



FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA

CARRERA: LICENCIATURA EN CIENCIAS DE LA COMPUTACION

MATERIA: ALGORITMOS I

TRABAJO: PORTAFOLIO DE RECOPIACION DE ALGORITMOS

DOCENTE ENCARGAD@: ANA LISSETTE GIRON DE BERMUDEZ

PRESENTAN:

ISAI MEJIA MENDEZ

DAVID ANSELMO LEMUS TURCIOS

ROSA ELIZABETH VENTURA VIGIL

DENIS EMERSON ZAMORA MARTINEZ

INDICE

Concepto	Nº Pagina.
Introducción.....	i
Objetivos.....	1
Definición de algoritmos.....	2
Estructura secuencial.....	6
Estructuras condicionales:	
• ->Estructura Condicional Simples.....	11
• ->Estructura Condicional Compuestas.....	16
• ->Estructura Condicional Anidadas.....	21
Estructuras Repetitivas (for).....	28
Estructuras Iterativas (While, Repeat).....	35
Estructuras De Datos Arreglos Unidimensionales.....	42

INTRODUCCION

El presente trabajo de investigación pretende dar a conocer la diferente estructura **de control** que permiten abordar las soluciones de problemas que requieran **tratamiento lógico**.

En el primer punto veremos cómo desarrollar ejercicios con estructuras secuenciales, y aspectos relevantes de esta estructura.

El siguiente punto será el desarrollo de las estructuras condicionales, en la cual veremos las estructuras simples, compuestas y anidadas, en las cuales desarrollaremos ejercicios para ver la diferencia que tienen una de la otra entre ellas como también la del primer punto antes expuesto.

Seguidamente veremos la estructura repetitiva (for), en la cual tocaremos en q casos se deben utilizar.

Veremos la estructura iterativa con las con las condiciones (while, repeat), que están junto a las repetitivas pero cambian cuando decidimos en que caso las utilizaremos tanto como el concepto que abarcan cada una de ellas.

Y como punto final, estaremos exponiendo los arreglos unidimensionales los cuales son más complejos tanto en la ejecución como en la elaboración de los mismos porque se trabajan con vectores y reserva de memoria.

Promoviéndolas como herramientas de programación informática de fácil comprensión para el uso de los/las estudiantes.


Dentro del proceso de programación encontraremos el programa (PseInt) este nos sirve para hacernos más factible el trabajo, el cual nos permite tener una facilidad de uso de programación.

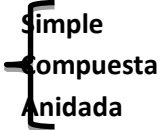
OBJETIVOS

El objetivo de este trabajo es presentar las diferentes estructuras de control de datos que se han visto a lo largo del ciclo; y como consecuencia la capacidad adquirida para escribir nuestros primeros algoritmos y programar el código ayudados de PsInt que ha sido una herramienta fundamental en el desarrollo de esta materia.

En concreto estas son las sentencias que pretendemos exponerles a los demás:


 E. Secuencial

 E. Condicional



Simple
Compuesta
Anidada

 E. Repetitiva (for)

 E. Iterativa (while, repeat)

 E. Arreglos.

Presentación de algoritmos

Los algoritmos son un conjunto de instrucciones y pasos lógicos que la computadora debe seguir para resolver un problema.

Los algoritmos se centran en el estudio de técnicas que se aplican a problemas que pueden ser solucionados por medio de un tratamiento lógico y algorítmico.

Que entre sus características se destacan por la necesidad de un análisis profundo y sistemático, como también de un pensamiento flexible y estructurado para alcanzar una solución idónea al problema a tratar.

En esta etapa se estudia un conjunto de técnicas y herramientas metodológicas que permiten al estudiante alcanzar un alto nivel de razonamiento en la lógica antes descrita y estructurarlo para que desemboque en su capacidad para la solución de dicho problemas.

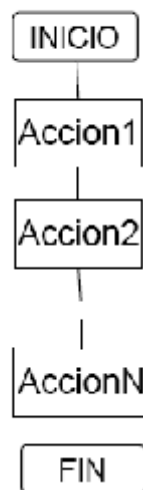
Los algoritmos los conocemos como tales gracias al Matemático y Astrónomo Persa-Árabe Muhammad Musa Al-Khwarizmi; del siglo IX, el cual escribió el tratado de las ecuaciones y números.

El desarrollo del presente documento lleva consigo una serie de etapas que describen de forma sistemática y lógica los pasos a ejecutar dentro de la planificación y solución de los problemas que hemos planteados con diferentes estructuras de datos.

✚ a) *Estructura Secuencial*

La estructura secuencial es aquella en la que una acción (instrucción) sigue a otra en secuencia. Las tareas se suceden de tal modo que la salida de una es la entrada de la siguiente y así sucesivamente hasta el fin del proceso.

En Diagrama de flujo se realiza así:



➤ Ejemplos de solución de algoritmos con Estructura Secuencial

1- Elaborar un algoritmo que calcule el valor de “x” en una ecuación cuadrática.

The image shows the PSeInt software interface. The main window displays a C++ program for solving a quadratic equation. The code includes input prompts for coefficients a, b, and c, followed by the quadratic formula calculation and output of the roots. A conditional block checks the sign of the coefficients to format the equation display. An execution window is overlaid on the code, showing the program's output for a=65, b=78, and c=-167.

```

1 Proceso Ecuacion_Cuadratica
2   Escribir "Ingrese el valor de a: ";
3   Leer a;
4   Escribir "Ingrese el valor de b: ";
5   Leer b;
6   Escribir "Ingrese el valor de c: ";
7   Leer c;
8   Escribir "Solucion segun ecuacion cuadratica";
9   x1<-(-b+RC((b*b)-(4*a*c)))/(2*a);
10  x2<-(-b-RC((b*b)-(4*a*c)))/(2*a);
11  Escribir "Su ecuacion es: ";
12
13  //desde aqui codigo para controlar el signo del numero, y mostrarlo en pantalla
14  signo1<-" ";
15  signo2<-" ";
16  signo3<-" ";
17  Si a>0 Entonces
18  ... signo1<"+ ";
19  FinSi
20  Si b>0 Entonces
21  ... signo2<"+ ";
22  FinSi
23  Si c>0 Entonces
24  ... signo3<"+ ";
25  FinSi
26  // fin codigo.
27
28  Escribir signo1,a,"X^2",signo2,b,"X",signo3,c,"=0";
29  Escribir "El valor de x1 es: ", x1;
30  Escribir "El valor de x2 es: ", x2;
31 FinProceso
32

```

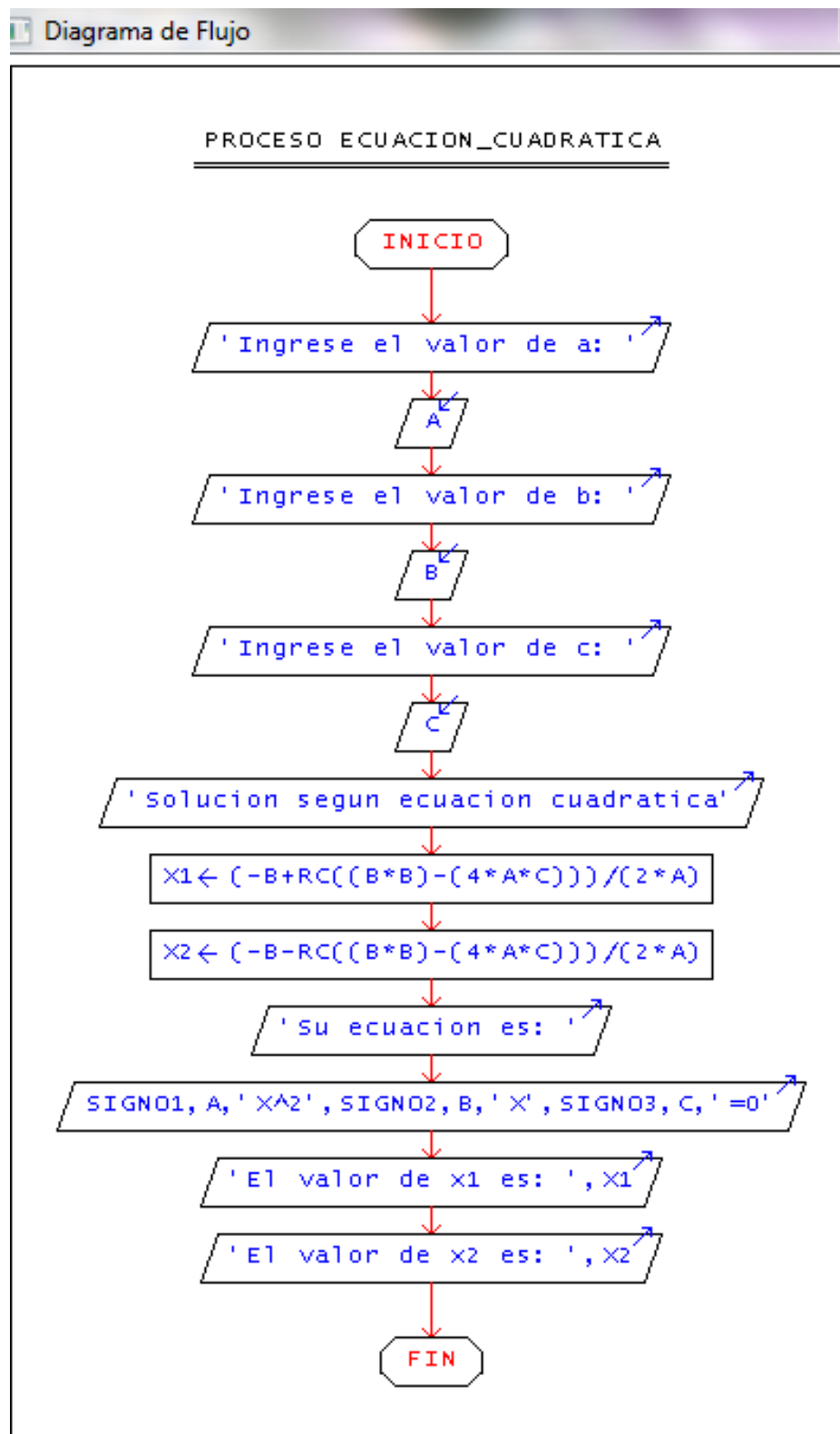
Execution Output:

```

*** Ejecucion Iniciada. ***
Ingrese el valor de a:
> 65
Ingrese el valor de b:
> 78
Ingrese el valor de c:
> -167
Solucion segun ecuacion cuadratica
Su ecuacion es:
+65X^2+78X-167=0
El valor de x1 es: 1.1114995674
El valor de x2 es: -2.3114995674
*** Ejecucion Finalizada. ***

```

En Diagrama de flujo



2- Elaborar un algoritmo que calcule la media aritmética de 5 números introducidos desde el teclado.

The screenshot displays the Turbo Pascal IDE with the following code in the editor:

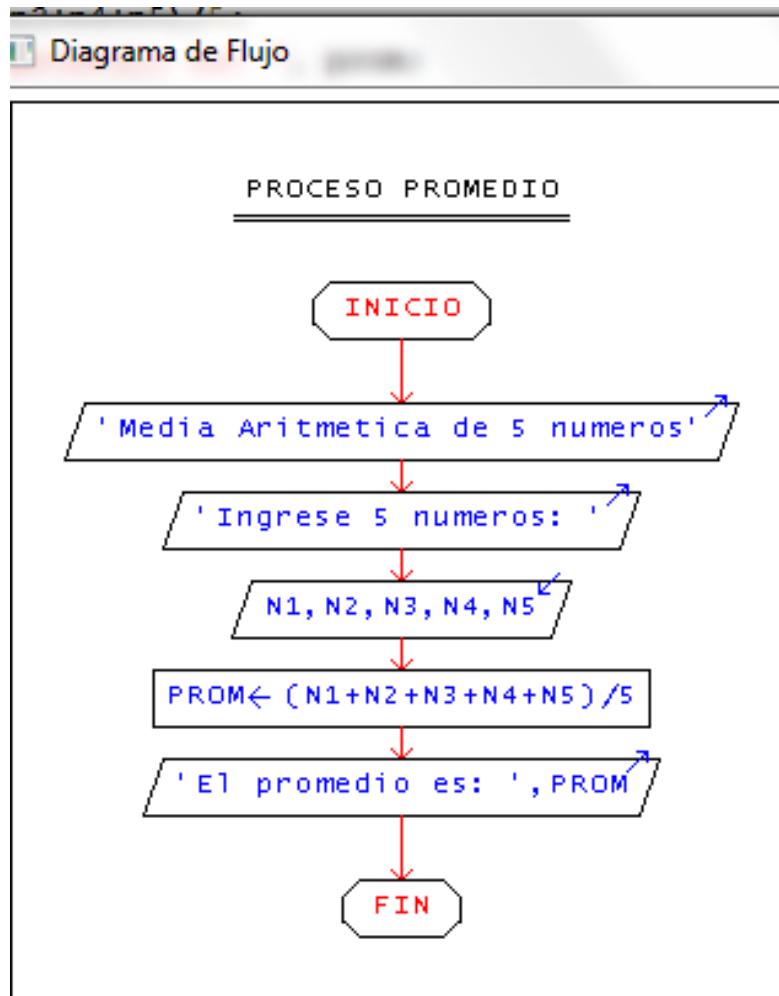
```
1 Proceso promedio
2   Escribir "Media Aritmetica de 5 numeros";
3   Escribir "Ingrese 5 numeros: ";
4   Leer n1,n2,n3,n4,n5;
5   prom<-(n1+n2+n3+n4+n5)/5;
6   Escribir "El promedio es: ", prom;
7 FinProceso
8
```

The console window shows the execution output:

```
*** Ejecucion Iniciada. ***
Media Aritmetica de 5 numeros
Ingrese 5 numeros:
65
78
34
47
0.0001
El promedio es: 43.20002
*** Ejecucion Finalizada. ***
```

On the right side of the IDE, a 'Comandos' (Commands) panel is visible, containing various control icons such as 'Escribir', 'Leer', 'Asignar', 'Si-Entonces', 'Segun', 'Mientras', 'Repetir', and 'Para'.

