

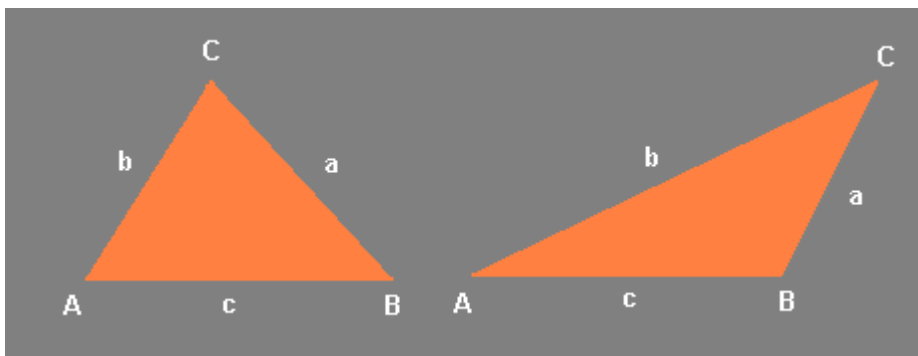
Laboratorio de Programación # 2

EJERCICIO 1:

- RESOLUCIÓN DE TRIÁNGULOS, ÁREAS Y MEDIANAS.

Base teórica

Introducción: Un triángulo que no es rectángulo se le llama oblicuángulo. Los **elementos de un triángulo oblicuángulo** son los **tres ángulos A, B y C** y los **tres lados** respectivos, opuestos a los anteriores, **a, b y c**.



-Un problema de resolución de triángulos oblicuángulos consiste en hallar tres de sus elementos, lados o ángulos, cuando se conocen los otros tres (uno de los cuales tiene que ser un lado) de otro modo será un triángulo indefinido.

-Oblicuángulo se contrapone a rectángulo, en sentido estricto. Pero cuando se habla de triángulos oblicuángulos no se pretende excluir al triángulo rectángulo en el estudio, que queda asumido como caso particular. Es decir, cuando el triángulo es rectángulo, porque se dice expresamente que lo es, el problema se reduce, tiene un tratamiento particular y no necesariamente se aplican las técnicas generales de resolución que vamos a ver seguidamente. *Se utilizan tres propiedades:*

Suma de los ángulos de un triángulo:

$$A + B + C = 180^\circ$$

Teorema del seno:

$$\frac{a}{\text{sen}(A)} = \frac{b}{\text{sen}(B)} = \frac{c}{\text{sen}(C)}$$

Teorema del coseno:

$$a^2 = b^2 + c^2 - 2 * b * c * \cos(A)$$

$$b^2 = a^2 + c^2 - 2 * a * c * \cos(B)$$

$$c^2 = a^2 + b^2 - 2 * a * b * \cos(C)$$

Casos en la resolución de triángulos:

Caso 1

Los tres lados conocidos ...

Ejemplo: datos: a, b, c incógnitas: A, B, C

$$A = \arccos\left(\frac{b^2 + c^2 - a^2}{2 * b * c}\right); B = \arccos\left(\frac{a^2 + c^2 - b^2}{2 * a * c}\right)$$

$$C = 180 - A - B$$

Caso 2

Un lado y los ángulos adyacentes conocidos:

Ejemplo: datos: a, B, C incógnitas: b, c, A

$$A = 180 - B - C; b = \left(\frac{a * \sin(B)}{\sin(A)}\right); c = \sqrt{a^2 + b^2 - 2 * a * b * \cos(C)}$$

Caso 3

Dos lados y el ángulo formado entre ellos:

Ejemplo: datos: a, b, C incógnitas: c, A, B

$$c = \sqrt{a^2 + b^2 - 2 * a * b * \cos(C)}; A = \arccos\left(\frac{b^2 + c^2 - a^2}{2 * b * c}\right)$$

$$B = 180 - A - C$$

Caso 4

Dos lados y el ángulo opuesto a uno de ellos:

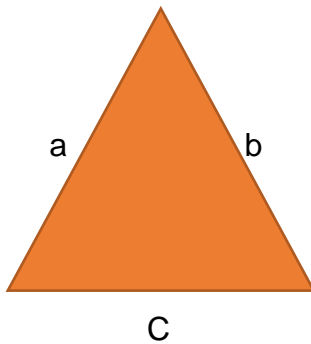
Ejemplo: datos: a, b, A incógnitas: c, B, C

$$c = \sqrt{a^2 + b^2 - 2 * a * b * \cos(C)} ; B = \left(\frac{a^2 + c^2 - b^2}{2 * a * c} \right)$$

$$C = 180 - A - B$$

-Cálculo de áreas de un triángulo (por formula de HERON)

Dado el triángulo:



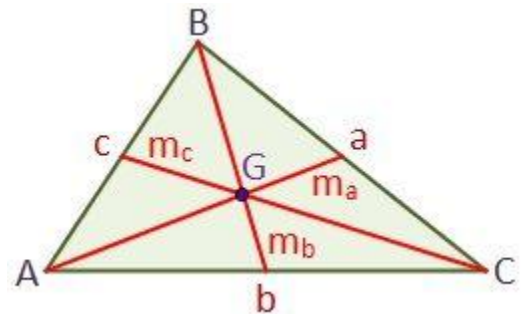
1. Con los 3 lados se halla el perímetro del triángulo. (P)
2. Con el valor del perímetro se halla el semiperímetro. (S)
3. Finalmente, con el semiperímetro se halla el área (A).

$$P = a + b + c ; S = \frac{P}{2} ; A = \sqrt{S * (S - a) * (S - b) * (S - c)}$$

Cálculo de medianas

La **mediana** de un triángulo es el segmento que une uno de sus vértices con el centro del lado opuesto.

Hay tres **medianas** (m_a, m_b, m_c), según de que vértice parta ésta. La longitud de las medianas se calcula a partir del **teorema de la mediana**:

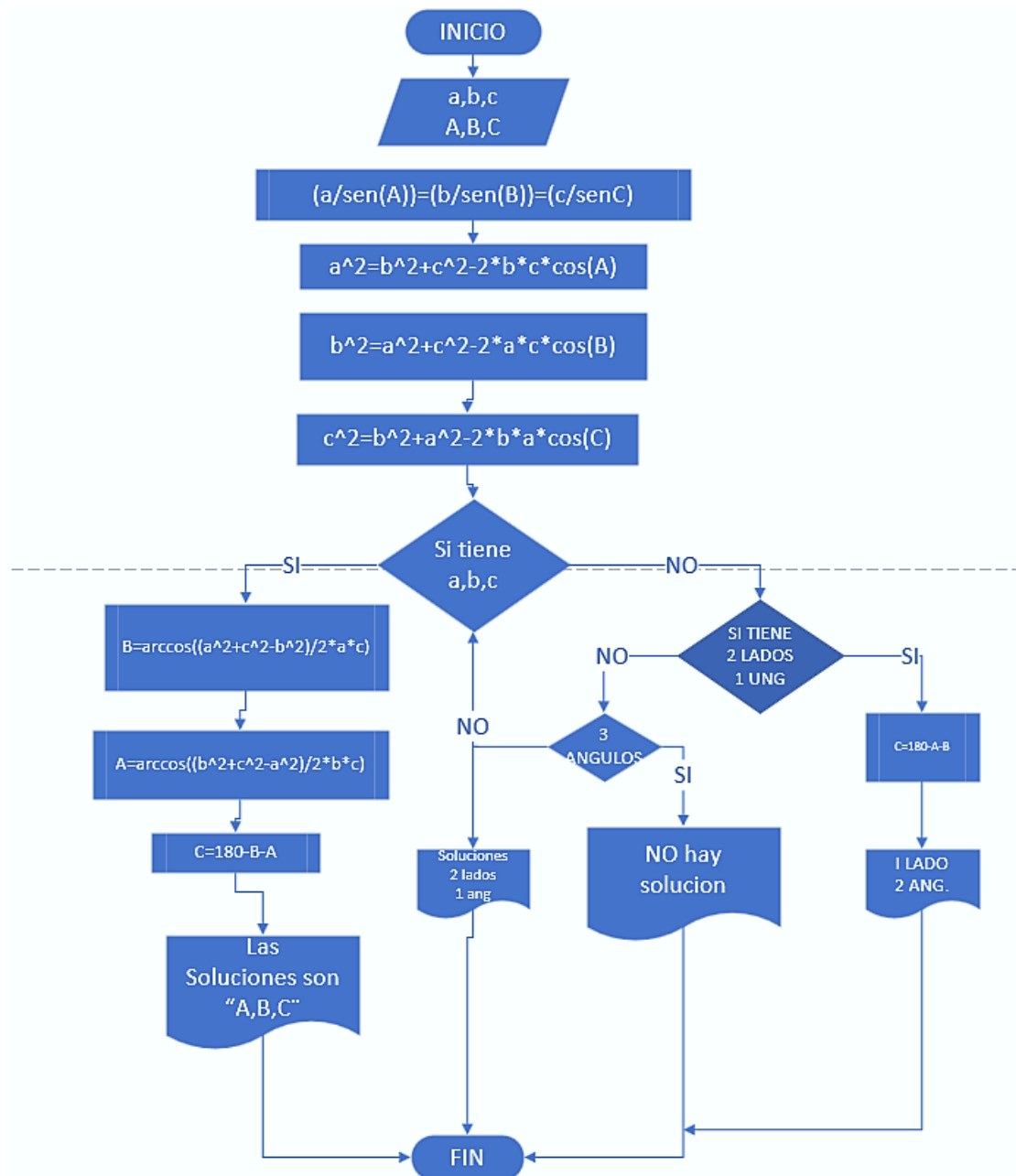


$$m_a = \frac{\sqrt{2(b^2 + c^2) - a^2}}{2}$$

$$m_b = \frac{\sqrt{2(a^2 + c^2) - b^2}}{2}$$

$$m_c = \frac{\sqrt{2(a^2 + b^2) - c^2}}{2}$$

Diagrama de flujo



Listado del código

// 1. Resolución de triángulos dados 3 datos: a) tres lados

// b) tres ángulos

// c) dos lados y un ángulo

// d) dos ángulos y un lado

// Resolver los otros tres valores, calcular el area o superficie del triangulo y medianas

