

En esta ocasión tenemos un tema práctico en el que podrán ejercitar sus conocimientos y conceptos aprendidos en la unidad. Deberán crear los algoritmos que sirvan para resolver las situaciones planteadas en los diferentes ejercicios que hemos escogido para la misma.

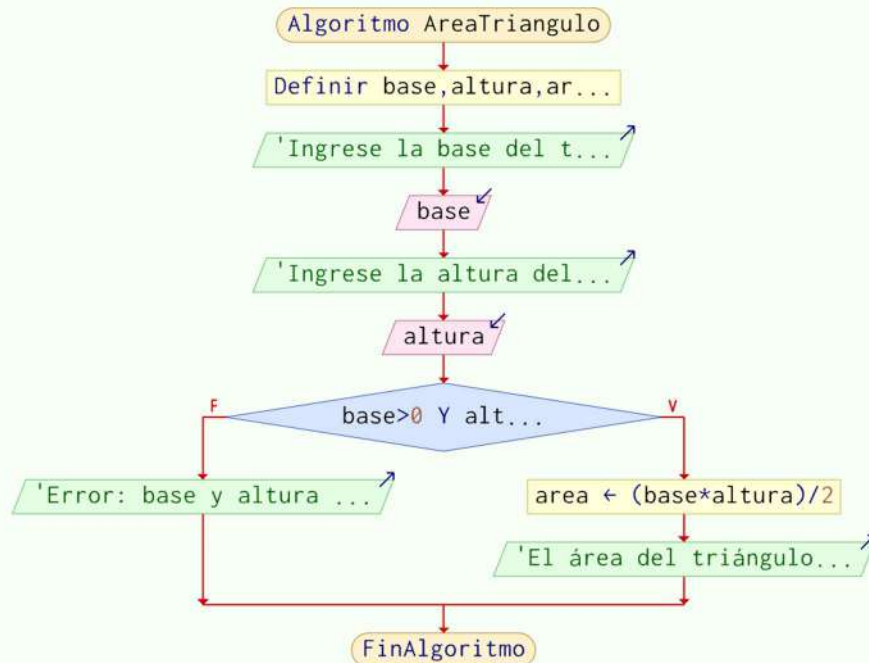
Estos algoritmos deben ser representados de manera gráfica o escrita, es decir que pueden representarlo de la manera que prefieran: por pseudocódigo o por diagrama de flujo.

Los ejercicios son los siguientes:

I- REALIZAR:

1) Determinar el área de un triángulo, introducir por teclado la Altura y Base del mismo. Este resultado deberá ser presentado por pantalla.

```
1  Algoritmo AreaTriangulo
2      Definir base, altura, area Como Real
3
4      Escribir "Ingrese la base del triángulo:"
5      Leer base
6      Escribir "Ingrese la altura del triángulo:"
7      Leer altura
8
9      Si base > 0 Y altura > 0 Entonces
10         area ← (base * altura) / 2
11         Escribir "El área del triángulo es: ", area
12     SiNo
13         Escribir "Error: base y altura deben ser mayores que cero."
14     FinSi
15 FinAlgoritmo
16
```