En diagrama de flujo



 b) Estructura Condicional

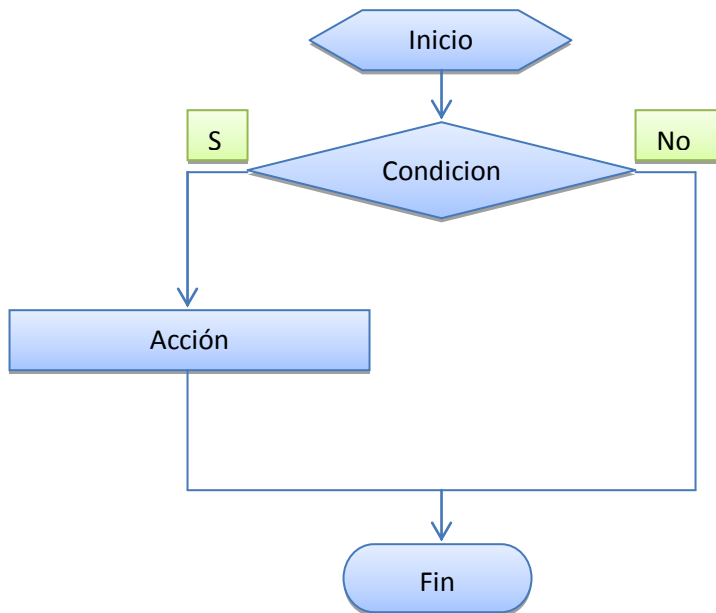
- *Estructura Condicional Simple:*

En estas estructuras existe un punto en el cual es necesario establecer una pregunta para decidir si ciertas acciones deben realizarse o no.

Se trata de una estructura CONDICIONAL SIMPLE porque por el camino del verdadero hay actividades y por el camino del falso no hay actividades.

Por el camino del verdadero pueden existir varias operaciones, entradas y salidas.

En diagrama de flujo se representa así:



Ejemplos de solución de algoritmos con estructura condicional simple

- 1- Elaborar un algoritmo que capture la edad de una persona y que imprima si este es mayor de edad.

The screenshot displays the PSeInt software interface. The main window shows a pseudocode algorithm for calculating age and checking if it's greater than 18. The code is as follows:

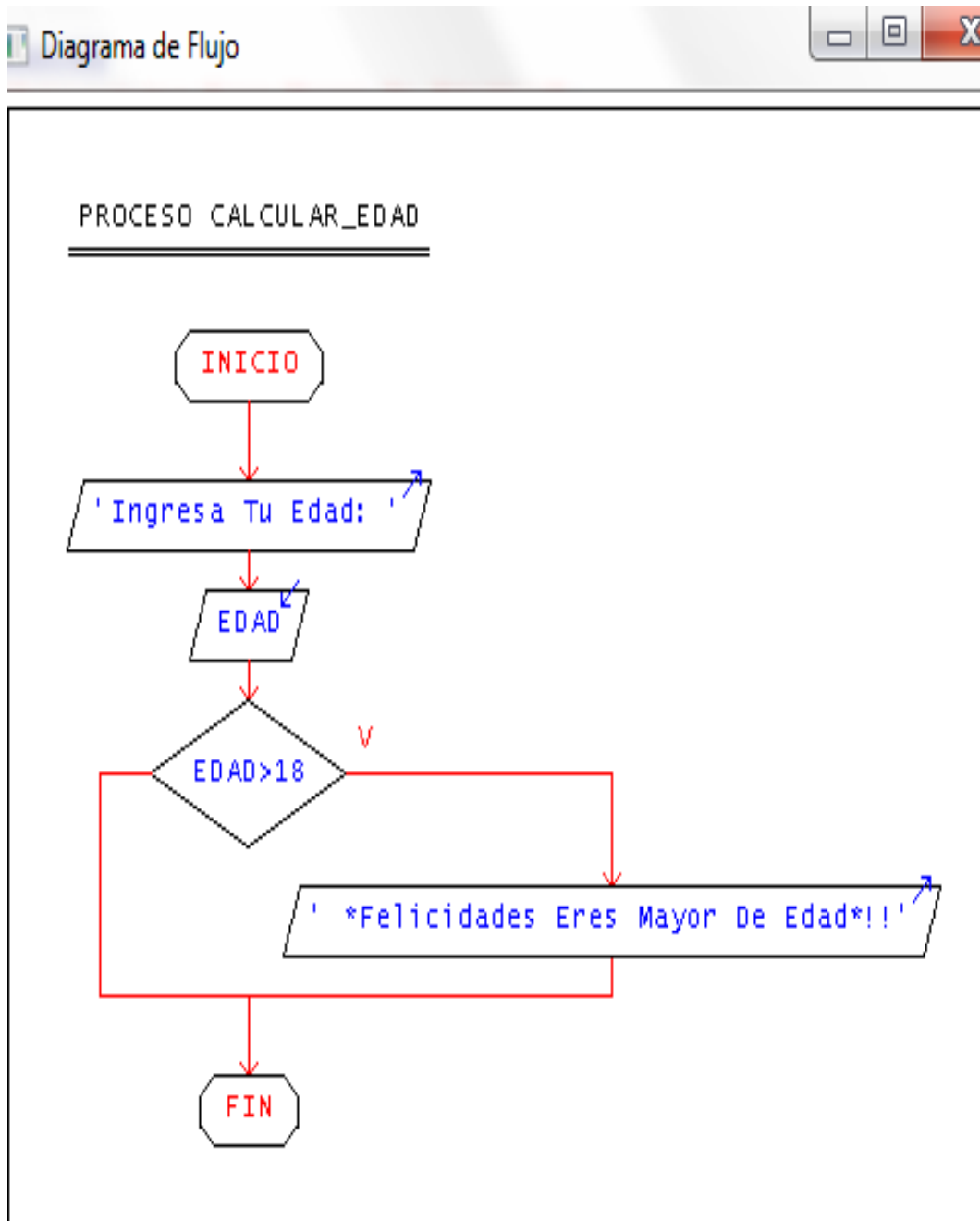
```
1 Proceso Calcular_Edad
2
3   Escribir "Ingresa Tu Edad: ";
4   Leer edad;
5
6   Si edad>18 Entonces
7     Escribir " *Felicidades Eres Mayor De Edad*!!";
8
9   FinSi
10
11 FinProceso
12
```

An execution window titled "CAProgram Files\PSeInt\pseint.exe" is overlaid on the code, showing the following output:

```
*** Ejecucion Iniciada. ***
Ingresa Tu Edad:
> 23
 *Felicidades Eres Mayor De Edad*!!
*** Ejecucion Finalizada. ***
```

On the right side of the PSeInt window, there is a "Comandos" (Commands) panel with various icons for operations like "Escribir", "Leer", "Asignar", "Si-Entonces", "Segun", "Mientras", "Repetir", and "Para".

En diagrama de flujo



2- Elaborar un algoritmo que sea capaz de convertir un numero negativo a positivo

The image shows the PSeInt software interface with a flowchart and its execution. The flowchart is a simple linear process:

```

1 Proceso conversion
2   Escribir "Conversion De Numero Negativo a Positivo";
3   Escribir "Ingrese Un Numero Negativo: ";
4
5   Leer n;
6   Si n<0 Entonces
7     positivo<-n*-1;
8
9     Escribir "El Positivo De ", n, " Es: ", positivo;
10
11  FinSi
12 FinProceso
13

```

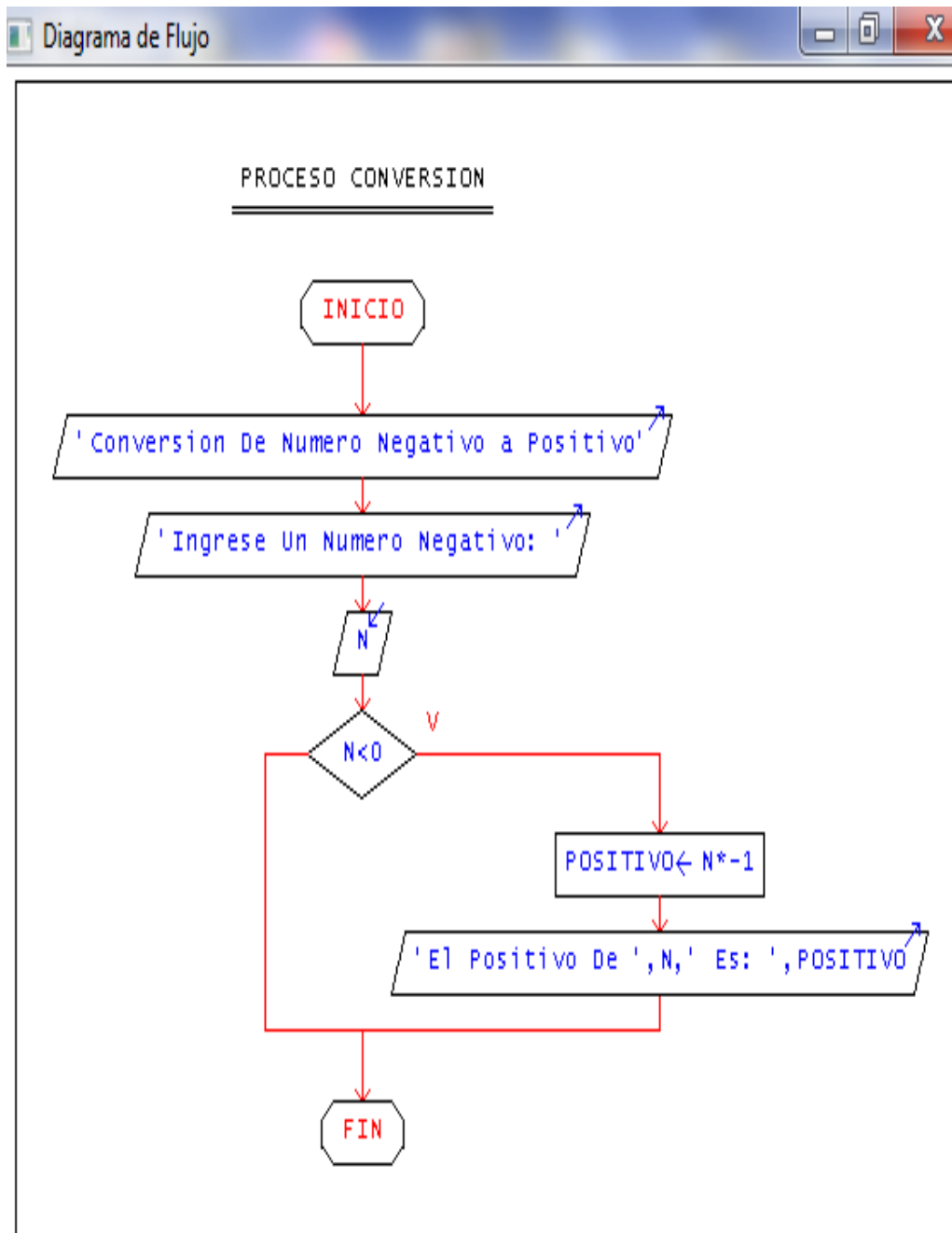
The execution window shows the following output:

```

*** Ejecucion Iniciada. ***
Conversion De Numero Negativo a Positivo
Ingrese Un Numero Negativo:
> -342
El Positivo De -342 Es: 342
*** Ejecucion Finalizada. ***

```

En diagrama de Flujo

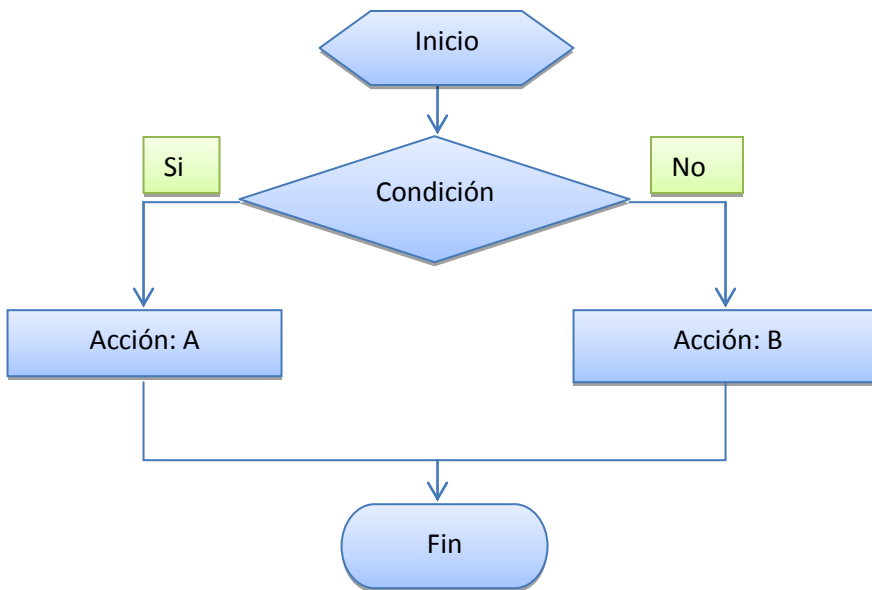


- *Estructura condicional compuesta*

Son estructuras lógicas que permiten controlar la ejecución de varias acciones y se utilizan cuando se tienen dos opciones de acción en la condición.

Es decir tenemos actividades por el verdadero y por el falso de la condición. Lo más importante que hay que tener en cuenta que se realizan las actividades de la rama del verdadero o las del falso, NUNCA se realizan las actividades de las dos ramas.

