

Ciclos sirve para:

for, do while, switch case, if else

### **Ciclo FOR:**

Ejecuta un bloque de instrucciones "n" veces, es recomendable la utilización de este ciclo cuando se conoce el numero de repeticiones de una tarea es decir cuando se sabe en numero de veces que se va a repetir algo

#### **Ejemplo:**

For(i=0, i < 10 , i=++)

Donde se establece como variable centinela la variable "i" y se inicia con 0, la condición de paro de ciclo es que "i" no sea mayor que 10 y el incremento o actualización de "i" es un incremento de 1 en 1.

### **Ciclo Do While:**

Los ciclos do-while son una estructura de control cíclica, los cuales nos permiten ejecutar una o varias líneas de código de forma repetitiva sin necesidad de tener un valor inicial e incluso a veces sin siquiera conocer cuando se va a dar el valor final, hasta aquí son similares a los ciclos while, sin embargo el ciclo do-while nos permite añadir cierta ventaja adicional y esta consiste que nos da la posibilidad de ejecutar primero el bloque de instrucciones antes de evaluar la condición necesaria, de este modo los ciclos do-while, son más efectivos para algunas situaciones específicas. En resumen un ciclo do-while, es una estructura de control cíclica que permite ejecutar de manera repetitiva un bloque de instrucciones sin evaluar de forma inmediata una condición

especifica, sino evaluándola justo después de ejecutar por primera vez el bloque de instrucciones

Ejemplo:

```
#include <stdio.h>
```

```
main() { /* visualizar los números del 0 al 9.*/ int digito=0;
```

```
do
```

```
    printf("%d ",digito++);while (digito<=9);
```

```
}
```

Visualizará: 0 1 2 3 4 5 6 7 8 9

Por tanto, se trata de repetir el cuerpo del bucle (grupo de sentencias), hasta que la condición se haga falsa (valor nulo), o mejor dicho, vamos a ejecutar el grupo de sentencias mientras la condición sea cierta (valor no nulo), entonces, como mínimo siempre vamos a ejecutar el cuerpo del bucle al menos una vez, ya que la condición de salida (<exp-condición>) la encontramos al final.

### **Switch Case:**

La estructura condicional switch ... case se utiliza cuando queremos evitarnos las llamadas escaleras de decisiones. La estructura if nos puede proporcionar, únicamente, dos resultados, uno para verdadero y otro para falso. Una estructura switch ... case, por su parte, nos permite elegir entre muchas opciones. Las palabras reservadas en C para la condicional switch ... case son:

switch

case

default

Y necesita de una instrucción más, no perteneciente a la programación estructurada en sí, pero que en este caso es necesaria para esta estructura:

break

### **Ejemplo:**

```
int a;
```

```
switch(a)
```

```
{
```

```
case 0:
```

```
    //operaciones a realizar en caso de que a valga 0;
```

```
    break;
```

```
case 1:
```

```
    //mismo proceso
```

```
    break;
```

```
...
```

```
case n:
```

```
    //n tiene que ser una constante, o numérica {0,1,2 ....} o  
    una definida #define n 2
```

```
    break;
```

default:

```
//en caso de que no se de ninguna de las anteriores  
hacer...
```

```
}
```

esto seria equivalente a:

```
int a;
```

```
if(a==0)
```

```
//operaciones para 0
```

```
else if(a==1)
```

```
//operaciones para 1
```

```
else if(a==n)
```

```
//operaciones para n
```

```
else
```

```
//cualquier otro caso
```

If Else:

La estructura if / else lo que hace es ejecutar una acción si el resultado de la evaluación de la expresión es verdadera y otra acción si el resultado de la evaluación es falsa.

La diferencia con utilizar sólo la estructura if es que si la expresión evaluada es verdadera sólo en ese caso se ejecuta una acción de otro modo se pasa de largo. En cambio en la estructura if / else si la expresión es falsa entonces se ejecuta otra acción.

if (expresión(es))

else (sentencias)

En síntesis lo que hace esta estructura es realizar una acción si la expresión es verdadera y otra si es falsa.

### Ejemplo:

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
if ( edad > 20 )
```

```
    cout<<"Eres mayor de edad" ;
```

```
else
```

```
    cout<<"No eres mayor de edad";
```

```
}
```

El diagrama de flujo correspondiente a esta estructura es el siguiente:

