**METODO SIMPLEX PARA SOLUCION DE PROBLEMAS**

**GONZÁLEZ MORALES**

**FIC/UNACH**

**PROGRAMACIÓN**

**PROYECTO INDIVIDUAL**

**DE PROGRAMACION LINEAL.**

El Método Simplex publicado por George Dantzig en 1947 consiste en un algoritmo iterativo que secuencialmente a través de iteraciones se va aproximando al óptimo del problema de Programación Lineal en caso de existir esta última.

La primera implementación computacional del Método Simplex es el año 1952 para un problema de 71 variables y 48 ecuaciones. Su resolución tarda 18 horas. Luego, en 1956, un código llamado RSLP1, implementado en un IBM con 4Kb en RAM, admite la resolución de modelos con 255 restricciones.

El Método Simplex hace uso de la propiedad de que la solución óptima de un problema de Programación Lineal se encuentra en un vértice o frontera del dominio de puntos factibles (esto último en casos muy especiales), por lo cual, la búsqueda secuencial del algoritmo se basa en la evaluación progresiva de estos vértices hasta encontrar el óptimo. Cabe destacar que para aplicar el Método Simplex a un modelo lineal, este debe estar en un formato especial conocido como**formato estándar** el cual definiremos a continuación.

**FORMA ESTÁNDAR DE UN MODELO DE PROGRAMACIÓN LINEAL**

Consideremos un modelo de Programación Lineal en su forma estándar, que denotaremos en lo que sigue por:

* **Min          c1x1  + c2x2  + ... + cnxn**
* **sa            a11x1 + a12x2 + ... + a1nxn = b1**
* **a21x1 + a22x2 + ... + a2nxn = b2**
* **...          ...                  ...**
* **am1x1 + am2x2 + ... + amnxn = bm**
* **xi >=  0,   i = 1, 2, ..., n    y    m <= n**

Matricialmente escrito como:

**Min**    **cTx**  
**s.a Ax=b**  
**x >=  0**

No existe pérdida de generalidad en asumir que un modelo de PL viene dado en su forma estándar:

* **EJEMPLO**
* **P)            Max        9u + 2v + 5z**
* **sa            4u + 3v + 6z <=  50**
* **u + 2v - 3z >=  8**
* **2u - 4v + z = 5**
* **u,v >=  0**
* **z e IR**

1. Siempre es posible llevar un problema de maximización a uno de minimización. Si **f(x)** es la función objetivo a maximizar y**x\***es la solución óptima **f(x\*) >= f(x)**, para todo x factible. **-f(x\*) <= - f(x)**, para todo x factible. En consecuencia: **x\*** es también mínimo de  **-f(x)**
2. Cada restricción del tipo <= puede ser llevada a una ecuación de igualdad usando una (nueva) **variable de** **holgura**no negativa, con coeficiente nulo en la función objetivo.
3. Cada restricción del tipo >= puede ser llevada a una ecuación de igualdad usando una (nueva) **variable de** **exceso** no negativa, con coeficiente nulo en la función objetivo.
4. Siempre es posible escribir una variable libre de signo como la diferencia de dos variables no negativas.

Considerando la siguiente notación: **u = x1, v = x2, z = x3 - x4, s1 = x5 (holgura), s2 = x6 (exceso)**, el problema P) puede ser escrito en forma equivalente como:

* **Min         - 9x1 - 2x2 - 5x3 + 5x4 + 0x5 + 0x6**
* **sa:              4x1 + 3x2 + 6x3 - 6x4 +    x5          = 50**
* **x1 + 2x2 - 3x3 + 3x4             - x6  =  8**
* **2x1 - 4x2 +  x3   -   x4                     =  5**
* **xi >=  0,    i=1,2,3,4,5,6.**

