

MAS EJERCICIOS

1) Escriba un programa en C que permita al usuario cargar 12 números enteros (entre 1 y 100) en un arreglo. Luego, realice las siguientes tareas utilizando funciones:

- Una función para cargar el arreglo
- Una función para calcular el promedio
- Una función debe contar cuántos números son mayores al promedio de todos los números ingresados y retornar ese valor.
- Otra función debe imprimir solo los números pares.

Requisitos:

- Utilizar al menos dos funciones además del main.
- Validar que los números ingresados estén en el rango de 1 a 100. (El programa debe volver a pedirlo si se va de rango)
- No usar variables globales.

Pista para validar:

Usás un do...while porque querés **pedir al menos una vez**.
Comprobás si num está **fuera del rango permitido**.
Si está mal, mostrás un mensaje y lo pedís de nuevo.

2) Escribí un programa en C que permita al usuario ingresar **8 números enteros de tres cifras** (entre 100 y 999). Luego, utilizando funciones:

1. Cargar el arreglo con los 8 números, **validando que estén entre 100 y 999**.
2. Contar cuántos de esos números tienen **al menos un dígito repetido**.
3. Calcular y mostrar la **suma total de todos los dígitos** de los 8 números.
4. Mostrar en pantalla aquellos números que tienen **todos sus dígitos diferentes**.

No usar variables globales.

Usar al menos **2 funciones además del main()**.

Validar el rango de entrada correctamente.

Pistas

- Para **obtener los dígitos** de un número de 3 cifras, podés usar divisiones/mod (ej: centena = num / 100, decena = (num / 10) % 10, unidad = num % 10).

3) Escribí un programa en C que permita al usuario ingresar **10 números enteros entre 1 y 1000**. Luego, utilizando funciones:

1. Cargar los 10 números en un arreglo, **validando que estén en el rango de 1 a 1000**.
2. Mostrar en pantalla únicamente los **números primos** que fueron ingresados.
3. Contar cuántos de los números ingresados son **múltiplos de 5**.
4. Calcular el **promedio de los números que no son primos ni múltiplos de 5**.

No usar variables globales.

Usar al menos **2 funciones además del main()**.

Validar correctamente la entrada.

Pistas:

- Para verificar si un número es **primo**, hacé una función tipo `int esPrimo(int num)` que devuelva 1 si es primo, 0 si no. Los números primos son aquellos que **SOLO** pueden dividirse por 1 y por si mismos (división entera).