[illegible]

```
cout<<"\t/_ )3* _____2*( _\\ "<<endl<<endl;

cout<<"\t\t LADO 3"<<endl;

cout<<"\t-----"<<endl;

cout<<"\tingrese los datos conforme se denota en el

TRIANGULO MOSTRADO : "<<endl;

cout<<"\t(En caso de no poseer un dato ingresar cero)"<<endl;

cout<<"\t(Ingresar los angulos en grados positivos)"<<endl;

cout<<"\n\t\tLADOS"<<endl;

cout<<"\t-----"<<endl;

for(int i=1;i<=3;i++){

    do{        cout<<" --> Lado < "<<i<<" > = ";

                cin>>lado[i];

            }while(lado[i]<0);

            if(lado[i]!=0){cl=cl+1;

                        gl[cl]=i;

                    }

}cout<<"\n\t\tANGULOS"<<endl;

cout<<"\t-----"<<endl;

for(int i=1;i<=3;i++){

    do{        cout<<" --> Angulo < "<<i<<" > en grados = ";

                cin>>angulo[i];

            }while(angulo[i]<0);

            if(angulo[i]!=0){ca=ca+1;

                        ga[ca]=i;

                    }

}cout<<"\n";cout<<"\n";

cout<<"SE TIENE "<<cl<<" LADOS Y "<<ca<<" ANGULOS COMO

DATOS"<<endl<<endl;

if(cl==3){cout<<"\t\t) Tres lados"<<endl;// LEY DE COSENOS

        angulo[3]= acos( double

(((pow((lado[3]),2)))+(pow((lado[1]),2))-

(pow((lado[2]),2)))/(2*lado[3]*lado[1])));angulo[3]=angulo[3]*180/PI;        //
```

```

                                angulo[1]= acos( double
(((pow((lado[2]),2))+pow((lado[1]),2))-
(pow((lado[3]),2)))/(2*lado[2]*lado[1]));angulo[1]=angulo[1]*180/PI;

                                angulo[2]=180-angulo[3]-angulo[1];

                                }

                                if(ca==3 and cl==0){cout<<"\t\tb) Tres angulos"<<endl;

                                cout<<"\t\tNO ES POSIBLE CALCULAR..."<<endl;

                                cout<<"\tDEBIDO A QUE EXISTEN INFINITOS
TRIANGULOS DE DIVERSOS TAMANOS, QUE PUEDEN POSEER LOS MISMOS ANGULOS"<<endl;

                                }

                                // LEY DE SENOS con un angulo que NO forman estos
                                // LEY DE COSENOS con un angulo que SI forman estos

                                if(cl==2 and ca==1){cout<<"\t\tc) Dos lados y un angulo"<<endl;

                                for(int i=1;i<=3;i++){

                                        if(gl[i]==1 and gl[i+1]==2 and ga[i]==1){// LEY
DE COSENOS Y LEY DE SENOS

                                                lado[3]=
sqrt(pow((lado[1]),2)+(pow((lado[2]),2))-(2*lado[1]*lado[2]*(cos(double (angulo[1]*(PI/180))))));

                                                angulo[3]= asin(double
((lado[2]*(sin(double (angulo[1]*(PI/180)))))/lado[3]));angulo[3]=angulo[3]*180/PI;

                                                angulo[2]=180-angulo[1]-angulo[3];

                                                }

                                        if(gl[i]==1 and gl[i+1]==3 and ga[i]==1){// LEY
DE SENOS

                                                angulo[2]= asin(double
((lado[1]*(sin(double (angulo[1]*(PI/180)))))/lado[3]));angulo[2]=angulo[2]*180/PI;

                                                angulo[3]=180-angulo[1]-angulo[2];

                                                lado[2]=lado[3]*(sin(double
(angulo[3]*(PI/180))))/(sin(double (angulo[1]*(PI/180))));

                                                }

                                        if(gl[i]==2 and gl[i+1]==3 and ga[i]==1){// LEY
DE SENOS

                                                angulo[3]= asin(double
((lado[2]*(sin(double (angulo[1]*(PI/180)))))/lado[3]));angulo[3]=angulo[3]*180/PI;

                                                angulo[2]=180-angulo[1]-angulo[3];

                                                lado[1]=lado[3]*(sin(double
(angulo[2]*(PI/180))))/(sin(double (angulo[1]*(PI/180))));

```

```

    }

    if(gl[i]==1 and gl[i+1]==2 and ga[i]==2){// LEY
DE SENOS

        angulo[3]= asin(double
((lado[2]*(sin(double (angulo[2]*(PI/180)))))/lado[1]));angulo[3]=angulo[3]*180/PI;

        angulo[1]=180-angulo[2]-angulo[3];

        lado[3]=lado[1]*(sin(double
(angulo[1]*(PI/180))))/(sin(double (angulo[2]*(PI/180))));

    }

    if(gl[i]==1 and gl[i+1]==3 and ga[i]==2){// LEY
DE SENOS

        angulo[1]= asin(double
((lado[3]*(sin(double (angulo[2]*(PI/180)))))/lado[1]));angulo[1]=angulo[1]*180/PI;

        angulo[3]=180-angulo[1]-angulo[2];

        lado[2]=lado[3]*(sin(double
(angulo[3]*(PI/180))))/(sin(double (angulo[3]*(PI/180))));

    }

    if(gl[i]==2 and gl[i+1]==3 and ga[i]==2){// LEY
DE COSENOS Y LEY DE SENOS

        lado[1]=
sqrt(pow((lado[2]),2)+(pow((lado[3]),2))-(2*lado[2]*lado[3]*(cos(double (angulo[2]*(PI/180))))));

        angulo[3]= asin(double
((lado[2]*(sin(double (angulo[2]*(PI/180)))))/lado[1]));angulo[3]=angulo[3]*180/PI;

        angulo[1]=180-angulo[3]-angulo[2];

    }

    if(gl[i]==1 and gl[i+1]==2 and ga[i]==3){// LEY
DE SENOS

        angulo[2]= asin(double
((lado[1]*(sin(double (angulo[3]*(PI/180)))))/lado[2]));angulo[2]=angulo[2]*180/PI;

        angulo[1]=180-angulo[3]-angulo[2];

        lado[3]=lado[2]*(sin(double
(angulo[1]*(PI/180))))/(sin(double (angulo[3]*(PI/180))));

    }

    if(gl[i]==1 and gl[i+1]==3 and ga[i]==3){// LEY
DE TANGENTES Y LEY DE SENOS

        x=((lado[1]-lado[3])*(tan(double
(((180-angulo[3])/2)*(PI/180))))/(lado[1]+lado[3]));x=atan(double (x));

```



```

(x*(180/PI));

angulo[2]=((180-angulo[3])/2)-

angulo[1]=(2*(x*(180/PI)))+angulo[2];

lado[2]=lado[1]*(sin(double

(angulo[3]*(PI/180))))/(sin(double (angulo[1]*(PI/180))));

}

if(gl[i]==2 and gl[i+1]==3 and ga[i]==3){// LEY

DE SENOS

angulo[1]= asin(double

((lado[3]*(sin(double (angulo[3]*(PI/180)))/lado[2]));angulo[1]=angulo[1]*180/PI;

angulo[2]=180-angulo[3]-angulo[1];

lado[1]=lado[2]*(sin(double

(angulo[2]*(PI/180)))/sin(double (angulo[3]*(PI/180))));

}

}

}

if(cl==1 and ca==2){cout<<"\t\t Dos angulos y un lado"<<endl;

for(int i=1;i<=3;i++){

if(gl[i]==1 and ga[i]==1 and ga[i+1]==2){// LEY

DE COSENOS Y LEY DE SENOS

angulo[3]=180-angulo[1]-angulo[2];

lado[3]= lado[1] * (sin(double

(angulo[1]*(PI/180)))/sin(double (angulo[2]*(PI/180))));

lado[2]=

sqrt(pow((lado[1]),2)+(pow((lado[3]),2))-(2*lado[1]*lado[3]*(cos(double (angulo[3]*(PI/180))))));

}

if(gl[i]==1 and ga[i]==1 and ga[i+1]==3){// LEY

DE COSENOS Y LEY DE SENOS

angulo[2]=180-angulo[1]-angulo[3];

lado[3]= lado[1] * (sin(double

(angulo[1]*(PI/180)))/sin(double (angulo[2]*(PI/180))));

lado[2]=

sqrt(pow((lado[1]),2)+(pow((lado[3]),2))-(2*lado[1]*lado[3]*(cos(double (angulo[3]*(PI/180))))));

}

if(gl[i]==1 and ga[i]==2 and ga[i+1]==3){// LEY

DE COSENOS Y LEY DE SENOS

angulo[1]=180-angulo[2]-angulo[3];

```

```

lado[3]= lado[1] * (sin(double
(angulo[1]*(PI/180)))/(sin(double (angulo[2]*(PI/180))));

lado[2]=
sqrt(pow((lado[1]),2)+(pow((lado[3]),2))-(2*lado[1]*lado[3]*(cos(double (angulo[3]*(PI/180))))));
}
if(gl[i]==2 and ga[i]==1 and ga[i+1]==2){// LEY
DE SENOS
angulo[3]=180-angulo[1]-angulo[2];
lado[3]= lado[2] * (sin(double
(angulo[1]*(PI/180)))/(sin(double (angulo[3]*(PI/180))));
lado[1]= lado[3] * (sin(double
(angulo[2]*(PI/180)))/(sin(double (angulo[1]*(PI/180))));
}
if(gl[i]==2 and ga[i]==1 and ga[i+1]==3){// LEY
DE COSENOS Y LEY DE SENOS
angulo[2]=180-angulo[1]-angulo[3];
lado[3]= lado[2] * (sin(double
(angulo[1]*(PI/180)))/(sin(double (angulo[3]*(PI/180))));
lado[2]=
sqrt(pow((lado[2]),2)+(pow((lado[3]),2))-(2*lado[2]*lado[3]*(cos(double (angulo[2]*(PI/180))))));
}
if(gl[i]==2 and ga[i]==2 and ga[i+1]==3){// LEY
DE SENOS
angulo[1]=180-angulo[2]-angulo[3];
lado[3]= lado[2] * (sin(double
(angulo[1]*(PI/180)))/(sin(double (angulo[3]*(PI/180))));
lado[1]= lado[2] * (sin(double
(angulo[2]*(PI/180)))/(sin(double (angulo[3]*(PI/180))));
}
if(gl[i]==3 and ga[i]==1 and ga[i+1]==2){// LEY
DE COSENOS Y LEY DE SENOS
angulo[3]=180-angulo[1]-angulo[2];
lado[1]= lado[3] * (sin(double
(angulo[2]*(PI/180)))/(sin(double (angulo[1]*(PI/180))));
lado[2]=
sqrt(pow((lado[1]),2)+(pow((lado[3]),2))-(2*lado[1]*lado[3]*(cos(double (angulo[3]*(PI/180))))));
}