Su estructura es la siguiente:



Ejemplos de solución de algoritmos con Estructuras Condicional Compuestas

- 1- Elaborar un algoritmo que solicite un número y si este se encuentra en el rango del 1 al 10 e imprima un mensaje "Numero Valido" de lo contrario que imprima un mensaje "Numero Invalido"

The image shows a screenshot of the PSaint IDE. The main window displays a Pascal program with the following code:

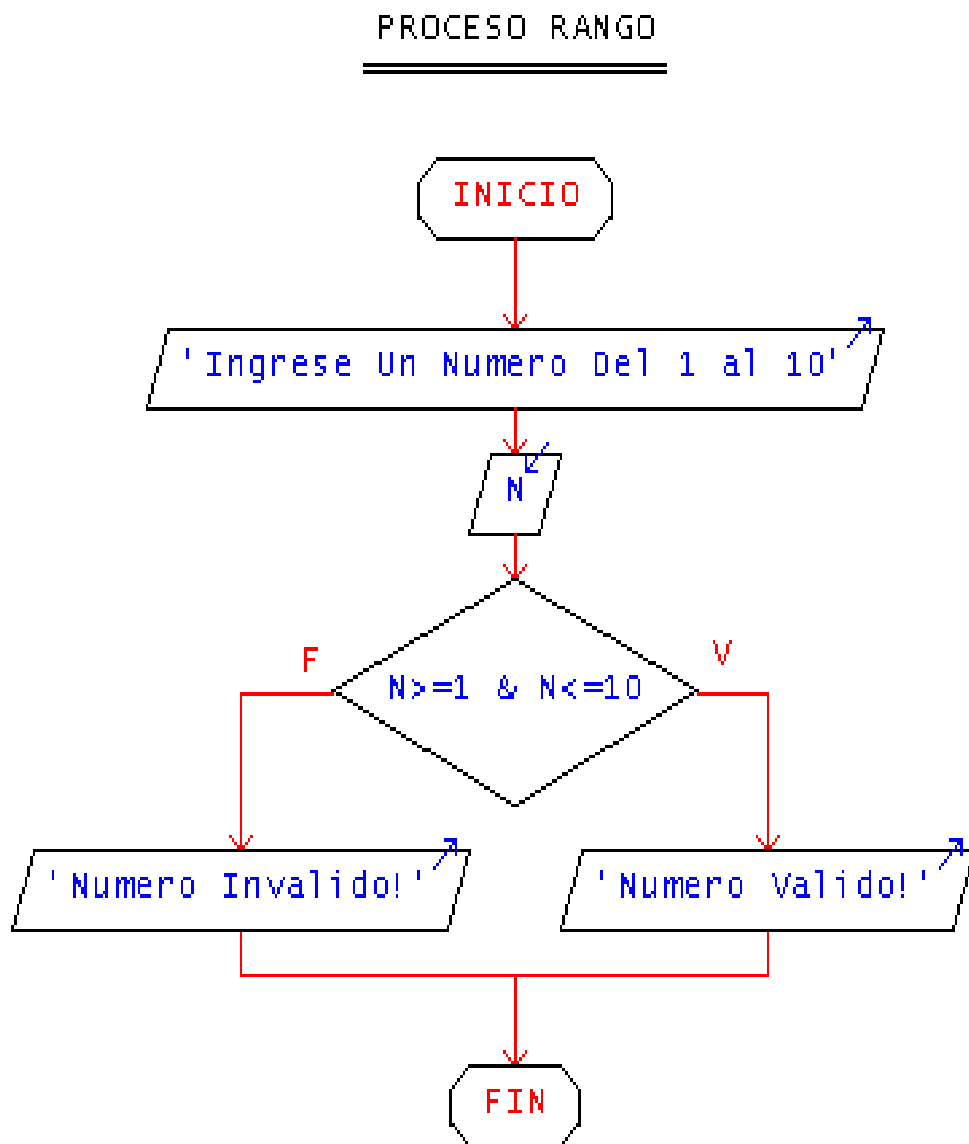
```
1 Proceso Rango
2
3   Escribir "Ingrese Un Numero Del 1 al 10";
4   Leer n;
5
6   Si n>=1 y n<=10 Entonces
7     Escribir "Numero Valido!";
8   Sino
9     Escribir "Numero Invalido!";
10
11 FinSi
12
13 FinProceso
14
```

An execution window titled "C:\Program Files\PSaint\psaint.exe" is overlaid on the code, showing the following output:

```
*** Ejecucion Iniciada. ***
Ingrese Un Numero Del 1 al 10
> 10
Numero Valido!
*** Ejecucion Finalizada. ***
```

The IDE interface includes a menu bar (Archivo, Editar, Configurar, Ejecutar, Ayuda), a toolbar, and a right-hand sidebar with a "Comandos" panel containing various control flow icons such as "Escribir", "Leer", "Asignar", "Si-Entonces", "Segun", "Mientras", "Repetir", and "Para".

En Diagrama de Flujo



2- Elaborar un algoritmos que solicite 3 números de una ecuación cuadrática, y que imprima su ecuación y el valor de x_1 y x_2

The image shows the PSeInt software interface with a flowchart on the right and a code editor on the left. The code editor contains the following pseudocode:

```

1 Proceso Cuadratica
2   Escribir "Ingrese el Valor de a: ";
3   Leer a;
4   Escribir "Ingrese el Valor de b: ";
5   Leer b;
6   Escribir "Ingrese el Valor de c: ";
7   Leer c;
8   Si a<=0 Entonces
9     Escribir " Error No Se Puede Continuar Con La Solucion Introduzca Otro Valor**";
10  Sino
11    Escribir "Solucion Segun Ecuacion Cuadratica";
12     $x1 <- (-b + \sqrt{(b*b) - (4*a*c)}) / (2*a)$ ;
13     $x2 <- (-b - \sqrt{(b*b) - (4*a*c)}) / (2*a)$ ;
14    Escribir "Su Ecuacion Es: ";
15
16    //desde aqui codigo para controlar el signo del numero, y mostrarlo en pantalla
17    signo1<="-";
18    signo2<="-";
19    signo3<="-";
20    Si a>0 Entonces
21      signo1<="+";
22    FinSi
23    Si b>0 Entonces
24      signo2<="+";
25    FinSi
26    Si c>0 Entonces
27      signo3<="+";
28    FinSi
29    // fin codigo.
30    Escribir signo1,a,"X^2",signo2,b,"X",signo3,c,"=0";
31    Escribir "El Valor De x1 Es: ", x1;
32    Escribir "El Valor De x2 Es: ", x2;
33    FinSi
34 FinProceso

```

The execution window shows the following output:

```

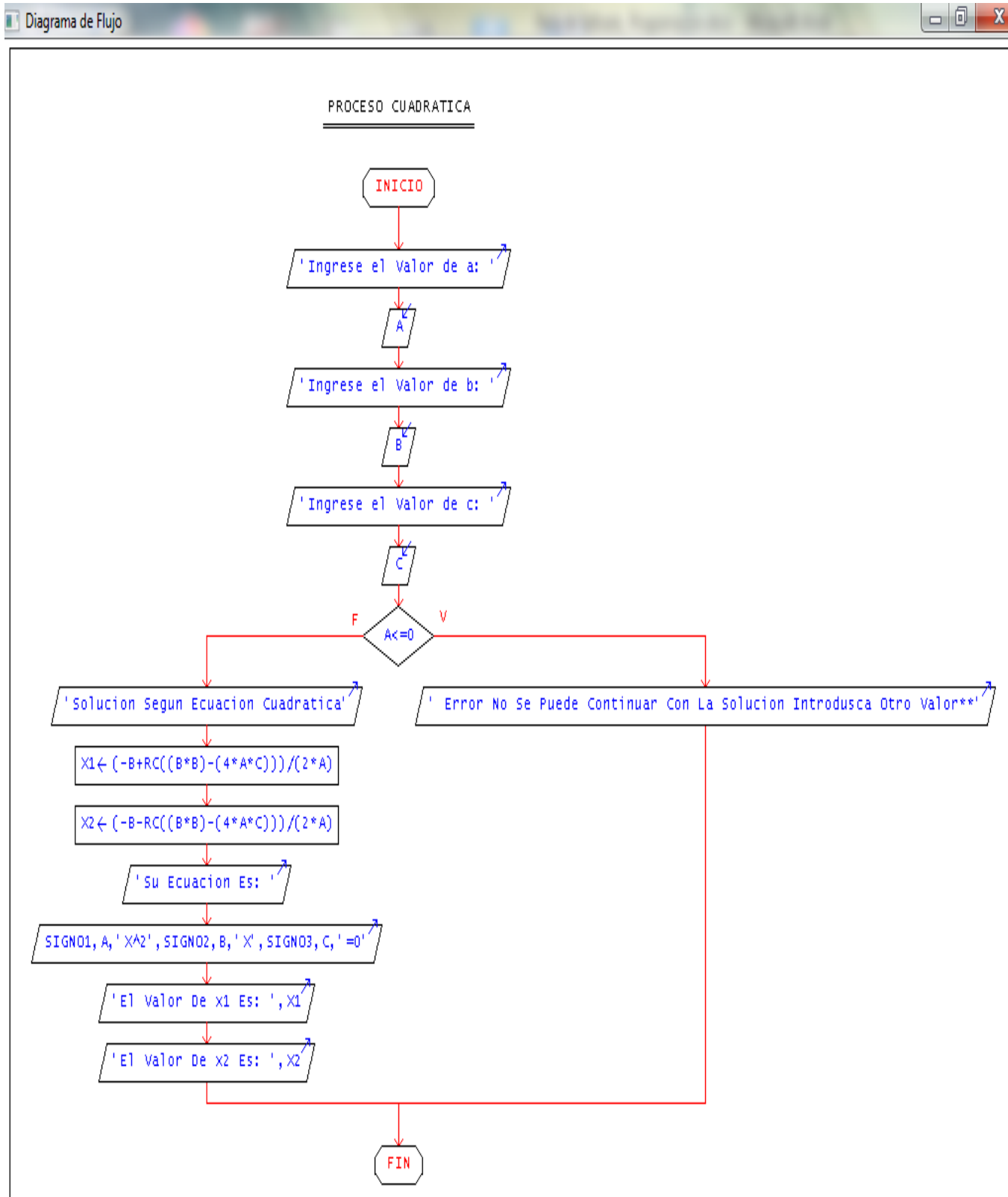
*** Ejecucion Iniciada. ***
Ingrese el Valor de a:
> 12
Ingrese el Valor de b:
> 7
Ingrese el Valor de c:
> -49
Solucion Segun Ecuacion Cuadratica
Su Ecuacion Es:
+12X^2+7X-49=0
El Valor De x1 Es: 1.75
El Valor De x2 Es: -2.333333333333
*** Ejecucion Finalizada. ***

```

The flowchart on the right side of the PSeInt interface includes the following steps:

- Inicio
- Comandos: Escribir
- Dato1: Leer
- Comandos: Asignar
- Si-Entonces
- Comandos: Segun
- Mientras
- Repetir
- Para
- Fin

En Diagrama De Flujo

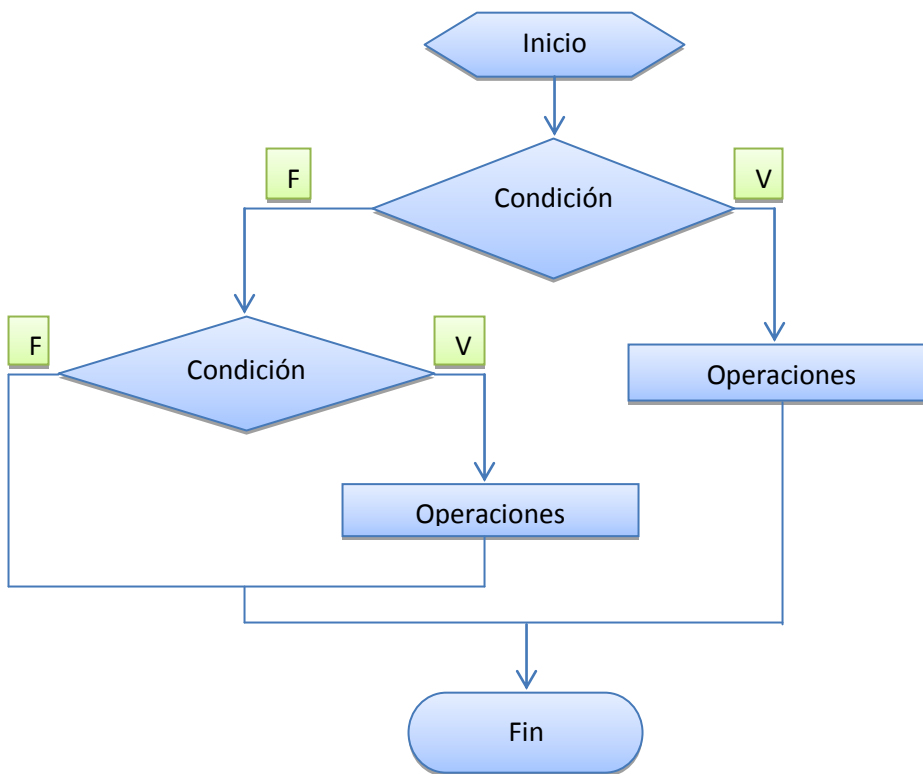


- Estructura Condicional Anidadas

Decimos que una estructura condicional es anidada cuando por la rama del verdadero o el falso de una estructura condicional hay otra estructura condicional.

El diagrama de flujo que se presenta contiene dos estructuras condicionales. La principal se trata de una estructura condicional compuesta y la segunda es una estructura condicional simple y está contenida por la rama del falso de la primera estructura.

Su estructura es la siguiente



Ejemplos de algoritmos con Estructuras Condicional Anidadas

- 1- Elaborar un algoritmo que sea capaz de hacer las cuatro operaciones, suma, resta, división y multiplicación y que imprima la operación que se le indique desde el teclado.

