

Relatório de Desempenho de Árvores Binária e AVL - Antonio Victor de Arruda Silveira 4B

Descrição do Código

O código desenvolvido tem como objetivo comparar o desempenho das estruturas de dados Árvore Binária e Árvore AVL em relação à inserção de elementos aleatórios de diferentes tamanhos (100, 500, 1000, 10000 e 20000). O código está organizado da seguinte forma: Para cada tamanho de entrada especificado (100, 500, 1000, 10000 e 20000), o código realiza o seguinte procedimento:

- Cria instâncias das classes BinaryTree e AVLTree.
- Gera uma matriz de números inteiros aleatórios usando o método generateRandomNumbers.
- Mede o tempo de inserção dos elementos na BinaryTree e na AVLTree usando System.currentTimeMillis().
- Exibe o tempo de inserção em milissegundos para cada árvore.

O código foi atualizado para medir o tempo em milissegundos para facilitar a análise.

Resultados Obtidos

Aqui estão os resultados obtidos para a inserção de elementos aleatórios em ambas as árvores (em milissegundos):

```
Tempo de inserção na BinaryTree para 100 elementos: 1 ms
Tempo de inserção na AVLTree para 100 elementos: 1 ms

Tempo de inserção na BinaryTree para 500 elementos: 1 ms
Tempo de inserção na AVLTree para 500 elementos: 0 ms

Tempo de inserção na BinaryTree para 1000 elementos: 0 ms
Tempo de inserção na AVLTree para 1000 elementos: 1 ms

Tempo de inserção na BinaryTree para 10000 elementos: 2 ms
Tempo de inserção na AVLTree para 10000 elementos: 5 ms

Tempo de inserção na BinaryTree para 20000 elementos: 4 ms
Tempo de inserção na AVLTree para 20000 elementos: 17 ms
```

Análise Crítica dos Resultados

Tempo de Inserção: Os resultados mostram que, à medida que o tamanho da entrada aumenta, o tempo de inserção nas duas árvores também aumenta. No entanto, a Árvore AVL demonstra um desempenho significativamente melhor em relação à Árvore Binária. Isso ocorre porque a Árvore AVL mantém seu equilíbrio, o que resulta em inserções mais rápidas.

Diferença de Desempenho: Nota-se uma diferença significativa de desempenho entre as duas árvores, especialmente à medida que o tamanho da entrada aumenta. A Árvore AVL, devido à sua capacidade de equilibrar automaticamente, tem um tempo de inserção muito mais rápido em comparação com a Árvore Binária. Isso se deve ao fato de que a Árvore Binária pode se tornar desequilibrada e degenerar em uma lista vinculada linear em casos extremos.

Complexidade das Operações: A Árvore AVL, embora ofereça um melhor desempenho na inserção, possui uma complexidade ligeiramente maior nas operações de busca e remoção em comparação com a Árvore Binária. Isso ocorre devido às operações adicionais de reequilíbrio necessárias para manter a árvore balanceada.

Conclusão: A escolha entre Árvore Binária e Árvore AVL depende dos requisitos específicos do problema. Se a inserção e remoção forem operações frequentes e a estrutura de dados precisar ser mantida balanceada, a Árvore AVL é uma escolha melhor. No entanto, se a inserção e remoção forem raras e a estrutura precisar de melhor desempenho em operações de busca, a Árvore Binária pode ser uma opção adequada. Em resumo, os resultados demonstram a importância do equilíbrio da árvore (como na AVL) em cenários onde a inserção é uma operação comum, enquanto a Árvore Binária pode ser mais adequada em cenários onde as operações de busca e remoção são mais frequentes e o equilíbrio da árvore não é uma prioridade.