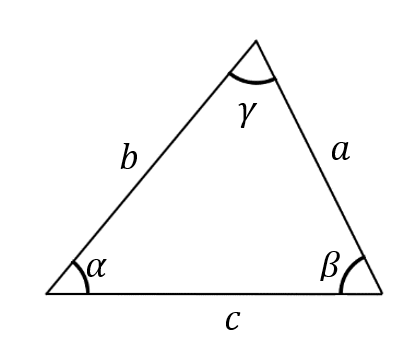
Practica #2

Nombre: Univ. Calcina Choque Juan

Materia: Lab. Programación

1. Resolución de triángulos

PLANTEO:



Caso 1. Dato: los tres lados:

Te pide hallar los tres ángulos se procede de la siguiente manera:

Ley de cosenos:

Hallamos

Caso 2. Dato: 2 lados 1 ángulo

1. Si te dan como datos los dos lados y ángulo opuesto a ellos

Se usa la ley de senos:

1. Si te dan como datos los dos lados y el ángulo entre ellos

Usamos la ley de tangentes y la suma de ángulos

Ejemplo:

Si los datos son

(2) en (1)

(2)+(3)

Caso 3. Dato: 2 ángulos 1 lado

Usamos la suma de ángulos y la ley de senos

Caso 4. Dato: 3 ángulos

Resp. Se necesita 1 lado por lo menos

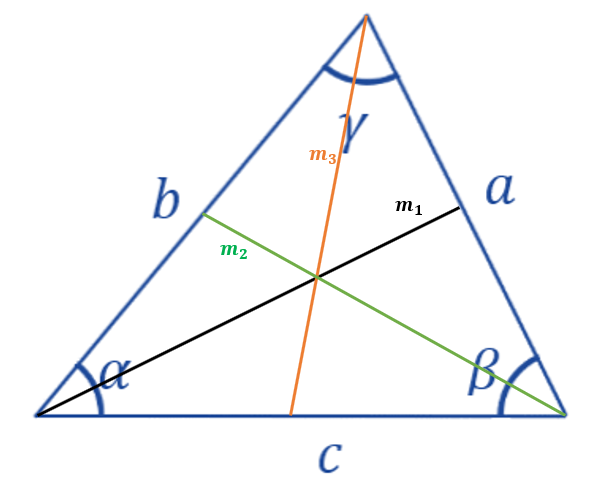
Área

Por el teorema de Heron:

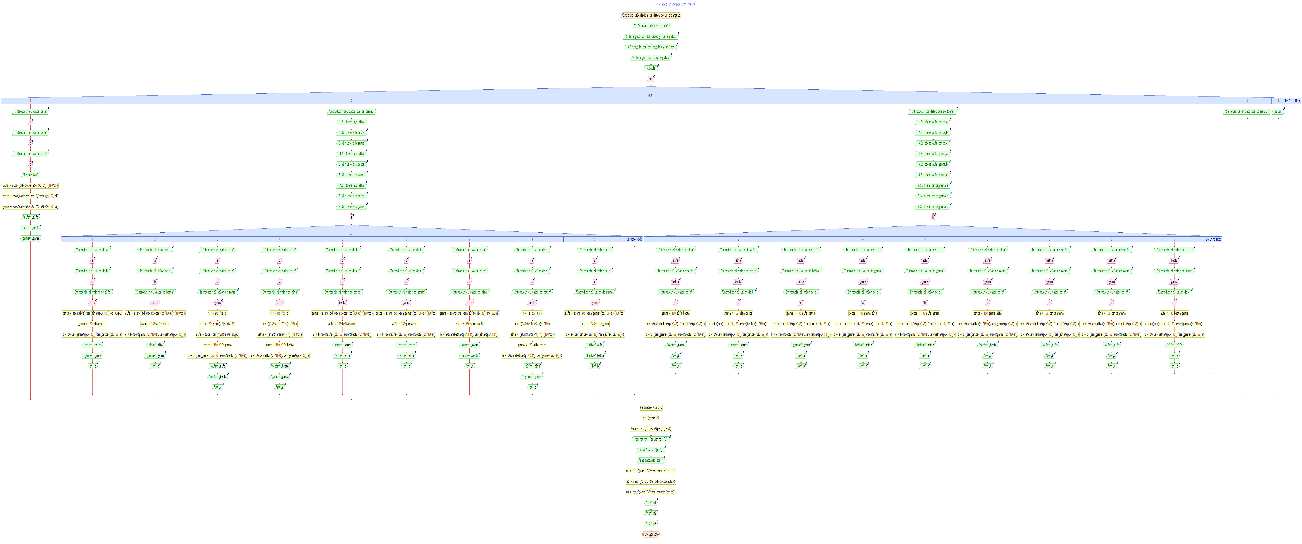
Donde

Perímetro

Medianas



ALGORITMO:



PROGRAMA;

// Univ Calcina Choque Juan Carlos

#include<iostream>

#include<cmath>

using namespace std;

int main()

{

float a,alfa,area,b,betta,c,gamma,m1,m2,m3,n,m,p,perimetro,pi;

int z,h,num;

cout << "\t\tUNIV CALCINA CHOQUE JUAN CARLOS" << endl;

cout << "\tCALCULADORA DE DATOS DE UN TRIANGULO" << endl;

cout << "1.Triangulo de tres lados" << endl;

cout << "2.Triangulo con dos lados y un angulos" << endl;

cout << "3.Triangulo con dos angulos y un lado" << endl;

cout << "4.Triangulo con tres angulos" << endl;

cout << "6.Salir" << endl;

cin >> num;

switch (num)

{

case 1:

cout << "Introducir el valor de a" << endl;

cin >> a;

cout << "Introducir el valor de b" << endl;

cin >> b;

cout << "Introducir el valor de c" << endl;

cin >> c;

cout << "Las Respuestas son" << endl;

alfa = acos((b\*b+c\*c-a\*a)/(2\*b\*c))\*(180/M\_PI);

betta = acos((a\*a+c\*c-b\*b)/(2\*a\*c))\*(180/M\_PI);

gamma = acos((a\*a+b\*b-c\*c)/(2\*a\*b))\*(180/M\_PI);

cout << "\nalfa=" << alfa << endl;

cout << "betta=" << betta << endl;

cout << "gamma=" << gamma << endl;

break;

case 2:

cout << "Introducir los datos que se tiene:" << endl;

cout << "1.Si tiene a,b,alfa" << endl;

cout << "2.Si tiene a,b,betta" << endl;

cout << "3.Si tiene a,b,gamma" << endl;

cout << "4.Si tiene b,c,alfa" << endl;

cout << "5.Si tiene b,c,betta" << endl;

cout << "6.Si tiene b,c,gamma" << endl;

cout << "7.Si tiene a,c,alfa" << endl;

cout << "8.Si tiene a,c,betta" << endl;

cout << "9.Si tiene a,c,gamma" << endl;

cin >> z;

switch (z) {

case 1:

cout << "Introducir el valor de a" << endl;

cin >> a;

cout << "Introducir el valor de b" << endl;

cin >> b;

cout << "Introducir el valord de alfa" << endl;

cin >> alfa;

betta = asin((b/a)\*sin(alfa\*(M\_PI/180)))\*(180/M\_PI);

gamma = 180-alfa-betta;

c = b\*(sin(gamma\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\nbetta=" << betta << endl;

cout << "gamma=" << gamma << endl;

cout << "c=" << c << endl;

break;

case 2:

cout << "Introducir el valor de a" << endl;

cin >> a;

cout << "Introducir el valor de b" << endl;

cin >> b;

cout << "Introducir el valor de betta" << endl;

cin >> betta;

alfa = asin((a/b)\*sin(betta\*(M\_PI/180)))\*(180/M\_PI);

gamma = 180-alfa-betta;

c = b\*(sin(gamma\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\nalfa=" << alfa << endl;

cout << "gamma=" << gamma << endl;

cout << "c=" << c << endl;

break;

case 3:

cout << "Introducir el valor de a" << endl;