The screenshot shows the PSeInt software interface with a flowchart algorithm for a calculator. The algorithm is as follows:

```

1 Proceso Tabla
2   Escribir "Elija La Opcion Que Quiera";
3   Escribir "-----";
4   Escribir "1=Suma";
5   Escribir "2=Resta";
6   Escribir "3=Multiplicacion";
7   Escribir "4=Division";
8   Escribir "-----";
9   Leer a;
10  Escribir "-----";
11  Escribir "Ingrese Dos Numeros";
12  Leer b;
13  Mientras b<0 Hacer
14    Escribir "Error, Ingrese numero valido";
15    Leer b;
16  FinMientras
17  Leer c;
18  Mientras c<0 Hacer
19    Escribir "Error, Ingrese numero valido";
20    Leer c;
21  FinMientras
22  Si a=1 Entonces
23    Escribir "Suma";
24    d<-b+c;
25  Sino
26    Si a=2 Entonces
27      Escribir "Resta";
28      d<-b-c;
29    Sino
30      Si a=3 Entonces
31        Escribir "Multiplicacion";
32        d<-b*c;
33    Sino

```

The right sidebar of the PSeInt window contains a 'Comandos' (Commands) panel with the following icons and labels:

- Hola! Escribir
- Dato1 Leer
- A+B+ Asignar
- Si-Entonces
- Segun
- Mientras
- Repetir

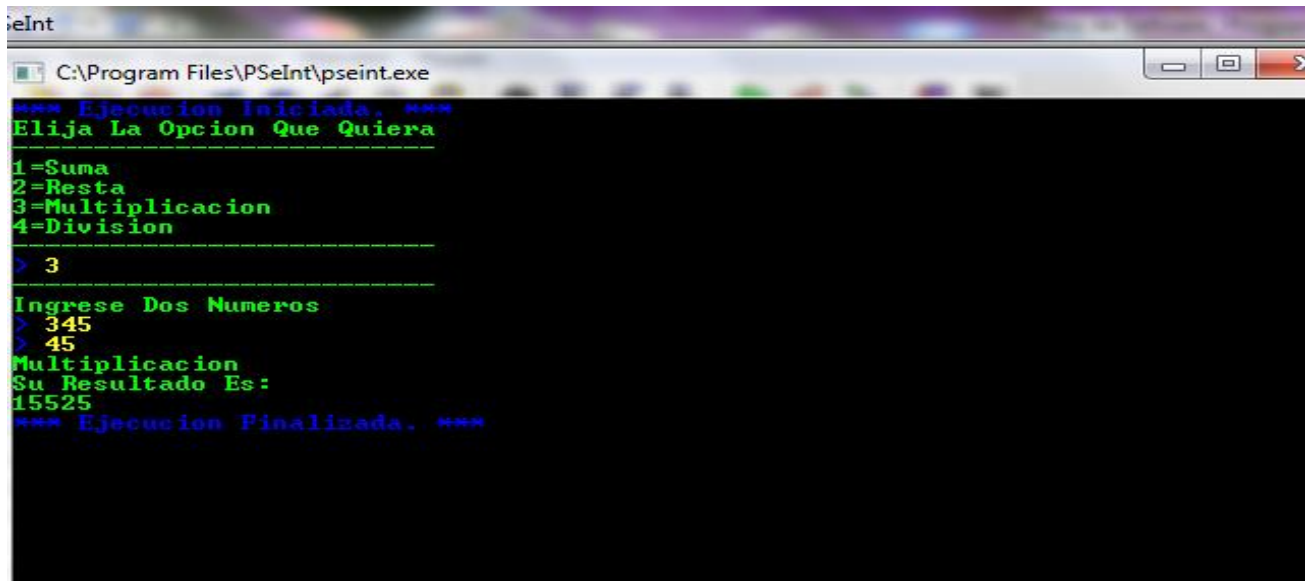




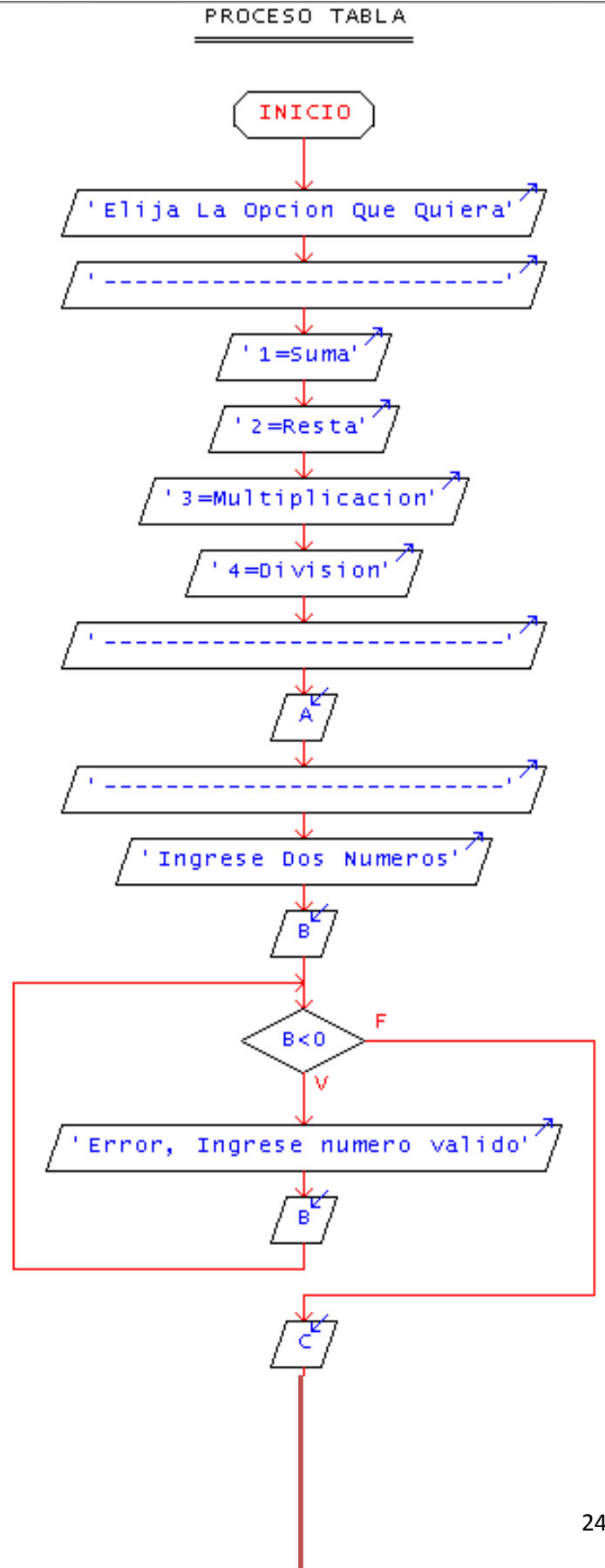
```

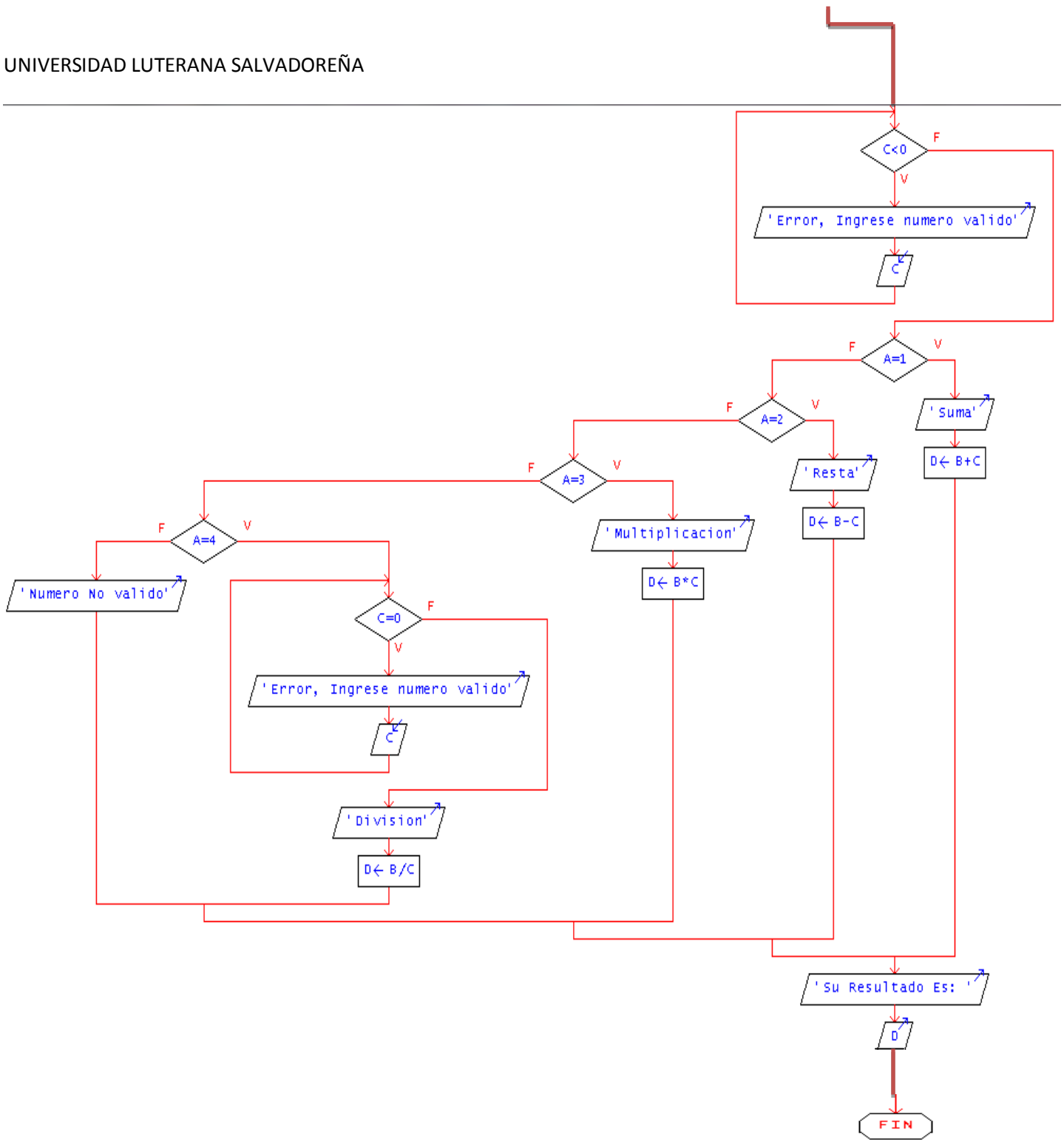
34      Si a=4 Entonces
35          Mientras c=0 Hacer
36              Escribir "Error, Ingrese numero valido";
37              Leer c;
38              FinMientras
39              Escribir "Division";
40              d<-b/c;
41          Sino
42              Escribir "Numero No valido";
43          FinSi
44      FinSi
45  FinSi
46  FinSi
47  Escribir "Su Resultado Es: " ;
48  Escribir d;
49
50 FinProceso
51

```



En diagrama de flujo





2- Elaborar un algoritmo que capture 3 notas y evalúe el promedio. Deberá imprimir en que rango se encuentra el promedio de acuerdo a lo siguiente:

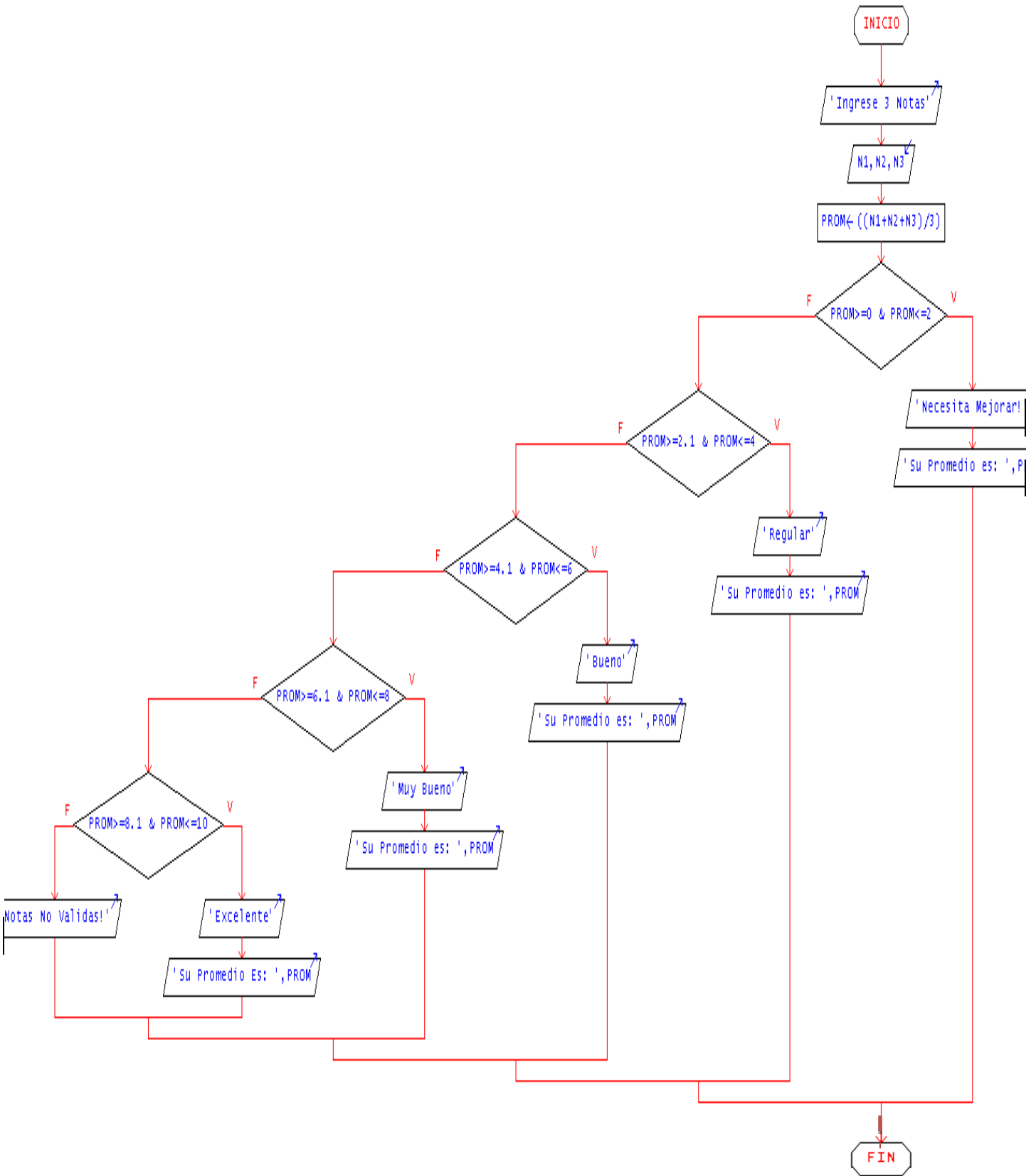
- 0 a 2 Necesita mejorar.
- 2.1 a 4 Regular.
- 4.1 a 6 Bueno.
- 6.1 a 8 Muy Bueno.
- 8.1 a 10 Excelente

```

1 Proceso Promedio
2   Escribir "Ingrese 3 Notas";
3   Leer n1,n2,n3;
4   prom<-((n1+n2+n3)/3);
5   Si prom>=0 y prom<=2 Entonces
6     Escribir "Necesita Mejorar!";
7     Escribir "Su Promedio es: ", prom;
8   Sino
9     Si prom>=2.1 y prom<=4 Entonces
10      Escribir "Regular";
11      Escribir "Su Promedio es: ", prom;
12    Sino
13      Si prom>=4.1 y prom<=6 Entonces
14        Escribir "Bueno";
15        Escribir "Su Promedio es: ", prom;
16      Sino
17        Si prom>=6.1 y prom<=8 Entonces
18          Escribir "Muy Bueno";
19          Escribir "Su Promedio es: ", prom;
20        Sino
21          Si prom>=8.1 y prom<=10 Entonces
22            Escribir "Excelente";
23            Escribir "Su Promedio Es: ", prom;
24          Sino
25            Escribir "Notas No Validas!";
26          FinSi
27        FinSi
28      FinSi
29    FinSi
30  FinSi
31 FinProceso
32

```

En diagrama de flujo

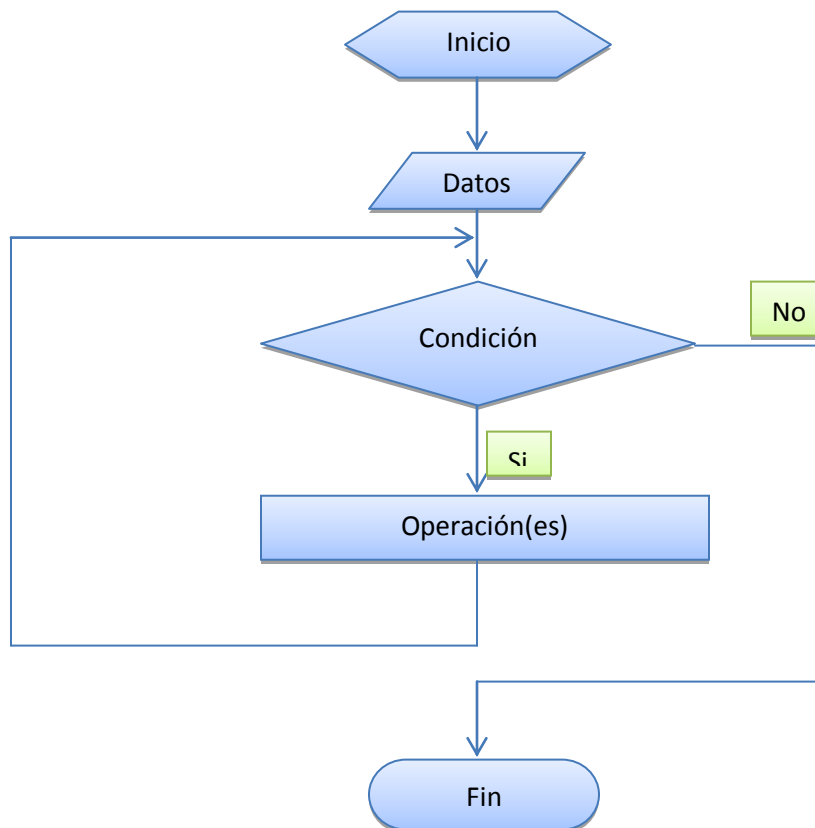


✚ c) Estructuras repetitivas (for.)

