

Program Control Statements. Part 2

Iterativa: sirven para repetir las sentencias de un bloque

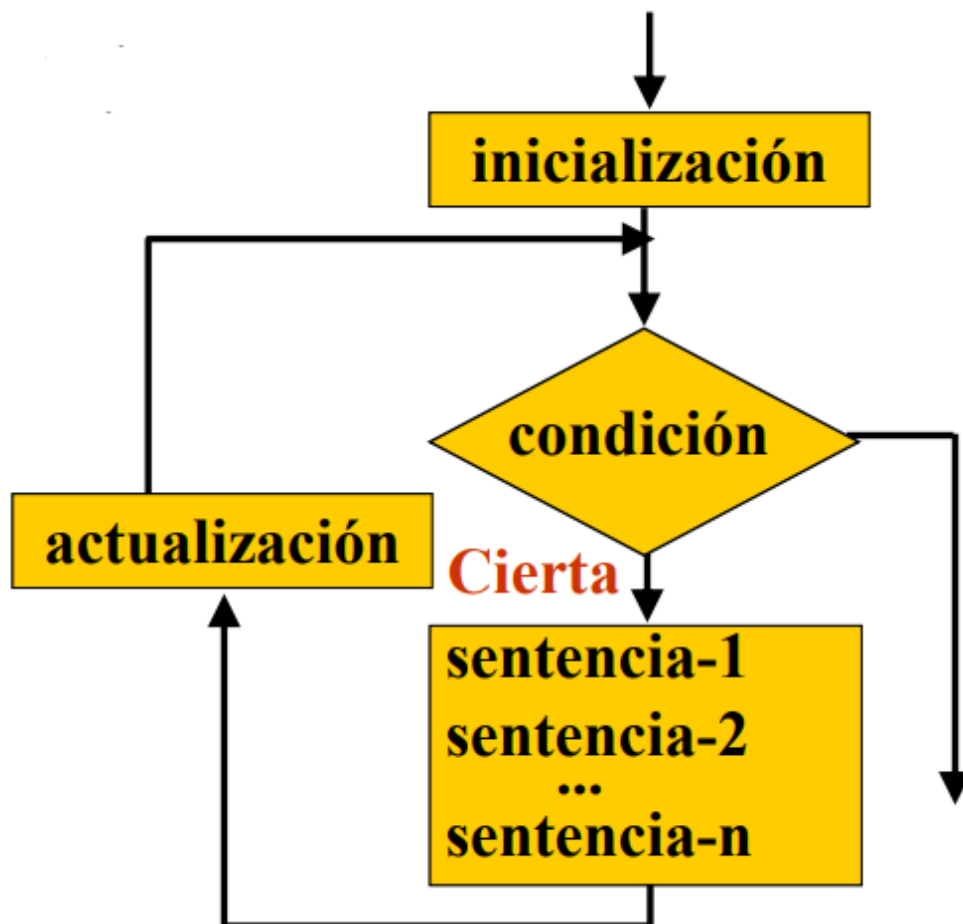
Las estructuras iterativas se denominan comúnmente bucles

Cada repetición de las sentencias de un bloque se denomina iteración. Toda estructura puede contener cualquier conjunto de instrucciones

1. The for Loop

Se empieza ejecutando la inicialización (INI) y evaluando la condición (CON)

Mientras la condición se evalúa a cierta se ejecuta el bloque de sentencias y se ejecuta la actualización o iteración. En muchas situaciones las iteraciones se realizan un número determinado de veces, normalmente se conoce de antemano cuántas veces se hará el proceso que se repite. En este tipo de iteración, siempre se necesita una variable que lleve el control de las veces que se repite el proceso, por esto se les conoce también como ciclos con variable de control



```
for(initialization; condition; iteration)
    statement;
```

For repeating a block, the general form is

```
for(initialization; condition; iteration)
{
    statement sequence
}
```

Ejemplo:

```
public class RaicesCuadradas {
    public static void main(String args[]) {
        int suma = 0;
        for (int num = 1; num < 100; num++) {
            suma += num;
        }
        System.out.println(suma);
        suma=(100*99)/2; //(n*(n-1))/2
        System.out.println(suma);
    }
}
```

Ejemplo de For al revés:

```
class DecrFor {
    public static void main(String args[]) {
        int x;
        for (x = 100; x > -100; x -= 5) {
            System.out.println(x);
        }
    }
}
```

Ejercicios:

- i. Dado un número entero de entrada, muestra su tabla de multiplicar.
- ii. Muestra en pantalla los 50 primeros múltiplos de 5.
- iii. Programa que obtiene una secuencia de 10 números negativos enteros (aleatorios) y muestra al final un mensaje indicando cuál es el menor de los números leídos.

1.2. Some Variations on the for Loop

Ejemplo:

```
class Comma {
    public static void main(String args[]) {
        int i, j;
        for (i = 0, j = 10; i < j; i++, j--)
        {
            System.out.println("i and j: " + i + " " + j);
        }
    }
}
```

Ejemplo:

```
import java.util.Scanner;
class ForTest {
    public static void main(String args[])
    {
        Scanner sc= new Scanner(System.in);
        int i;
        char letra;
        System.out.println("Press S to stop.");
        for (i = 0; (letra=sc.next().charAt(0)) != 'S'; i++)
        {
            System.out.println("Pass #" + i);
        }
    }
}
```


1.3. Missing Pieces

class Empty1 {

```
    public static void main(String args[]) {
        int i;
        for (i = 0; i < 10;) {
            System.out.println("Pass #" + i);
            i++; // increment loop control var
        }
    }
}
```

```
class Empty2 {
    public static void main(String args[]) {
        int i;
        i = 0; // move initialization out of loop
        for (; i < 10;) {
            System.out.println("Pass #" + i);
            i++; // increment loop control var
        }
    }
}
```

```
class Empty3 {
    public static void main(String args[]) {
        int i;
        int sum = 0;
        for (i = 1; i <= 3; sum += i++) ; //primero suma y luego actualiza
        System.out.println("Sum is " + sum);
    }
}
```



1.4. Declaring Loop Control Variables Inside the for Loop

```
for (int i = 1; i <= 5; i++)
{
    ...
}
```

Ejercicios:

- iv. Programa que lee 10 números e indica si todos los números leídos son iguales o no.
- v. Programa que lea 20 números y averigüe la media aritmética de todos ellos.
- vi. Programa que lee 40 números. Al finalizar mostrará el máximo introducido, y cuántas veces se repite dicho máximo.
- vii. Programa que lee 9 números, y en el caso en el que el primero introducido sea igual al último, nos muestra la media. En caso contrario, no se mostrará nada.
- viii. Programa que muestra todos los divisores de un número introducido. Entre los divisores no se incluirá el propio número.

-
- ix. Programa que muestre las tablas de multiplicar de los 10 primeros números naturales (números del 1 al 10)
 - x. Calcula el factorial de un número leído por teclado
 - xi. Calcula y muestra el factorial de los 10 primeros números naturales
 - xii. Programa que averigua si un número leído es primo o no.
 - xiii. Programa que muestra los números primos existentes entre el 1 y el 100.
 - xiv. Programa que muestra los 100 primeros número primos.