

3.- Realiza el programa que encuentre el valor de la pendiente de la recta que pasa a través de dos puntos dados P1, P2, cuyas coordenadas son: (X_1, Y_1) y (X_2, Y_2) .

$$m = (Y_2 - Y_1) / (X_2 - X_1)$$

Ecuación 1: $a_1x + b_1y = c_1$

Ecuación 2: $a_2x + b_2y = c_2$

B.- PROGRAMACIÓN CONDICIONAL.

La instrucción IF-THEN-ELSE, que se utilizó en BASIC para establecer la selección de una de las dos opciones en el proceso de un programa, se puede usar en QB, pero si en el programa se presentan más de dos opciones se puede utilizar una nueva instrucción que sólo trabaja en QB: SELECT CASE.

SELECT CASE.- Permite establecer dos o más opciones en un mismo programa.

Formato: SELECT CASE expresión

CASE condición 1

CASE condición 2

CASE

CASE

CASE ELSE (opcional)

END SELECT

Ejemplos de formatos que se utilizan en las condiciones:

CASE IS < x cuando la condición es menor que x

CASE IS =15 cuando la condición es igual a 15

CASE 10 to 100 cuando la condición está en el rango de 10 a 100

CASE IS > =100 cuando la condición es mayor o igual que 100

CASE " A " cuando la condición es igual a la constante A

CASE 7 cuando la condición es igual a 7

CASE 3,9 cuando la condición es igual a 3 ó 9

Al igual que el IF-THEN-ELSE, el SELECT CASE acepta los mismos operadores relacionales y lógicos: =, >, <, >=, <=, AND, OR, NOT

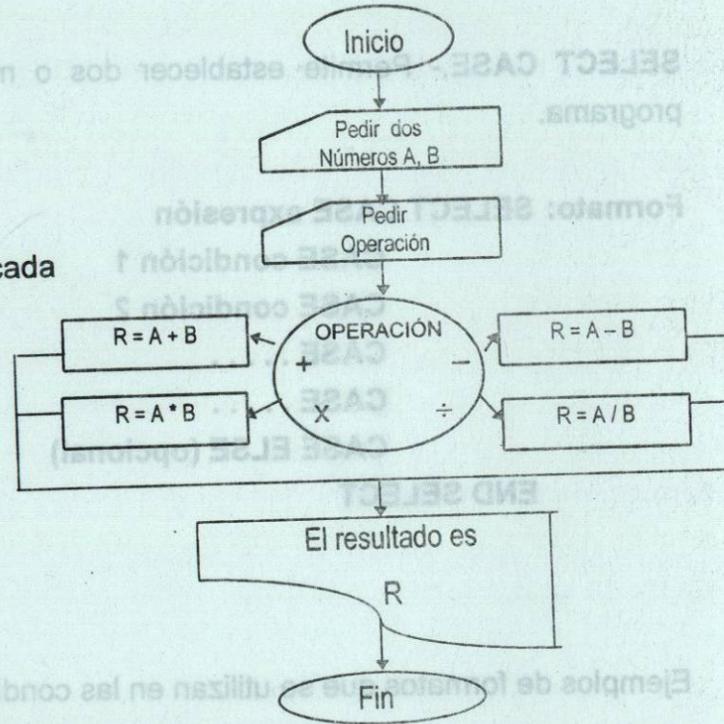
a.- Ejemplos

Ejemplo No.1.- Realiza el algoritmo, diagrama de flujo, codificación en QB de un programa que, con dos números cualesquiera, pueda efectuar una de las operaciones fundamentales de la aritmética.

ALGORITMO

- 1.- Inicio
- 2.- Pedir dos números
- 3.- Pedir la operación
- 4.- Seleccionar opción
- 5.- Efectuar operación indicada
- 6.- Imprimir resultado
- 7.- Fin

