# Exercícios Práticos – Bubble Sort em C (com ponteiros)

## Exercício 1 – Bubble Sort com Ponteiros

Objetivo:

Implemente o algoritmo Bubble Sort em C usando ponteiros para ordenar um vetor de 20 inteiros fornecidos pelo usuário.

Requisitos:

- Criar uma função bubbleSort(int \*arr, int n) que ordene os valores com ponteiros.

- Usar scanf e printf com aritmética de ponteiros (ex: \*(ptr + i)).

- Exibir o vetor ordenado após a ordenação.

## Exercício 2 – Bubble Sort sem Recursão

Objetivo:

Implemente o algoritmo Bubble Sort sem uso de recursão, utilizando apenas laços for.

Requisitos:

- A lógica do algoritmo Bubble Sort é naturalmente iterativa, não exigindo modificações especiais.

- Utilizar troca com ponteiros para manipular os elementos.

## Exercício 3 – Bubble Sort com Alocação Dinâmica

Objetivo:

Crie um programa que aloque dinamicamente um vetor de n inteiros, leia os valores e ordene com Bubble Sort.

Requisitos:

- Utilizar malloc para criar dinamicamente o vetor.

- Implementar a ordenação com bubbleSort(int \*arr, int n).

- Liberar a memória com free() ao final.

- Usar ponteiros para acessar e modificar os elementos.

Exemplo de entrada esperada:

Digite o número de elementos: 10

Digite os 10 valores: 5 3 7 1 9 2 8 6 0 4

Saída esperada:

Array ordenado: 0 1 2 3 4 5 6 7 8 9