

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
CAMPUS TIMÓTEO**

José Geraldo Duarte Júnior
Pedro Arthur Diniz Freitas
Raphael Gomes Wagner

ÁRVORES BINÁRIA: INSERÇÃO E BUSCA

Timóteo

2022

José Geraldo Duarte Júnior
Pedro Arthur Diniz Freitas
Raphael Gomes Wagner

ARVORES BINÁRIA: INSERÇÃO E BUSCA

Lista 7 de Algoritmos e Estrutura de Dados
sobre Árvores do tipo B e B*, no curso de
Engenharia de Computação no Centro Federal
de Educação Tecnológica de Minas Gerais

Orientador: Gustavo Martins

Timóteo

2022

- .
- .
- .

Dedico a
Todos menos alguns...

Agradecimentos

Agradeço a todos pela dedicação por terem passado em AED I comigo..

1 Introdução

Esse trabalho foi desenvolvido em conjunto, feito por Raphael Gomes José Geraldo e Pedro Arthur, no intuito de executar processos de inserção e remoção de objetos em árvores do tipo B e B* (B estrela), utilizando de implementações fornecidas pelo professor e pesquisador Nivio Ziviani, porem com pequenas alterações para facilitar as aplicação destas nos meios desejados. Essa implementação foi utilizada para medir tempos de execução sobre a inserção e remoção de dados nas árvores citadas acima.

Para esta prática, imaginaremos um caso de construção de árvores B e B* para a paginação de arquivos em um sistema operacional. Para tanto, consideramos que o tamanho de um setor na unidade de armazenamento contém 512 Bytes, e uma página contém 4 kBytes. Com isso consideramos a existência de dois modelos de arquivos, sendo arquivos de 1kB e arquivos de 256B, assim temos a implementação de 1 árvore para cada caso de arquivo nos dois modelos de árvore.

2 Desenvolvimento

2.1 Inserção

Como dito anteriormente, a implementação desta árvore segue a lógica de que seriam inseridos arquivos de 1kB e de 256B, considerando também que o tamanho de um setor na unidade de armazenamento contém 512 Bytes, e uma página contém 4 kB. Isso resultou em 2 modelos de árvores para cada tipo, ou seja 2 árvores B instanciadas para receber sucessivamente arquivos de 1kB em uma e arquivos de 256B na outra. O mesmo caso ocorre nas árvores B*.

Logo as características resultantes destas árvores foram:

Árvore B - 1: $m = 4$;

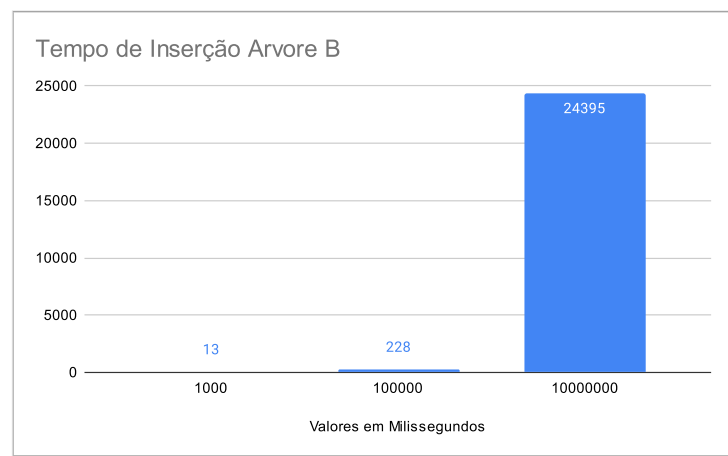
Árvore B - 2: $m = 8$;

Árvore B* - 1: $m = 4$;

Árvore B* - 2: $m = 8$.

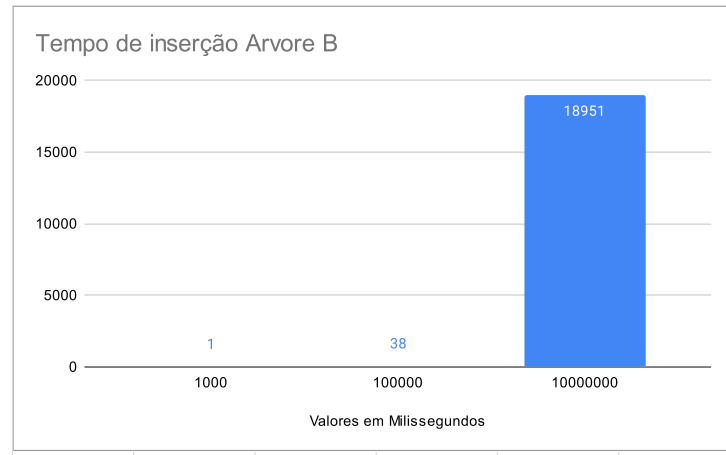
Os arquivos inseridos são representados por objetos do tipo *Meultem* desenvolvido também pelo professor e pesquisador da UFMG, Nívio Ziviani. Este objeto armazena um inteiro que foi gerado de forma pseudoaleatória utilizando de um algoritmo para esse fim.

