

# TP2-2024

---

El objetivo de esta práctica es que comiencen a plantear sus primeras funciones en C, por lo que es importante recordar que la entrada y salida va en el `main` y no en la función que están implementado.

## Recomendaciones para la resolución

---

- Cada punto debe ser entregado en un archivo independiente.
- El nombre de cada archivo debe ser `ejercicio` seguido del número de ejercicios más `.c`, de esta manera, el primer ejercicio será entonces `ejercicio1.c`.
- Recuerden tener en cuenta las **Cuestiones de Estilo**
- No podrán usar variables globales. Toda la información necesaria para el funcionamiento de las funciones a desarrollar tienen que ser pasado como argumentos de las mismas.
- Mantengan separado lo que es entrada, del algoritmo y la salida.
- Conviene documentar las funciones implementadas indicando el propósito de los argumentos y que es lo que retorna.

## Ejercicios

---

Todos los ejercicios a continuación, requieren la creación de un `main` que permita utilizar la función a implementar.

### 1. Contador de dígitos

Escribe una función que cuente y muestre en pantalla la cantidad de dígitos en un número entero positivo.

### 2. Suma de dígitos

Diseña e implementa una función que calcule la suma de los dígitos de un número entero positivo pasado como parámetro.

### 3. Tabla de Multiplicar

Desarrolla una función que reciba un número entero como parámetro y muestre en pantalla su tabla de multiplicar del 1 al 10.

### 4. Mayor de Tres Números

Diseña una función que tome tres números enteros como argumentos y devuelva el mayor de ellos.

## 5. Menor de Tres Números

Escribe una función similar a la anterior, pero que devuelva el menor de tres números enteros dados.

## 6. Suma de Números Primos

Crea una función que calcule la suma de los primeros N números primos positivos.

## 7. Potencia lenta

Desarrolla una función que calcule la potencia de un número elevado a una potencia dada, sin utilizar el operador de multiplicación.

## 8. Número defectivo

Escribe una función que determine si un número entero positivo es un número defectivo.

Un número defectivo o número deficiente es un número  $n$  para el que la suma de sus divisores es menor que  $2n$ . De manera equivalente, es un número para el que la suma de sus divisores propios (o suma alícuota) es menor que  $n$ .

Los primeros números defectivos se pueden consultar en:

<https://oeis.org/A005100>

## 9. Es Fibonacci?

Escribe una función que indique si un número pertenece o no a la serie de Fibonacci.

## 10. Cálculo de Suma de Dígitos en Serie de Fibonacci

Escribe una función que calcule la suma de los dígitos de los términos en una serie de Fibonacci hasta el N-ésimo término.

## 11. Freedom units

Para completar la conversión de grados Fahrenheit del TP1, ahora agregaremos la implementación de las otras 'freedom units'

Implementen las conversiones en ambos sentidos de:

Conversión de Litros a Galones: Implementa una función que convierta un volumen en litros a su equivalente en galones.

Conversión de Millas a Kilómetros: Diseña una función que convierta una distancia en millas a su equivalente en kilómetros.

## 12. Números perfectos

Escribir una función que indique si un número es perfecto.

Se dice que un número es perfecto cuando es igual a la suma de sus divisores propios menos sí mismo. Por ejemplo, el 15 no es un número perfecto, pero el 28 sí lo es.

Más detalle en [wikipedia](#) y el [OEIS](#).

## 13: Arreglos

Empleen el valor de `return` como `int` para indicar potenciales fallos, los ítems de este ejercicio deben estar implementados en un solo archivo.

### 13.1 Cargador de arreglos

Desarrollar una función que dado un arreglo de números enteros, y la cantidad de elementos, solicite valores por teclado para llenar las posiciones.

```
int cargar_arreglo(int capacidad, int arreglo[]);
```

### 13.2 Impresora de arreglos

Desarrollar una función que permita mostrar el contenido de un arreglo.

```
int muestra_arreglo(int capacidad, int arreglo[]);
```

### 13.3 Operaciones con arreglos

Implementar funciones que permitan hacer las siguientes operaciones sobre arreglos de números enteros.

- {13.3.1} Obtener la suma.
- {13.3.2} Obtener el promedio, como `double`.
- {13.3.3} Obtener el el valor mínimo del arreglo.

- {13.3.4} Obtener la posición del valor máximo.