cin >> a;

cout << "Introducir el valor de b" << endl;

cin >> b;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

m = (a+b)/(a-b);

n = ((180-gamma)/2)\*(M\_PI/180);

alfa = (atan(tan(n)/m)+n)\*(180/M\_PI);

betta = 180-alfa-gamma;

c = b\*(sin(gamma\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\nalfa=" << alfa << endl;

cout << "betta=" << betta << endl;

cout << "c=" << c << endl;

break;

case 4:

cout << "Introducir el valor de b" << endl;

cin >> b;

cout << "Introducir el valor de c" << endl;

cin >> c;

cout << "Introducir el valor de alfa" << endl;

cin >> alfa;

m = (b+c)/(b-c);

n = ((180-alfa)/2)\*(M\_PI/180);

betta = (atan(tan(n)/m)+n)\*(180/M\_PI);

gamma = 180-alfa-betta;

a = c\*(sin(alfa\*(M\_PI/180))/sin(gamma\*(M\_PI/180)));

cout << "\nbetta=" << betta << endl;

cout << "gamma=" << gamma << endl;

cout << "a=" << a << endl;

break;

case 5:

cout << "Introducir el valor de b" << endl;

cin >> b;

cout << "Introducir el valor de c" << endl;

cin >> c;

cout << "Introducir el valor de betta" << endl;

cin >> betta;

gamma = asin((c/b)\*sin(betta\*(M\_PI/180)))\*(180/M\_PI);

alfa = 180-betta-gamma;

a = b\*(sin(alfa\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\ngamma=" << gamma << endl;

cout << "alfa=" << alfa << endl;

cout << "a=" << a << endl;

break;

case 6:

cout << "Introducir el valor de b" << endl;

cin >> b;

cout << "Introducir el valor de c" << endl;

cin >> c;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

betta = asin((b/c)\*sin(gamma\*(M\_PI/180)))\*(180/M\_PI);

alfa = 180-gamma-betta;

a = b\*(sin(alfa\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\nbetta=" << betta << endl;

cout << "alfa=" << alfa << endl;

cout << "a=" << a << endl;

break;

case 7:

cout << "Introducir el valor de a" << endl;

cin >> a;

cout << "Introducir el valor de c" << endl;

cin >> c;

cout << "Introducir el valor de alfa" << endl;

cin >> alfa;

gamma = asin((c/a)\*sin(alfa\*(M\_PI/180)))\*(180/M\_PI);

betta = 180-gamma-alfa;

b = a\*(sin(betta\*(M\_PI/180))/sin(alfa\*(M\_PI/180)));

cout << "\ngamma=" << gamma << endl;

cout << "betta=" << betta << endl;

cout << "b=" << b << endl;

break;

case 8:

cout << "Introducir el valor de a" << endl;

cin >> a;

cout << "Introducir el valor de c" << endl;

cin >> c;

cout << "Introducir el valor de betta" << endl;

cin >> betta;

m = (a+c)/(a-c);

n = ((180-betta)/2)\*(M\_PI/180);

alfa = (atan(tan(n)/m)+n)\*(180/M\_PI);

gamma = 180-alfa-betta;

b = c\*(sin(betta\*(M\_PI/180))/sin(gamma\*(M\_PI/180)));

cout << "\nalfa=" << alfa << endl;

cout << "gamma=" << gamma << endl;

cout << "b=" << b << endl;

break;

case 9:

cout << "Introducir el valor de a" << endl;

cin >> a;

cout << "Introducir el valor de c" << endl;

cin >> c;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

alfa = asin((a/c)\*sin(gamma\*(M\_PI/180)))\*(180/M\_PI);

betta = 180-alfa-gamma;

b = a\*(sin(betta\*(M\_PI/180))/sin(alfa\*(M\_PI/180)));

cout << "\nalfa=" << alfa << endl;

cout << "betta=" << betta << endl;

cout << "b=" << b << endl;

break;