Para conocer la metodología que se aplica en el método, vamos a resolver el siguiente problema:

|  |  |
| --- | --- |
| **Maximizar** | ***Z= f(x,y)= 3x + 2y*** |
| **sujeto a:** | ***2x + y http://www.investigacion-operaciones.com/menor.gif18*** |
|  | ***2x + 3y http://www.investigacion-operaciones.com/menor.gif 42*** |
|  | ***3x + y http://www.investigacion-operaciones.com/menor.gif24*** |
|  | **xhttp://www.investigacion-operaciones.com/mayor.gif0 , y http://www.investigacion-operaciones.com/mayor.gif0** |

Se consideran las siguientes fases:

**1. CONVERTIR LAS DESIGUALDADES EN IGUALDADES.**

Se introduce una variable de holgura por cada una de las restricciones, para convertirlas en igualdades, resultando el sistema de ecuaciones lineales:

|  |
| --- |
| ***2x + y + h = 18*** |
| ***2x + 3y + s = 42*** |
| ***3x +y + d = 24*** |

**2. IGUALAR LA FUNCIÓN OBJETIVO A CERO.**

***Z = 3x + 2y - 3x - 2y + Z = 0***

**3. ESCRIBIR LA TABLA INICIAL SIMPLEX.**

En las columnas aparecerán todas las variables del problema y, en las filas, los coeficientes de las igualdades obtenidas, una fila para cada restricción y la última fila con los coeficientes de la función objetivo:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabla 1. Iteración No.1 | | | | | | |
| Base | Variable de decisión | | Variable de holgura | | | Valores solución |
|  | **X** | **Y** | **H** | **S** | **d** |  |
| h | **2** | **1** | **1** | **0** | **0** | **18** |
| s | **2** | **3** | **0** | **1** | **0** | **42** |
| d | **3** | **1** | **0** | **0** | **1** | **24** |
| Z | **-3** | **-2** | **0** | **0** | **0** | **0** |

**4. ENCONTRAR LA VARIABLE DE DECISIÓN QUE ENTRA EN LA BASE Y LA VARIABLE DE HOLGURA QUE SALE DE LA BASE**

Para escoger la variable de decisión que entra en la base, nos fijamos en la última fila, la de los coeficientes de la función objetivo y escogemos la variable con el coeficiente negativo mayor (en valor absoluto).

En nuestro caso, la variable **x** de coeficiente **- 3**.

Si existiesen dos o más coeficientes iguales que cumplan la condición anterior, entonces se elige uno cualquiera de ellos.

Si en la última fila no existiese ningún coeficiente negativo, significa que se ha alcanzado la solución óptima. Por tanto, lo que va a determinar el final del proceso de aplicación del método del simplex, es que en la última fila no haya elementos negativos.

La columna de la variable que entra en la base se llama columna pivote (**Sombreada**).

Para encontrar la variable de holgura que tiene que salir de la base, se divide cada término de la última columna (valores solución) por el término correspondiente de la columna pivote, siempre que estos últimos sean mayores que cero. En nuestro caso:

Si hubiese algún elemento menor o igual que cero no se hace dicho cociente. En el caso de que todos los elementos fuesen menores o iguales a cero, entonces tendríamos una solución no acotada y no se puede seguir.

El término de la columna pivote que en la división anterior dé lugar al menor cociente positivo, el **3**, ya **8** es el menor, indica la fila de la variable de holgura que sale de la base, d. Esta fila se llama fila pivote (**sombreado**).

Si al calcular los cocientes, dos o más son iguales, indica que cualquiera de las variables correspondientes puede salir de la base.

En la intersección de la fila pivote y columna pivote tenemos el elemento pivote operacional, **3**.

**5. ENCONTRAR LOS COEFICIENTES DE LA NUEVA TABLA.**

Los nuevos coeficientes de x se obtienen dividiendo todos los coeficientes de la fila d por el pivote operacional, **3**, que es el que hay que convertir en **1**.