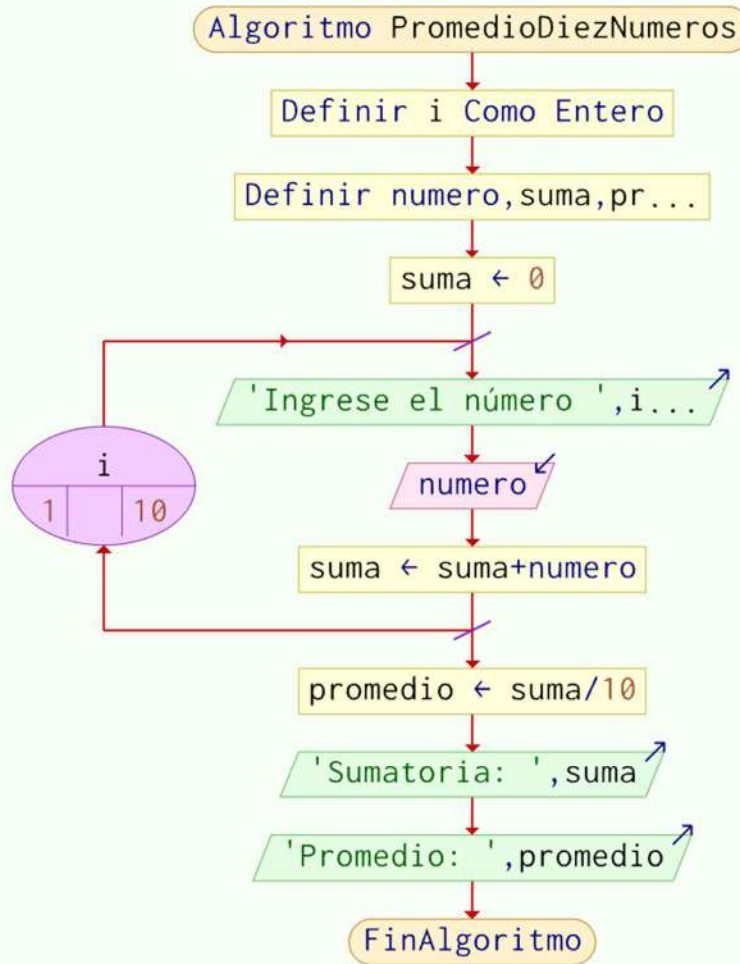
PSelnt - Ejecutando proceso AREATRIANGULO

```
*** Ejecución Iniciada. ***
Ingrese la base del triángulo:
> 4
Ingrese la altura del triángulo:
> 2
El área del triángulo es: 4
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar

2) Escribir algoritmo que pida diez números cualesquiera y que determine el promedio de los mismos, debiendo presentar como resultado la sumatoria y el promedio de éstos.

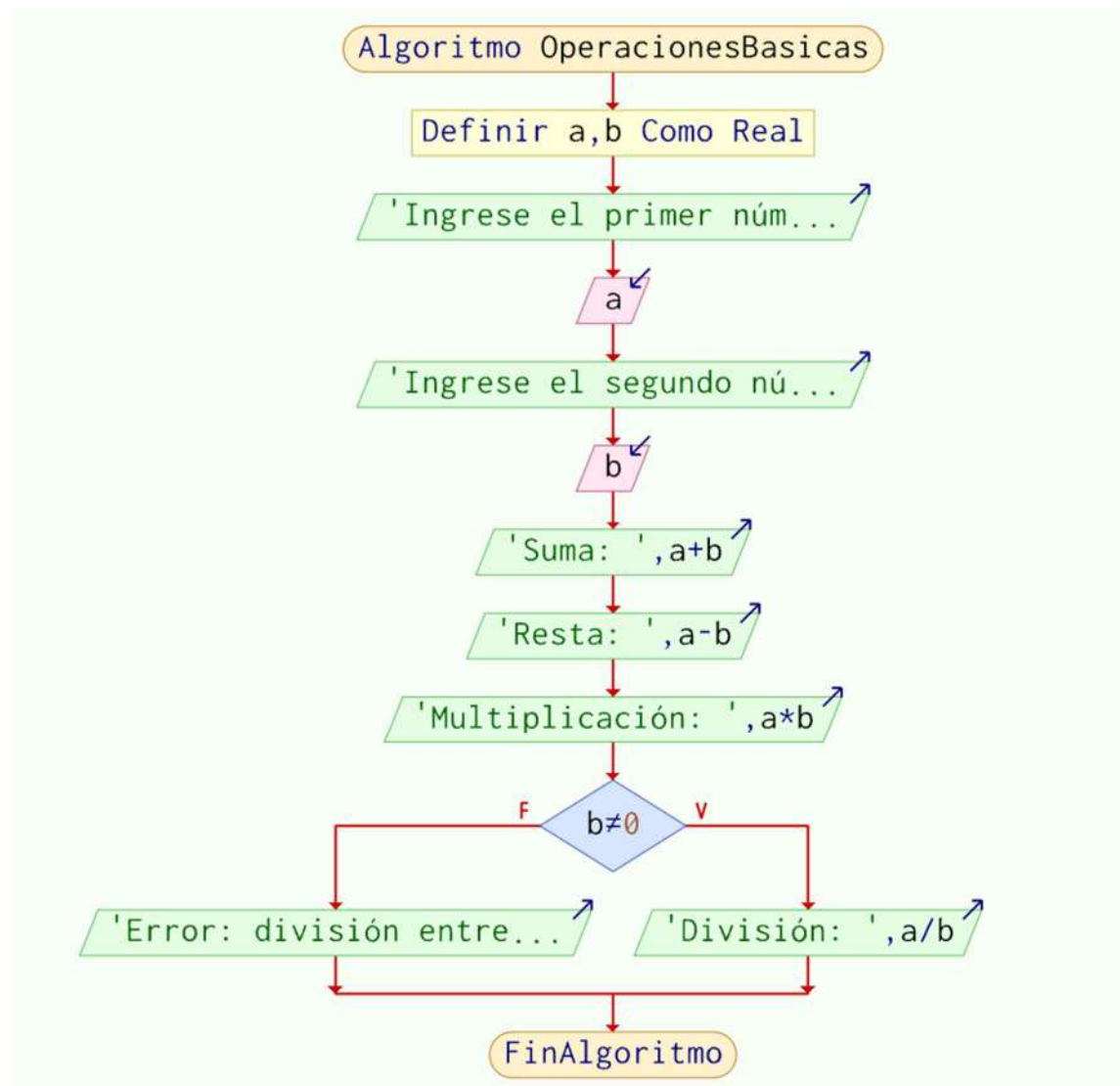
```
1  Algoritmo PromedioDiezNumeros
2      Definir i Como Entero
3      Definir numero,suma,promedio Como Real
4      suma ← 0
5      Para i←1 Hasta 10 Hacer
6          Escribir 'Ingrese el número ',i,':'
7          Leer numero
8          suma ← suma+numero
9      FinPara
10     promedio ← suma/10
11     Escribir 'Sumatoria: ',suma
12     Escribir 'Promedio: ',promedio
13 FinAlgoritmo
14
```