}

break;

case 3:

cout << "Introducir los datos que se tiene:" << endl;

cout << "1.Si tiene alfa,betta,a" << endl;

cout << "2.Si tiene alfa,betta,b" << endl;

cout << "3.Si tiene alfa,betta,c" << endl;

cout << "4.Si tiene alfa,gamma,a" << endl;

cout << "5.Si tiene alfa,gamma,b" << endl;

cout << "6.Si tiene alfa,gamma,c" << endl;

cout << "7.Si tiene betta,gamma,a" << endl;

cout << "8.Si tiene betta,gamma,b" << endl;

cout << "9.Si tiene betta,gamma,c" << endl;

cin >> h;

switch (h) {

case 1:

cout << "Introducir el valor de alfa" << endl;

cin >> alfa;

cout << "Introducir el valor de betta" << endl;

cin >> betta;

cout << "Introducir el valor de a" << endl;

cin >> a;

gamma = 180-alfa-betta;

b = a\*(sin(betta\*(M\_PI/180))/sin(alfa\*(M\_PI/180)));

c = b\*(sin(gamma\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\ngamma=" << gamma << endl;

cout << "b=" << b << endl;

cout << "c=" << c << endl;

break;

case 2:

cout << "Introducir el valor de alfa" << endl;

cin >> alfa;

cout << "Introducir el valor de betta" << endl;

cin >> betta;

cout << "Introducir el valor de b" << endl;

cin >> b;

gamma = 180-alfa-betta;

a = b\*(sin(alfa\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

c = b\*(sin(gamma\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\ngamma=" << gamma << endl;

cout << "a=" << a << endl;

cout << "c=" << c << endl;

break;

case 3:

cout << "Introducir el valor de alfa" << endl;

cin >> alfa;

cout << "Introducir el valor de betta" << endl;

cin >> betta;

cout << "Introducir el valor de c" << endl;

cin >> c;

gamma = 180-alfa-betta;

a = c\*(sin(alfa\*(M\_PI/180))/sin(gamma\*(M\_PI/180)));

b = c\*(sin(betta\*(M\_PI/180))/sin(gamma\*(M\_PI/180)));

cout << "\ngamma=" << gamma << endl;

cout << "a=" << a << endl;

cout << "b=" << b << endl;

break;

case 4:

cout << "Introducir el valor de alfa" << endl;

cin >> alfa;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

cout << "Introducir el valor de a" << endl;

cin >> a;

betta = 180-gamma-alfa;

b = a\*(sin(betta\*(M\_PI/180))/sin(alfa\*(M\_PI/180)));

c = a\*(sin(gamma\*(M\_PI/180))/sin(alfa\*(M\_PI/180)));

cout << "\nbetta=" << betta << endl;

cout << "b=" << b << endl;

cout << "c=" << c << endl;

break;

case 5:

cout << "Introducir el valor de alfa" << endl;

cin >> alfa;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

cout << "Introducir el valor de b" << endl;

cin >> b;

betta = 180-alfa-gamma;

a = b\*(sin(alfa\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

c = b\*(sin(gamma\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\nbetta=" << betta << endl;

cout << "a=" << a << endl;

cout << "c=" << c << endl;

break;

case 6:

cout << "Introducir el valor de alfa" << endl;

cin >> alfa;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

cout << "Introducir el valor de c" << endl;

cin >> c;

betta = 180-gamma-alfa;

a = c\*(sin(alfa\*(M\_PI/180))/sin(gamma\*(M\_PI/180)));

b = c\*(sin(betta\*(M\_PI/180))/sin(gamma\*(M\_PI/180)));

cout << "\nbetta=" << betta << endl;

cout << "a=" << a << endl;

cout << "b=" << b << endl;

break;

case 7:

cout << "Introducir el valor de betta" << endl;

cin >> betta;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

cout << "Introducir el valor de a" << endl;

cin >> a;

alfa = 180-betta-gamma;

b = a\*(sin(betta\*(M\_PI/180))/sin(alfa\*(M\_PI/180)));

c = a\*(sin(gamma\*(M\_PI/180))/sin(alfa\*(M\_PI/180)));

cout << "\nalfa=" << alfa << endl;