A continuación mediante la reducción gaussiana hacemos ceros los restantes términos de su columna, con lo que obtenemos los nuevos coeficientes de las otras filas incluyendo los de la función objetivo Z.

También se puede hacer utilizando el siguiente esquema:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Fila del pivote:  Resto de las filas:  Veámoslo con un ejemplo una vez calculada la fila del pivote.   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Vieja fila de s | 2 | 3 | 0 | 1 | 0 | 42 | |  | - | - | - | - | - | - | | Coeficiente | 2 | 2 | 2 | 2 | 2 | 2 | |  | X | x | x | x | x | X | | Nueva fila pivote | 1 | 1/3 | 0 | 0 | 1/3 | 8 | |  | = | = | = | = | = | = | | Nueva fila de “s” | **0** | **7/3** | **0** | **1** | **-2/3** | **26** | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabla 2. Iteración No.2 | | | | | | |
| Base | Variable de decisión | | Variable de holgura | | | Valores solución |
|  | **X** | **Y** | **H** | **S** | **d** |  |
| h | **0** | **1/3** | **1** | **0** | **-2/3** | **2** |
| s | **0** | **7/3** | **0** | **1** | **-2/3** | **20** |
| d | **1** | **1/3** | **0** | **0** | **1/3** | **8** |
| Z | **0** | **-1** | **0** | **0** | **1** | **24** |

Como en los elementos de la última fila hay uno negativo, -1, significa que no hemos llegado todavía a la solución óptima. Hay que repetir el proceso:

La variable que entra en la base es y, por ser la variable que corresponde al coeficiente **-1**.

Para calcular la variable que sale, dividimos los términos de la última columna entre los términos correspondientes de la nueva columna pivote:

Y como el menor cociente positivo es 6, tenemos que la variable de holgura que sale es **h**.

El elemento pivote, que ahora hay que hacer **1**, es **1/3**.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabla 3. Iteración No.3 | | | | | | |
| Base | Variable de decisión | | Variable de holgura | | | Valores solución |
|  | **X** | **Y** | **H** | **S** | **d** |  |
| h | **0** | **1** | **3** | **0** | **-2** | **6** |
| s | **0** | **0** | **-7** | **0** | **4** | **12** |
| d | **1** | **0** | **-1** | **0** | **1** | **6** |
| Z | **0** | **0** | **3** | **0** | **-1** | **30** |

Como en los elementos de la última fila hay uno negativo, **-1**, significa que no hemos llegado todavía a la solución óptima. Hay que repetir el proceso:

La variable que entra en la base es **d**, por ser la variable que corresponde al coeficiente **-1**.

Para calcular la variable que sale, dividimos los términos de la última columna entre los términos correspondientes de la nueva columna pivote:

Y como el menor cociente positivo es **3**, tenemos que la variable de holgura que sale es **s**.