```
PSelnt - Ejecutando proceso PROMEDIODIEZN...  
*** Ejecución Iniciada. ***  
Ingrese el número 1:  
> 2  
Ingrese el número 2:  
> 2  
Ingrese el número 3:  
> 2  
Ingrese el número 4:  
> 2  
Ingrese el número 5:  
> 2  
Ingrese el número 6:  
> 2  
Ingrese el número 7:  
> 2  
Ingrese el número 8:  
> 2  
Ingrese el número 9:  
> 2  
Ingrese el número 10:  
> 2  
Sumatoria: 20  
Promedio: 2  
*** Ejecución Finalizada. ***
```

3) Crear el algoritmo que sume, reste, multiplique y divida dos números y que despliegue el resultado de esas operaciones. Tomar en cuenta que la división entre cero (0) no es permitida. El algoritmo debe evaluar que el segundo valor sea diferente de cero.

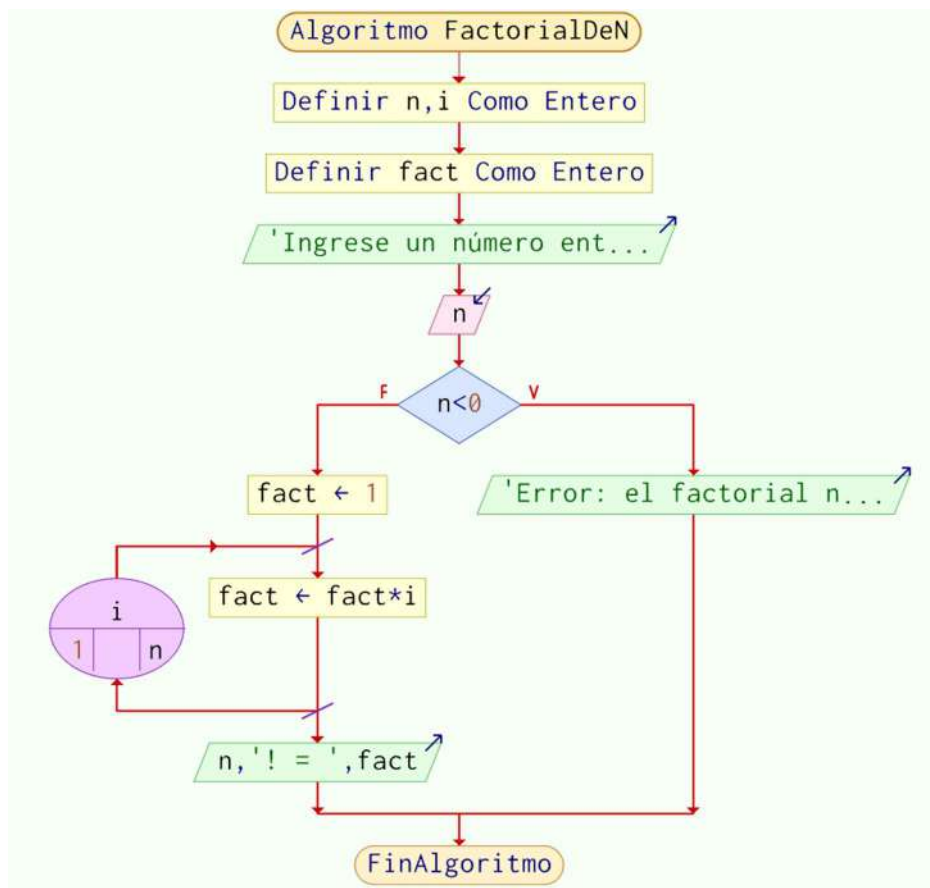
```
1  Algoritmo OperacionesBasicas
2      Definir a, b Como Real
3
4      Escribir "Ingrese el primer número (a):"
5      Leer a
6      Escribir "Ingrese el segundo número (b):"
7      Leer b
8
9      Escribir "Suma: ", a + b
10     Escribir "Resta: ", a - b
11     Escribir "Multiplicación: ", a * b
12
13     Si b ≠ 0 Entonces
14         Escribir "División: ", a / b
15     SiNo
16         Escribir "Error: división entre cero no permitida."
17     FinSi
18 FinAlgoritmo
19
```

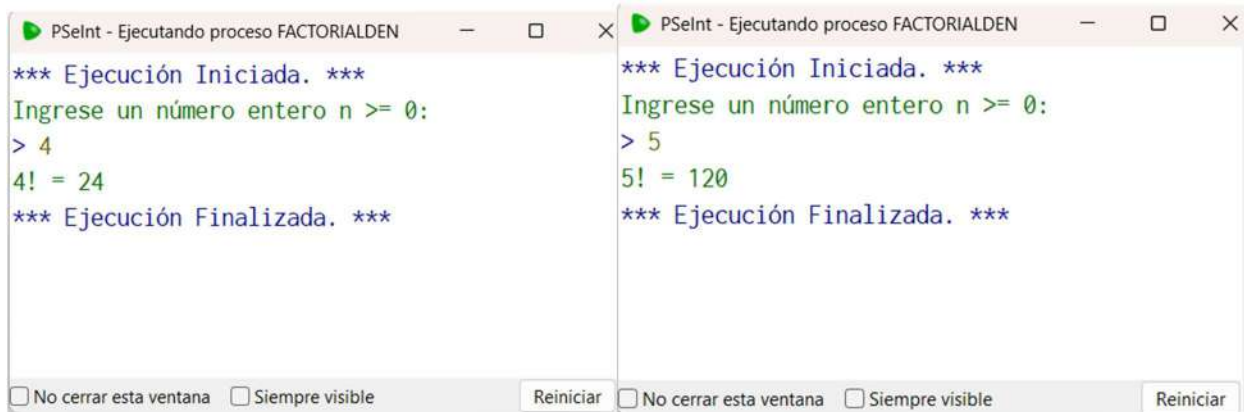


PSInt - Ejecutando proceso OPERACIONESBA...	PSInt - Ejecutando proceso OPERACIONESBA...
