

RESUMEN DE CONTENIDOS

Nº 8

ESTRUCTURAS DE CONTROL EN C++

SELECCIÓN

Observamos 2 tipos de estructuras de selección:

- condicional o de decisión o **if**
- selección múltiple o **switch**

If (Estructura condicional o de decisión)

La sintaxis en pseudocódigo sería:

```
SI condición
    ENTONCES  A
    SINO      B
FINSI
```

En C++:

```
if (expresión lógica)
    acción1;
else
    acción 2;
```

```
int c=0;
if ( c == 200 )
    c = c/2;
else c = 2*c;
```

Estructuras condicionales anidadas

En pseudocódigo:

SI condición1

ENTONCES A

SINO **SI** condición2

ENTONCES B

SINO C

FINSI

FINSI

En C++:

```
if (expresión_lógica 1)
{
    Acciones;
}
else if(expresión_lógica 2)
{
    Acciones;
}
else
{
    Acciones;
}
```

- Switch (Estructura de selección múltiple)

En pseudocódigo:

SEGUN E HACER

1: A

2: B

..

..

n: R

DEOTROMODO: U

FINSEGUN

Sintaxis	Ejemplo
<pre> switch (variable) { case valor1: acción_1; break; case valor2: acción_2; break; case valor3: acción_3; break; default: acción_m; } </pre>	<pre> switch (m) { case 1: m++; break; case 2: m=2*m; break; case 3: m = m / 2; break; default : m = 100; break; } </pre>

Ejemplo

```
switch ( m )
{
    case 1:
    case 4: m++;
    break;

    case 2: m=2*m;
    break;

    case 3: m = m / 2;
    break;

    default : m = 100;
    break;
}
```

La variable de control sólo puede ser variables de tipo **int** o **char**

ITERACIÓN

Observamos 2 tipos de estructuras de ITERACIÓN:

- While (Mientras)
- Do while (similar al Repetir)

- **WHILE (Mientras)**

MIENTRAS condición **HACER**

A

FINMIENTRAS

```
while (expresión lógica )
{
    acciones
}
```

```
int a=0;
while ( a<100 )
{
    cout << a<< "\n";
    a++;
}
```

- **DO-WHILE (puede asemejarse a la estructura Repetir)**

REPETIR

A

HASTAQUE condición

Do

{

acciones

} while (expresión lógica);

*Las acciones abarcadas por esta estructura se ejecutan repetidamente hasta que la expresión lógica arroje el resultado **cero** (falso).*

```
int b=0;
do
{
    b++ ;
    cout << b<< "\n" ;
} while ( b<100 );
```