



Funciones

Para qué sirven

Hasta el momento todos los programas que se han presentado constan de una única función, la función `main()`. En principio, sería posible escribir todo el código del programa dentro de ella, pero en la práctica pronto se descubre que hay fragmentos complicados de código que hay que repetir muy a menudo. Por tanto, es mejor escribir el fragmento que se repite una sola vez, e invocar su funcionamiento donde sea necesario. Ese fragmento, ahora independiente del flujo general del programa, es lo que se llama **función**.

Las funciones pueden devolver un resultado o no. En otros lenguajes de programación se llama **procedimientos** a las funciones que no devuelven valores. En C no hay tal distinción.

Manejo general

Se distinguen tres fases en la utilización de una función:

1. **Declaración de la función.** Se especifica qué tipos de datos recibirá la función y qué tipo devolverá. Las funciones se declaran escribiendo su **prototipo** antes de usarla por primera vez.
2. **Definición de la función.** En cualquier parte del código, siempre después de la declaración, se define la función escribiendo las sentencias que la forman.
3. **Invocación de la función.** Cuando se quiere activar la función basta escribir su nombre y especificar los datos con los que debe trabajar.

Declaración

Los valores que reciben las funciones para comenzar su trabajo se llaman **parámetros**. Pueden ser desde ninguno hasta una cantidad variable. En el prototipo de la función hay que escribir el tipo de dato que devolverá, el nombre de la función y la lista de los tipos de datos de todos los parámetros, separados por comas; opcionalmente, se pueden escribir los nombres de los parámetros. Si la función no devuelve ningún valor, se especifica el tipo **void**; y si no recibe ningún parámetro, la lista se sustituye por **void**.

Ejemplo

El prototipo de una función llamada **Lineal** que reciba tres parámetros **float** y devuelva un resultado **float** es éste:

```
float Lineal (float, float, float);
```

Definición

La definición de la función se realiza escribiendo el tipo de dato que devolverá, el nombre de la función y la lista de los tipos de datos y nombres de todos los parámetros, separados por comas; tras esto, entre llaves, se escribe el código de la función.

Si la función devuelve un valor, se usa una sentencia con **return** y el valor devuelto. Es posible usar varios **return** en una misma función, aunque se desaconseja hacerlo.

Ejemplo

Para definir una función que devuelva el resultado de una función lineal $y=ax+b$, para cada valor que se le dé de a , b y x , se escribe esto:

```
float Lineal (float a, float b, float x);  
{  
    return a * x + b;  
}
```

Invocación

Esta es la parte más sencilla: basta escribir el nombre de la función y los valores de los parámetros (si los hay), separados por comas. Si se desea usar el valor devuelto por la función, puede asignarse a una variable o usarlo en otra función; no es obligatorio usar el valor devuelto.

Ejemplo

Para asignar a la variable y el valor calculado por la función **Lineal** para $a=2$, $b=1.5$ y $x=0.3$ se escribe esta sentencia:

```
y = Lineal (2.0, 1.5, 0.3);
```

Un ejemplo completo

Para ver la utilidad de las funciones ahora se va a desarrollar un pequeño programa de ejemplo. El programa permite pasar de grados Celsius a grados Fahrenheit y viceversa. El programa se llamará **ConvCF** y se invocará **ConvCF x c** para convertir x grados Celsius a Fahrenheit y **ConvCF x f** para convertir x grados Fahrenheit a Celsius.

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

float Lineal (float, float, float);

int main (int argc, char *argv[])
{
    float Dato, Resultado;
    char Orden;

    Dato = atof (argv[1]);
    Orden = argv[2][0];

    if ( Orden == 'c' )
    {
        Resultado = Lineal (1.8, 32.0, Dato);
        printf ("%0.2f grados Celsius son %0.2f grados Fahrenheit.\n",
            Dato, Resultado);
    }

    if ( Orden == 'f' )
    {
        Resultado = Lineal (0.555556, -17.7778, Dato);
        printf ("%0.2f grados Fahrenheit son %0.2f grados Celsius.\n",
            Dato, Resultado);
    }

    return 0;
}

float Lineal (float a, float b, float x)
{
    return a * x + b;
}
```

