



CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS

Departamento de Computação - DECOM

Laboratório de Algoritmos e Estrutura de Dados II

Prática 03 - Implementação do TAD Symetric Binary B-Tree (SBB)

Thiago Ribeiro Corrêa - Engenharia da Computação

Professor: Otaviano Martins

Belo Horizonte

13 de dezembro de 2023

1 - ÁRVORE COM INSERÇÕES ORDENADAS

Para a árvore em que foram inseridos elementos ordenados foram obtidos os seguintes resultados:

Número de Elementos	Número de Comparações	Tempo (s)
1000	15	1,431
2000	16	1,428
3000	18	1,631
4000	17	1,467
5000	17	1,282
6000	19	1,661
7000	19	1,517
8000	18	1,550
9000	18	1,266

2 - ÁRVORE COM INSERÇÕES DESORDENADAS

Para a árvore em que foram inseridos elementos desordenados foram obtidos os seguintes resultados:

Número de Elementos	Número de Comparações	Tempo (s)
1000	10	1,278
2000	10	1,392
3000	11	1,685
4000	13	1,547
5000	11	1,408
6000	13	1,399
7000	14	1,836
8000	11	1,279
9000	14	1,685

3 - ANÁLISE DOS GRÁFICOS

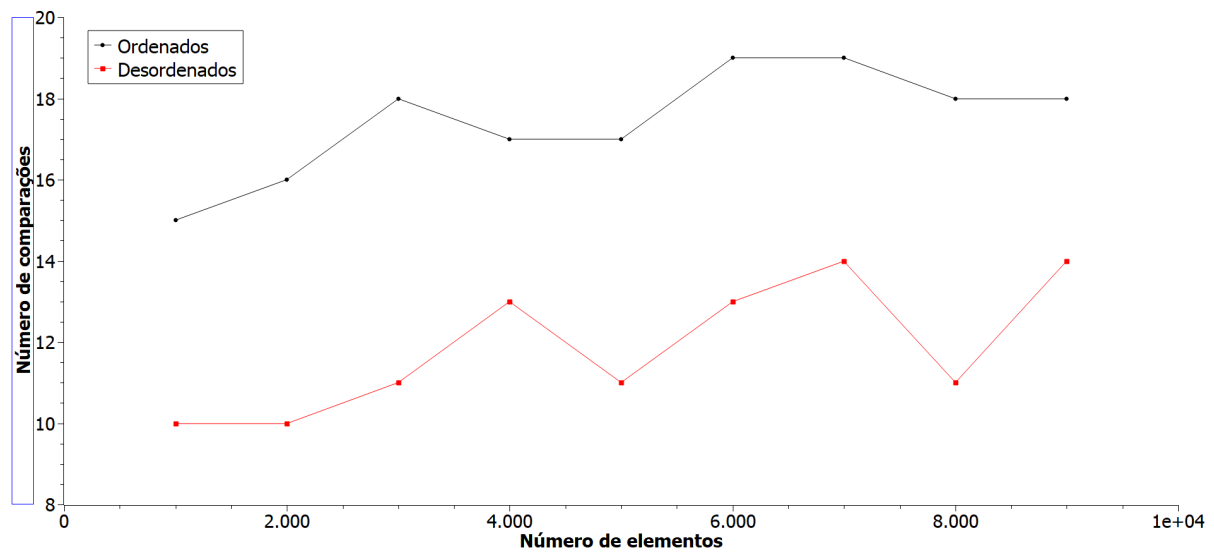


GRÁFICO 1 - Número de comparações *versus* Número de elementos

A partir do GRAF.1 que analisa o número de comparações para encontrar um elemento não existente em uma árvore SBB em função do número de elementos na árvore, é possível notar que a árvore desordenada possui melhores resultados. Isso pode ser explicado pois a árvore desordenada possibilita diferentes arranjos, os quais podem ser mais fáceis de se procurar um elemento.

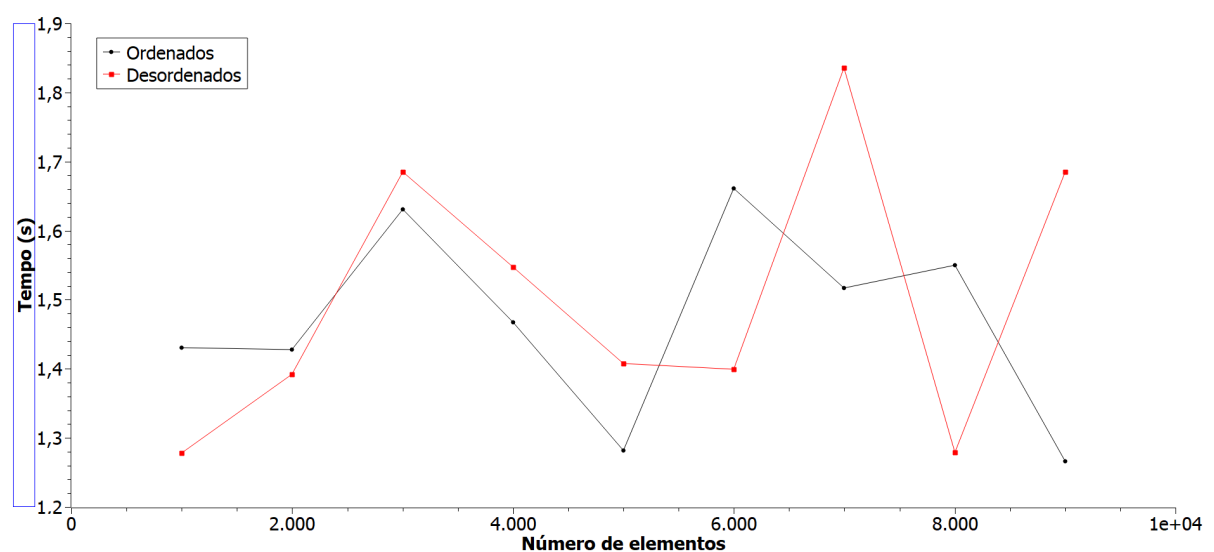


GRÁFICO 2 - Tempo *versus* Número de elementos

A partir do GRAF.2 é possível notar que o tempo de execução de um programa depende não apenas dele como também do que mais está sendo executado no computador no momento. É possível notar isso quando mesmo que menos comparações tenham sido feitas, em alguns casos o tempo gasto foi maior que o tempo para mais comparações.

4 - COMPARAÇÃO COM A PRÁTICA 2

Ao comparar o resultado da prática 3 com o da prática 2 é possível ver uma diferença discrepante no número de comparações feitas, principalmente para a árvore ordenada. Isso, pois a árvore SBB para elementos ordenados não aceita que os elementos fiquem todos em apenas uma diagonal como ocorre na árvore binária de pesquisa. Também é possível ver uma diferença para as demais árvores, ainda que a diferença não seja tão exagerada. Dessa forma, é possível concluir que a árvore SBB é mais eficiente para buscas, haja vista que está parcialmente balanceada.