```

```

DE COSENOS Y LEY DE SENOS
if(gl[i]==3 and ga[i]==1 and ga[i+1]==3){// LEY

    angulo[2]=180-angulo[1]-angulo[3];

    lado[1]= lado[3] * (sin(double
(angulo[2]*(PI/180)))/(sin(double (angulo[1]*(PI/180))));

    lado[2]=
sqrt(pow((lado[1]),2)+(pow((lado[3]),2))-(2*lado[1]*lado[3]*(cos(double (angulo[3]*(PI/180))))));

}

if(gl[i]==3 and ga[i]==2 and ga[i+1]==3){// LEY

DE COSENOS Y LEY DE SENOS

    angulo[1]=180-angulo[2]-angulo[3];

    lado[1]= lado[3] * (sin(double
(angulo[2]*(PI/180)))/(sin(double (angulo[1]*(PI/180))));

    lado[2]=
sqrt(pow((lado[1]),2)+(pow((lado[3]),2))-(2*lado[1]*lado[3]*(cos(double (angulo[3]*(PI/180))))));

}

}

}

if((cl+ca)>3){cout<<"Ingreso mas datos de los requeridos"<<endl;

}

area=0.5*lado[2]*lado[3]*((sin(double (angulo[2]*(PI/180))));

// MEDIANAS

mediana[1]=0.5*sqrt(2*((pow((lado[1]),2)))+(pow((lado[2]),2)))-
(pow((lado[3]),2)));

mediana[2]=0.5*sqrt(2*((pow((lado[2]),2)))+(pow((lado[3]),2)))-
(pow((lado[1]),2)));

mediana[3]=0.5*sqrt(2*((pow((lado[1]),2)))+(pow((lado[3]),2)))-
(pow((lado[2]),2)));

cout<<"\t\tPOR TANTO TENEMOS : "<<endl<<endl;

for(int i=1;i<=3;i++){

    cout<<"\t--> LADO < "<<i<<" > =
"<<lado[i]<<"[u]\t\t\t\tANGULO Alfa "<<i<<"* = "<<angulo[i]<<" grados"<<endl;

}cout<<"\n";cout<<"\n";

cout<<"\t--> MEDIANA A1-L3 = "<<mediana[1]<<" [u]"<<endl;

cout<<"\t--> MEDIANA A2-L1 = "<<mediana[2]<<" [u]"<<endl;

```

```

                                cout<<"\t--> MEDIANA A3-L2 = "<<mediana[3]<<"
[u]"<<endl<<endl;

                                cout<<"\t--> AREA = "<<area<<" [u^2]"<<endl;

                                cout<<"-----"
-----"<<endl;

                                break;

                                }

                                case 2:{cout<<"Hasta luego..."<<endl;

                                break;

                                }

                                default: cout<<"Error de entrada"<<endl;

                                }

                                cout<<"\nDESEA REALIZAR OTRA OPERACION (S/N) "<<endl;

                                cin>>op2;

                                }while(op2=='s' or op2=='S');

                                cout<<"\nGRACIAS..."<<endl;

                                return 0;

                                }

```

Capturas de pantalla

[illegible]

```
***** RESOLUCION DE TRIANGULOS *****  
-----  
1.- INGRESAR DATOS  
2.- SALIR  
  
INGRESE SU ELECCION:  
-----  
  
***** TRIANGULOS *****  
-----  
  
      \ 1* /  
     LADO 1   LADO 2  
    / 3* \_____/ 2* \  
   /_____\  \_____  
LADO 3
```

INGRESE LOS DATOS CONFORME SE DENOTA EN EL TRIANGULO MOSTRADO :
(En caso de no poseer un dato ingresar cero)
(Ingresar los angulos en grados positivos)

LADOS

```
--> Lado < 1 > = 0  
--> Lado < 2 > = 6  
--> Lado < 3 > = 0
```

ANGULOS

```
--> Angulo < 1 > en grados = 43  
--> Angulo < 2 > en grados = 72  
--> Angulo < 3 > en grados = 0
```

[illegible]

```

EJERCICIO #1 (C++).cpp
70 if(ca==3 and cl==0){cout<<"\t\tb) Tres angulos"<<endl;
71 cout<<"\t\tNO ES POSIBLE CALCULAR..."<<endl;
72 cout<<"\t\tDEBIDO A QUE EXISTEN INFINITOS TRIANGULOS DE DIVERSOS TAMANOS, QUE PUEDEN POSEER LOS MISMOS ANGULOS"<<endl;
73 }
74 // LEY DE SENOS con un angulo que NO forman estos
75 // LEY DE COSENOS con un angulo que SI forman estos
76 if(cl==2 and ca==1){cout<<"\t\tc) Dos lados y un angulo"<<endl;
77 for(int i=1;i<=3;i++){
78     if(gl[i]==1 and gl[i+1]==2 and ga[i]==1){// LEY DE COSENOS Y LEY DE SENOS
79         lado[3]= sqrt(pow((lado[1]),2)+pow((lado[2]),2)-(2*lado[1]*lado[2]*cos(double (angulo[1]*(PI/180))))));
80         angulo[3]= asin(double ((lado[2]*sin(double (angulo[1]*(PI/180))))/lado[3]));angulo[3]=angulo[3]*180/PI;
81         angulo[2]=180-angulo[1]-angulo[3];
82     }
83     if(gl[i]==1 and gl[i+1]==3 and ga[i]==1){// LEY DE SENOS
84         angulo[2]= asin(double ((lado[1]*sin(double (angulo[1]*(PI/180))))/lado[3]));angulo[2]=angulo[2]*180/PI;
85         angulo[3]=180-angulo[1]-angulo[2];
86         lado[2]=lado[3]*sin(double (angulo[3]*(PI/180)))/(sin(double (angulo[1]*(PI/180))));
87     }
88     if(gl[i]==2 and gl[i+1]==3 and ga[i]==1){// LEY DE SENOS
89         angulo[3]= asin(double ((lado[2]*sin(double (angulo[1]*(PI/180))))/lado[3]));angulo[3]=angulo[3]*180/PI;
90         angulo[2]=180-angulo[1]-angulo[3];
91         lado[1]=lado[3]*sin(double (angulo[2]*(PI/180)))/(sin(double (angulo[1]*(PI/180))));
92     }
93     if(gl[i]==1 and gl[i+1]==2 and ga[i]==2){// LEY DE SENOS
94         angulo[3]= asin(double ((lado[2]*sin(double (angulo[2]*(PI/180))))/lado[1]));angulo[3]=angulo[3]*180/PI;
95         angulo[1]=180-angulo[2]-angulo[3];
96         lado[3]=lado[1]*sin(double (angulo[1]*(PI/180)))/(sin(double (angulo[2]*(PI/180))));
97     }
98     if(gl[i]==1 and gl[i+1]==3 and ga[i]==2){// LEY DE SENOS
99         angulo[1]= asin(double ((lado[3]*sin(double (angulo[2]*(PI/180))))/lado[1]));angulo[1]=angulo[1]*180/PI;
100        angulo[3]=180-angulo[1]-angulo[2];
101        lado[2]=lado[3]*sin(double (angulo[3]*(PI/180)))/(sin(double (angulo[1]*(PI/180))));
102    }
103    if(gl[i]==2 and gl[i+1]==3 and ga[i]==2){// LEY DE COSENOS Y LEY DE SENOS
104        lado[1]= sqrt(pow((lado[2]),2)+pow((lado[3]),2)-(2*lado[2]*lado[3]*cos(double (angulo[2]*(PI/180))))));
105        angulo[3]= asin(double ((lado[2]*sin(double (angulo[2]*(PI/180))))/lado[1]));angulo[3]=angulo[3]*180/PI;
106        angulo[1]=180-angulo[2]-angulo[3];

```

EJERCICIO 2:

- MATRICES DE VANDERMONDE Y SU DETERMINANTE

Base teórica

En álgebra lineal con matriz de Vandermonde indicamos una matriz cuyas **FILAS** o **COLUMNAS** tienen elementos, a partir de 1, en progresión geométrica

$$A = \begin{vmatrix} 1 & a & a^2 \\ 1 & b & b^2 \\ 1 & c & c^2 \end{vmatrix} \quad B = \begin{vmatrix} 1 & 1 & 1 & 1 \\ a & b & c & d \\ a^2 & b^2 & c^2 & d^2 \\ a^3 & b^3 & c^3 & d^3 \end{vmatrix}$$

Y para hallar la determinante:

Determinante de Vandermonde

- ♦ O determinante da matriz de Vandermonde pode ser escrito da forma seguinte:

$$\begin{vmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ 1 & \dots & \dots & \dots \\ 1 & x_n & \dots & x_n^n \end{vmatrix} = \prod_{0 \leq i < j \leq n} (x_j - x_i)$$

MATRIZ

Un arreglo (matriz) es una colección ordenada de datos (tanto primitivos u objetos dependiendo del lenguaje). Los arreglos (matrices) se emplean para almacenar multiples valores en una sola variable, frente a las variables que sólo pueden almacenar un valor (por cada variable).

VECTOR

En programación, se le denomina vector, formación, matriz (en inglés array, del cual surge la mala traducción arreglo), a una zona de almacenamiento contiguo que contiene una serie de elementos del mismo tipo, los elementos de la matriz.

FOR

Un for en programación se usa cuando queremos repetir un conjunto de instrucciones un número finito de veces.

SWITCH

Usar switch ayuda a simplificar el código y evita confusiones, ya que organiza en varias ramas el código que va a ser ejecutado.

DO WHILE

es una estructura de repetición o iterativa, ejecutamos primero todo el bloque de instrucciones, y al final siempre comprobaremos la condición para salir.

IF

Un if en programación se utiliza para evaluar una expresión condicional: si se cumple la condición (es verdadera), ejecutará un bloque de código. Si es falsa, es posible ejecutar otras sentencias.

Como hemos dicho, la condición es una condición lógica, sólo devolverá true o false, y se ejecutará si su valor es true.

PARA LAS OPERACIONES

```
for (int i=0;i<n;i++){
```

```

        for (int j=0;j<n;j++){
            Matriz[i][j]=pow((Vector[i]),j);
        }
    }
}

```

PARA EL DETERMINANTE

```

For (int i=0;i<n-1;i++){
    for(int j=i+1;j<n;j++){
        det=det*(Vector[j]-Vector[i]);
    }
}

```

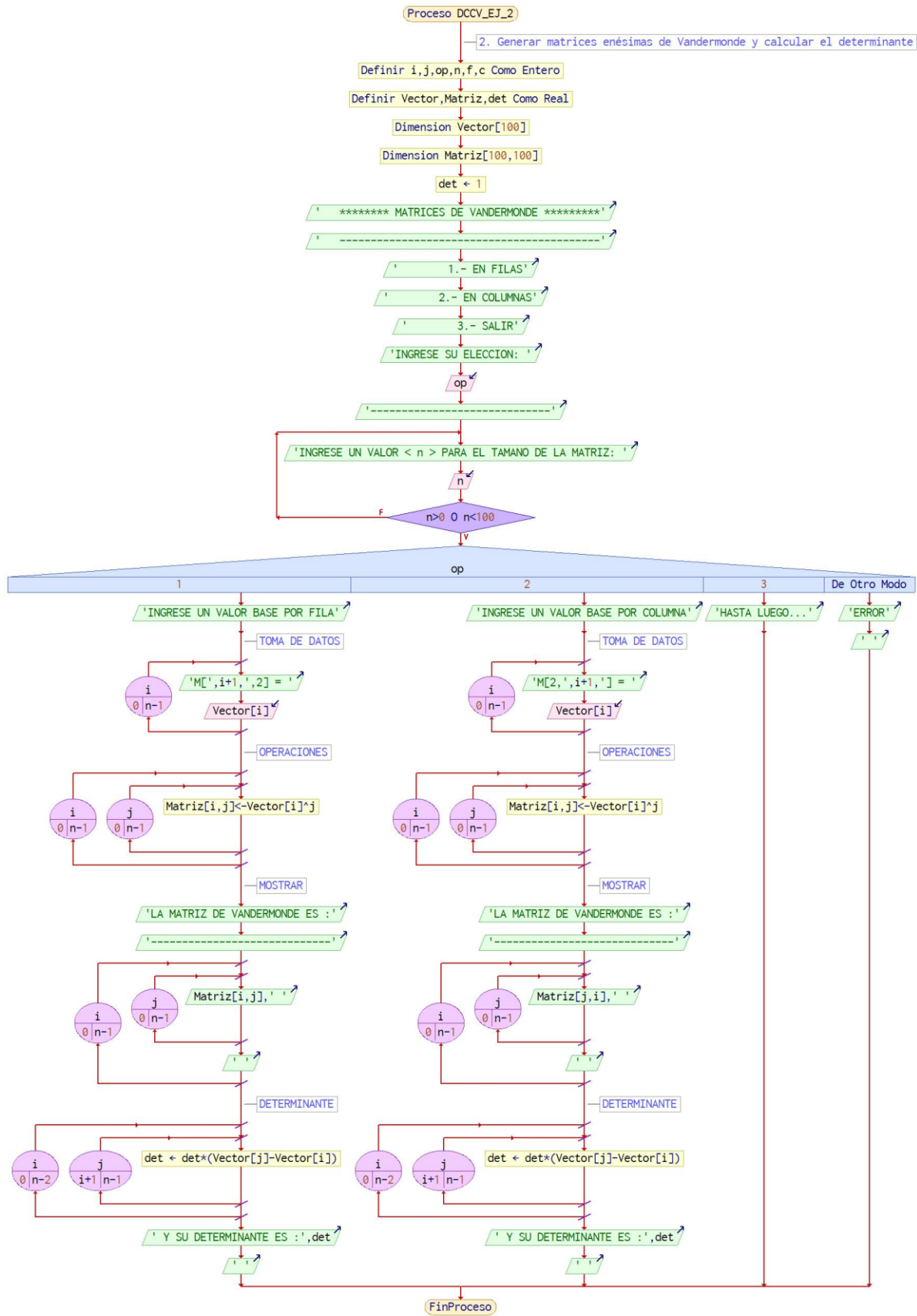
$$\begin{aligned}
 13. \quad |H| &= \begin{vmatrix} 1 & 1 & 1 \\ a & b & c \\ a^2 & b^2 & c^2 \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ 0 & b-a & c-a \\ 0 & b^2-a^2 & c^2-a^2 \end{vmatrix} = \begin{vmatrix} (b-a) & (c-a) \\ (b-a)(b+a) & (c-a)(b-a) \end{vmatrix} = \\
 & (b-a)(c-a) \begin{vmatrix} 1 & 1 \\ (b+a) & (c+a) \end{vmatrix} = (b-a)(c-a)(c-b)
 \end{aligned}$$

• Resta a cada fila "a" por la fila anterior.

$$H = \begin{vmatrix} 1 & 1 & 1 & 1 \\ a & b & c & d \\ a^2 & b^2 & c^2 & d^2 \\ a^3 & b^3 & c^3 & d^3 \end{vmatrix} = (b-a)(c-a)(d-a)(c-b)(d-b)(d-c)$$

Diagrama de flujo

Diagrama realizado en PSEINT



Listado del código

```
#include<iostream>
```

```
#include<stdio.h>
```

```
#include<math.h>
```

```
using namespace std;
```

```
int main(){
```

```
    int op,n,f,c;
```

```
    double det=1;
```

```
    float Vector[100];
```

```
    float Matriz[100][100];
```

```
    cout<<"\n\t***** MATRICES DE VANDERMONDE *****"<<endl;
```

```
    cout<<"\t-----"<<endl;
```

```
    cout<<"\t\t1.- EN FILAS"<<endl;
```

```
    cout<<"\t\t2.- EN COLUMNAS"<<endl;
```

```
    cout<<"\t\t3.- SALIR"<<endl<<endl;
```

```
    cout<<"INGRESE SU ELECCION: "<<endl;
```

```
    cin>>op;
```

```
    cout<<"-----"<<endl;
```

```
    do{cout<<"INGRESE UN VALOR < n > PARA EL TAMANO DE LA MATRIZ: "<<endl;
```

```
        cin>>n;
```

```
    }while(n<0 or n>100);
```

```
    switch(op){
```

```
        case 1:{cout<<"INGRESE UN VALOR BASE POR FILA : "<<endl;
```

```
                //TOMA DE DATOS
```

```
                for (int i=0;i<n;i++){
```

```
                    cout<<"M["<<i+1<<","2] = ";
```

```

        cin>>Vector[i];
    }

    //OPERACIONES
    for (int i=0;i<n;i++){
        for (int j=0;j<n;j++){
            Matriz[i][j]=pow((Vector[i]),j);
        }
    }

    //MOSTRAR
    cout<<"\n\tLA MATRIZ DE VANDERMONDE ES : "<<endl<<endl;
    cout<<"-----"<<endl;
    for (int i=0;i<n;i++){
        cout<<"\t";
        for (int j=0;j<n;j++){
            cout<<Matriz[i][j]<<"\t";
        }cout<<"\n";
    }

    //DETERMINANTE
    for(int i=0;i<n-1;i++){
        for(int j=i+1;j<n;j++){
            det=det*(Vector[j]-Vector[i]);
        }
    }

    cout<<"\n\tY SU DETERMINANTE ES : " <<det<<endl;
    break;
}

case 2:{cout<<"INGRESE UN VALOR BASE POR COLUMNA: "<<endl;

    //TOMA DE DATOS
    for (int i=0;i<n;i++){
        cout<<"M[2,"<<i+1<<"] = "<<endl;

```

```

        cin>>Vector[i];
    }

    //OPERACIONES
    for (int i=0;i<n;i++){
        for (int j=0;j<n;j++){
            Matriz[i][j]=pow((Vector[i]),j);
        }
    }

    //MOSTRAR
    cout<<"\n\tLA MATRIZ DE VANDERMONDE ES : "<<endl<<endl;
    cout<<"-----"<<endl;
    for (int i=0;i<n;i++){
        cout<<"\t";
        for (int j=0;j<n;j++){
            cout<<Matriz[j][i]<<"\t";
        }cout<<"\n";
    }

    //DETERMINANTE POR VANDERMONDE
    for(int i=0;i<n-1;i++){
        for(int j=i+1;j<n;j++){
            det=det*(Vector[j]-Vector[i]);
        }
    }

    cout<<"\n\tY SU DETERMINANTE ES : " <<det<<endl;

    break;
}

case 3:{cout<<"Hasta luego..."<<endl;
    break;
}

```

```

        default: cout<<"Error"<<endl;

    }

    return 0;

}

```

Capturas de pantalla

```

EJERCICIO #1.cpp × EJERCICIO #2.cpp × EJERCICIO #3.cpp × EJERCICIO #4.cpp ×
1 //2. Generar matrices enésimas de Vandermonde y calcular el determinante
2
3 #include<iostream>
4 #include<stdio.h>
5 #include<math.h>
6
7 using namespace std;
8
9 int main(){
10     int op,n,f,c;
11     double det=1;
12     float Vector[100];
13     float Matriz[100][100];
14
15     cout<<"\n\t***** MATRICES DE VANDERMONDE *****"<<endl;
16     cout<<"\t-----"<<endl;
17     cout<<"\t\t1.- EN FILAS"<<endl;
18     cout<<"\t\t2.- EN COLUMNAS"<<endl;
19     cout<<"\t\t3.- SALIR"<<endl<<endl;
20     cout<<"INGRESE SU ELECCION: "<<endl;
21     cin>>op;
22     cout<<"-----"<<endl;
23     do{cout<<"INGRESE UN VALOR < n > PARA EL TAMANO DE LA MATRIZ: "<<endl;
24         cin>>n;
25     }while(n<0 or n>100);
26     switch(op){
27     case 1:{cout<<"INGRESE UN VALOR BASE POR FILA : "<<endl;
28         //TOMA DE DATOS
29         for (int i=0;i<n;i++){
30             cout<<"M["<<i+1<<","2] = ";
31             cin>>Vector[i];
32         }
33         //CONSTRUCCION

```

```

***** MATRICES DE VANDERMONDE *****
-----
1.- EN FILAS
2.- EN COLUMNAS
3.- SALIR

INGRESE SU ELECCION:
1
-----
INGRESE UN VALOR < n > PARA EL TAMANO DE LA MATRIZ:
5
INGRESE UN VALOR BASE POR FILA :
M[1,2] = 2
M[2,2] = 3
M[3,2] = 4
M[4,2] = 5
M[5,2] = 6

LA MATRIZ DE VANDERMONDE ES :

-----
1      2      4      8      16
1      3      9      27     81
1      4      16     64     256
1      5      25     125     625
1      6      36     216     1296

Y SU DETERMINANTE ES : 288

-----
Process exited after 20.71 seconds with return value 0
Presione una tecla para continuar . . .