2.1.1 Árvore B - Arquivos 1kB



Quantidade de arquivos	Tempo em Milissegundos
10^3	13
10^5	228
10^7	24395

2.1.2 Arvore B - Arquivos 256B



Quantidade de arquivos	Tempo em Milissegundos
10^3	1
10^5	38
10^7	18951

2.1.3 Arvore B* - Arquivos 1kB

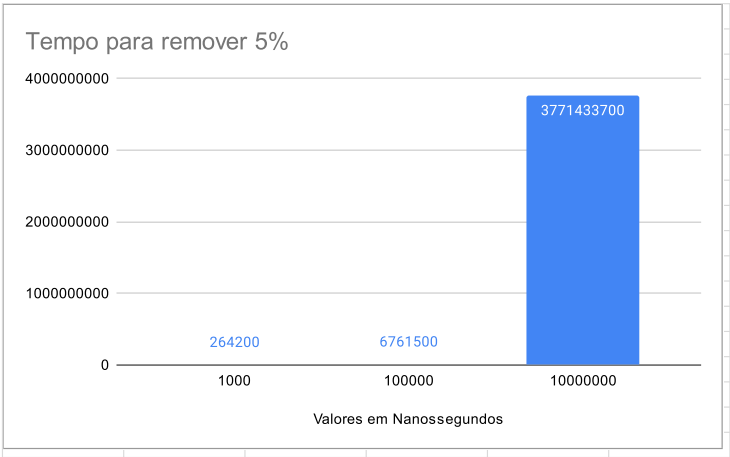
2.1.4 Arvore B* - Arquivos 256B

2.2 Remoção

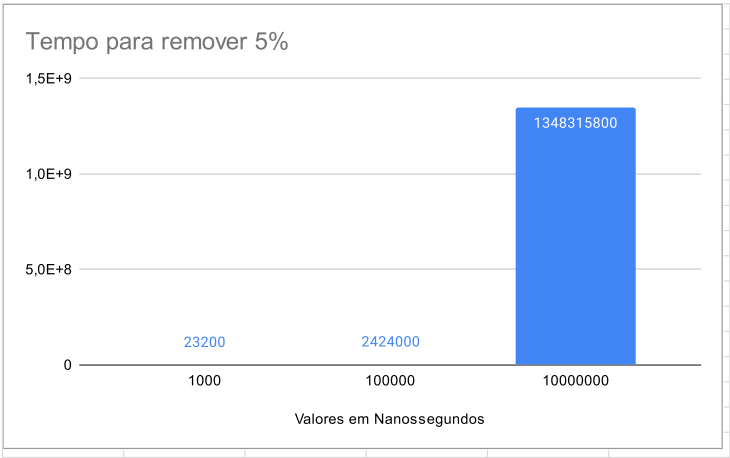
Após o processo de inserção daqueles objetos que representam os arquivos a serem inseridos na arvore, são executados testes de tempo de remoção de 5% dos mesmos. Gerando os seguintes resultados a seguir:

2.2.1 Arvore B - Arquivos 1kB

Quantidade de arquivos	Tempo em Nanossegundos
10^3	264200
10^5	6761500
10^7	3771433700



2.2.2 Arvore B - Arquivos 256B



Quantidade de arquivos	Tempo em Nanossegundos
10 ³	23200
10 ⁵	2424000
10 ⁷	1348315800

2.2.3 Arvore B* - Arquivos 1kB

2.2.4 Arvore B* - Arquivos 256B

3 Conclusão

Após a análise da prática e dos tempos de execução em todas as árvores, é notável que, se tomado os mesmos métodos para inserir e excluir, tanto o método de inclusão quanto o de exclusão com arquivos de 256B tiveram tempos de execução mais rápidos do que com arquivos de 1kB, e sabendo que 1kB são 1000 Bytes, por lógica teria um tempo de execução maior.

Sobre a árvore de B*, pela complexidade ao implementar a árvore e pela falta de conteúdo na internet para construção da árvore, não foi possível recolher os dados e eventualmente não foi possível construir o gráfico e tabela para a comparação dos dados.

Referências