En general, la estructura for se usa en aquellas situaciones en las cuales CONOCEMOS la cantidad de veces que queremos que se ejecute el bloque de instrucciones.

Esta estructura requiere una variable entera que cumple la función de un CONTADOR de vueltas como "inicialización contador", se suele colocar el nombre de la variable En la sección de "condición" se coloca la condición que deberá ser verdadera para que el ciclo continúe (en caso de un falso, el ciclo se detendrá). Y en la sección de "incremento contador" se coloca una instrucción que permite modificar el valor de la variable que hace de contador (para permitir que alguna vez la condición sea falsa).

Su estructura es la siguiente



Ejemplos de Estructura Repetitiva (for)

- 1- Elaborar un algoritmo que ingrese desde teclado el nombre de n empleados de una empresa, y que imprima el salario base más comisión, tomando en cuenta que por cada venta se le dará el 10% de su salario en concepto de comisión.

```

1 Proceso Comision
2   Escribir "Introduzca Cuantos Salarios De Trabajadores Va a Efectuar";
3   leer trabajadores;
4   Para n<-1 Hasta trabajadores Hacer
5       Escribir "Escriba El Nombre Del Trabajador", n;
6       leer Nom;
7       Escribir "Introduzca El Sueldo Base";
8       Leer Suel;
9       Escribir "Introduzca La Primera Venta";
10      Leer Ven1;
11      Escribir "Introduzca La Segunda Venta";
12      Leer Ven2;
13      Escribir "Introduzca La Tercera Venta";
14      Leer Ven3;
15      comis<-(ven1+ven2+ven3)*0.10;
16      Escribir "+++++";
17      Escribir "";
18      escribir "En Conceptos De Comision Es De $:";
19      Escribir comis;
20      total<-suel+comis;
21      Escribir "Su Sueldo Mas Comision Es $:";
22      Escribir total;
23      Escribir "+++++";
24      Escribir "";
25   FinPara
26 FinProceso
27

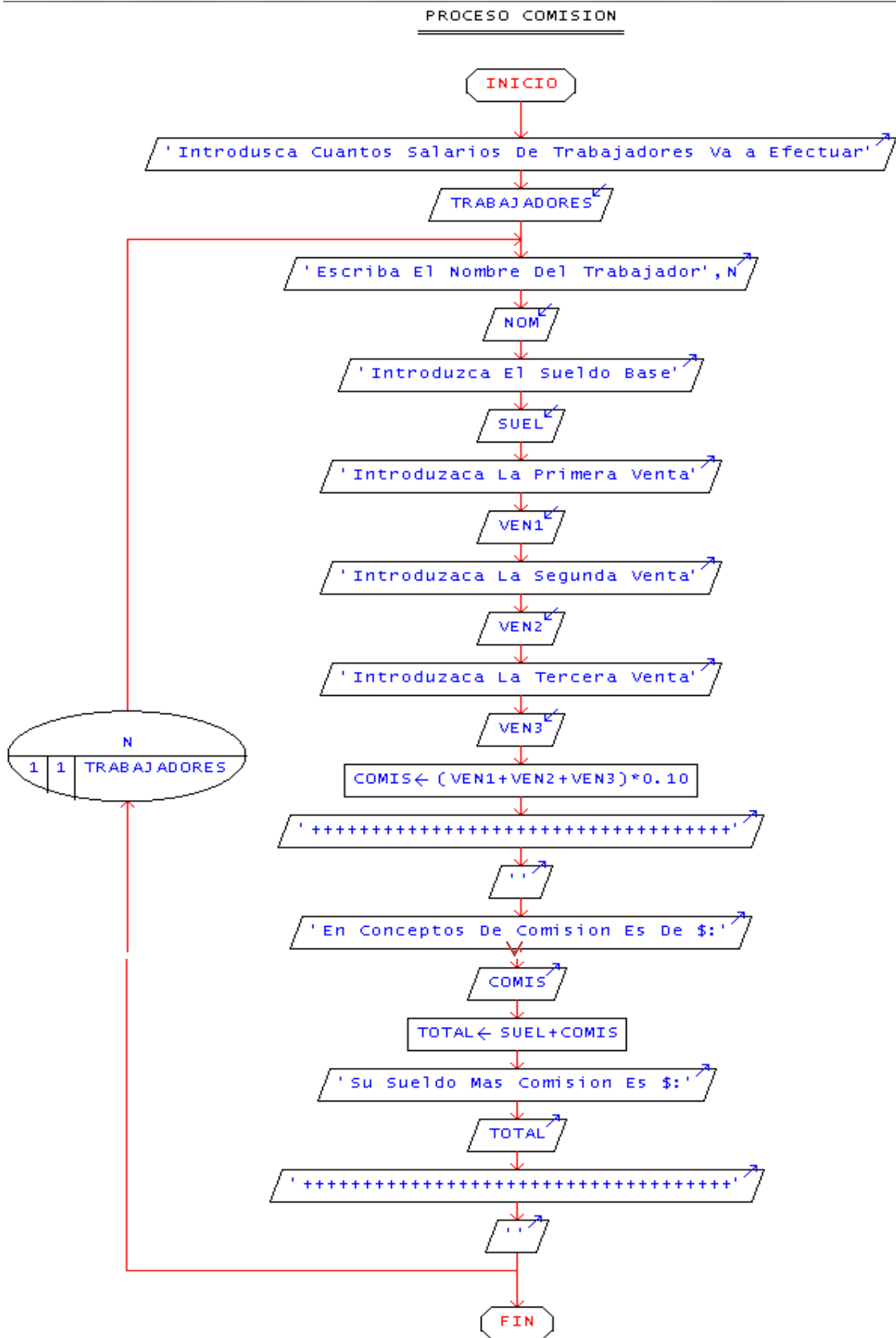
```

```

*** Ejecucion Iniciada. ***
Introduzca Cuantos Salarios De Trabajadores Va a Efectuar
> 34
Escriba El Nombre Del Trabajador1
> Jhon Mc Thopson
Introduzca El Sueldo Base
> 2134
Introduzca La Primera Venta
> 10000
Introduzca La Segunda Venta
> 1243
Introduzca La Tercera Venta
> 500
+++++
En Conceptos De Comision Es De $:
1174.3
Su Sueldo Mas Comision Es $:
3308.3
+++++
Escriba El Nombre Del Trabajador2
>

```

En diagrama de flujo



2- En una empresa se requiere hacer un aumento salarial a sus empleados los cuales estarán aumentados según la siguiente tabla.

- De \$ 0 a 1,000 aumento del 12%
- De \$1,000 a 2,500 aumento de 10%
- De \$2,500 en adelante aumento de 0.08%.

Hacer que imprima el nuevo salario según el rango del empleado.

The image shows a screenshot of a Pascal IDE (PSeInt) with a program titled 'e.repitiva.for2'. The program is designed to calculate salary increases for three employees based on their current salary ranges. The code uses nested conditional statements (if-then-else) to apply different percentage increases: 12% for salaries up to 1000, 10% for salaries between 1000 and 2500, and 0.08% for salaries above 2500. The program prompts the user for the employee's name and current salary, then displays the calculated increase and the new salary.

```

1 Proceso Aumento_Salarial
2   Para numemp<-1 Hasta 3 Hacer
3     Escribir "Ingrese Nombre Del Trabajador _", numemp;
4     Leer nom;
5     Escribir "Ingrese Salario del Trabajador _", numemp;
6     Leer Sal;
7     Si Sal>0 y Sal<1000 Entonces
8       aum<-sal*0.12;
9       ns<-sal+aum;
10      Escribir "El Aumento Es $: ", aum;
11      Escribir "Nuevo Salario Del Trabajador Es $: ",ns;
12    Sino
13      Si sal>=1000 y sal<=2500 Entonces
14        aum<-sal*0.10;
15        ns<-sal+aum;
16        Escribir "El Aumento Es $: ", aum;
17        Escribir "Nuevo Salario Del Trabajador Es $: ", ns;
18      Sino
19        Si sal>2500 Entonces
20          aum<-sal*0.08;
21          ns<-sal+aum;
22          Escribir "El Aumento Es $: ", aum;
23          Escribir "Nuevo Salario Del Trabajador Es $: ",ns;
24        FinSi
25      FinSi
26    FinPara
27  FinProceso
28
29

```

The execution window shows the following output:

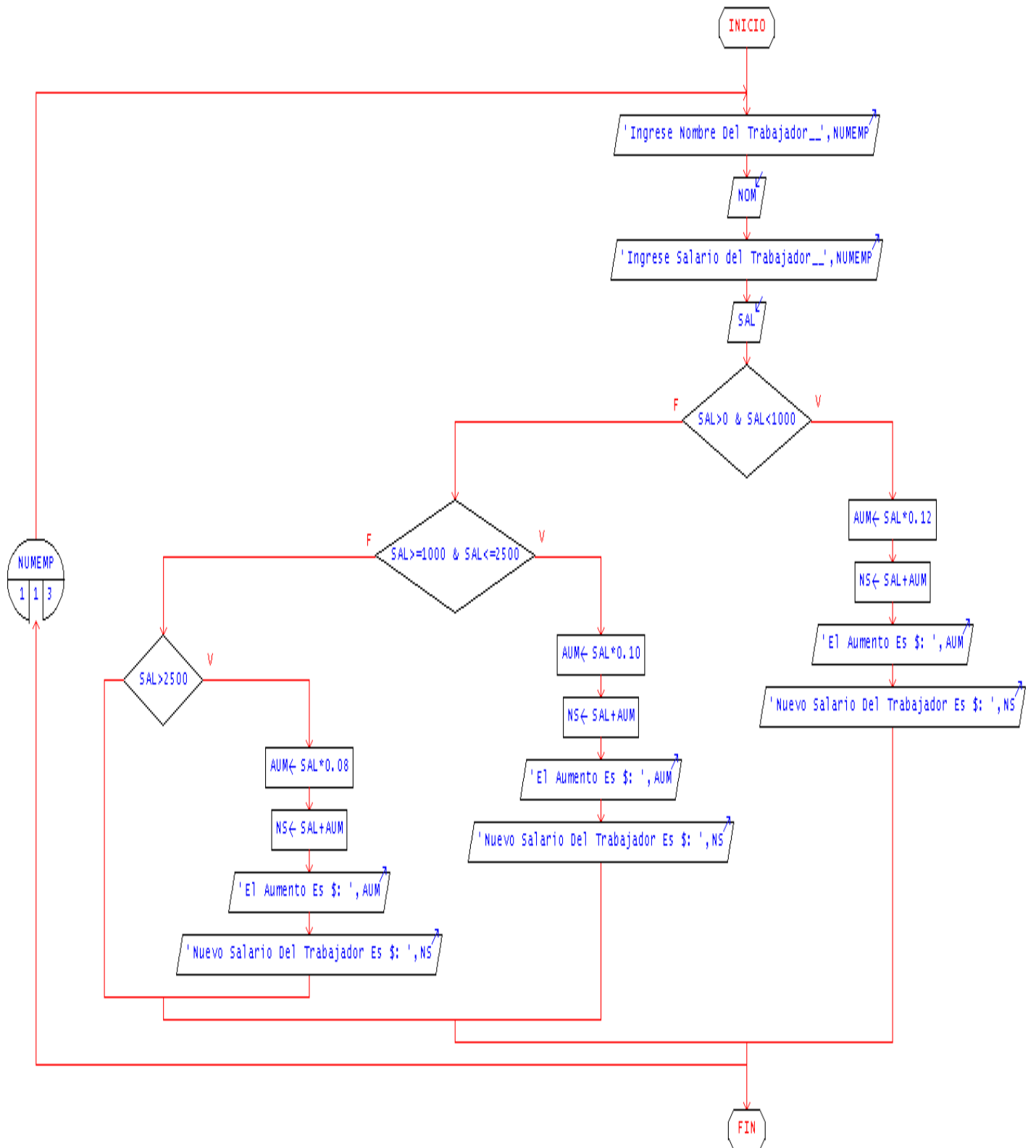
```

*** Ejecucion Iniciada. ***
Ingrese Nombre Del Trabajador_1
> Mattew Junior
Ingrese Salario del Trabajador_1
> 800
El Aumento Es $: 96
Nuevo Salario Del Trabajador Es $: 896
Ingrese Nombre Del Trabajador_2
> Stiven Cougneth
Ingrese Salario del Trabajador_2
> 1300
El Aumento Es $: 130
Nuevo Salario Del Trabajador Es $: 1430
Ingrese Nombre Del Trabajador_3
> Lucas Coughyl
Ingrese Salario del Trabajador_3
> 2900
El Aumento Es $: 232
Nuevo Salario Del Trabajador Es $: 3132
*** Ejecucion Finalizada. ***

```

En diagrama de flujo

PROCESO AUMENTO_SALARIAL



- 3- Elaborar un algoritmo para una Universidad, que tendrá una promoción para estudiantes de nuevo ingreso, esta promoción será solo para aquellos que su promedio de educación media haya sido ≥ 8 a 10, se le hará un descuento del 30% de descuento del curso de lo contrario pagará el curso completo. Hacer que el algoritmo imprima la cantidad de descuento del curso a base de materias que cursara dicho estudiante.