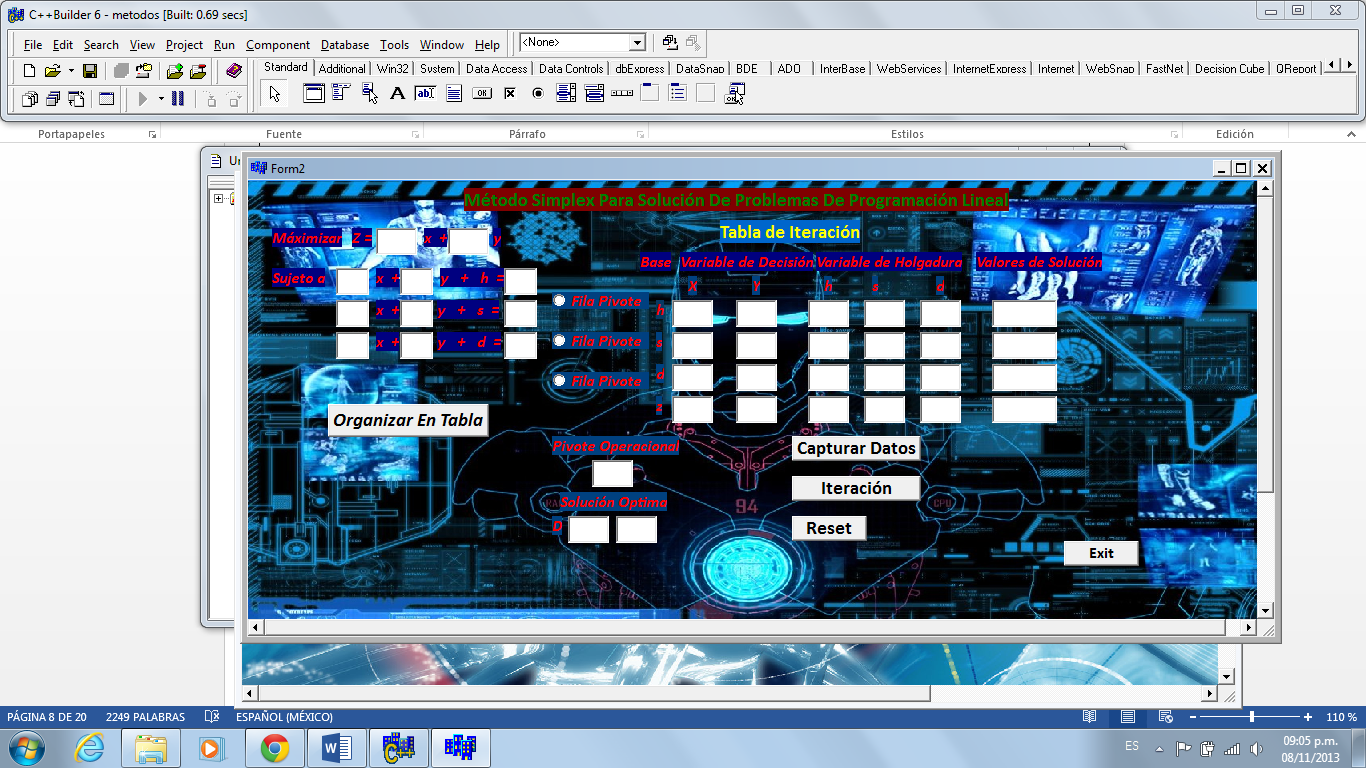
El elemento pivote, que ahora hay que hacer **1**, es **4**.

Obtenemos la tabla:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tabla 4. Final del Proceso | | | | | | |
| Base | Variable de decisión | | Variable de holgura | | | Valores solución |
|  | **X** | **Y** | **H** | **S** | **d** |  |
| h | **0** | **1** | **-1/2** | **0** | **0** | **12** |
| s | **0** | **0** | **-7/4** | **0** | **1** | **3** |
| d | **1** | **0** | **-3/4** | **0** | **0** | **3** |
| Z | **0** | **0** | **5/4** | **0** | **0** | **33** |

Como todos los coeficientes de la fila de la función objetivo son positivos, hemos llegado a la solución óptima.

La solución óptima viene dada por el valor de **Z** en la columna de los valores solución, en nuestro caso: **33**. En la misma columna se puede observar el vértice donde se alcanza, observando las filas correspondientes a las variables de decisión que han entrado en la base: **D (3,12)**

**INTERFAZ GRÁFICA:**

**PROGRAMACIÓN:**

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

#include <vcl.h>

#pragma hdrstop

#include "Unit1.h"

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

#pragma package(smart\_init)

