

PROGRAMACIÓN ESTRUCTURADA EN C++

ESTRUCTURAS DE CONTROL

JONATHAN JEAN PAUL ARANCIBIA SANABRIA

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5
6     if(condicion)
7     {
8         //sentencias a ejecutar
9     }
10
11     return 0;
12 }
```

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5
6     if(condicion) //condicion puede ser cualquier expresion booleana
7     {
8         //sentencias a ejecutar en caso 'condicion' sea verdadero
9     }
10
11     return 0;
12 }
```

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5
6     if(condicion)
7     {
8         //sentencias a ejecutar en caso 'condicion' sea verdadero
9     }
10    else
11    {
12        //sentencias a ejecutar en caso 'condicion' sea falso
13    }
14    return 0;
15 }
```

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5
6     if(condicion) //condicion puede ser cualquier expresion booleana
7         //sentencia a ejecutar en caso 'condicion' sea verdadero
8     else
9         //sentencia a ejecutar en caso 'condicion' sea falso
10
11     //recordar que se pueden omitir las {} del 'if' y el 'else'
12     //siempre y cuando contengan una y solo una sentencia
13     return 0;
14 }
```

```
if(condicion_1)
{
    //sentencias a ejecutar solo en caso 'condicion_1' sea verdadero
}
else if(condicion_2)
{
    //sentencias a ejecutar solo en caso 'condicion_2' sea verdadero
}
else if(condicion_3)
{
    //sentencias a ejecutar solo en caso 'condicion_3' sea verdadero
}
else
{
    //sentencias a ejecutar solo en caso las anteriores condiciones sean falsas
}
```

Señále lo incorrecto con respecto a la estructura if-else.

- 1 El if necesita una condición (expresión booleana) para que, en caso se cumpla (sea verdadero), se ejecute el bloque de instrucciones correspondiente.
- 2 No es posible colocar un if dentro de otro if.
- 3 El bloque de instrucciones del if se ejecuta cuando la condición sea verdadera, y el bloque del else se ejecuta cuando sea falsa.

¿Es posible escribir un **if** sin un **else**?

1

Sí

2

No

¿Qué tipos de variable puede comprobar un **switch** ?

- 1 Solo enteros.
- 2 Enteros y reales.
- 3 Solo caracteres.
- 4 Caracteres y enteros.

¿Cuál es la diferencia entre el **if** y el **switch** ?

1

En el if se pueden analizar expresiones booleanas y en el switch solo se comprueba una variable.

2

En el if se comprueba solo una variable y en el switch solo se analiza una expresión booleana completa.

¿Para que se sirve el **break** dentro del **switch** ?

- 1 Hace que termine la estructura switch y deja de leer las siguientes instrucciones.
- 2 Es una palabra reservada que se usa para indicar el código que se ejecutará en caso ningún caso se cumpla.
- 3 Se usa para indicar los distintos casos del switch.

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5
6     while(condicion) //condicion puede ser cualquier expresion booleana
7     {
8         //sentencias a ejecutar mientras 'condicion' sea verdadero
9     }
10
11     return 0;
12 }
```

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5
6     for(inicializacion ; condicion ; variacion)
7     {
8         //sentencias a ejecutar siempre y cuando 'condicion' sea verdadero
9     }
10
11     return 0;
12 }
```

¿Es necesario inicializar una variable que será contador? ¿Por qué?

1

Sí es necesario inicializar una variable que será usada como contador porque el valor inicial de una variable es indeterminado.

2

No es necesario inicializar una variable que será usada como contador porque su valor inicial siempre es 0.

¿Cuándo es más conveniente usar la estructura **for**?

1

Cuando trabajamos con contadores y sabemos cuántas veces queremos que se repita el código, con una variable por ejemplo.

2

No es recomendable usar el ciclo for.

3

Cuando se trata de repetir un bloque de instrucciones mientras se cumpla una condición.

Señale el orden correcto en las partes que se divide la estructura for.

for(1ro parte ; 2da parte ; 3era parte)

1

Declaración e inicialización ; Comparación o límite ; Incremento o decremento

2

Comparación o límite ; Declaración e inicialización ; Incremento o decremento

3

Comparación o límite ; Incremento o decremento ; Declaración e inicialización

PROGRAMACIÓN ESTRUCTURADA EN C++

GRACIAS 😊

JONATHAN JEAN PAUL ARANCIBIA SANABRIA