



## Sentencias de control (2)

### La sentencia while

Esta sentencia permite la repetición de una sentencia mientras una condición sea verdadera. Así que ésta es su estructura:

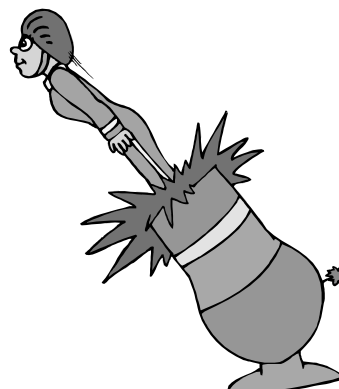
```
while ( Condición )  
    Sentencia
```

**Condición** es una condición lógica y **Sentencia** puede ser una sentencia simple, compuesta o vacía. Primeramente se evalúa **Condición**; si es cierta, se ejecuta **Sentencia** y se vuelve a evaluar **Condición**, siguiendo de ese modo hasta que **Condición** sea falsa. Pero si **Condición** es falsa la primera vez, **Sentencia** no se ejecuta nunca.

### Ejemplo

El siguiente programa pide un número al usuario y le va calculando la raíz cuadrada mientras su diferencia con 1 sea mayor que una milésima.

```
#include <stdio.h>    /* printf() scanf() */  
#include <math.h>    /* sqrt() */  
  
int main (void)  
{  
  
    int  Numero;  
    float Raiz;  
  
    printf ("Escribe un número: ");  
    scanf ("%d", &Numero);  
    Raiz = Numero;  
  
    while ( Raiz-1 >= 0.001 )  
    {  
        Raiz = sqrt (Raiz);  
        printf ("Raíz: %f\n", Raiz);  
    }  
  
    return 0;  
}
```



### La sentencia do

Esta sentencia es muy similar a la sentencia **while**. La diferencia es que la condición se evalúa después de ejecutar la sentencia, con lo que se garantiza que ésta se ejecutará al menos una vez, mientras que en una sentencia **while** es perfectamente posible que la sentencia no se ejecute ni una sola vez. Ésta es la estructura de la sentencia **do**:

```
do Sentencia  
while ( Condición );
```

## Ejemplo

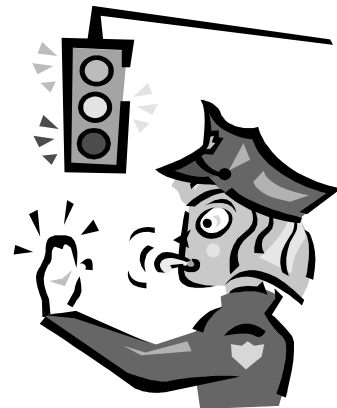
El programa que se muestra a continuación pide números al usuario y va mostrando sus cuadrados hasta que el usuario introduce el número 0.

```
#include <stdio.h> /* printf() scanf() */

int main (void)
{
    int Numero, Cuadrado;

    do
    {
        printf ("Escribe un número: ");
        scanf ("%d", &Numero);
        Cuadrado = Numero * Numero;
        printf ("Su cuadrado es %d\n", Cuadrado);
    } while ( Numero != 0 );

    return 0;
}
```



## La sentencia for

Esta es la sentencia que en los lenguajes de programación permite repetir un número determinado de veces una porción del código. En C es especialmente potente, y permite gran expresividad. Su formato general es éste:

```
for ( SentenciaInicial ; Condición ; SentenciaIncremento )
    Sentencia
```

Y se ejecuta de esta manera:

1. Se ejecuta **SentenciaInicial**.
2. Se comprueba **Condición**.
3. Si **Condición** es cierta, se ejecutan **Sentencia** y **SentenciaIncremento** (por ese orden) y se vuelve al paso 2.
4. Si **Condición** es falsa, concluye la ejecución de la sentencia **for**.

## Ejemplo

El programa que aparece ahora pide al usuario dos números y muestra los cuadrados de todos los números que se encuentran entre ellos, ambos incluidos.

```
#include <stdio.h> /* printf() scanf() */

int main (void)
{
    int i, NumeroInicial, NumeroFinal, Cuadrado;

    printf ("Escribe el primer número: ");
    scanf ("%d", &NumeroInicial);
    printf ("Escribe el segundo número: ");
    scanf ("%d", &NumeroFinal);

    for ( i = NumeroInicial ; i <= NumeroFinal ; i++ )
    {
        Cuadrado = i * i;
        printf ("El cuadrado de %d es %d\n", i, Cuadrado);
    }

    return 0;
}
```