```

```

EJERCICIO #1.cpp x EJERCICIO #2.cpp x EJERCICIO #3.cpp x EJERCICIO #4.cpp x
37 }
38 //MOSTRAR
39 cout<<"\n\tLA MATRIZ DE VANDERMONDE ES : "<<endl<<endl;
40 cout<<"-----"<<endl;
41 for (int i=0;i<n;i++){
42     cout<<"\t";
43     for (int j=0;j<n;j++){
44         cout<<Matriz[i][j]<<"\t";
45     }cout<<"\n";
46 }
47 //DETERMINANTE
48 for(int i=0;i<n-1;i++){
49     for(int j=i+1;j<n;j++){
50         det=det*(Vector[j]-Vector[i]);
51     }
52 }
53 cout<<"\n\tY SU DETERMINANTE ES : " <<det<<endl;
54 break;
55 }
56 case 2:{cout<<"INGRESE UN VALOR BASE POR COLUMNA: "<<endl;
57     //TOMA DE DATOS
58     for (int i=0;i<n;i++){
59         cout<<"M[2,"<<i+1<<" ] = "<<endl;
60         cin>>Vector[i];
61     }
62     //OPERACIONES
63     for (int i=0;i<n;i++){
64         for (int j=0;j<n;j++){
65             Matriz[i][j]=pow((Vector[i]),j);
66         }
67     }
68     //MOSTRAR
69     cout<<"\n\tLA MATRIZ DE VANDERMONDE ES : "<<endl<<endl;
70     cout<<"-----"<<endl;
71 }

```

```

C:\Users\HP\OneDrive\Escritorio\CRIS\LAB PROG\LABORATORIO ...
***** MATRICES DE VANDERMONDE *****
-----
1.- EN FILAS
2.- EN COLUMNAS
3.- SALIR

INGRESE SU ELECCION:
1
-----
INGRESE UN VALOR < n > PARA EL TAMANO DE LA MATRIZ:
5
INGRESE UN VALOR BASE POR FILA :
M[1,2] = 2
M[2,2] = 3
M[3,2] = 4
M[4,2] = 5
M[5,2] = 6

LA MATRIZ DE VANDERMONDE ES :
-----
1      2      4      8      16
1      3      9      27     81
1      4      16     64     256
1      5      25     125    625
1      6      36     216    1296

Y SU DETERMINANTE ES : 288
-----
Process exited after 20.71 seconds with return value 0
Presione una tecla para continuar . . .

```

```

EJERCICIO #1.cpp x EJERCICIO #2.cpp x EJERCICIO #3.cpp x EJERCICIO #4.cpp x
59 for (int i=0;i<n;i++){
60     cout<<"M[2,"<<i+1<<" ] = "<<endl;
61     cin>>Vector[i];
62 }
63 //OPERACIONES
64 for (int i=0;i<n;i++){
65     for (int j=0;j<n;j++){
66         Matriz[i][j]=pow((Vector[i]),j);
67     }
68 }
69 //MOSTRAR
70 cout<<"\n\tLA MATRIZ DE VANDERMONDE ES : "<<endl<<endl;
71 cout<<"-----"<<endl;
72 for (int i=0;i<n;i++){
73     cout<<"\t";
74     for (int j=0;j<n;j++){
75         cout<<Matriz[j][i]<<"\t";
76     }cout<<"\n";
77 }
78 //DETERMINANTE POR VANDERMONDE
79 for(int i=0;i<n-1;i++){
80     for(int j=i+1;j<n;j++){
81         det=det*(Vector[j]-Vector[i]);
82     }
83 }
84
85 cout<<"\n\tY SU DETERMINANTE ES : " <<det<<endl;
86 break;
87 }
88 case 3:{cout<<"Hasta luego..."<<endl;
89     break;
90 }
91 default: cout<<"Error"<<endl;
92 }
93 return 0;
94 }

```

```

C:\Users\HP\OneDrive\Escritorio\CRIS\LAB PROG\LABORATORIO ...
***** MATRICES DE VANDERMONDE *****
-----
1.- EN FILAS
2.- EN COLUMNAS
3.- SALIR

INGRESE SU ELECCION:
1
-----
INGRESE UN VALOR < n > PARA EL TAMANO DE LA MATRIZ:
5
INGRESE UN VALOR BASE POR FILA :
M[1,2] = 2
M[2,2] = 3
M[3,2] = 4
M[4,2] = 5
M[5,2] = 6

LA MATRIZ DE VANDERMONDE ES :
-----
1      2      4      8      16
1      3      9      27     81
1      4      16     64     256
1      5      25     125    625
1      6      36     216    1296

Y SU DETERMINANTE ES : 288
-----
Process exited after 20.71 seconds with return value 0
Presione una tecla para continuar . . .

```

EJERCICIO 2:

- DIVISION DE POLINOMIOS POR HORNER

Base teórica

Ejemplo 2: Dividir $D(x) = 25x^5 - x^2 + 4x^3 - 5x^4 + 8$ por $d(x) = 5x^2 - 3 + 2x$


Solución:

Ordenando y completando los términos del dividendo y divisor:

$$D(x) = 25x^5 - 5x^4 + 4x^3 - x^2 + 0x + 8, \quad d(x) = 5x^2 + 2x - 3$$

Coeficiente principal del d(x)		Coeficientes del D(x)					
5		25	-5	4	-1	0	8
Demás coeficientes del d(x) con signo cambiado	-2		-10	15			
	3			6	-9		
			-15		-10	15	
				25	-20	8	-12
		5	-3	5	-4	23	-4
		coeficientes del cociente q(x)				coeficientes del resto	

$q(x) = 5x^3 - 3x^2 + 5x - 4$
 $r(x) = 23x - 4$



APLICACION DE HORNER

// DIVISION

```
for(int i=0;i<=n2;i++){
```

```
    Cociente[i+1]=MH[0][i+1]/MH[i][i];
```

```
}
```

```
for(int k=0;k<=n1+1;k++){
```

```
    // MULTIPLICACION 1
```

```
    for(int i=k+1;i<=k+1;i++){
```

```
        for(int j=k+1;j<=n2+k;j++){
```

```
            if(i==resto+1){break;
```

```
        }
```

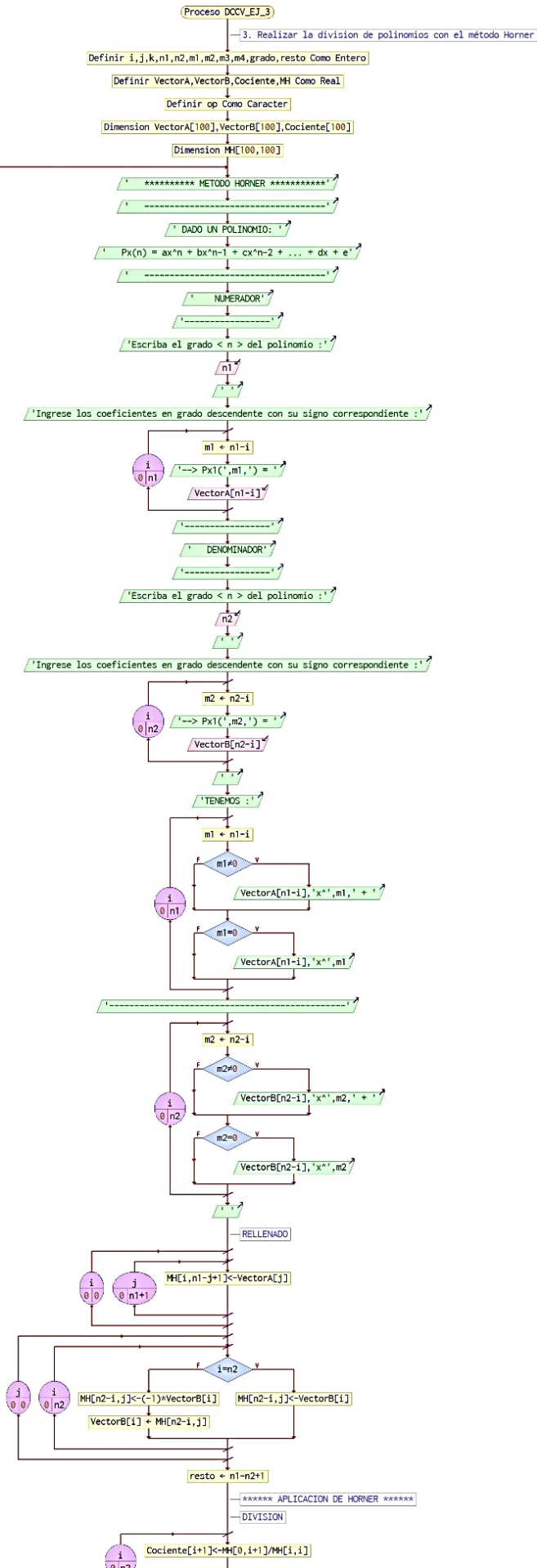
```

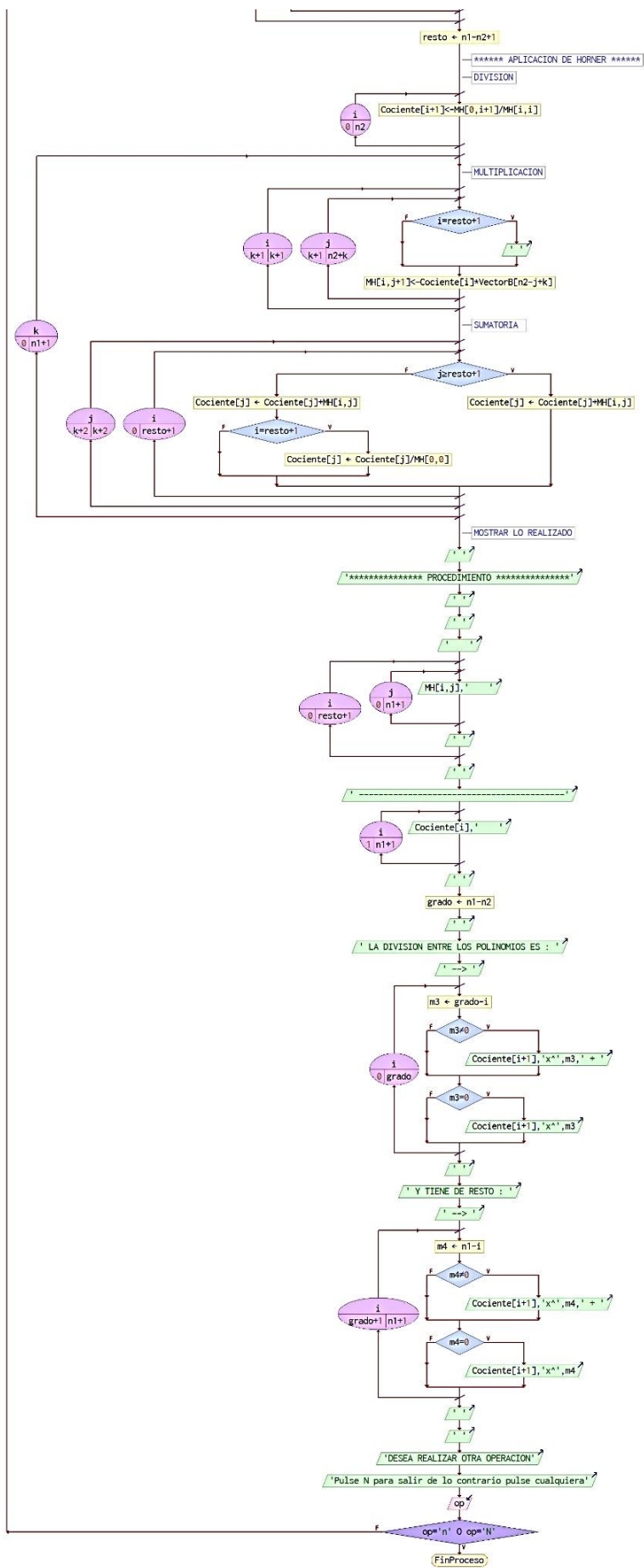
        MH[i][j+1]=Cociente[i]*VectorB[n2-j+k];
    }
}
// SUMATORIA 1
for(int j=k+2;j<=k+2;j++){Cociente[j]=0;
    for(int i=0;i<=resto+1;i++){
        if(j>=resto+1){
            Cociente[j]=Cociente[j]+MH[i][j];
        }else{
            Cociente[j]=Cociente[j]+MH[i][j];
            if(i==resto+1){
                Cociente[j]=Cociente[j]/MH[0][0];
            }
        }
    }
}
}
}

```

Diagrama de flujo

Diagrama realizado en PSEINT





Listado del código

```
#include <iostream>

#include <conio.h>

#include <math.h>

using namespace std;

int main(){

    int n1,n2,m1,m2,m3,m4,grado,resto;

    float VectorA[100],VectorB[100],Cociente[100];

    float MH[100][100];

    char op;

    do{

        cout<<"\n\t\t***** METODO HORNER *****"<<endl;

        cout<<"-----"<<endl;

        cout<<"DADO UN POLINOMIO: "<<endl;

        cout<<"\t\tPx(n) = ax^n + bx^n-1 + cx^n-2 + ... + dx + e"<<endl;

        cout<<"-----"<<endl;

        cout<<"\t\tNUMERADOR"<<endl;

        cout<<"-----"<<endl;

        cout<<"Escriba el grado < n > del polinomio : "<<endl;

        cin>>n1;cout<<"\n";

        cout<<"Ingrese los coeficientes en grado descendente con su signo correspondiente  
:"<<endl;

        for(int i=0;i<=n1;i++){

            m1=n1-i;

            cout<<"\n--> Px1("<m1<") = ";

            cin>>VectorA[n1-i];
```

```

    }

    cout<<"-----"<<endl;

    cout<<"\tDENOMINADOR"<<endl;

    cout<<"-----"<<endl;

    cout<<"Escriba el grado < n > del polinomio : "<<endl;

    cin>>n2;cout<<"\n";

    cout<<"Ingrese los coeficientes en grado descendente con su signo correspondiente
    : "<<endl;

    for(int i=0;i<=n2;i++){

        m2=n2-i;

        cout<<"\n--> Px2(" <<m2<<" ) = ";

        cin>>VectorB[n2-i];

    }cout<<"\n";

    cout<<"\tTENEMOS: " <<endl;cout<<"\n\t\t";

    for(int i=0;i<=n1;i++){

        m1=n1-i;

        if(m1!=0){

            cout<<VectorA[n1-i]<<"x^"<<m1<<" + ";

        }

        if(m1==0){

            cout<<VectorA[n1-i]<<"x^"<<m1;

        }

    }cout<<"\n";

    cout<<"\t--> -----" <<endl;cout<<"\t\t";

    for(int i=0;i<=n2;i++){

        m2=n2-i;

        if(m2!=0){

            cout<<VectorB[n2-i]<<"x^"<<m2<<" + ";

        }

        if(m2==0){

```

```

        cout<<VectorB[n2-i]<<"x^"<<m2;

    }
}cout<<"\n";cout<<"\n";

// RELLENADO
for(int i=0;i<=0;i++){

    for(int j=0;j<=n1+1;j++){

        MH[i][n1-j+1]=VectorA[j];

    }

}

for(int j=0;j<=0;j++){

    for(int i=0;i<=n2;i++){

        if(i==n2){

            MH[n2-i][j]=VectorB[i];

        }else{

            MH[n2-i][j]=(-1)*VectorB[i];

            VectorB[i]=MH[n2-i][j];

        }

    }

}

resto=n1-n2+1;

// ***** APLICACION DE HORNER *****

// DIVISION

for(int i=0;i<=n2;i++){

    Cociente[i+1]=MH[0][i+1]/MH[i][i];

}

for(int k=0;k<=n1+1;k++){

    // MULTIPLICACION 1

    for(int i=k+1;i<=k+1;i++){

        for(int j=k+1;j<=n2+k;j++){

            if(i==resto+1){break;

```

```

        }

        MH[i][j+1]=Cociente[i]*VectorB[n2-j+k];

    }

}

// SUMATORIA 1
for(int j=k+2;j<=k+2;j++){Cociente[j]=0;

    for(int i=0;i<=resto+1;i++){

        if(j>=resto+1){

            Cociente[j]=Cociente[j]+MH[i][j];

        }else{

            Cociente[j]=Cociente[j]+MH[i][j];

            if(i==resto+1){

                Cociente[j]=Cociente[j]/MH[0][0];

            }

        }

    }

}

}

// MOSTRAR LO REALIZADO
cout<<"\n\t***** PROCEDIMIENTO *****"<<endl<<endl;

for(int i=0;i<=resto+1;i++){cout<<"\t";

    for(int j=0;j<=n1+1;j++){

        cout<<MH[i][j]<<"\t";

    }cout<<"\n";

}cout<<"\n";cout<<"-----"

"<<endl;cout<<"\t\t";

for(int i=1;i<=n1+1;i++){

    cout<<Cociente[i]<<"\t";

}cout<<"\n";

grado=n1-n2;

```

```

        cout<<"\n LA DIVISION ENTRE LOS POLINOMIOS ES : "<<endl<<endl;
        cout<<"\t--> ";
        for(int i=0;i<=grado;i++){
            m3=grado-i;
            if(m3!=0){
                cout<<Cociente[i+1]<<"x^"<<m3<<" + ";
            }
            if(m3==0){
                cout<<Cociente[i+1]<<"x^"<<m3;
            }
        }cout<<"\n";
        cout<<"\n Y TIENE RESTO : "<<endl<<endl;
        cout<<"\t--> ";
        for(int i=grado+1;i<=n1+1;i++){
            m4=n1-i;
            if(m4!=0){
                cout<<Cociente[i+1]<<"x^"<<m4<<" + ";
            }
            if(m4==0){
                cout<<Cociente[i+1]<<"x^"<<m4;
                break;
            }
        }cout<<"\n\n";
        cout<<"\nDESEA REALIZAR OTRA OPERACION (S/N) "<<endl;
        cin>>op;
    }while(op=='s' or op=='S');
    cout<<"\nGRACIAS..."<<endl;

    return 0;

```

}

Capturas de pantalla

EJERCICIO #1.cpp × EJERCICIO #2.cpp × EJERCICIO #3.cpp × EJERCICIO #4.cpp ×

```
1 //3. Realizar la division de polinomios con el método Horner
2
3 #include <iostream>
4 #include <conio.h>
5 #include <math.h>
6 using namespace std;
7
8 int main(){
9     int n1,n2,m1,m2,m3,m4,grado,resto;
10    float VectorA[100],VectorB[100],Cociente[100];
11    float MH[100][100];
12    char op;
13
14    do{
15        cout<<"\n\t\t***** METODO HORNER *****"<<endl;
16        cout<<"-----"<<endl;
17        cout<<"DADO UN POLINOMIO: "<<endl;
18        cout<<"\t\tP(x) = ax^n + bx^n-1 + cx^n-2 + ... + dx + e"<<endl;
19        cout<<"-----"<<endl;
20        cout<<"\t\tNUMERADOR"<<endl;
21        cout<<"-----"<<endl;
22        cout<<"Escriba el grado < n > del polinomio : "<<endl;
23        cin>>n1;cout<<"\n";
24        cout<<"Ingrese los coeficientes en grado descendente con su signo correspondiente : "<<endl;
25        for(int i=0;i<=n1;i++){
26            m1=n1-i;
27            cout<<"\n--> Px1("&<m1<<" = ";
28            cin>>VectorA[n1-i];
29        }
30        cout<<"-----"<<endl;
31        cout<<"\t\tDENOMINADOR"<<endl;
32        cout<<"-----"<<endl;
33        cout<<"Escriba el grado < n > del polinomio : "<<endl;
34        cin>>n2;cout<<"\n";
35        cout<<"Ingrese los coeficientes en grado descendente con su signo correspondiente : "<<endl;
36        for(int i=0;i<=n2;i++){
37            m2=n2-i;
```

C:\Users\HP\OneDrive\Escritorio\CRIS\LAB PROG\LABORATORIO # 2\EJERCICIO 3...

```
***** METODO HORNER *****
DADO UN POLINOMIO:
P(x) = ax^n + bx^n-1 + cx^n-2 + ... + dx + e
-----
NUMERADOR
Escriba el grado < n > del polinomio :
4
Ingrese los coeficientes en grado descendente con su signo correspondiente :
--> Px1(4) = 38
--> Px1(3) = -65
--> Px1(2) = 0
--> Px1(1) = 0
--> Px1(0) = 27
-----
DENOMINADOR
Escriba el grado < n > del polinomio :
2
Ingrese los coeficientes en grado descendente con su signo correspondiente :
--> Px2(2) = 2
--> Px2(1) = -5
--> Px2(0) = 3
-----
TENEMOS:
-->
38x^4 + -65x^3 + 0x^2 + 0x^1 + 27x^0
-----
2x^2 + -5x^1 + 3x^0
```

EJERCICIO #1.cpp × EJERCICIO #2.cpp × EJERCICIO #3.cpp × EJERCICIO #4.cpp ×

```
34 cin>>n2;cout<<"\n";
35 cout<<"Ingrese los coeficientes en grado descendente con su signo correspondiente : "<<endl;
36 for(int i=0;i<=n2;i++){
37     m2=n2-i;
38     cout<<"\n--> Px2("&<m2<<" = ";
39     cin>>VectorB[n2-i];
40 }cout<<"\n";
41 cout<<"\t\tTENEMOS: "<<endl;cout<<"\n\t\t";
42 for(int i=0;i<=n1;i++){
43     m1=n1-i;
44     if(m1!=0){
45         cout<<VectorA[n1-i]<<"x^"<<m1<<" + ";
46     }
47     if(m1==0){
48         cout<<VectorA[n1-i]<<"x^"<<m1;
49     }
50 }cout<<"\n";
51 cout<<"\t\t-----"<<endl;cout<<"\t\t";
52 for(int i=0;i<=n2;i++){
53     m2=n2-i;
54     if(m2!=0){
55         cout<<VectorB[n2-i]<<"x^"<<m2<<" + ";
56     }
57     if(m2==0){
58         cout<<VectorB[n2-i]<<"x^"<<m2;
59     }
60 }cout<<"\n";cout<<"\n";
61 // RELLENADO
62 for(int i=0;i<=0;i++){
63     for(int j=0;j<=n1+1;j++){
64         MH[i][n1-j+1]=VectorA[j];
65     }
66 }
67 for(int j=0;j<=0;j++){
68     for(int i=0;i<=n2;i++){
69         if(i==n2){
70             MH[n2-i][j]=VectorB[i];
71         }
```

C:\Users\HP\OneDrive\Escritorio\CRIS\LAB PROG\LABORATORIO # 2\EJERCICIO 3...

```
Escriba el grado < n > del polinomio :
2
Ingrese los coeficientes en grado descendente con su signo correspondiente :
--> Px2(2) = 2
--> Px2(1) = -5
--> Px2(0) = 3
-----
TENEMOS:
-->
38x^4 + -65x^3 + 0x^2 + 0x^1 + 27x^0
-----
2x^2 + -5x^1 + 3x^0

***** PROCEDIMIENTO *****
2    38    -65    0    0    27
5    0    95    -57    0    0
-3    0    0    75    -45    0
0    0    0    0    45    -27
0    0    0    0    0    0
-----
19    15    9    0    0

LA DIVISION ENTRE LOS POLINOMIOS ES :
--> 19x^2 + 15x^1 + 9x^0

Y TIENE RESTO :
--> 0x^1 + 0x^0

DESEA REALIZAR OTRA OPERACION (S/N)
```


EJERCICIO # 1.cpp
EJERCICIO # 2.cpp
EJERCICIO # 3.cpp
EJERCICIO # 4.cpp

```

61 // RELLENADO
62 for(int i=0; i<=n1; i++){
63     for(int j=0; j<=n1+1; j++){
64         MH[i][n1-j+1]=VectorA[j];
65     }
66 }
67 for(int j=0; j<=n2; j++){
68     for(int i=0; i<=n2; i++){
69         if(i==n2){
70             MH[n2-i][j]=VectorB[i];
71         }else{
72             MH[n2-i][j]=(-1)*VectorB[i];
73             VectorB[i]=MH[n2-i][j];
74         }
75     }
76 }
77 resto=n1-n2+1;
78 // ***** APLICACION DE HORNER *****
79 // DIVISION
80 for(int i=0; i<=n2; i++){
81     Cociente[i+1]=MH[0][i+1]/MH[i][i];
82 }
83 for(int k=0; k<=n1+1; k++){
84     // MULTIPLICACION 1
85     for(int i=k+1; i<=k+1; i++){
86         for(int j=k+1; j<=n2+k; j++){
87             if(i==resto+1){break;}
88         }
89         MH[i][j+1]=Cociente[i]*VectorB[n2-j+k];
90     }
91 }
92 // SUMATORIA 1
93 for(int j=k+2; j<=k+2; j++){Cociente[j]=0;}
94 for(int i=0; i<=resto+1; i++){
95     if(j>=resto+1){
96         Cociente[j]=Cociente[j]+MH[i][j];
97     }else{
98         Cociente[i]=Cociente[i]+MH[i][i];
99     }

```

```

-----
Escriba el grado < n > del polinomio :
2
Ingrese los coeficientes en grado descendente con su signo correspondiente :
--> Px2(2) = 2
--> Px2(1) = -5
--> Px2(0) = 3

TENEMOS:
38x^4 + -65x^3 + 0x^2 + 0x^1 + 27x^0
-->
-----
2x^2 + -5x^1 + 3x^0

***** PROCEDIMIENTO *****
2  38  -65  0  0  27
5  0  95  -57  0  0
-3  0  0  75  -45  0
0  0  0  0  45  -27
0  0  0  0  0  0
-----
19  15  9  0  0

LA DIVISION ENTRE LOS POLINOMIOS ES :
--> 19x^2 + 15x^1 + 9x^0

Y TIENE RESTO :
--> 0x^1 + 0x^0

DESEA REALIZAR OTRA OPERACION (S/N)

```

EJERCICIO 4:

- RESOLUCION DE UN SISTEMA DE ECUACIONES LINEALES (2X2)

Base teórica

Introducción: Recordemos que los **Sistemas de Ecuaciones Lineales 2x2** son aquello que se componen de **dos** ecuaciones con **dos** incógnitas, y existen varios métodos para llegar a su solución en caso de existir.

Un método que se utiliza es el siguiente:

Método de (CRAMER)

Antes de iniciar con el paso a paso de este método, es pertinente recordar qué es una matriz 2x2 y qué es un determinante.

Una matriz 2x2 no es más que un arreglo de elementos que posee dos columnas y dos filas

Matriz 2x2.
 Dos filas y dos columnas

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

Y un determinante de una matriz 2x2 consiste en restar el producto de las diagonales de la matriz:

$$\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc$$

Veamos que sí es la resta del producto de las diagonales:

$$\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \boxed{ad} - bc$$

$$\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - \boxed{bc}$$

Paso 1. Se prepara la matriz de los coeficientes y se halla el determinante

Identificamos los coeficientes de las incógnitas y construimos la **matriz M** con ellos:

$$\begin{array}{l} \boxed{2x + 3y = 20} \quad \text{Ecuación 1} \\ \boxed{x - 2y = 3} \quad \text{Ecuación 2} \end{array}$$

Matriz de los coeficientes.

$$\begin{array}{cc} x & y \\ \downarrow & \downarrow \\ M = \begin{bmatrix} 2 & 3 \\ 1 & -2 \end{bmatrix} \end{array}$$

Calculamos su determinante:

$$|M| = (2)(-2) - (3)(1)$$

$$|M| = -4 - 3 = -7$$

Bien, ya tenemos que el determinante de la matriz de coeficientes es -7

Paso 2. Se prepara la matriz de la incógnita **x**, y se halla el determinante

La matriz de la **incógnita X** es la misma matriz de coeficientes con una diferencia. En lugar de colocar los coeficientes de **X**, se ubican los valores numéricos que quedaron al otro lado de las ecuaciones.

$$\begin{array}{lcl} 2x + 3y = 20 & \text{Ecuación 1} \\ x - 2y = 3 & \text{Ecuación 2} \end{array}$$

Matriz de los coeficientes.

Veamos:

Ya con esto tenemos la **Matriz de X**, y procedemos a calcular su determinante:

$$M = \begin{bmatrix} 2 & 3 \\ 1 & -2 \end{bmatrix}$$

$$M_x = \begin{bmatrix} 20 & 3 \\ 3 & -2 \end{bmatrix}$$

$$M_x = \begin{bmatrix} 20 & 3 \\ 3 & -2 \end{bmatrix}$$

$$|M_x| = (20)(-2) - (3)(3)$$

$$|M_x| = -40 - 9 = -49$$

El determinante de la **Matriz X** es -49

Paso 3. Se prepara la matriz de la incógnita **y**, y se halla el determinante

La matriz de la **incógnita Y** es la misma matriz de coeficientes con una diferencia. En lugar de colocar los coeficientes de **Y**, se ubican los valores numéricos que quedaron al otro lado de las ecuaciones.

Veamos:

$$\begin{array}{l} 2x + 3y = 20 \quad \text{Ecuación 1} \\ x - 2y = 3 \quad \text{Ecuación 2} \end{array}$$

Matriz de los coeficientes.

$$M = \begin{bmatrix} 2 & 3 \\ 1 & -2 \end{bmatrix}$$

Diagram showing the coefficient matrix M. The element 3 in the first row, second column is crossed out with a large X. Arrows point from the labels 'x' and 'y' above to the first and second columns of the matrix respectively.

$$M_y = \begin{bmatrix} 2 & 20 \\ 1 & 3 \end{bmatrix}$$

Ya con esto tenemos la **Matriz de Y**, y procedemos a calcular su determinante:

$$M_y = \begin{bmatrix} 2 & 20 \\ 1 & 3 \end{bmatrix}$$

$$\begin{aligned} |M_y| &= (2)(3) - (20)(1) \\ |M_y| &= 6 - 20 = -14 \end{aligned}$$

El determinante de la **Matriz Y** es **-14**

Paso 4. Hallamos el valor de las incógnitas.

El **valor de Y** va a ser igual al determinante de la **matriz Y** dividido en el determinante de la **matriz de coeficientes**:

$$y = \frac{|M_y|}{|M|}$$

El **valor de X** va a ser igual al determinante de la **matriz X** dividido en el determinante de la **matriz de coeficientes**:

$$x = \frac{|M_x|}{|M|}$$

Resolvemos:

$$y = \frac{|M_y|}{|M|} = \frac{-14}{-7} = 2$$

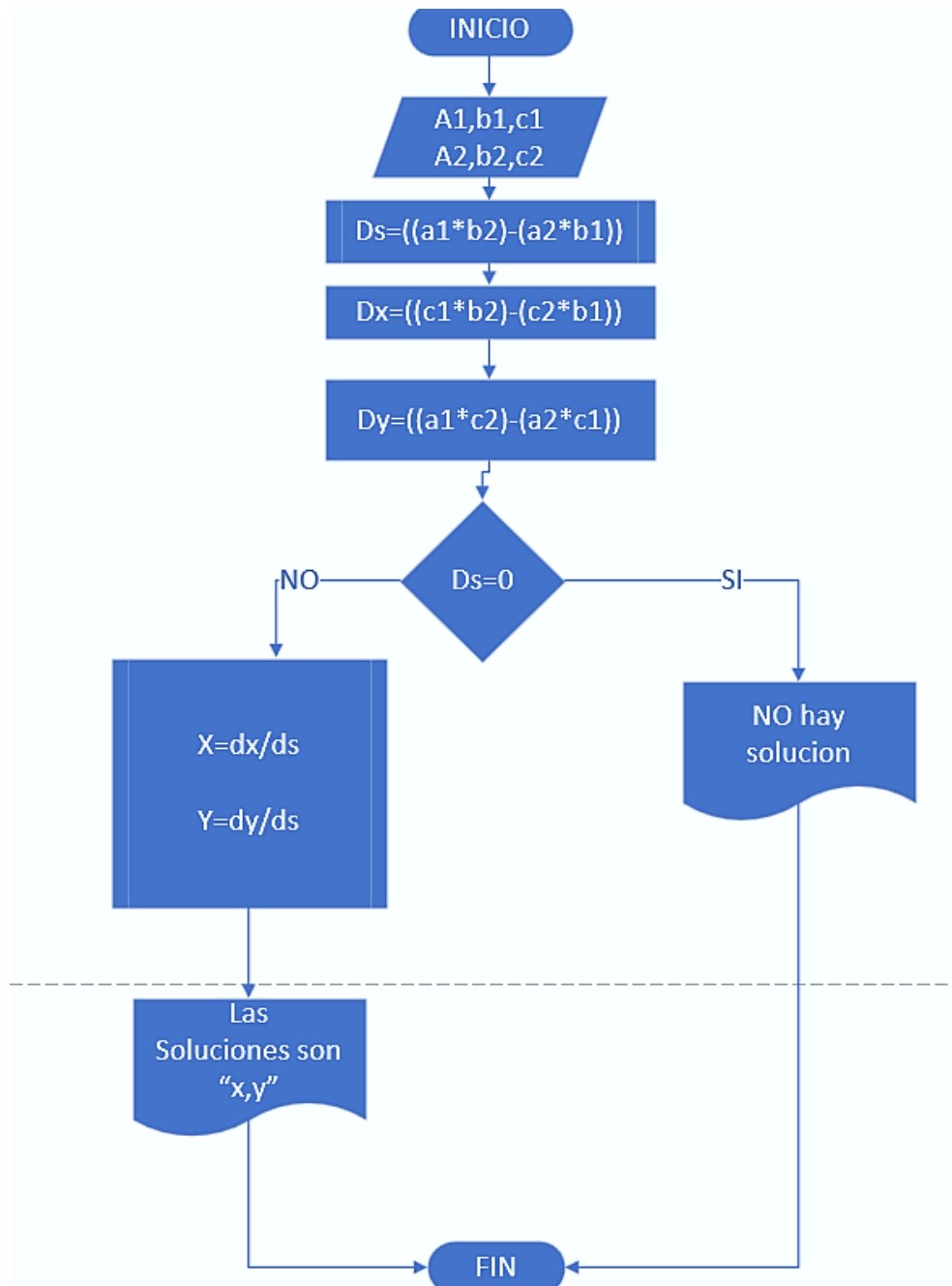
$$x = \frac{|M_x|}{|M|} = \frac{-49}{-7} = 7$$

Paso 5. Verificación de la solución del sistema.

Nuestra solución:

$$\begin{array}{l} y = 2 \\ x = 7 \end{array}$$

Diagrama de flujo



Listado del código

//4. Resolver un sistema de 2 ecuaciones y 2 incógnitas

```
#include<iostream>
```

```

#include<math.h>

#include<string.h>

using namespace std;

int main(){

    float a1,b1,c1,a2,b2,c2;

    float delta=0,dx=0,dy=0;

    float x=0,y=0;

    char op;

    do{

        cout<<"\nSI SE TIENE DOS ECUACIONES DE LA FORMA :"<<endl;

        cout<<"\t\tax + by = c"<<endl;

        cout<<"-----"<<endl;

        cout<<"\tPRIMERA ECUACION"<<endl;

        cout<<"-----"<<endl;

        cout<<"Escriba el coeficiente de < a > x:"<<endl;

        cin>>a1;

        cout<<"Escriba el coeficiente de < b > y:"<<endl;

        cin>>b1;

        cout<<"Escriba la constante < c >:"<<endl;

        cin>>c1;

        cout<<"-----"<<endl;

        cout<<"\tSEGUNDA ECUACION"<<endl;

        cout<<"-----"<<endl;

        cout<<"Escriba el coeficiente de < a > x:"<<endl;

        cin>>a2;

        cout<<"Escriba el coeficiente de < b > y:"<<endl;

        cin>>b2;

        cout<<"Escriba la constante < c >:"<<endl;

        cin>>c2;
    }
}

```

```

// USO DEL METODO DETERMINANTE
delta=((a1*b2)-(a2*b1));
dx=((c1*b2)-(c2*b1));
dy=((a1*c2)-(a2*c1));
if(delta!=0){
    x=dx/delta;
    y=dy/delta;
    cout<<"\nPor tanto, las soluciones del sistema son:"<<endl<<endl;
    cout<<"\ttx = "<<x<<endl;
    cout<<"\tty = "<<y<<endl;
}
if(delta==0){
    cout<<"\n\tNo es posible resolver ya que presenta una division entre
cero"<<endl;
}
cout<<"\nDESEA REALIZAR OTRA OPERACION (S/N) "<<endl;
cin>>op;
}while(op=='s' or op=='S');
cout<<"\nGRACIAS..."<<endl;

return 0;

}

```


Capturas de pantalla

```
EJERCICIO #1 (C++).cpp  EJERCICIO #4 (C++).cpp
1  //4. Resolver un sistema de 2 ecuaciones y 2 incógnitas
2
3  #include<iostream>
4  #include<math.h>
5  #include<string.h>
6  using namespace std;
7  int main(){
8      float a1,b1,c1,a2,b2,c2;
9      float delta=0,dx=0,dy=0;
10     float x=0,y=0;
11     char op;
12
13     do{
14         cout<<"\nSI SE TIENE DOS ECUACIONES DE LA FORMA :<<endl;
15         cout<<"\t\tax + by = c"<<endl;
16         cout<<"-----"<<endl;
17         cout<<"\tPRIMERA ECUACION"<<endl;
18         cout<<"-----"<<endl;
19         cout<<"Escriba el coeficiente de < a > x:"<<endl;
20         cin>>a1;
21         cout<<"Escriba el coeficiente de < b > y:"<<endl;
22         cin>>b1;
23         cout<<"Escriba la constante < c >:"<<endl;
24         cin>>c1;
25         cout<<"-----"<<endl;
26         cout<<"\tSEGUNDA ECUACION"<<endl;
27         cout<<"-----"<<endl;
28         cout<<"Escriba el coeficiente de < a > x:"<<endl;
29         cin>>a2;
30         cout<<"Escriba el coeficiente de < b > y:"<<endl;
31         cin>>b2;
32         cout<<"Escriba la constante < c >:"<<endl;
33         cin>>c2;
34
35         // USO DEL METODO DETERMINANTE
36         delta=((a1*b2)-(a2*b1));
37         dx=((c1*b2)-(c2*b1));
38         dy=((a1*c2)-(a2*c1));
```

```
C:\Users\HP\OneDrive\Escritorio\CRIS\...
SI SE TIENE DOS ECUACIONES DE LA FORMA :
      ax + by = c
-----
PRIMERA ECUACION
-----
Escriba el coeficiente de < a > x:
3
Escriba el coeficiente de < b > y:
4
Escriba la constante < c >:
15
-----
SEGUNDA ECUACION
-----
Escriba el coeficiente de < a > x:
2
Escriba el coeficiente de < b > y:
5
Escriba la constante < c >:
17
-----
Por tanto, las soluciones del sistema son:
      x = 1
      y = 3
-----
DESEA REALIZAR OTRA OPERACION (S/N)
```

```
EJERCICIO #1 (C++).cpp  EJERCICIO #4 (C++).cpp
22     cin>>b1;
23     cout<<"Escriba la constante < c >:"<<endl;
24     cin>>c1;
25     cout<<"-----"<<endl;
26     cout<<"\tSEGUNDA ECUACION"<<endl;
27     cout<<"-----"<<endl;
28     cout<<"Escriba el coeficiente de < a > x:"<<endl;
29     cin>>a2;
30     cout<<"Escriba el coeficiente de < b > y:"<<endl;
31     cin>>b2;
32     cout<<"Escriba la constante < c >:"<<endl;
33     cin>>c2;
34
35     // USO DEL METODO DETERMINANTE
36     delta=((a1*b2)-(a2*b1));
37     dx=((c1*b2)-(c2*b1));
38     dy=((a1*c2)-(a2*c1));
39     if(delta!=0){
40         x=dx/delta;
41         y=dy/delta;
42         cout<<"\nPor tanto, las soluciones del sistema son:"<<endl<<endl;
43         cout<<"\t\ttx = "<<x<<endl;
44         cout<<"\t\tty = "<<y<<endl;
45     }
46     if(delta==0){
47         cout<<"\n\tNo es posible resolver ya que presenta una division entre cero"<<endl;
48     }
49     cout<<"\nDESEA REALIZAR OTRA OPERACION (S/N) "<<endl;
50     cin>>op;
51 }while(op=='s' or op=='S');
52 cout<<"\nGRACIAS..."<<endl;
53
54
55 return 0;
56 }
```

```
C:\Users\HP\OneDrive\Escritorio\CRIS\...
SI SE TIENE DOS ECUACIONES DE LA FORMA :
      ax + by = c
-----
PRIMERA ECUACION
-----
Escriba el coeficiente de < a > x:
3
Escriba el coeficiente de < b > y:
4
Escriba la constante < c >:
15
-----
SEGUNDA ECUACION
-----
Escriba el coeficiente de < a > x:
2
Escriba el coeficiente de < b > y:
5
Escriba la constante < c >:
17
-----
Por tanto, las soluciones del sistema son:
      x = 1
      y = 3
-----
DESEA REALIZAR OTRA OPERACION (S/N)
```