



U.A. PROGRAMACION MODULAR	
Número de practica: 3	Academia de Programación
Título: Estructuras Repetitivas	
Objetivo: El alumno realiza ejercicios donde hace uso de las estructuras repetitivas: for, while y do while.	

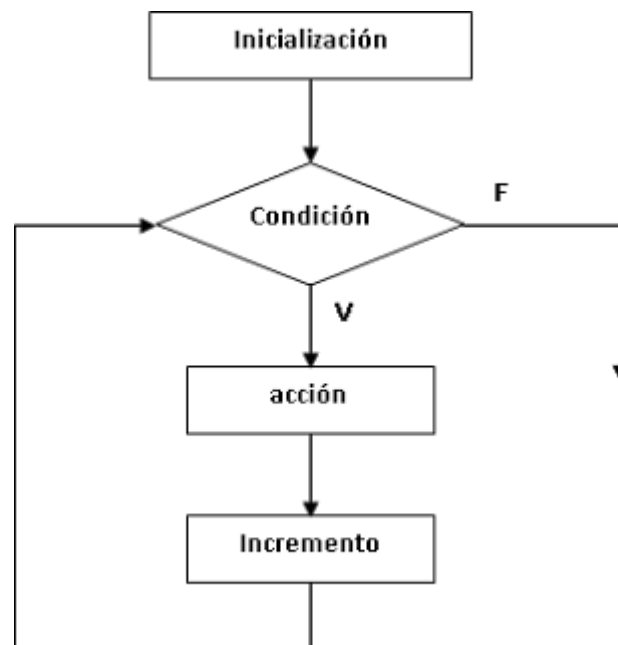
## Estructuras Repetitivas

Las estructuras repetitivas se utilizan cuando se quiere que un conjunto de instrucciones se ejecuten un cierto número finito de veces, por ejemplo, escribir algo en pantalla cierta cantidad de veces, mover un objeto de un punto a otra cierta cantidad de pasos, o hacer una operación matemática cierta cantidad de veces. Se les llama bucle o ciclo a todo proceso que se repite cierto número de veces dentro de un pseudocódigo o un programa y las estructuras repetitivas nos permiten hacerlo de forma sencilla.

A cada repetición de bloque de instrucciones se le conoce como iteración.

### Estructura de repetición FOR

Este tipo de estructuras incluye un contador como parte de su estructura, lo cual, quiere decir que se conoce el número de veces que se repetirá el cuerpo del bucle.



SINTAXIS:

Para una sola acción:

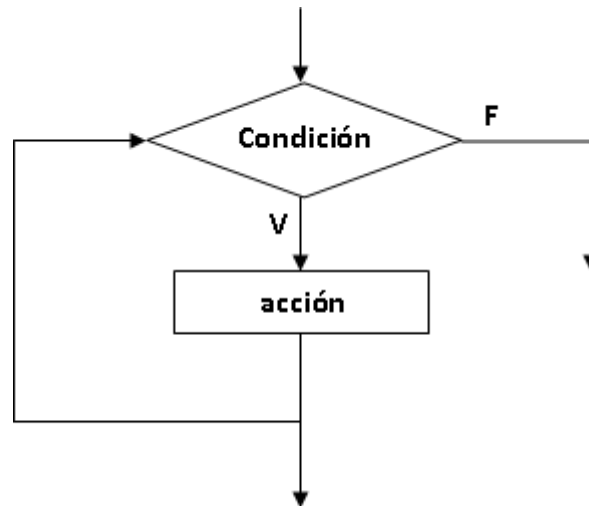
```
for ( inicio; condición; incremento )  
    acción;
```

Para varias acciones:

```
for ( inicio; condición; incremento ) {  
    acción 1;  
    acción 2;  
    .  
    .  
    .  
    acción n;  
}
```

### Estructura de repetición WHILE

Esta estructura primero verifica la condición y luego ejecuta la acción. La acción puede ser una acción simple o una acción compuesta (bloque de instrucciones encerradas entre llaves).