The image shows a Pascal IDE window titled 'PSeInt' with a file named 'e.repetitiva.for3'. The code is as follows:

```

1 Proceso Pago_Colegiatura
2   Para numep<-1 Hasta 3 Hacer
3     Escribir "Ingrese El Promedio",numep;
4     Leer prom;
5     Escribir "Ingrese El Pago Por Materia",numep;
6     Leer mat;
7     Escribir "Cuantas Materias Va a Cursar",numep;
8     Leer cur;
9     Si prom<=8 y prom<=10 Entonces
10      desc<-mat*cur*0.30;
11      totalp<-mat*cur-desc;
12      Escribir "Se Le Hara Un Descuento De $: ", desc;
13      Escribir "Su Colegiatura Con Descuento a Pagar Es $: ",totalp;
14    Sino
15      Si prom<=0 y prom<8 Entonces
16        desc<-mat*cur*0.0;
17        totalp<-mat*cur-desc;
18        Escribir "No Se Se Hara Descuento!";
19        Escribir "Su Colegiatura a Pagar Es $: ",totalp;
20      Sino
21        Escribir "Error";
22      FinSi
23    FinSi
24  FinPara
25 FinProceso
26

```

An overlaid terminal window shows the execution of the program with the following input and output:

```

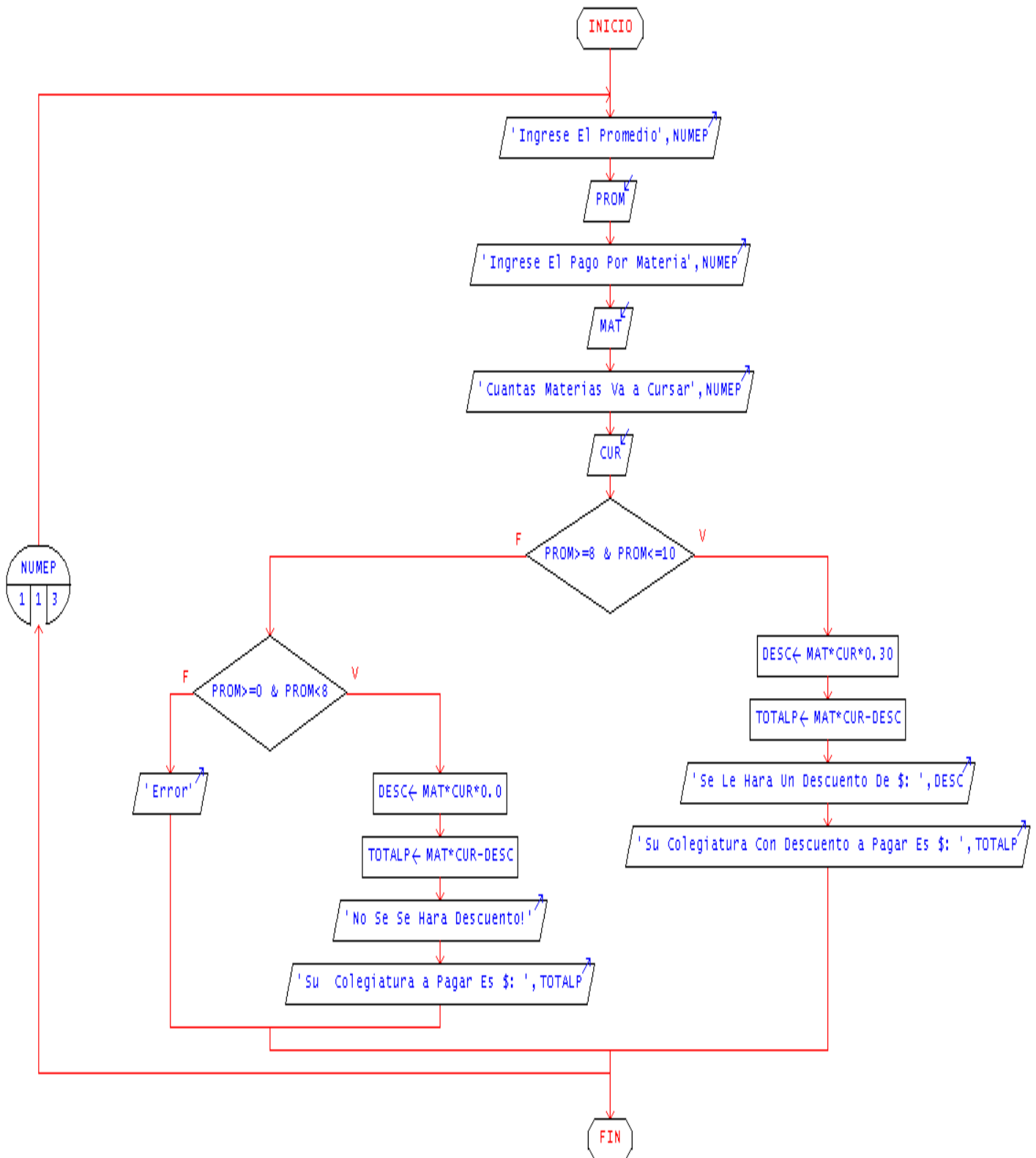
Ingrese El Promedio
> 9.02
Ingrese El Pago Por Materia
> 15
Cuantas Materias Va a Cursar1
> 5
Se Le Hara Un Descuento De $: 22.5
Su Colegiatura Con Descuento a Pagar Es $: 52.5
Ingrese El Promedio2
> 6
Ingrese El Pago Por Materia2
> 20
Cuantas Materias Va a Cursar2
> 4
No Se Se Hara Descuento!
Su Colegiatura a Pagar Es $: 80
Ingrese El Promedio3
> 6.7
Ingrese El Pago Por Materia3
> 25
Cuantas Materias Va a Cursar3
> 3
No Se Se Hara Descuento!
Su Colegiatura a Pagar Es $: 75
*** Ejecucion Finalizada. ***

```

The IDE interface includes a menu bar (Archivo, Editar, Configurar, Ejecutar, Ayuda), a toolbar with various icons, and a right-hand sidebar with a 'Comandos' panel containing flowchart symbols like 'Escribir', 'Leer', 'Asignar', 'Si-Entonces', 'Segun', 'Mientras', 'Repetir', and 'Para'.

En diagrama de flujo

PROCESO PAGO_COLEGIATURA



d) Estructura Iterativa (*while_repeat*) = *mientras y repetir*

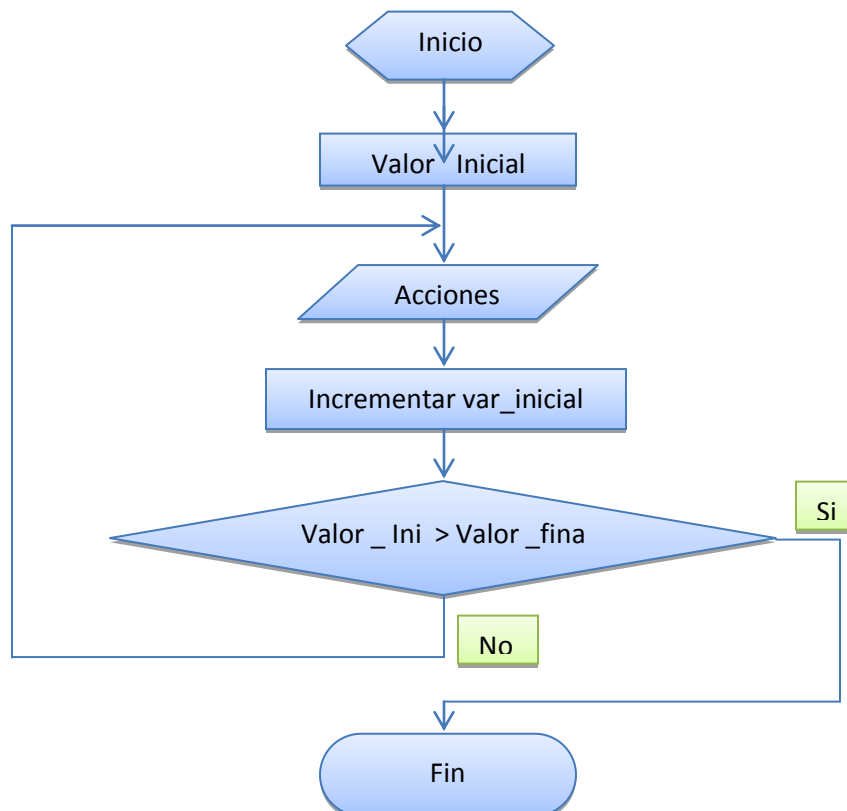
Se llama *Mientras* a la estructura algorítmica que se ejecuta mientras la condición ejecuta la instrucción especificada, llamada el cuerpo del bucle.

Este proceso de evaluación de la expresión booleana y ejecución del cuerpo se repite mientras la expresión sea cierta.

Se llama *Repetir* a la estructura algorítmica que se ejecuta un número definido de veces hasta que la condición se torna verdadera.

La diferencia entre ambas es que la condición se sitúa al principio (*Mientras*) o al final (*Repetir*) de la secuencia de instrucciones. Entonces, en el primero, el bucle continúa mientras la condición es verdadera (la cual se comprueba antes de ejecutar la acción) y en el segundo, el bucle continúa hasta que la condición se hace verdadera (la condición se comprueba después de ejecutar la acción, es decir, se ejecutará al menos una vez).

Su estructura es la siguiente



Ejemplos de algoritmos con Estructuras Iterativas

- 1- Elaborar un algoritmo para una empresa que tiene una oferta para sus clientes, en la que el cliente por cada compra saca una bolita de una tómbola y el descuento se hace de la siguiente manera. Si la bola es roja el descuento será del 40%, si la bola es amarilla será del 25%, y si es blanca la compra será gratis.

The image shows a screenshot of the PSaint IDE. The main window displays a flowchart algorithm for a supermarket process. The algorithm starts with a 'Proceso super_mercado' and includes a 'Mientras' loop that repeats as long as the user responds with 'N'. Inside the loop, it prompts for the purchase amount and the ball color. It then uses 'Si-Entonces' (if-then) structures to calculate discounts: 40% for red, 25% for yellow, and 0% for white. The total amount to be paid is calculated, and the user is asked if they want to continue. The process ends with 'FinProceso'.

```

1 Proceso super_mercado
2   resp<-"S";
3   Mientras resp<>"N" Y resp<>"n" Hacer
4     Escribir "Ingrese Gasto En Compra $";
5     Leer gastos;
6     Escribir "Ingrese Color De La Bolita";
7     Leer color;
8     Si color="Roja" Entonces
9       desc<-gastos*0.40;
10      tap<-gastos-desc;
11      Escribir "Se Le Hara Un Descuento De $: ", desc;
12      Escribir "Total a Pagar Es $: ", tap;
13     Sino
14       Si color="Amarilla" Entonces
15         desc<-gastos*0.25;
16         tap<-gastos-desc;
17         Escribir "Se Le Hara Un Descuento De $: ", desc;
18         Escribir "Total a Pagar Es $: ", tap;
19       Sino
20         Si color="Blanca" Entonces
21           desc<-gastos;
22           tap<-gastos-desc
23           Escribir "Felicidades Ganastes Tu Compra Gratis** $: ", desc;
24           Escribir "Total a Pagar Es $: ", tap;
25         FinSi
26       FinSi
27     FinSi
28     Escribir "Desea Continuar (S/N)";
29     Leer resp;
30   FinMientras
31 FinProceso
  
