

Ejercicio: Algoritmos de Ordenamiento en C

Objetivo

Implementar y comparar tres algoritmos de ordenamiento (Bubble Sort, Insertion Sort y Selection Sort) utilizando un array de calificaciones predefinido.

Descripción

En este ejercicio, trabajarás con un array de 10 calificaciones ya definido. Deberás implementar tres funciones diferentes de ordenamiento y un menú que permita elegir qué algoritmo utilizar. Para cada algoritmo, deberás contar y mostrar:

- El número de comparaciones realizadas
- El número de intercambios ejecutados
- El array ordenado resultante

Requisitos

1. Datos iniciales:

- Utiliza el siguiente array de calificaciones: `float notas[10] = {7.5, 5.0, 9.2, 6.8, 8.1, 4.3, 7.0, 5.5, 9.9, 6.0};`

2. Implementa las siguientes funciones:

- `void bubbleSort(float arr[], int n)`
- `void insertionSort(float arr[], int n)`
- `void selectionSort(float arr[], int n)`
- `void imprimirArray(float arr[], int n)`

3. Crea un menú con las siguientes opciones:

- Ordenar usando Bubble Sort
- Ordenar usando Insertion Sort
- Ordenar usando Selection Sort
- Mostrar array original
- Salir

4. Para cada algoritmo implementado:

- Copiá el array original a un array temporal para no modificar los datos originales
- Contá el número de comparaciones entre elementos dentro de la función
- Contá el número de intercambios realizados dentro de la función
- Imprimí estas estadísticas al final de la función (o sea antes de que la función termine)
- Mostrá el array resultante