# **ALGORITMOS**

**CURSO:** SOFTWARE

ALUMNO: ARIEL CALDERÓN CUEVA

DOCENTE: MSc. ENRIQUE BAÑO LEÓN

### **FUNCIONES EN LENGUAJE C**

En C, las funciones son bloques de código que realizan una tarea específica y pueden ser llamadas desde otros lugares en el programa. Las funciones en C se definen con un encabezado que especifica el tipo de retorno, el nombre de la función y los parámetros que toma (si los hay), seguido por el cuerpo de la función que contiene las instrucciones a ejecutar.

## Ejemplo: Función para Calcular el Factorial

```
1 #include <stdio.h>
                                                                                               /tmp/AtaAnBR5no.o
                                                                                              El factorial de 5 es 120
3 // Declaración de la función factorial
4 int factorial(int n):
5
6 * int main() {
      int num = 5;
   // Llamada a la función factorial
int resultado = factorial(num);
9
10
11
12 // Impresión del resultado
13
      printf("El factorial de %d es %d\n", num, resultado);
14
15
      return 0;
16 }
17
18 // Definición de la función factorial
19 * int factorial(int n) {
20 * if (n == 0 || n == 1) {
21
          return 1;
22* } else {
   return n * factorial(n - 1);
23
24
```

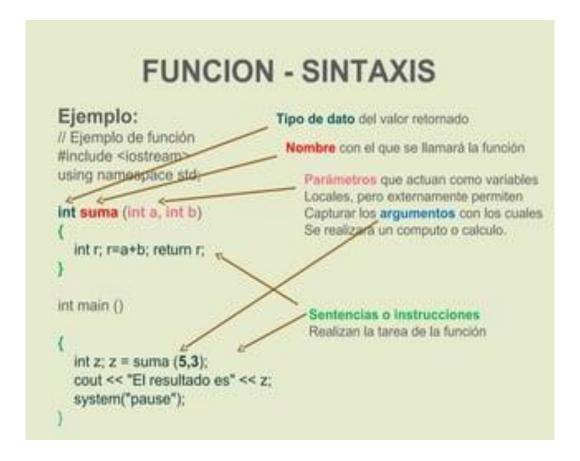
Este programa define una función llamada factorial que calcula el factorial de un número dado. La función se llama desde la función **main**, y el resultado se imprime en la consola. La recursividad se utiliza en la función factorial para calcular el factorial de manera eficiente.

## Ejemplo: Función para Intercambiar Valores

```
1 #include <stdio.h>
                                                                                         /tmp/AtaAnBR5no.o
                                                                                         Antes del intercambio: x = 5, y = 10
3 // Declaración de la función para intercambiar valores
                                                                                         Después del intercambio: x = 10, y = 5
4 void intercambiar(int *a, int *b);
5
6 * int main() {
      int x = 5, y = 10;
8
     // Antes del intercambio
9
   printf("Antes del intercambio: x = %d, y = %d\n", x, y);
10
11
12 // Llamada a la función para intercambiar valores
13 intercambiar(&x, &y);
14
15 // Después del intercambio
16 printf("Después del intercambio: x = %d, y = %d\n", x, y);
17
18
      return 0;
19 }
20
21 // Definición de la función para intercambiar valores
22 * void intercambiar(int *a, int *b) {
23    int temp = *a;
24 *a = *b;
25 *b = temp;
26 }
```

En este ejemplo, se define una función llamada intercambiar que toma dos punteros a enteros como parámetros y realiza un intercambio de los valores a los que apuntan. La función se llama desde la función **main**, y después del intercambio, se imprime el resultado en la consola.

Estos ejemplos ilustran cómo definir y utilizar funciones en C para realizar tareas específicas y modularizar el código.



#### Bibliografía:

- Logica de programacion- https://www.youtube.com
- <u>Funciones en C https://platzi.com</u>
- Funciones en Lenguaje C https://disenowebakus.net
- Estructuras de control en C https://www.youtube.com