SINTAXIS:

Para una sola acción:

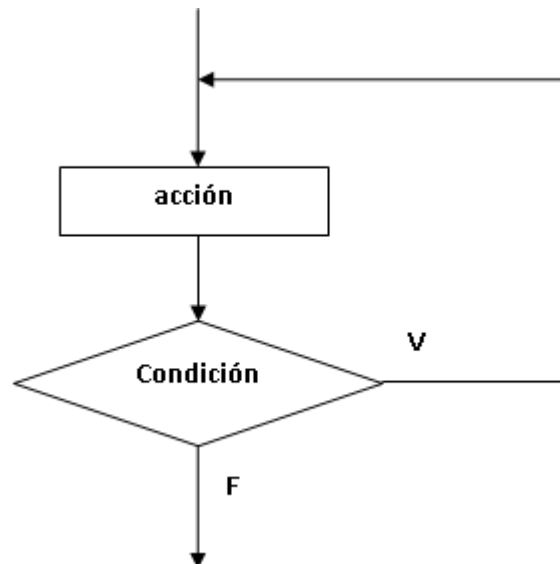
```
while ( condición )  
    acción;
```

Para varias acciones:

```
while ( condición ) {  
    acción 1;  
    acción 2;  
    .  
    .  
    .  
    acción n;  
}
```

### Estructura de repetición DO WHILE

Este tipo de estructura primero ejecuta la acción y luego verifica la condición. La acción puede ser simple o compuesta.



### SINTAXIS:

Para una sola acción:

```
do  
    acción;  
while ( condición );
```

Para varias acciones:

```
do {  
    acción 1;  
    acción 2;  
    .  
    .  
    .  
    acción n;  
} while ( condición );
```

## Ejemplos

### Ejemplo 1 de while

Desarrollar un programa que permita la carga de 10 valores por teclado y nos muestre posteriormente la suma de los valores ingresados y su promedio.

### Ejemplo 2 de while

Una planta que fabrica perfiles de hierro posee un lote de una cantidad desconocida de piezas. Por lo que se debe de preguntar si se verificara otra pieza.

Crear un programa que pida ingresar por teclado la cantidad de piezas a procesar y luego ingrese la longitud de cada perfil; sabiendo que la pieza cuya longitud esté comprendida en el rango de 1.20 y 1.30 son aptas. Imprimir por pantalla la cantidad de piezas aptas que hay en el lote.

### Ejemplo 3 do . . while

Una planta de fábrica de perfiles necesita saber, de las piezas fabricadas:

- Cuántas piezas tienen un peso entre 9.8 Kg. y 10.2 Kg.?, cuántas con más de 10.2 Kg.? y cuántas con menos de 9.8 Kg.?
- La cantidad total de piezas procesadas.

Realizar un programa que permita ingresar el peso (en kilogramos) de piezas. El proceso termina cuando ingresamos el valor 0.

### Ejemplo 4 for.

Crea un programa que realice la tabla de multiplicar de un numero ingresado desde el teclado.

### Ejemplo 5 for

Escribir un programa que lea 50 notas de alumnos y nos informe cuántos tienen notas mayores o iguales a 6 (aprobados) y cuántos menores (reprobados)

### Ejemplo 6 for

Escribir un programa que lea 50 notas de alumnos y nos informe cuántos tienen notas mayores o iguales a 6 (aprobados) y cuántos menores (reprobados), adicionalmente obtener la calificación mayor y la menor.

## Ejercicios

1. Hacer un programa que lea los nombres y las edades de varias personas (n), al finalizar el programa debe mostrar el nombre y la edad de la persona mayor, también el nombre y la edad de la persona menor.
2. Diseña un programa que permita ingresar de una en una las edades de un conjunto de personas y muestre luego de cada ingreso:
  - a) El número de personas mayores de edad.
  - b) El número de personas menores de edad.
  - c) La mayor edad ingresada.
  - d) La menor edad ingresada.

No se cuenta con el valor exacto de las personas.

3. En un banco se procesan datos de las cuentas corrientes de sus clientes. De cada cuenta corriente se conoce: número de cuenta y saldo actual. El ingreso de datos debe finalizar al ingresar un valor negativo en el número de cuenta.  
Se pide realizar un programa que lea los datos de las cuentas corrientes e informe:
  - a) De cada cuenta: número de cuenta y estado de la cuenta según su saldo, sabiendo que:  
Estado de la cuenta  
'Acreeedor' si el saldo es >0.  
'Deudor' si el saldo es <0.  
'Nulo' si el saldo es =0.
  - c) La suma total de los saldos acreedores.
4. Escriba un programa que lea mediante teclado un número entero positivo n y mediante un método recursivo sume los n primeros términos de la siguiente serie:

$$\sum_{k=1}^n \frac{1}{k^3} = 1 + \frac{1}{2^3} + \frac{1}{3^3} + \frac{1}{4^3} + \frac{1}{5^3} + \dots + \frac{1}{n^3}$$

El programa deberá mostrar por pantalla el resultado de la suma con 4 dígitos decimales de precisión. En caso de que el número introducido por el usuario no sea positivo, el programa volverá a pedir que se introduzca un nuevo número hasta que el número introducido sea positivo. Utilice la clase Scanner para efectuar la lectura de datos y el método printf para la salida de datos.