

# PROGRAMACIÓN EN LENGUAJE C

MANEJO DE FUNCIONES

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

La utilización de funciones nos permite dividir un programa extenso en pequeños segmentos que realizan tareas concretas.

Esto propicia:

- \* La facilidad de realizar las mismas tareas varias veces.
- \* Mejorar la legibilidad del programa.

Una vez iniciada la ejecución del programa, desde la función *main()* se puede llamar a otras funciones y, también, desde estas funciones a otras.

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

Cuando se llama a una función, se ejecutan las instrucciones que constituyen dicha función. Una vez que se ejecutan las instrucciones de la función, se devuelve el control del programa a la siguiente operación (si existe) inmediatamente después de la que provocó la llamada a la función.

### Funciones

De biblioteca. Funciones predefinidas en el lenguaje

Del usuario. Definidas por el programador

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

### Estructura general de una función en C

```
tipo_dato_retorno nombreFunción( lista_parámetros)  
{  
     cuerpo_de_la_función  
    return expresión  
}
```

*tipo\_dato\_retorno* representa el tipo de dato del valor que devuelve la función. De forma predeterminada, se considera que toda función devuelve un tipo entero (int). Si la función no regresa ningún valor se usa la palabra reservada *void*.

*nombreFunción* representa el nombre que se le asigna a la función.

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

*lista\_parámetros*: es la lista de declaración de los parámetros que son pasados a la función. Éstos se separan por comas. Debemos tener en cuenta que pueden existir funciones que no utilicen parámetros.

*cuerpo\_de\_la\_función*: está compuesto por un conjunto de sentencias que llevan a cabo la tarea específica para la cual ha sido creada la función.

*return expresión*: mediante la palabra reservada *return* se devuelve el valor de la función, en este caso representado por *expresión*. Si la función no regresa ningún valor, esta sentencia se puede omitir.

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

### Ejemplo 1

```
#include <stdio.h>

void letrero(){
    printf("*****\n");
    printf("    Bienvenido al curso de    \n");
    printf("    programación en lenguaje C  \n");
    printf("*****\n");
}

int main(){
    letrero();
}
```

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

### Ejemplo 2

```
/*Función que calcula el cubo de un número entero */  
long cubo(int base)  
{  
    int potencia;  
    potencia = base * base * base;  
    return potencia;  
}
```

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

### Ejercicio 1

Elaborar un programa que calcule el cubo de los primeros 10 números naturales. Definir una función que calcule el cubo de un número.

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

### Declaración o prototipo de una función.

Se emplean para indicar qué funciones utilizará el programa, ya sea definidas por el usuario o de biblioteca. Además de esta forma no importa en dónde se ubique la definición de la función.

La declaración de una función se conoce también como prototipo de la función. En el prototipo de una función se tienen que especificar los parámetros de la función, así como el tipo de dato que devuelve.

Se colocan fuera de las funciones.

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

### Sintaxis

*tipo\_dato \_retorno nombreFunción( lista\_parámetros);*

Se puede observar que el prototipo de una función es la primera línea o *cabecera* de la definición de la función. De aquí que los archivos que contienen los prototipos de funciones de biblioteca tienen la extensión *.h* (de header). Por ejemplo *stdio.h*

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015

# PROGRAMACIÓN EN LENGUAJE C

## MANEJO DE FUNCIONES

### Ejercicio 2

Elaborar un programa que calcule el número de combinaciones de “m” elementos, formando subgrupos de “n” elementos.

Para calcular el número de combinaciones se aplica la siguiente fórmula:

$$C_{m,n} = \frac{m!}{n! * (m - n)!}$$

Elba Karen Sáenz García  
Laura Sandoval Montaña

Enero/2015