DIAGRAMA DE FLUJO



CODIFICACIÓN QB

```
CLS
INPUT "Dame el valor del primer número: ",A
INPUT "Dame el valor del segundo número (Diferente a cero): ",B
INPUT "Operación a ejecutar: +,-,*,/: ",OP$

SELECT CASE OP$
CASE "+"
    R=A+B
CASE "-"
    R=A-B
CASE "*"
    R=A*B
CASE "/"
    R=A/B
END SELECT

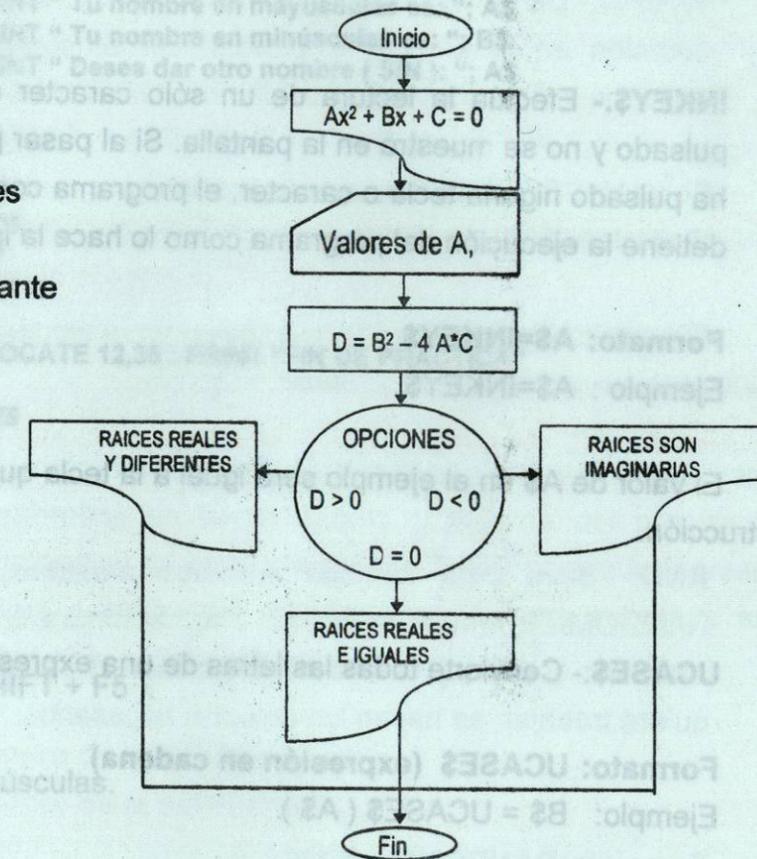
PRINT "La operación tiene como respuesta: ";R
END
```

Ejemplo No. 2.- Realiza el algoritmo, diagrama de flujo y codificación QB de un programa que determine si la ecuación de 2do. grado $Ax^2+Bx+C=0$ tiene raíces: reales y diferentes, reales e iguales o imaginarias.

ALGORITMO

- 1 Inicio
- 2.- Imprime ecuación cuadrática general
- 3.- Pregunta los valores de los coeficientes
- 4.- Evalúa el discriminante
- 5.- Selecciona opción
- 6.- Imprime resultado
- 7.- Fin

DIAGRAMA DE FLUJO



CODIFICACIÓN QB

```
CLS
LOCATE 3,10 : PRINT"La Ecuación a trabajar es Ax^2+Bx+C=0"
LOCATE 5,10 : INPUT"Dame los valores de A,B,C: ",A,B,C
D=B^2-4*A*C

SELECT CASE D
CASE IS = 0
    LOCATE 5,15 : PRINT"Las raíces son reales e iguales"
CASE IS > 0
    LOCATE 5,15 : PRINT"Las raíces son reales y diferentes"
CASE IS < 0
    LOCATE 5,15 : PRINT"Las raíces son imaginarias"
END SELECT

END
```

b.- Nuevas Instrucciones

A continuación veremos nuevas instrucciones que se pueden aplicar en QB, después de ver su definición, formato y ejemplo, realizaremos un programa para explicarlas mejor.

- INKEY\$.- Efectúa la lectura de un sólo carácter del teclado que se haya pulsado y no se muestra en la pantalla. Si al pasar por esta instrucción no se ha pulsado ninguna tecla o carácter, el programa continúa; esta instrucción no detiene la ejecución del programa como lo hace la instrucción INPUT.

Formato: A\$=INKEY\$

Ejemplo: A\$=INKEY\$

El valor de A\$ en el ejemplo será igual a la tecla que se pulse al llegar a esta instrucción.

- UCASE\$.- Convierte todas las letras de una expresión en mayúsculas.

Formato: UCASE\$ (expresión en cadena)

Ejemplo: B\$ = UCASE\$ (A\$)

En el ejemplo el contenido de B\$ será guardado con letras mayúsculas, independientemente si A\$ se encuentre con mayúsculas o minúsculas.

- LCASE\$.- Convierte todas las letras de una expresión en minúsculas.

Formato: LCASE\$ (expresión en cadena)

Ejemplo: B\$ = LCASE\$ (B\$)

En el ejemplo el contenido de B\$ será guardado con letras minúsculas, independientemente si B\$ se encuentre con mayúsculas o minúsculas.

TECLEA:

```
20 CLS
   LOCATE 4,10: INPUT " Dame tu nombre completo: ", A$
   A$ = UCASE$ ( A$ )
   B$ = LCASE$ ( A$ )
   LOCATE 10,15 : PRINT " Tu nombre en mayúsculas es: "; A$
   LOCATE 12,15 : PRINT " Tu nombre en minúsculas es: "; B$
   LOCATE 20,28 : PRINT " Desea dar otro nombre ( S/N ): "; A$
25 C$ = INKEY$
```

SELECT CASE C\$

CASE ""

GOTO 25

CASE "S", "s"

GOTO 20

CASE "N"

CLS : LOCATE 12,35 : PRINT "FIN DE PRÁCTICA"

CASE ELSE

GOTO 25

END SELECT

END

- Corre el programa (SHIFT + F5).

