

Trabalho 01
Estrutura de Dados
Prof. Roberto Cabral
21 de outubro de 2022

5 algoritmos:
BubbleSort
InserctionSort
SelectionSort
MergeSort
QuickSort

1. Problema

(Parte 1). Nesta atividade, deve-se implementar todos os algoritmos de ordenação estudados em sala de aula. Deve-se ter uma versão interativa e uma versão recursiva desses algoritmos¹. O código deve ser organizado da seguinte forma:

- `ordenacao.h`, onde estarão as assinaturas das funções de ordenação.
- `ordenacao.c`, onde estarão as implementações das funções de ordenação.
- `main.c`, onde todas as funções deverão ser testadas.

Para cada um desses algoritmos deve-se implementar as seguintes operações:

- Dado um vetor de inteiros, ordenar o vetor em ordem crescente.
- Dado um vetor de inteiros, ordene o vetor de modo que o vetor resultante esteja na seguinte ordem: primeiro os números ímpares, depois os números pares; sendo que os números ímpares deverão estar na ordem crescente e os pares na ordem decrescente.

(Parte 2). Além disso, escolha um algoritmo de ordenação e implemente as seguintes operações:

- Dado um vetor de Alunos (um aluno possui um nome e uma matrícula), ordene o vetor em ordem alfabética.
- Dado um vetor de Alunos (um aluno possui um nome e uma matrícula), ordene o vetor pela matrícula.

2. Testes

É esperado, para a parte 1, que cada função seja analisada para entradas de tamanhos distintos (uma sugestão é testar para conjuntos de dados de 2^8 até 2^{20} elementos).

Esta análise consiste em comparar os algoritmos considerando duas métricas de desempenho: (i) número de comparações e (ii) tempo total gasto para ordenação. As entradas deverão ser conjuntos de elementos aleatórios.

Deve-se entregar um relatório com o custo de desempenho de cada uma das funções e dissertar no relatório uma pequena comparação entre os algoritmos/implementações.

¹para o quicksort, espera-se apenas a versão recursiva

3. Informações adicionais:

- Deverá ser submetido, juntamente com o código, um relatório técnico explicando tudo que foi feito no trabalho. Dentre outras coisas, o relatório deve ter:
 - Uma descrição detalhada de cada um dos algoritmos de ordenação;
 - Gráficos mostrando comparações entre os algoritmos para distintos tamanhos de entrada;
 - Comparação entre os algoritmos interativos e recursivos;
 - Uma seção descrevendo como o trabalho foi dividido entre as duplas;
 - Uma seção descrevendo otimizações que foram feitas no código;
 - Uma seção de dificuldades encontrada.
- O trabalho deverá ser feito em duplas (**as notas serão individuais**);
- O trabalho deverá ser entregue, pelo moodle, até o dia **09 de Novembro**;
- O número máximo de páginas do relatório são 10 páginas.
- A apresentação do trabalho será feita em horário definido pelo professor.

Obs.: Qualquer indício de plágio resultará em nota ZERO para todos os envolvidos.