

Prática 3 Ordenação

Profa. Maria Camila Nardini Barioni

camila.barioni@ufu.br

Bloco B - sala 1B137

2º semestre de 2023

Informações gerais

- Esta prática tem por objetivo reforçar o conhecimento do(a) aluno(a) com relação ao algoritmo de ordenação Heapsort estudado em sala
 - Data da entrega: 09/02/2024
 - Grupo de até 3 (dois) alunos(as). Incluir nome e matrícula como comentário no código
 - Linguagem de programação a ser usada: C
 - O que deve ser entregue: Link para o repositório do código (Replit.com) e documento .pdf com as respostas das questões descritas na atividade
 - Essa prática vale 02 pontos

Exercícios

1. Responda às seguintes questões sobre o método de ordenação heapsort estudado em sala.
 - a) Suponha que todos os elementos de um **heap máximo** são distintos. Onde pode estar o segundo maior elemento? Onde não pode estar o segundo maior elemento? E o terceiro maior? Explique!
 - b) Um vetor em ordem crescente é um **heap máximo**? Um vetor em ordem decrescente é um **heap máximo**? Explique!
 - c) Mostre que o heapsort não é estável.

Exercícios

2. Implemente o algoritmo heapsort discutido em sala de aula para organizar vetores de números inteiros e compare o seu desempenho com o algoritmo shellsort. Para realizar essa comparação, construa uma tabela de referência contendo o tempo obtido para cada método de ordenação, considerando:
- a) três tipos de entrada de dados: aleatória, em ordem ascendente e em ordem descendente
 - b) para cada tipo de entrada de dados, use pelo menos três tamanhos de entrada iniciando com vetor de tamanho 10.000.

Deve ser possível escolher o tipo de entrada de dados e o tamanho do vetor por meio de um menu de opções. Na comparação dos métodos, discuta as vantagens e desvantagens observadas. Você pode consultar os slides da aula teórica para apoiar a sua comparação.

Exercícios (esse vale 1 ponto extra)

3. Considere a função `max_heapify()` vista em aula que tem o objetivo de descer corrigindo a partir de uma dada posição. Faça uma função não-recursiva para realizar o mesmo objetivo.

A questão 3 pode ser entregue até 10/02 na tarefa do Teams "Exercício 3 Ordenação – Ponto extra"

Substitua a função `max_heapify()` pela sua versão não recursiva no código do algoritmo `heapsort` e apresente exemplos de execução no `main`.