



En lenguajes de programación, las estructuras de control permiten modificar el flujo de ejecución de las instrucciones de un programa.

Con las estructuras de control se puede:

- De acuerdo a una condición, ejecutar un grupo u otro de sentencias (If-Then-Else)
- De acuerdo al valor de una variable, ejecutar un grupo u otro de sentencias (Select-Case)
- Ejecutar un grupo de sentencias mientras se cumpla una condición (Do-While)
- Ejecutar un grupo de sentencias hasta que se cumpla una condición (Do-Until)
- Ejecutar un grupo de sentencias un número determinado de veces (For-Next)

Todas las estructuras de control tienen un único punto de entrada y un único punto de salida. Las estructuras de control se puede clasificar en : secuenciales, iterativas y de control avanzadas. Esto es una de las cosas que permite que la programación se rija por los principios de la programación estructurada.

Los lenguajes de programación modernos tienen estructuras de control similares. Básicamente lo que varía entre las estructuras de control de los diferentes lenguajes es su sintaxis, cada lenguaje tiene una sintaxis propia para expresar la estructura.

Otros lenguajes ofrecen estructuras diferentes, como por ejemplo los comandos guardados.

Algunas estructuras de control en el lenguaje Java

Antecedentes

El término «estructuras de control», viene del campo de la ciencia computacional. Cuando se presentan implementaciones de Java para las estructuras de control, nos referimos a ellas con la terminología de la Especificación del lenguaje Java, que se refiera a ella como instrucciones.



Ejecución secuencial

Pero por lo general las instrucciones se ejecutan una después de la otra, en el orden en que están escritas, es decir, en secuencia. Este proceso se conoce como ejecución secuencial.

Transferencia de control

En Java, como en otros lenguajes de programación por excelencia como C y C++, el programador puede especificar que las siguientes instrucciones a ejecutarse tal vez no sea la siguiente en secuencia. Esto se conoce como transferencia de control. Hay que tener en cuenta que la instrucción goto es una palabra reservada pero no se utiliza ni se recomienda. Un programa bien estructurado no necesita de esta instrucción.¹

De selección

Las estructuras de control de selección, ejecutan un bloque de instrucciones u otro, o saltan a un subprograma o subrutina según se cumpla o no una condición.

Estructura de control

Las estructuras de control, denominadas también sentencias de control, permiten tomar decisiones y realizar un proceso repetidas veces. Se trata de estructuras muy importantes, ya que son las encargadas de controlar el flujo de un programa, según los requerimientos del mismo.

Selección if simple

Se trata de una estructura de control que permite redirigir un curso de acción según la evaluación de una condición simple, sea falsa o verdadera.

Si la condición es verdadera, se ejecuta el bloque de sentencias 1, de lo contrario, se ejecuta el bloque de sentencias 2.



```
IF (Condición) THEN
    (Bloque de sentencias 1)
ELSE
    (Bloque de sentencias 2)
END IF
```

Se pueden plantear múltiples condiciones simultáneamente, si se cumple la (Condición 1), se ejecuta (Bloque de sentencias 1) en caso contrario se comprueba la (Condición 2), si es cierta se ejecuta (Bloque de sentencias 2), y así sucesivamente hasta n condiciones, si ninguna de ellas es cumple se ejecuta (Bloque de sentencias else).

```
IF (Condición 1) THEN
    (Bloque de sentencias 1)
ELSEIF (Condición 2) THEN
    (Bloque de sentencias 2)

.....

ELSEIF (Condición n) THEN
    (Bloque de sentencias n)
ELSE
    (Bloque de sentencias ELSE)
```



```
END IF
```

Select-Case

Esta sentencia permite ejecutar una de entre varias acciones en función del valor de una expresión. Es una alternativa a *if then else* cuando se compara la misma expresión con diferentes valores.

- Se evalúa la expresión, dando como resultado un número.
- Luego, se recorren los «Case» dentro de la estructura buscando que el número coincida con uno de los valores.
- Es necesario que coincidan todos sus valores.
- Cuando se encuentra la primera coincidencia, se ejecuta el bloque de sentencias correspondiente y se sale de la estructura Select-Case.
- Si no se encuentra ninguna coincidencia con ningún valor, se ejecuta el bloque de sentencias de la sección «Case Else».

```
SELECT (Expresión)
  CASE Valor1
    (Bloque de sentencias 1)
  CASE Valor2
    (Bloque de sentencias 2)
  CASE Valor n
    (Bloque de sentencias n)
  CASE ELSE
    (Bloque de sentencias "Else")
```



```
END SELECT
```

Estructuras de control iterativas

Las estructuras de control iterativas o de repetición, inician o repiten un bloque de instrucciones si se cumple una condición o mientras se cumple una condición.

Do-While

Mientras la condición sea verdadera, se ejecutarán las sentencias del bloque.

```
DO WHILE (Condición)  
    (Bloque de sentencias)  
LOOP
```

que también puede expresarse:

```
WHILE (Condición)  
    (Bloque de sentencias)
```



```
WEND
```

Do-Until

Se ejecuta el bloque de sentencias, hasta que la condición sea verdadera

```
DO  
  (Bloque de sentencias)  
LOOP UNTIL (Condición)
```

For-Next

Artículo principal: Bucle for

La sentencia *For* da lugar a un lazo o bucle, y permite ejecutar un conjunto de sentencias cierto número de veces.

- Primero, se evalúan las expresiones 1 y 2, dando como resultado dos números.
- La variable del bucle recorrerá los valores desde el número dado por la expresión 1 hasta el número dado por la expresión 2.
- El bloque de sentencias se ejecutará en cada uno de los valores que tome la variable del bucle.



```
FOR (Variable) = (Expresión1) TO (Expresión2) STEP (Salto)
  (Bloque de sentencias)
NEXT
```

Estructuras anidadas

Las estructuras de control básicas pueden anidarse, es decir pueden ponerse una dentro de otra.

Estructura For-Next dentro de una estructura If-Then-Else

```
IF A > B THEN
  FOR X = 1 TO 5
    (Bloque de sentencias 1)
  NEXT
ELSE
  (Bloque de instrucciones 2)
END IF
```

Estructura If-Then-Else dentro de estructura For-Next



```
FOR x = 10 TO 20 STEP 2
  IF A == C THEN
    (Bloque de instrucciones)
  ELSE
    (Bloque de instrucciones)
  END IF
NEXT
```

Estructura For-Next que está dentro de estructura Do-While

```
DO WHILE A > 0
  FOR X = 1 TO 10
    (Bloque de instrucciones)
  NEXT
  A = A - 1
LOOP
```