

SEMANA 12 - PRÁCTICA

Funciones.

Parámetros.

Paso por valor y paso por referencia.

Parámetros de tipo array.

Realizar un programa en C para cada uno de los siguientes enunciados:

1. Crear un programa en C que dé la bienvenida al usuario solicitándole su inicial y mostrando un mensaje que incluya dicha inicial. Para llevar a cabo esta tarea, desarrolla una función llamada `mensajeBienvenida`, que reciba como argumento la inicial ingresada por el usuario. Esta función deberá imprimir un mensaje de bienvenida en la consola, incluyendo la inicial proporcionada como parámetro. En el programa principal (`main`), se solicitará al usuario que ingrese su inicial y luego se llamará a la función `mensajeBienvenida`, pasándole la inicial ingresada como argumento.
2. Crear una función llamada `calcularAreaCuadrado` que reciba el lado de un cuadrado como parámetro y devuelva el área de ese cuadrado. Luego, desarrollar un programa que invoque a esta función y muestre el resultado.
3. Crea una función llamada `esPar` que reciba un número entero como parámetro y devuelva 1 si el número es par, o 0 si es impar. Después, crea un programa que utilice esta función y muestre los mensajes en consola "El número ingresado es par" o "El número ingresado es impar", según el valor retornado.

Para los siguientes ejercicios también desarrollar un programa que invoque dichas funciones...

4. Crear una función que calcule el cubo de un número real (`float`). El resultado deberá ser otro número real.
5. Crear una función llamada "signo", que reciba un número real y devuelva un número entero con el valor: -1 si el número es negativo, 1 si es positivo, o 0 si es cero.
6. Crear una función que devuelva la primera letra de una cadena de texto.
7. Crear una función "esPrimo", que reciba un número y devuelva el valor 1 si es un número primo, o 0 en caso contrario.

8. Crear una función que reciba un número cualquiera y devuelva como resultado la suma de sus dígitos.
9. Crear una función que reciba un caracter y un número, y escriba un “triángulo” formado por ese caracter, que tenga como anchura inicial la que se ha indicado. Por ejemplo, si la caracter es * y la anchura es 4, debería escribir:

**
*
 10. Realizar un programa que llame a una función que pida al usuario ingresar el nombre, el apellido y la edad de una persona. Luego llamar a otra función que muestre por pantalla los datos de la persona.
 11. Realizar un programa que le pida al usuario que ingrese dos números, luego llamar a una función que calcule la suma de ambos y la devuelva al programa principal. Desde el programa principal, mostrar el resultado de la suma.
 12. Realizar un programa que le pida al usuario ingresar los 3 lados de un triángulo e informe si es equilátero, isósceles o escaleno. Esto debe hacerse llamando a una función que reciba los lados como parámetros de entrada y muestre por pantalla el tipo de triángulo (Ejemplo: “El triángulo ingresado es un escaleno”).
 13. Realizar un programa que permita cargar los datos de un cliente: Nombre, Apellido, DNI.
La carga del DNI debe validarse a partir de una función llamada ValidarDNI, la cual recibirá como parámetro de entrada el DNI ingresado y devolverá un valor de tipo int (0 o 1).
Nota: El DNI debe estar formado por números y tener una longitud mínima de 6 caracteres y máxima de 8.
 14. Realizar un programa que le pida al usuario que ingrese una lista de 10 alumnos y los resultados del examen final. Luego el programa llamará a:
 - una función que calcule el promedio de notas,
 - otra función que calcule la mayor nota,
 - y otra que calcule la menor nota.Luego el sistema informará estos tres valores al usuario.

15. Dada la siguiente lista de corredores de karting y el resultado de cada uno:

Corredor	Posición final
Pedro Gomez	7
Martín Ceres	6
Ernestina Diaz	1
Gastón Bautista	2
Dalma Alvarez	8
Patricio Fernandez	3
Faustina Martinez	10
Andrea Perez	9
Hilda Lopez	4
León García	5

Realizar un programa que muestre la siguiente opción de menú al usuario:

- 1 - Mostrar lista de corredores
- 2 - Mostrar los tres mejores
- 3 - Mostrar lista completa de resultados
- 4 - Salir

La opción 1 se llevará a cabo mediante una función llamada ListaCorredores que mostrará al usuario los nombres y apellidos de los corredores.

La opción 2 llama a otra función llamada ListaMejores que mostrará a los corredores que salieron en las primeras 3 posiciones.

La opción 3 llamará a una función llamada ListaPosiciones que mostrará la lista de corredores con sus posiciones.

La opción 4 es la salida del sistema.

Nota: La lista de corredores y los resultados de la carrera se guardarán en un arreglo.

El sistema debe volver a mostrar el menú de opciones una vez que se complete cada solicitud del usuario, y hasta tanto el usuario ingrese la opción de salida.