#pragma resource "\*.dfm"

double a,b,c,d,e,f,g,h,i,j,k,l,m,n,r,s,t,u,v,w,xx,yy,zz,q;

double hx,hy,hh,hs,hd,hv,sx,sy,sh,ss,sd,sv,dx,dy,dh,ds,dd,dv,zx,zy,zh,zs,zd,zv;

double cph,cps,cpd,cep,clp,chp,cep1,chp1,clp1,vlg1,vlg2,vlg3;

double pivote,hx1,hy1,hh1,hs1,hd1,hv1,sx1,sy1,sh1,ss1,sd1,sv1;

double dx1,dy1,dh1,ds1,dd1,dv1,zx1,zy1,zh1,zs1,zd1,zv1;

TForm1 \*Form1;

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

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

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

**BUTTON 1 “EXIT”**

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

Close();

}

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

**BUTTON 2 “ORGANIZAR EN TABLA”**

void \_\_fastcall TForm1::Button2Click(TObject \*Sender)

{

a= Edit1->Text.ToDouble();

b= Edit2->Text.ToDouble();

c= Edit3->Text.ToDouble();

d= Edit4->Text.ToDouble();

e= Edit5->Text.ToDouble();

f= Edit6->Text.ToDouble();

g= Edit7->Text.ToDouble();

h= Edit8->Text.ToDouble();

i= Edit9->Text.ToDouble();

j= Edit10->Text.ToDouble();

k= Edit11->Text.ToDouble();

Edit12->Text=AnsiString(c);

Edit13->Text=AnsiString(d);

q= 1;

Edit14->Text=AnsiString(q);

r= 0;

Edit15->Text=AnsiString(r);

s= 0;

Edit16->Text=AnsiString(s);

Edit17->Text=AnsiString(e);

Edit18->Text=AnsiString(f);

Edit19->Text=AnsiString(g);

t=0;

Edit20->Text=AnsiString(t);

u=1;

Edit21->Text=AnsiString(u);

v=0;

Edit22->Text=AnsiString(v);

Edit23->Text=AnsiString(h);

Edit24->Text=AnsiString(i);

Edit25->Text=AnsiString(j);

w=0;

Edit26->Text=AnsiString(w);

xx=0;

Edit27->Text=AnsiString(xx);

yy=1;

Edit28->Text=AnsiString(yy);

Edit29->Text=AnsiString(k);

a= -1\*a;

Edit30->Text=AnsiString(a);

b= -1\*b;

Edit31->Text=AnsiString(b);

zz=0;

Edit32->Text=AnsiString(zz);

Edit33->Text=AnsiString(zz);

Edit34->Text=AnsiString(zz);

Edit35->Text=AnsiString(zz);

}

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

**BUTTON 3 “CAPTURAR DATOS”**

void \_\_fastcall TForm1::Button3Click(TObject \*Sender)

{

hx=Edit12->Text.ToDouble();

hy=Edit13->Text.ToDouble();

hh=Edit14->Text.ToDouble();

hs=Edit15->Text.ToDouble();

hd=Edit16->Text.ToDouble();

hv=Edit17->Text.ToDouble();

sx=Edit18->Text.ToDouble();

sy=Edit19->Text.ToDouble();

sh=Edit20->Text.ToDouble();

ss=Edit21->Text.ToDouble();

sd=Edit22->Text.ToDouble();

sv=Edit23->Text.ToDouble();

dx=Edit24->Text.ToDouble();

dy=Edit25->Text.ToDouble();

dh=Edit26->Text.ToDouble();

ds=Edit27->Text.ToDouble();

dd=Edit28->Text.ToDouble();

dv=Edit29->Text.ToDouble();

zx=Edit30->Text.ToDouble();

zy=Edit31->Text.ToDouble();

zh=Edit32->Text.ToDouble();

zs=Edit33->Text.ToDouble();

zd=Edit34->Text.ToDouble();

zv=Edit35->Text.ToDouble();

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

**BUTTON 5 “ITERACIÓN”**

void \_\_fastcall TForm1::Button5Click(TObject \*Sender)

{

pivote=Edit36->Text.ToDouble();

if(RadioButton1->Checked==true)

{

hx1 = hx/pivote;

hy1 = hy/pivote;

hh1 = hh/pivote;

hs1 = hs/pivote;

hd1 = hd/pivote;

hv1 = hv/pivote;

Edit12->Text=AnsiString(hx1);

Edit13->Text=AnsiString(hy1);

Edit14->Text=AnsiString(hh1);

Edit15->Text=AnsiString(hs1);

Edit16->Text=AnsiString(hd1);

Edit17->Text=AnsiString(hv1);

sx1 = sx - (sx\*hx1);

sy1 = sy - (sx\*hy1);

sh1 = sh - (sx\*hh1);

ss1 = ss - (sx\*hs1);

sd1 = sd - (sx\*hd1);

sv1 = sv - (sx\*hv1);

Edit18->Text=AnsiString(sx1);

Edit19->Text=AnsiString(sy1);

Edit20->Text=AnsiString(sh1);

Edit21->Text=AnsiString(ss1);

Edit22->Text=AnsiString(sd1);

Edit23->Text=AnsiString(sv1);

dx1 = dx - (dx\*hx1);

dy1 = dy - (dx\*hy1);

dh1 = dh - (dx\*hh1);

ds1 = ds - (dx\*hs1);

dd1 = dd - (dx\*hd1);

dv1 = dv - (dx\*hv1);

Edit24->Text=AnsiString(dx1);

Edit25->Text=AnsiString(dy1);

Edit26->Text=AnsiString(dh1);

Edit27->Text=AnsiString(ds1);

Edit28->Text=AnsiString(dd1);

Edit29->Text=AnsiString(dv1);

zx1 = zx - (zx\*hx1);

zy1 = zy - (zx\*hy1);

zh1 = zh - (zx\*hh1);

zs1 = zs - (zx\*hs1);

zd1 = zd - (zx\*hd1);

zv1 = zv - (zx\*hv1);

Edit30->Text=AnsiString(zx1);

Edit31->Text=AnsiString(zy1);

Edit32->Text=AnsiString(zh1);

Edit33->Text=AnsiString(zs1);

Edit34->Text=AnsiString(zd1);

Edit35->Text=AnsiString(zv1);

}

if(RadioButton2->Checked==true)

{

sx1 = sx/pivote;

sy1 = sy/pivote;

sh1 = sh/pivote;

ss1 = ss/pivote;

sd1 = sd/pivote;

sv1 = sv/pivote;

Edit18->Text=AnsiString(sx1);

Edit19->Text=AnsiString(sy1);

Edit20->Text=AnsiString(sh1);

Edit21->Text=AnsiString(ss1);

Edit22->Text=AnsiString(sd1);

Edit23->Text=AnsiString(sv1);

hx1 = hx - (hx\*sx1);

hy1 = hy - (hx\*sy1);

hh1 = hh - (hx\*sh1);

hs1 = hs - (hx\*ss1);

hd1 = hd - (hx\*sd1);

hv1 = hv - (hx\*sv1);

Edit18->Text=AnsiString(hx1);

Edit19->Text=AnsiString(hy1);

Edit20->Text=AnsiString(hh1);

Edit21->Text=AnsiString(hs1);

Edit22->Text=AnsiString(hd1);

Edit23->Text=AnsiString(hv1);

dx1 = dx - (dx\*sx1);

dy1 = dy - (dx\*sy1);

dh1 = dh - (dx\*sh1);

ds1 = ds - (dx\*ss1);

dd1 = dd - (dx\*sd1);

dv1 = dv - (dx\*sv1);

Edit24->Text=AnsiString(dx1);

Edit25->Text=AnsiString(dy1);

Edit26->Text=AnsiString(dh1);

Edit27->Text=AnsiString(ds1);

Edit28->Text=AnsiString(dd1);

Edit29->Text=AnsiString(dv1);

zx1 = zx - (zx\*sx1);

zy1 = zy - (zx\*sy1);

zh1 = zh - (zx\*sh1);

zs1 = zs - (zx\*ss1);

zd1 = zd - (zx\*sd1);

zv1 = zv - (zx\*sv1);

