 + (Z_2 - Z_1) t$$

ENTONCES PARA ENCONTRAR LA INTERSECCION CON EL

PLANO $Z=Z_0$, HACEMOS:

$$Z_0 = Z_1 + (Z_2 - Z_1) t$$

DE DONDE:

$$t = (Z_0 - Z_1) / (Z_2 - Z_1)$$

Y ASI:

$$X_0 = X_1 + (X_2 - X_1) (Z_0 - Z_1) / (Z_2 - Z_1)$$

$$Y_0 = Y_1 + (Y_2 - Y_1) (Z_0 - Z_1) / (Z_2 - Z_1)$$

ESTE PAR DE ECUACIONES SON LA BASE PARA DEFINIR EL PUNTO DE INTERSECCION BUSCADO.

REFERENCIAS BIBLIOGRAFICAS

- 1.- HARRINGTON S.,COMPUTER GRAPHICS A PROGRAMMING APPROACH,
MC GRAW-HILL,
- 2.- JAMES E. GEORGE , "THE DISPLAY OF ENGINEERING AND SCIENTIFI
DATA",COMPUTER GRAPHICS, 1(3), (1981)49
- 3.- JEAN-LAURENT MALLET, "PRESENTATION D'UN ENSEMBLE DE METHODE
TECHNIQUES DE LA CARTOGRAPHIE AUTOMATIQUE NUMERIQUE" ,
SCIENCES DE LA TERRE,SERIE (INFORMATIQUE GEOLOGIQUE),
NANCY UNIVERSITE, (4), (1974)
- 4.- L.J. SEGERLIND,APPLIED FINITE ELEMENT ANALYSIS,WILEY & SON
(1976) CAP. 12 Y 18.

ENCUADERNACIONES

"GAMA"

TAPIA No. 861 - 75 (L.SQ. CON ARISTA)

T 72-58-82