```

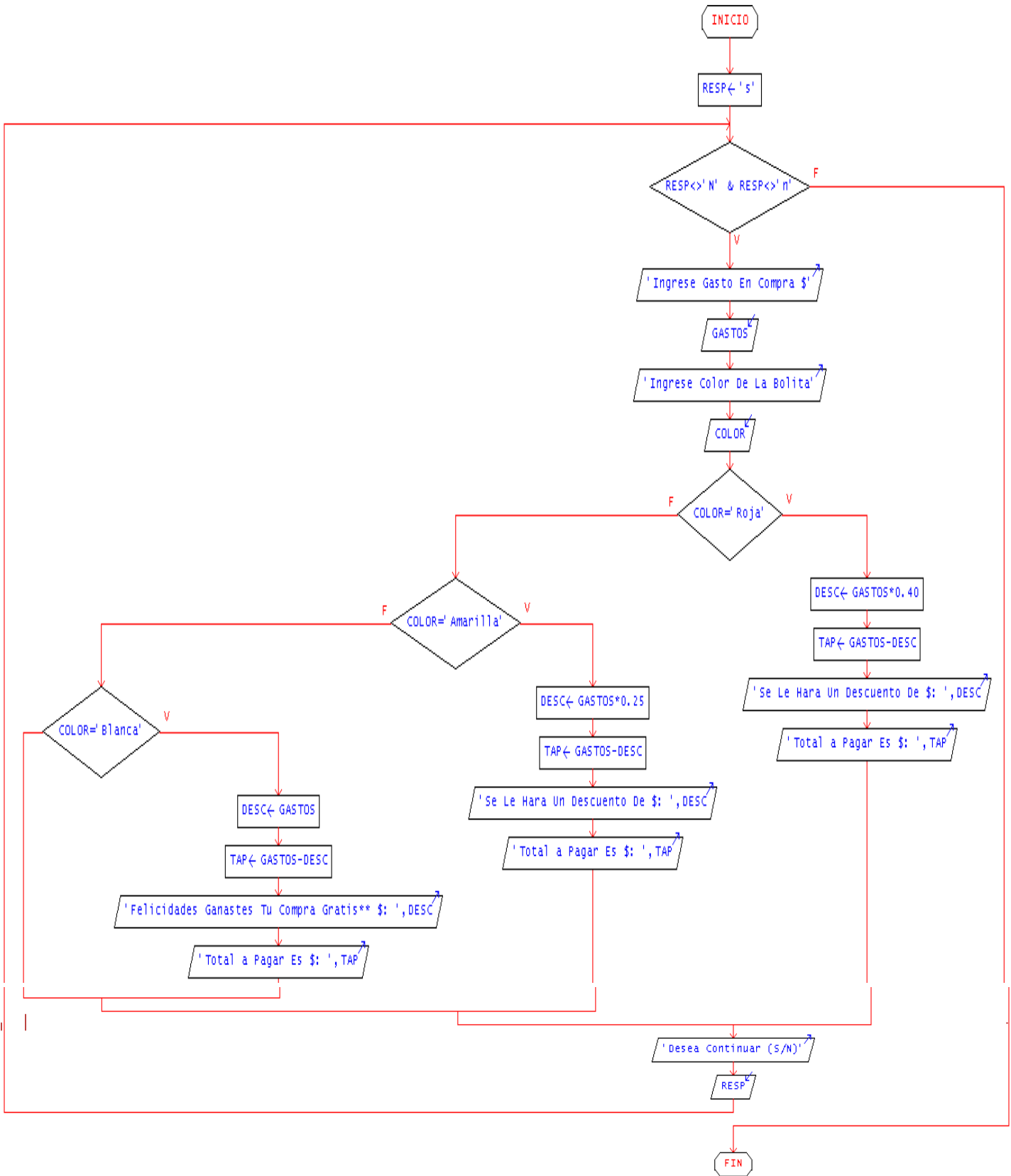
An execution window titled 'C:\Program Files\PSaint\psaint.exe' shows the following output:

```

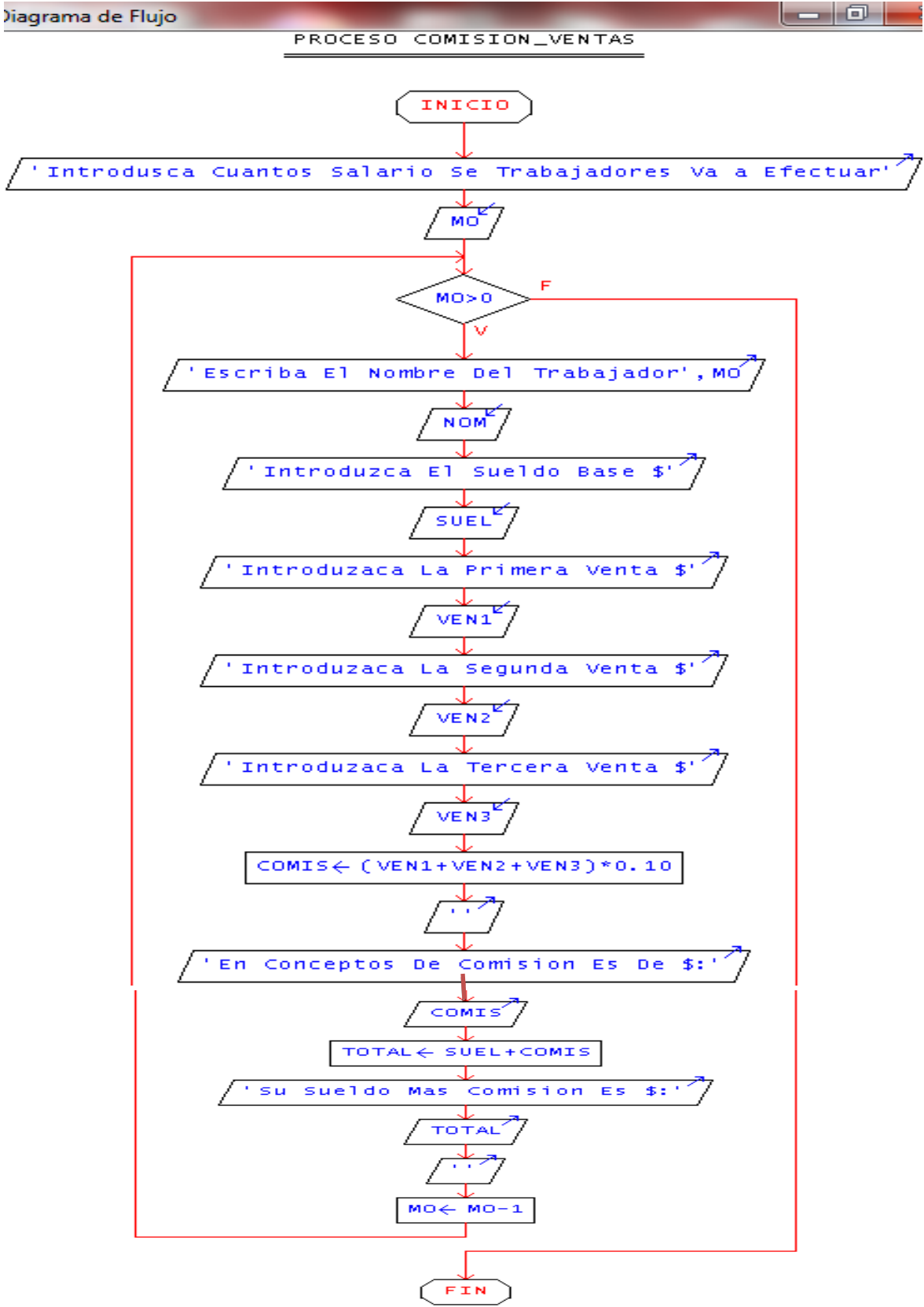
*** Ejecucion Iniciada. ***
Ingrese Gasto En Compra $
> 35
Ingrese Color De La Bolita
> Amarilla
Se Le Hara Un Descuento De $: 8.75
Total a Pagar Es $: 26.25
Desea Continuar (S/N)
> S
Ingrese Gasto En Compra $
> 245
Ingrese Color De La Bolita
> Blanca
Felicidades Ganastes Tu Compra Gratis** $: 245
Total a Pagar Es $: 0
Desea Continuar (S/N)
> S
Ingrese Gasto En Compra $
> 78
Ingrese Color De La Bolita
> Roja
Se Le Hara Un Descuento De $: 31.2
Total a Pagar Es $: 46.8
  
```

On the right side of the IDE, there is a 'Comandos' (Commands) panel with various flowchart symbols: 'Hola!' Escribir, 'Dato1' Leer, 'A+B+' Asignar, 'Si-Entonces', 'Segun', 'Mientras', 'Repetir', and 'Para'.

En diagrama de flujo



En diagrama de flujo



- 3- Elaborar un algoritmo para un juego de adivinador en el que tenga 3 intentos para acertar, que pida un número entre el 1 y 4 si el número está en el rango que imprima un mensaje en pantalla "Has Ganado".

The image shows the PSeInt IDE interface. The main window displays a flowchart for a number guessing game. The flowchart starts with 'Proceso Adivinador', followed by 'intento<-3;', a 'Repetir' loop containing 'Escribir "Adivina Un Numero Del 1 al 4, Tienes ", intento , " Intentos";', 'Leer num;', a 'Si num<=4 Entonces' block with 'Escribir "You Win**";' and 'intento<-0;', a 'Sino' block with 'intento<-intento-1;', and 'FinSi'. The process ends with 'Hasta Que intento=0' and 'FinProceso'.

An execution window titled 'C:\Program Files\PSeInt\pseint.exe' shows the output of the program:

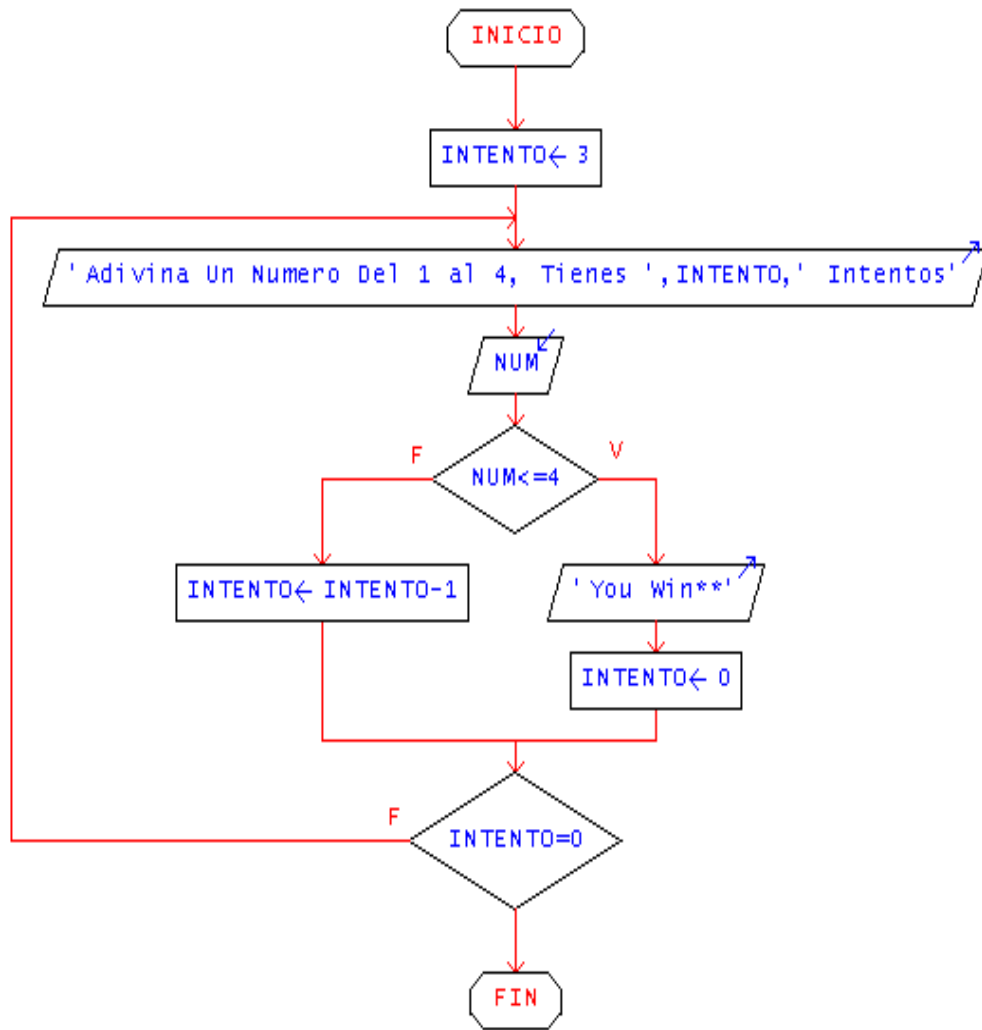

```

    *** Ejecucion Iniciada. ***
    Adivina Un Numero Del 1 al 4, Tienes 3 Intentos
    > 9
    Adivina Un Numero Del 1 al 4, Tienes 2 Intentos
    > 6
    Adivina Un Numero Del 1 al 4, Tienes 1 Intentos
    > 3
    You Win**
    *** Ejecucion Finalizada. ***
    
```

On the right side of the IDE, there is a 'Comandos' (Commands) panel with various flowchart symbols: 'Hola!' Escribir, 'Dato1' Leer, 'A←B+' Asignar, 'Si-Entonces', 'Segun', 'Mientras', 'Repetir', and 'Para'.

En diagrama de flujo

PROCESO ADIVINADOR



e) Arreglos Unidimensionales

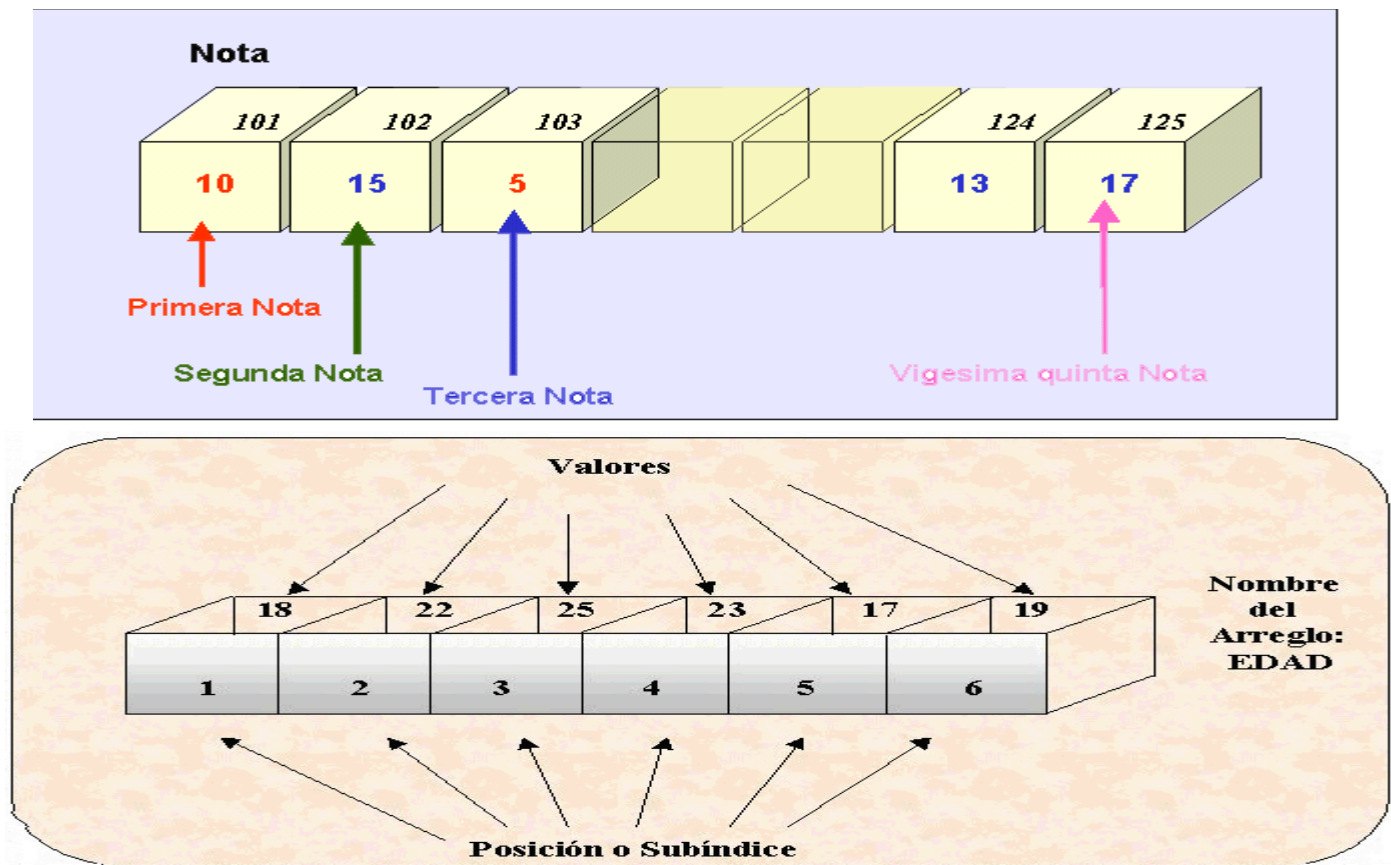
Un arreglo es un tipo de datos estructurado que está formado por una colección finita y ordenada de datos del mismo tipo.