Edit30->Text=AnsiString(zx1);

Edit31->Text=AnsiString(zy1);

Edit32->Text=AnsiString(zh1);

Edit33->Text=AnsiString(zs1);

Edit34->Text=AnsiString(zd1);

Edit35->Text=AnsiString(zv1);

Edit46->Text=AnsiString(dv1);

Edit47->Text=AnsiString(hv1);

}

if(RadioButton3->Checked==true)

{

dx1 = dx/pivote;

dy1 = dy/pivote;

dh1 = dh/pivote;

ds1 = ds/pivote;

dd1 = dd/pivote;

dv1 = dv/pivote;

Edit24->Text=AnsiString(dx1);

Edit25->Text=AnsiString(dy1);

Edit26->Text=AnsiString(dh1);

Edit27->Text=AnsiString(ds1);

Edit28->Text=AnsiString(dd1);

Edit29->Text=AnsiString(dv1);

hx1 = hx - (hx\*dx1);

hy1 = hy - (hx\*dy1);

hh1 = hh - (hx\*dh1);

hs1 = hs - (hx\*ds1);

hd1 = hd - (hx\*dd1);

hv1 = hv - (hx\*dv1);

Edit12->Text=AnsiString(hx1);

Edit13->Text=AnsiString(hy1);

Edit14->Text=AnsiString(hh1);

Edit15->Text=AnsiString(hs1);

Edit16->Text=AnsiString(hd1);

Edit17->Text=AnsiString(hv1);

sx1 = sx - (sx\*dx1);

sy1 = sy - (sx\*dy1);

sh1 = sh - (sx\*dh1);

ss1 = ss - (sx\*ds1);

sd1 = sd - (sx\*dd1);

sv1 = sv - (sx\*dv1);

Edit18->Text=AnsiString(sx1);

Edit19->Text=AnsiString(sy1);

Edit20->Text=AnsiString(sh1);

Edit21->Text=AnsiString(ss1);

Edit22->Text=AnsiString(sd1);

Edit23->Text=AnsiString(sv1);

zx1 = zx - (zx\*dx1);

zy1 = zy - (zx\*dy1);

zh1 = zh - (zx\*dh1);

zs1 = zs - (zx\*ds1);

zd1 = zd - (zx\*dd1);

zv1 = zv - (zx\*dv1);

Edit30->Text=AnsiString(zx1);

Edit31->Text=AnsiString(zy1);

Edit32->Text=AnsiString(zh1);

Edit33->Text=AnsiString(zs1);

Edit34->Text=AnsiString(zd1);

Edit35->Text=AnsiString(zv1);

}

}

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

**BUTTON 6 “RESET”**

void \_\_fastcall TForm1::Button6Click(TObject \*Sender)

{

Edit1->Text="";

Edit2->Text="";

Edit3->Text="";

Edit4->Text="";

Edit5->Text="";

Edit6->Text="";

Edit7->Text="";

Edit8->Text="";

Edit9->Text="";

Edit10->Text="";

Edit11->Text="";

Edit12->Text="";

Edit13->Text="";

Edit14->Text="";

Edit15->Text="";

Edit16->Text="";

Edit17->Text="";

Edit18->Text="";

Edit19->Text="";

Edit20->Text="";

Edit21->Text="";

Edit22->Text="";

Edit23->Text="";

Edit24->Text="";

Edit25->Text="";

Edit26->Text="";

Edit27->Text="";

Edit28->Text="";

Edit29->Text="";

Edit30->Text="";

Edit31->Text="";

Edit32->Text="";

Edit33->Text="";

Edit34->Text="";

Edit35->Text="";

Edit36->Text="";

Edit46->Text="";

Edit47->Text="";

}

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

**CORRER EL PROGRAMA.**