*** Ejecución Iniciada. *** Ingrese el primer número (a): > 4 Ingrese el segundo número (b): > 2 Suma: 6 Resta: 2 Multiplicación: 8 División: 2 *** Ejecución Finalizada. ***	*** Ejecución Iniciada. *** Ingrese el primer número (a): > 4 Ingrese el segundo número (b): > 0 Suma: 4 Resta: 4 Multiplicación: 0 Error: división entre cero no permitida. *** Ejecución Finalizada. ***
<input type="checkbox"/> No cerrar esta ventana <input type="checkbox"/> Siempre visible <input type="button" value="Reiniciar"/>	<input type="checkbox"/> No cerrar esta ventana <input type="checkbox"/> Siempre visible <input type="button" value="Reiniciar"/>

4) Crear algoritmo para determinar la factorial de un número dado y que imprima el resultado.

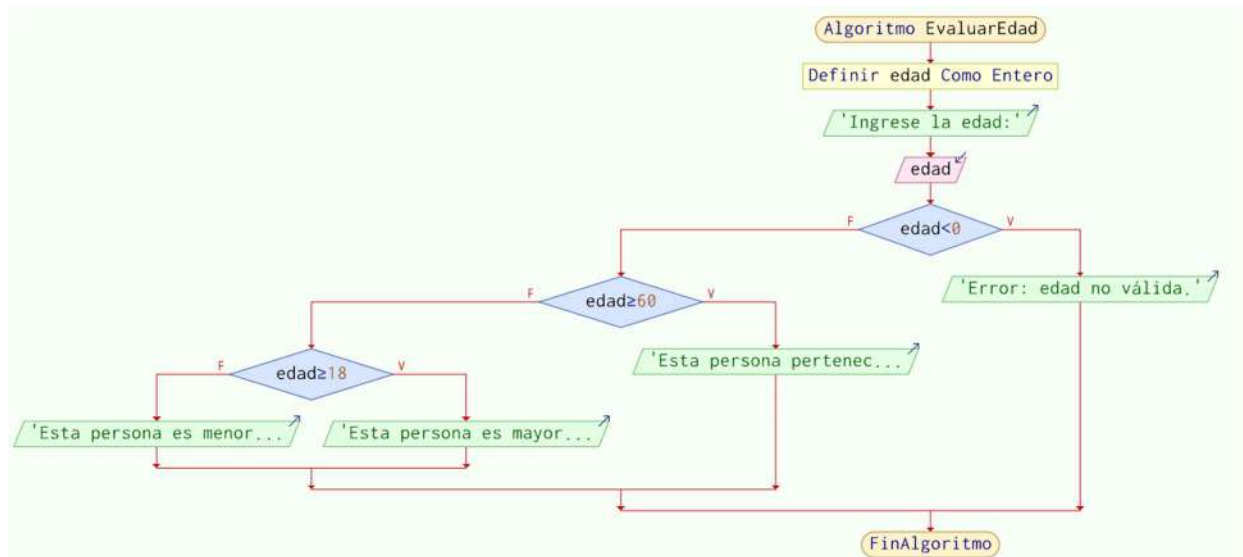
```
1  Algoritmo FactorialDeN
2      Definir n, i Como Entero
3      Definir fact Como Entero
4
5      Escribir "Ingrese un número entero n >= 0:"
6      Leer n
7
8      Si n < 0 Entonces
9          Escribir "Error: el factorial no está definido para números negativos."
10     SiNo
11         fact ← 1
12         Para i ← 1 Hasta n Hacer
13             fact ← fact * i
14         FinPara
15         Escribir n, "! = ", fact
16     FinSi
17 FinAlgoritmo
```





5) Escribir algoritmo que pida la edad de la persona por teclado y que imprima el resultado de evaluar la mayoría de edad: "Esta persona es menor de edad" o "Esta persona es mayor de edad" y, si es mayor de 60, que el mensaje impreso diga: "Esta persona pertenece a la tercera edad". (if anidado).