El tipo de acceso a los arreglos unidimensionales es el acceso directo, es decir, podemos acceder a cualquier elemento del arreglo sin tener que consultar a elementos anteriores o posteriores, esto mediante el uso de un índice para cada elemento del arreglo que nos da su posición relativa.

Para implementar arreglos se debe reservar espacio en memoria.

Los arreglos nos permiten hacer un conjunto de operaciones para manipular los datos guardados en ellos, estas operaciones son: ordenar, buscar, insertar, eliminar, modificar entre otras.

Su estructura es la siguiente



Ejemplos con Estructuras De Datos de Arreglos Unidimensionales.

- 1- Crear un arreglo de números llamado "Números", que almacene los siguientes datos: 20, 14, 8, 0, 5, 19, 4 y que determine la posición de cada dato.

The screenshot shows a programming environment with a code editor and a console window. The code in the editor defines an array of 7 elements and prints each element with its index. The console window shows the output of the program execution.

```

1 Proceso Arreglo_Numero
2
3 Dimension Numeros[7];
4
5 Numeros[1]<-20;
6 Numeros[2]<-14;
7 Numeros[3]<-8;
8 Numeros[4]<-0;
9 Numeros[5]<-5;
10 Numeros[6]<-19;
11 Numeros[7]<-4;
12
13 Para i<-1 Hasta 7 Con Paso 1 Hacer
14     Escribir "El Elemento En La Posicion: ",i,"Es: ", numeros[i];
15 FinPara
16
17 FinProceso
18

```

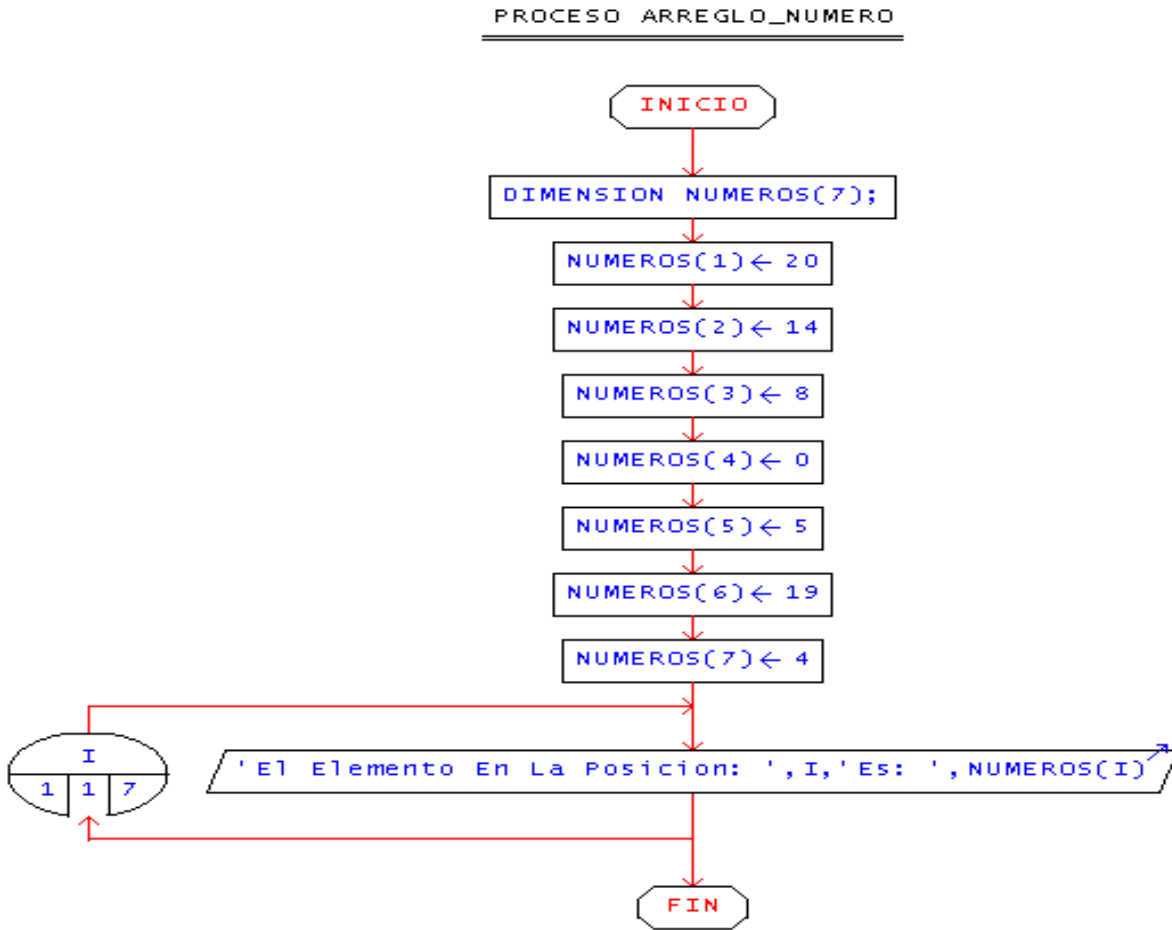
Console Output:

```

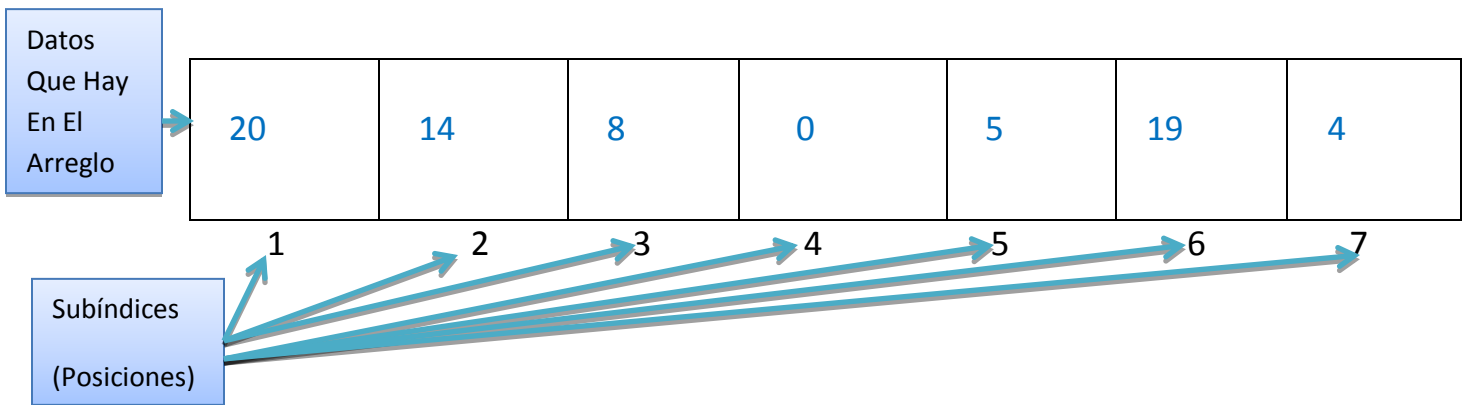
*** Ejecucion Iniciada. ***
El Elemento En La Posicion: 1Es: 20
El Elemento En La Posicion: 2Es: 14
El Elemento En La Posicion: 3Es: 8
El Elemento En La Posicion: 4Es: 0
El Elemento En La Posicion: 5Es: 5
El Elemento En La Posicion: 6Es: 19
El Elemento En La Posicion: 7Es: 4
*** Ejecucion Finalizada. ***

```

En diagrama de flujo



En Vector



2- Llenar un arreglo con x cantidades de números. Preguntar al usuario cuantas cantidades va a ingresar, y sumar todos estos elementos del vector.

The image shows the PSeInt IDE interface with a Pascal program and its execution output. The program is titled 'e. arreglos unidimensionales2' and contains the following code:

```

1 Proceso Suma_Arreglo
2
3   Dimension arreglo[100];
4
5   acum<-0
6
7   Escribir "Digite El Tamaño Del Vector";
8   Leer t;
9
10  Para i<-1 Hasta t Con Paso 1 Hacer
11  .....
12     Escribir "Digite El Numero";
13     Leer arreglo[i];
14
15     acum<-acum+arreglo[i];
16
17  FinPara
18
19  Para i<-1 Hasta t Con Paso 1 Hacer
20  .....
21     Escribir "La Suma: ",arreglo[i];
22
23  FinPara
24
25  Escribir "Es: ",acum;
26
27 FinProceso
    
```

The execution window shows the following output:

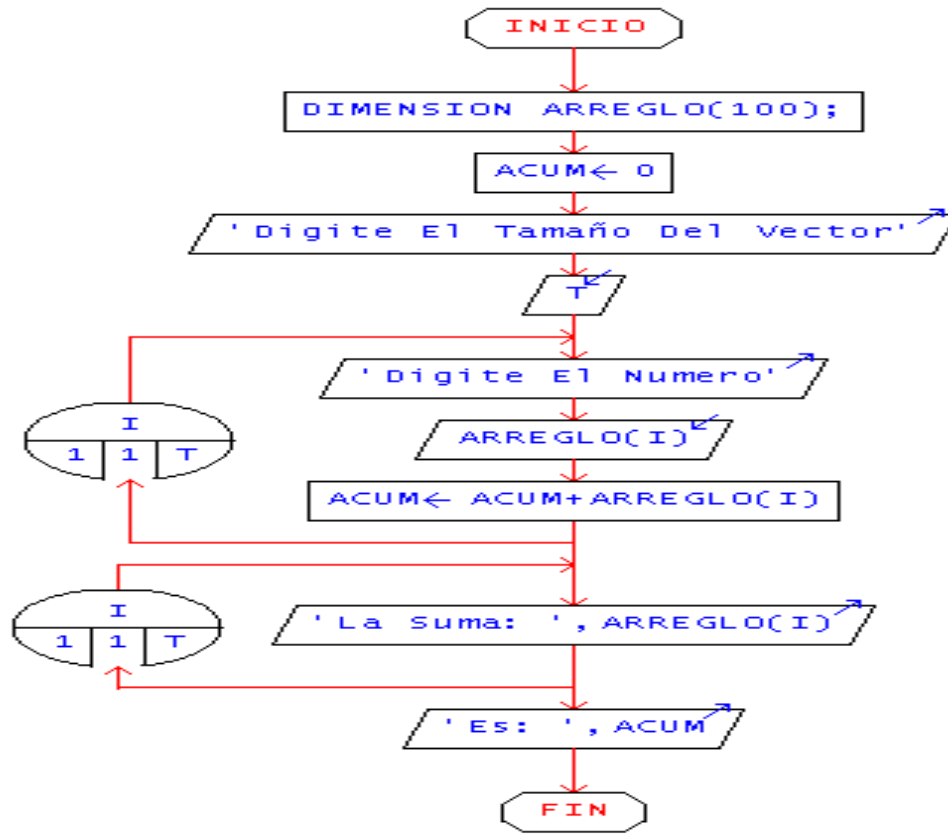
```

*** Ejecucion Iniciada. ***
Digite El Tamaño Del Vector
> 5
Digite El Numero
> 567
Digite El Numero
> 23
Digite El Numero
> 65
Digite El Numero
> ?
Digite El Numero
> 887
La Suma: 567
La Suma: 23
La Suma: 65
La Suma: ?
La Suma: 887
Es: 1549
*** Ejecucion Finalizada. ***
    
```

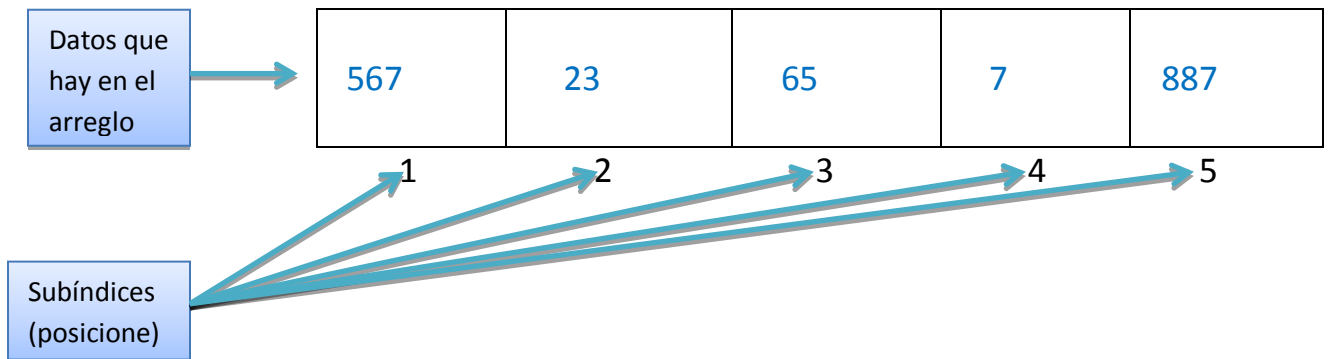
The IDE interface includes a menu bar (Archivo, Editar, Configurar, Ejecutar, Ayuda), a toolbar with various icons, and a right-hand sidebar with a 'Comandos' panel containing flowchart symbols like 'Escribir', 'Leer', 'Asignar', 'Si-Entonces', 'Segun', 'Mientras', 'Repetir', and 'Para'.

En diagrama de flujo

PROCESO SUMA_ARREGLO



En Vector



Conclusiones

Habiendo conocido todo lo relacionado a Estructuras de Datos desde su creación hasta como ejecutar el código en un programa hemos identificado las diferentes estructuras, características y algunos métodos de ejecutar los algoritmos podemos decir que estamos preparados para poder en primer lugar formular nuestro algoritmo junto a ello ver en que estructura se hará para luego desarrollarlo con un método de los pasos lógicos posible en la elaboración de dicho programa.

Aunque la practica hace la experiencia y se amplían nuestros conocimientos quedamos en la obligación de seguir practicando para que en un futuro cercano podamos lograr el vasto conocimiento y poder hacer uso de la lógica para programar que en la actualidad es muy importante para nuestro beneficio intelectual como laboral y dar los resultados que de nosotros se espera.

Referencias

<http://www.algoritmos>

<http://www.arreglosunidimensionales>

http://www.arreglosenpsint100915151154_phpapp02

www.uls/eva/videosdeestructurasdedatos/youtube