- Dale tu nombre en minúsculas.

- ¿Que sucedió? _____

- Daie tu nombre en mayúsculas.

- ¿Que sucedió? _____

- Guarda el programa en tu disco de trabajo.

NOTA: Como se observa el programa no está validado o sea que puede tener errores en la captura de información.

1020124173

b.- Nuevas Instrucciones

Ahora veremos otras instrucciones que aplicaremos en el programa que se encuentra después de la explicación.

- ▣ **SOUND.-** Da un sonido en una frecuencia desde 37 hasta 32767 Hertz y una duración en la que va a permanecer el sonido durante el tiempo que nosotros indiquemos, cada unidad de duración equivale a 1/18 de segundo.

Formato: SOUND frecuencia, duración

Ejemplos: SOUND 1000, 9

El sonido que producirá la instrucción en el ejemplo tendrá una frecuencia de 1000 Hertz y una duración de medio segundo.

- ▣ **RND.-** Sirve para semejar números aleatorios de 0 a 0.999999, con RANDOMIZE TIMER la función RND obtendrá un valor de semilla diferente cada vez que se ejecute, pero dependiendo de la escala de números que se quiera trabajar, se hacen los arreglos necesarios.

Formato: RANDOMIZE TIMER

RND

Ejemplo: RND genera un número de 0 a 0.999999
 100*RND genera un número de 0 a 99.9999
 1000*RND genera un número de 0 a 999.999

En los ejemplos anteriores es importante recalcar el uso del RANDOMIZE TIMER, porque si éste no se coloca al antes del RND, el valor será el mismo cuantas veces se genere.

- ▣ **INT.-** Sirve para eliminar la parte decimal del número, dejando sólo la parte entera.

Formato: INT(expresión numérica)

Ejemplos: INT(75.32) obtiene el número 75
 INT(0.54) obtiene el 0
 INT(1000*RND) obtiene un número de 0 a 999
 INT(100*RND)+1 obtiene un número de 1 a 100

En el último ejemplo se puede observar que se tuvo que hacer un ajuste aritmético para obtener un rango apropiado de trabajo.

- ▣ **GOSUB.-** Se utiliza en programas cuando se tienen operaciones repetitivas; se puede crear una subrutina en forma interna o externa del programa principal. Al final de la subrutina se concluye con RETURN para que regrese a la línea del programa después del GOSUB; cuando una subrutina es grabada como externa es necesario que sea en modo texto.

Formato: GOSUB número de subrutina

GOSUB nombre de la subrutina

Ejemplo : GOSUB 5000

GOSUB SONIDO

Como se observa en los dos ejemplos el nombre de la subrutina puede ser un número, por lo regular es una cantidad grande para no interferir con algunas etiquetas de un programa, o un nombre que vaya de acuerdo a la operación de la subrutina.

TECLEA:

```

20  CLS
    RANDOMIZE TIMER
    LOCATE 4,10: INPUT " Dame tu nombre: ", A$
    A$ = UCASE$( A$ )
    A = INT ( 10 * RND ) + 1
    LOCATE 6, 10: INPUT " Adivina un número del 1 al 10: ", B

    SELECT CASE A
      CASE B
        GOSUB 50
        LOCATE 12,28 : PRINT " Felicidades lo adivinaste "
      CASE ELSE
        SOUND 55,36
        LOCATE 12,28 : PRINT " Lo siento no es el número "
    END SELECT

    LOCATE 14,28 : PRINT " El número que pensé es: " ; A
    LOCATE 16,28 : PRINT " El número de " ; A$ ; " es: " ; B
25  LOCATE 22,25 : PRINT SPC (50);
    LOCATE 22,25 : PRINT INPUT " Quieres probar otra vez (S/N): " , B$

    SELECT CASE UCASE$(B$)
      CASE "S"
        GOTO 20
      CASE "N"
        CLS: LOCATE 12,35 : PRINT " FIN DE JUEGO "
      CASE ELSE
        SOUND 55,36 : GOTO 25
    END SELECT

    END

50  SOUND 300,3 : SOUND 600,2 : SOUND 900,1
    SOUND 300,1 : SOUND 600,2 : SOUND 900,3
    SOUND 300,3 : SOUND 600,2 : SOUND 900,1

```

☞ Corre el programa (SHIFT + F5).

☞ ¿Que sucedió? _____

☞ Guarda el programa en tu disco de trabajo.

NOTA: Como se observa el programa está validado o sea que puede detectar errores en la captura de información.

c.- Ejercicios

Desarrolla el algoritmo, el diagrama de flujo y la codificación QB para los siguientes programas.

1.- Realiza un programa que determine los valores de las raíces de una ecuación cuadrática $Ax^2+Bx+C=0$ y diga la naturaleza de las raíces.

2.- Sergio

3.- Ricardo

4.- Sigifredo

5.- Tomás

6.- Ulises

7.- Verónica

8.- Yazmin