

# Parcial 1 número

Axel Daniel, Ramirez Urbina, 202010083

*Escuela de Mecánica Eléctrica, Facultad de Ingeniería,  
Universidad de San Carlos de Guatemala*

## A. Resumen

El programa en lenguaje C diseñado permite al usuario ingresar un número y determina si es positivo, negativo o cero. Además de esta funcionalidad principal, ofrece opciones adicionales como el almacenamiento de los datos en un archivo de texto llamado "salida.txt", acceso al historial de datos, la posibilidad de borrar dicho historial y salir del programa.

El programa comienza solicitando al usuario que ingrese su nombre para personalizar los mensajes. Luego, presenta un menú interactivo con las siguientes opciones:

- Ingreso ejecución programa:** Permite al usuario ingresar un número, y el programa determina su clasificación (positivo, negativo o cero). El resultado se muestra en pantalla y se guarda en el archivo "salida.txt".
- Historial de datos:** Muestra en pantalla el contenido del archivo "salida.txt", que contiene el historial de los números ingresados y su clasificación.
- Borrado de datos:** Elimina el contenido del archivo "salida.txt", limpiando así el historial de datos.
- Salir:** Finaliza la ejecución del programa, mostrando un mensaje de despedida personalizado con el nombre del usuario.

El programa utiliza estructuras de control, funciones y manejo de archivos para implementar las funcionalidades de manera modular y ordenada. Se incluyen mensajes informativos para guiar al usuario durante su interacción con el programa, y se manejan posibles errores de ejecución para mejorar la experiencia del usuario.

## B. Código

```
C numero.c > ...
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4
5  // Estructura para almacenar los datos del usuario
6  struct Usuario {
7      char nombre[50];
8  };
9
10 // Función para ingresar el nombre del usuario
11 void ingresarNombre(struct Usuario *usuario) {
12     printf("Ingrese su nombre: ");
13     scanf("%s", usuario->nombre);
14 }
15
16 // Función para determinar si un número es positivo, negativo o cero
17 void determinarNumero() {
18     float numero;
19     printf("\nIngrese un numero: ");
20     scanf("%f", &numero);
21
22     // Determinar si el número es positivo, negativo o cero
23     if (numero > 0) {
24         printf("El numero %.2f es positivo.\n", numero);
25     } else if (numero < 0) {
26         printf("El numero %.2f es negativo.\n", numero);
27     } else {
28         printf("El numero es cero.\n");
29     }
30
31     // Guardar la clasificación en el archivo de salida
32     FILE *archivo = fopen("salida.txt", "a");
33     if (archivo != NULL) {
34         fprintf(archivo, "Numero: %.2f\n", numero);
35         fclose(archivo);
36     } else {
37         printf("Error al guardar la clasificacion del numero.\n");
38     }
39 }
40
41 // Función para mostrar el historial de datos
42 void mostrarHistorial() {
43     printf("\n--- Historial de Datos ---\n");
44
45     FILE *archivo = fopen("salida.txt", "r");
46     if (archivo != NULL) {
47         char linea[100];
48         while (fgets(linea, sizeof(linea), archivo)) {
```

```

49         printf("%s", linea);
50     }
51     fclose(archivo);
52 } else {
53     printf("No hay historial de datos.\n");
54 }
55 }
56
57 // Función para borrar el historial de datos
58 void borrarHistorial() {
59     if (remove("salida.txt") == 0) {
60         printf("\nHistorial de datos borrado exitosamente.\n");
61     } else {
62         printf("\nError al borrar el historial de datos.\n");
63     }
64 }
65
66 int main() {
67     int opcion;
68     struct Usuario usuario;
69
70     printf("¡Bienvenido al programa de clasificacion de numeros!\n");
71
72     // Ingresar el nombre del usuario al inicio del programa
73     ingresarNombre(&usuario);
74
75     do {
76         printf("\nMenu:\n");
77         printf("1. Ingresar numero y determinar clasificacion\n");
78         printf("2. Ver historial de datos\n");
79         printf("3. Borrar historial de datos\n");
80         printf("4. Salir\n");
81         printf("Seleccione una opcion: ");
82         scanf("%d", &opcion);
83
84         switch (opcion) {
85             case 1:
86                 determinarNumero();
87                 break;
88             case 2:
89                 mostrarHistorial();
90                 break;
91             case 3:
92                 borrarHistorial();
93                 break;
94             case 4:
95                 printf("\nGracias por usar el programa, %s! ¡Hasta luego!\n", usuario.nombre);
96                 break;
97             default:
98                 printf("\nOpcion no valida. Por favor, seleccione una opcion valida.\n");
99                 break;
100         }
101     } while (opcion != 4);
102
103     return 0;
104 }
105

```

### C. Resultados

```

Menu:
1. Ingresar numero y determinar clasificacion
2. Ver historial de datos
3. Borrar historial de datos
4. Salir
Seleccione una opcion: 1

Ingrese un numero: 2
El numero 2.00 es positivo.

Menu:
1. Ingresar numero y determinar clasificacion
2. Ver historial de datos
3. Borrar historial de datos
4. Salir
Seleccione una opcion: 2

--- Historial de Datos ---
Numero: 2.00

Menu:
1. Ingresar numero y determinar clasificacion
2. Ver historial de datos
3. Borrar historial de datos
4. Salir
Seleccione una opcion: 3

Historial de datos borrado exitosamente.

Menu:
1. Ingresar numero y determinar clasificacion
2. Ver historial de datos
3. Borrar historial de datos
4. Salir
Seleccione una opcion: 4

¡Gracias por usar el programa, Axel! ¡Hasta luego!

```

Documento en GitHub

<https://github.com/AxelRamirez12/parcial1.git>