

Nomes: _____

```
// Ordenação por seleção:
void trocar(int *a, int *b) {
    int aux = *a;
    *a = *b;
    *b = aux;
}

int menor_elemento(int vetor[], int n, int primeiro) {
    int i, menor = primeiro;
    for (i = primeiro + 1; i < n; i++) {
        if (vetor[i] < vetor[menor])
            menor = i;
    }
    return menor;
}

int ordenar_selecao(int vetor[], int n) {
    int i, menor;
    for (i = 0; i < n; i++) {
        menor = menor_elemento(vetor, n, i);
        trocar(&vetor[i], &vetor[menor]);
    }
}

// Ordenação por seleção:
int posicao_elemento(int vetor[], int ultimo, int elemento) {
    int i;
    for (i = 0; i <= ultimo && vetor[i] <= elemento; i++);
    return i;
}

void deslocar_subvetor(int vetor[], int primeiro, int ultimo) {
    int i;
    for (i = ultimo; i >= primeiro; i--) {
        vetor[i+1] = vetor[i];
    }
}

int ordenar_insercao(int vetor[], int n) {
    int i, posicao;
    int elemento;
    for (i = 1; i < n; i++) {
        elemento = vetor[i];
        posicao = posicao_elemento(vetor, i-1, elemento);
        deslocar_subvetor(vetor, posicao, i-1);
        vetor[posicao] = elemento;
    }
}

// Estrutura
typedef struct {
    char nome[20]; int idade; enum { MAS, FEM } sexo;
} Pessoa;
```

1. Reescreva a função `ordenar_insercao` para que ela **não** utilize funções auxiliares. Acima, selecionamos antes e deslocamos um subvetor depois. É possível fazer os dois de uma só vez. Qual a vantagem?
2. Na função `ordenar_selecao`, é realmente necessária a última iteração do laço de repetição? Por quê? E para a função `ordenar_insercao`?
3. Escreva uma função para ordenar um vetor de Pessoa em ordem decrescente de idade e, havendo empate, em ordem crescente de nome.
4. Suponha que existe um vetor de pessoas. Queremos criar um vetor de mulheres ordenado por idade em ordem decrescente e, havendo empate, em ordem crescente de nome. Não queremos modificar o vetor original mas também não queremos desperdiçar espaço nem duplicar informação. Para isso: (i) crie um vetor de ponteiros para pessoas; (ii) modifique a questão 3 para que ela receba um vetor de ponteiros. Implemente essa estratégia e explique suas vantagens e como ela funciona.
5. Ao invés de usar ponteiros, você poderia usar índices para o vetor original. Explique as vantagens e desvantagens.