Autoestudio 6: Procedimientos

Un procedimiento es un tipo de función que no devuelve un valor, es decir, una función que **no tiene un** **return**. El tipo de dato de declaración de un procedimiento es **void** que significa vacío.

# Definición de un Procedimiento

Un procedimiento se define igual que una función, con la diferencia de que siempre será tipo **void** y no tendrá return.

**void imprime\_edad (int a)**

**{**

**printf(“hola tengo %i años”, a);**

**}**

# Utilizando/llamando un procedimiento

Un procedimiento se llama igual que una función, sin embargo como no devuelve un resultado entonces no podemos asignarle a variables.

En un programa se podría mandar a llamar el procedimiento de las siguientes formas:

**int edad = 10;**

**imprime\_edad (8);**

**imprime\_edad(edad);**

**imprime\_edad (8\*2+edad);**

El manejo de parámetros en es igual que en las funciones, es decir un procedimiento puede recibir cualquier número de parámetros o ningún parámetro.

Normalmente utilizamos procedimientos cuando no necesitamos devolver ningún dato o cuando trabajamos con arreglos (tema futuro). Por ejemplo para desplegar mensajes grandes y repetitivos al usuario, o cuando deseamos realizar una modificación a los valores de algunas variables o estructuras.

# Variables

Cuando trabajamos con procedimientos a veces utilizamos variables de tipos globales, estas son variables que pueden ser utilizadas en todo el programa, en cualquier función estas variables se definen en la parte superior del programa, después de los *includes* y antes de la declaración de cualquier función o procedimiento y del *main*.

# Si deseas saber más puedes consultar, donde se explican los temas a mayor detalle:

(Joyanes Aguilar, L. y Zahonero Martínez, I. *Programación en C, C++, Java y UML*. México : McGraw Hill, 2010.) las siguientes secciones:

* Concepto de función. (pp. 190)
* Estructura de una función. (pp. 191)