cout << "b=" << b << endl;

cout << "c=" << c << endl;

break;

case 8:

cout << "Introducir el valor de betta" << endl;

cin >> betta;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

cout << "Introducir el valor de b" << endl;

cin >> b;

alfa = 180-betta-gamma;

a = b\*(sin(alfa\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

c = b\*(sin(gamma\*(M\_PI/180))/sin(betta\*(M\_PI/180)));

cout << "\nalfa=" << alfa << endl;

cout << "a=" << a << endl;

cout << "c=" << c << endl;

break;

case 9:

cout << "Introducir el valor de betta" << endl;

cin >> betta;

cout << "Introducir el valor de gamma" << endl;

cin >> gamma;

cout << "Introducir el valor de c" << endl;

cin >> c;

alfa = 180-betta-gamma;

a = c\*(sin(alfa\*(M\_PI/180))/sin(gamma\*(M\_PI/180)));

b = c\*(sin(betta\*(M\_PI/180))/sin(gamma\*(M\_PI/180)));

cout << "\nalfa=" << alfa << endl;

cout << "a=" << a << endl;

cout << "b=" << b << endl;

break;

}

break;

case 4:

cout << "\nSe necesita un lado por lo menos " << endl;

break;

case 5:

cout << "salir" << endl;

break;

}

perimetro = a+b+c;

p = (a+b+c)/2;

area = sqrtf(p\*(p-a)\*(p-b)\*(p-c));

cout << "\nPerimetro=" << perimetro << "[u]" << endl;

cout << "\nArea=" << area << "[u^2]" << endl;

cout << "\nLas medianas son:" << endl;

m1 = sqrtf(((a\*a)/4)+c\*c-a\*c\*cos(betta));

m2 = sqrtf(((b\*b)/4)+c\*c-b\*c\*cos(alfa));

m3 = sqrtf(((c\*c)/4)+a\*a-a\*c\*cos(betta));

cout << "m1=" << m1 << endl;

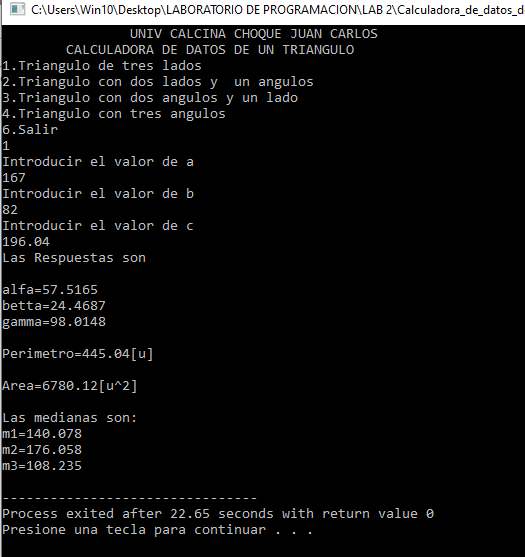
cout << "m2=" << m2 << endl;

cout << "m3=" << m3 << endl;

return 0;

}

**EJECUCION DEL PROGRAMA**



1. Generar Matriz de Vandermonde y valor de su determinante

La matriz de Vandermonde:

Hallamos el determinante de la matriz de Vandermonde:

Aplicando propiedades para reducir la matriz

ALGORITMO:

INICIO

Inicio

Introducir la dimensión de la matriz

n=

i=0; i<=n; i++

i

i=0; i<=n; i++

j=0; j<=n; j++

j==0

j

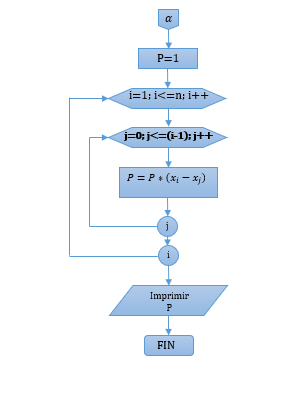
i

i=0; i<=n; i++

j=0; j<=n; j++

j

i



PROGRAMA:

//Generacion de la matriz de Vandermonde y su determinante

//Univ. Calcina Choque Juan Carlos

#include<stdio.h>

#include<conio.h>

#include<math.h>

#define N 30

main()

{

int n,i,j,A[N][N],P;

float X[N];

printf("\t\tUniv. Calcina Choque Juan Carlos\n");

printf("\tGeneracion de la matriz de Vandermonde y su determinante\n");

printf("Leer Dimesion del vector n: ");

scanf("%d",&n);

// Introduccion de Datos

for(i=0;i<=n;i++)

{

printf("\nLeer entradas del vector X[%d]: ",i);

scanf("%f",&X[i]);

}

for(i=0;i<=n;i++)

{

for(j=0;j<=n;j++)

{

if(j==0)

{

A[i][j]=1;

}

else

{

A[i][j]=pow(X[i],j);

}

}

}

printf("\n\nV= ");

for(i=0;i<=n;i++)

{

for(j=0;j<=n;j++)

{

printf("\t%d\t",A[i][j]);

}

printf("\n");

}

P=1;

for(i=1;i<=n;i++)

{

for(j=0;j<=(i-1);j++)

{

P=P\*(X[i]-X[j]);

}

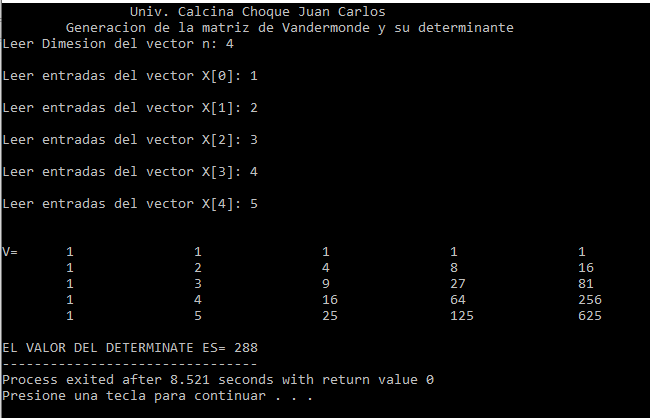
}

printf("\nEL VALOR DEL DETERMINATE ES= %d",P);

return 0;

}

**EJECUCION DEL PROGRAMA**



1. División de polinomios por el método de Horner

Si:

Donde:

Ejemplo: para

1. Resolver un sistema de ecuaciones de 2 incógnitas

Planteo:

El sistema de ecuaciones es:

Utilizamos el método de Cramer:

Las soluciones serán:

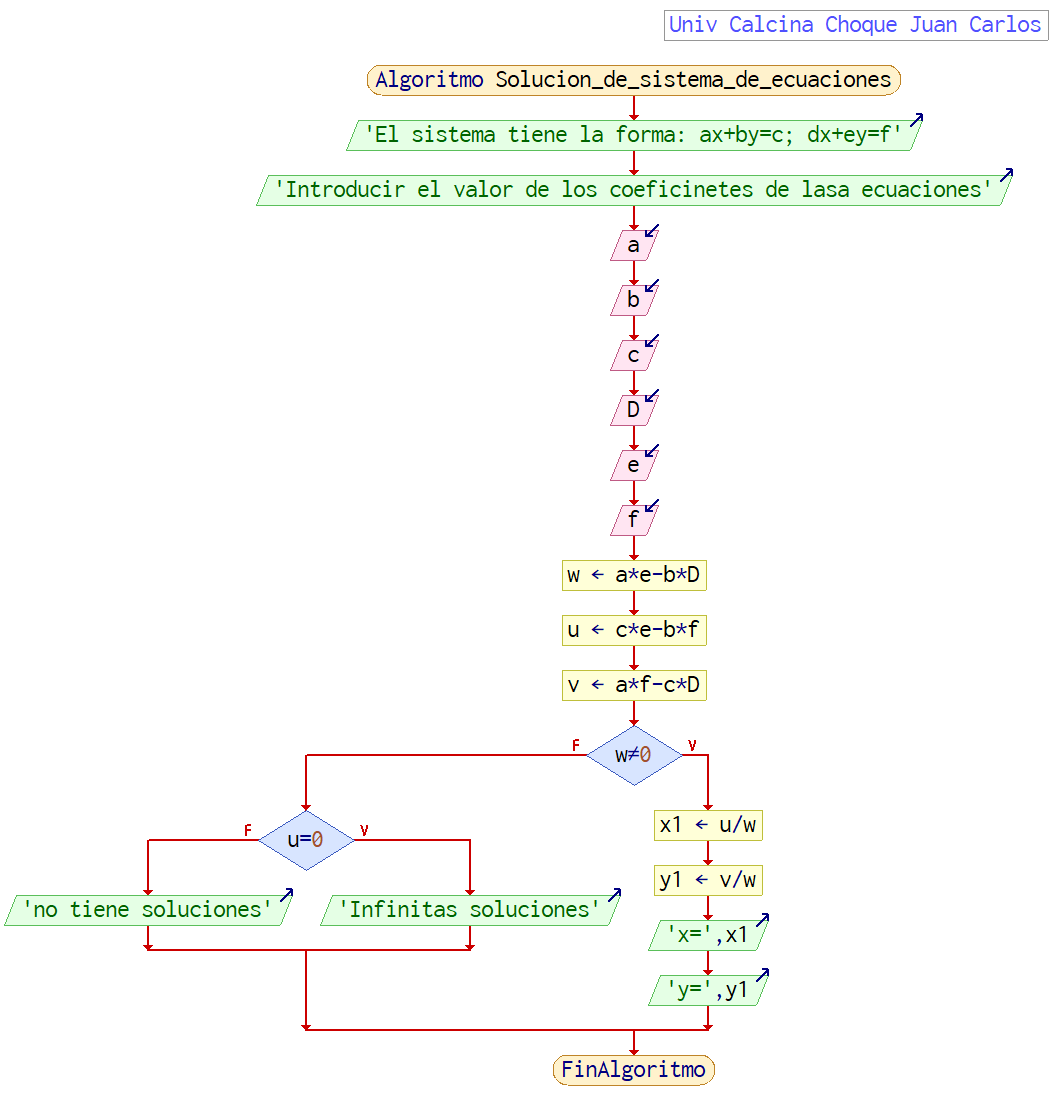
Pero:



La solución será:

1. Si
2. Si

Algoritmo:



PROGRAMA:

// UNIV. CALCINA CHOQUE JUAN CARLOS

// RESOLVER UN SISTEMA DE ECUACIONES

#include<stdio.h>

#include<math.h>

main()

{

float a,b,c,d,e,f;

float u,v,w,x,y;

printf("\t\t\tUNIV. CALCINA CHOQUE JUAN CARLOS\n");

printf("\tCALCULADORA PARA RESOLVER SISTEMA DE ECUACIONES CON DOS INCOGNITAS\n\n");

printf("\t\tEl sistema de ecuaciones sera de la forma:\n\n");

printf("\t\t\t\tax+by=c\n");

printf("\t\t\t\tdx+ey=f\n\n");

printf("Introducir el valor de a= ");

scanf("%f",&a);

printf("Introducir el valor de b= ");

scanf("%f",&b);

printf("Introducir el valor de c= ");

scanf("%f",&c);

printf("Introducir el valor de d= ");

scanf("%f",&d);

printf("Introducir el valor de e= ");

scanf("%f",&e);

printf("Introducir el valor de f= ");

scanf("%f",&f);

w=a\*e-b\*d;

u=c\*e-b\*f;

v=a\*f-c\*d;

printf("LA RESPUESTA ES: \n\n");

if(w!=0)

{

x=u/w;

y=v/w;

printf("\tx= %.3f\n",x);

printf("\ty= %.3f",y);

}

else

{

if(u==0,v==0)

{

printf("\tINFINITAS SOLUCIONES");

}

else printf("\tNO TIENE SOLUCION (SISTEMA INCOMPATIBLE)");

}

return 0;

}

**EJECUCION DEL PROGRAMA**