```
1  Algoritmo EvaluarEdad
2      Definir edad Como Entero
3
4      Escribir "Ingrese la edad:"
5      Leer edad
6
7      Si edad < 0 Entonces
8          Escribir "Error: edad no válida."
9      SiNo
10         Si edad ≥ 60 Entonces
11             Escribir "Esta persona pertenece a la tercera edad"
12         SiNo
13             Si edad ≥ 18 Entonces
14                 Escribir "Esta persona es mayor de edad"
15             SiNo
16                 Escribir "Esta persona es menor de edad"
17             FinSi
18         FinSi
19     FinSi
20 FinAlgoritmo
```



PSeInt - Ejecutando proceso EVALUAREDA

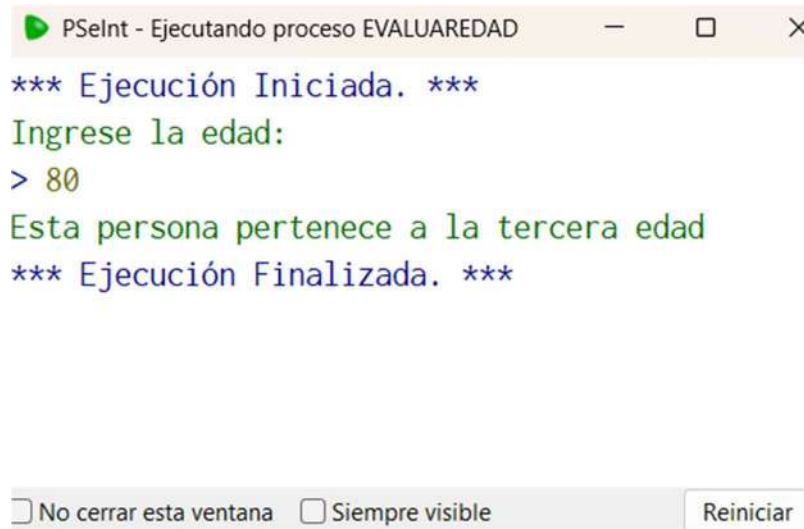
```
*** Ejecución Iniciada. ***
Ingrese la edad:
> 16
Esta persona es menor de edad
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar

PSeInt - Ejecutando proceso EVALUAREDA

```
*** Ejecución Iniciada. ***
Ingrese la edad:
> 25
Esta persona es mayor de edad
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar



```
*** Ejecución Iniciada. ***  
Ingrese la edad:  
> 80  
Esta persona pertenece a la tercera edad  
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☐ Siempre visible

II- RESPONDER

1. ¿Qué son los algoritmos computacionales y mencione sus características principales?

Un algoritmo computacional es un conjunto finito y ordenado de pasos o instrucciones precisas que describen cómo resolver un problema o realizar una tarea. En computación, el algoritmo se diseña para que pueda ser implementado en un programa y ejecutado por una máquina.

Características principales:

- Finitud: Debe terminar después de un número limitado de pasos.
- Definitud / precisión: Cada paso es claro, sin ambigüedades.
- Entrada: Acepta cero o más valores de entrada.
- Salida: Produce al menos una salida (resultado).
- Efectividad: Cada operación es elemental y realizable.
- Determinismo: Dadas las mismas entradas, produce las mismas salidas (salvo algoritmos aleatorios).
- Corrección: Resuelve el problema planteado para las entradas válidas.
- Eficiencia: Usa recursos (tiempo y memoria) de manera razonable.

2. ¿Cuáles son las diferentes formas en que se representa un algoritmo y exprese que forma te parece más sencilla o clara? Justifique

Formas de representación:

- **Lenguaje natural:** Descripción en palabras.
- **Pseudocódigo:** Instrucciones similares a un lenguaje de programación, pero más simple.
- **Diagramas de flujo:** Representación gráfica con símbolos (inicio, proceso, decisión).
- **Diagramas N-S (Nassi-Shneiderman):** Bloques estructurados para decisiones y ciclos.

La más sencilla y clara:

El pseudocódigo, porque:

- Es fácil de escribir y entender.
- Refleja la lógica del programa tal como se implementará.
- Permite pasar rápidamente al código real sin ambigüedades.

3. ¿Describa brevemente las etapas básicas que se llevan a cabo para la resolución de problemas?

- **Análisis del problema:** Comprender qué se necesita resolver.
- **Diseño de la solución:** Crear el algoritmo (pseudocódigo o diagrama).
- **Codificación:** Traducir el algoritmo a un lenguaje de programación.
- **Pruebas:** Verificar que el programa funciona correctamente.
- **Documentación y mantenimiento:** Registrar el diseño y corregir errores futuros.

4. ¿Defina estructura de control y como se clasifican?

Una estructura de control es el mecanismo que define cómo se ejecutan las instrucciones en un algoritmo o programa, permitiendo controlar el flujo de ejecución. Sin estas estructuras, el programa solo ejecutaría instrucciones de manera lineal (una tras otra), lo que limitaría su funcionalidad.

Clasificación:

- **Secuenciales:** Instrucciones que se ejecutan en orden.
- **Selectivas (decisión):** Permiten elegir entre alternativas (Si...Entonces, Segun).
- **Repetitivas (iteración):** Permiten repetir instrucciones (Para, Mientras, Repetir).

Software utilizado

