

Universidade de Vassouras

Engenharia de Software

Estrutura de Dados

Árvores AVL - Conceitos e Implementações

Alunos(a): Bruno de Oliveira - 202411333, Gabriel Macedo - 202412118,
João Candia - 202412113, Matheus Prates - 202411269, Natália Cardoso
- 202411325, Rosonatt Ferreira - 202411251, Ryan Guiwison - 202411281,
Samuel Soares - 202411296, Thiago Araújo - 202411274

Professor: André Saraiva

Saquarema - RJ

2025

Universidade de Vassouras

Alunos(a): Bruno de Oliveira - 202411333, Gabriel Macedo - 202412118,
João Candia - 202412113, Matheus Prates - 202411269, Natália Cardoso -
202411325, Rosonatt Ferreira - 202411251, Ryan Guiwison - 202411281,
Samuel Soares - 202411296, Thiago Araújo - 202411274

Árvores AVL - Conceitos e Implementações

Trabalho de Estrutura de Dados

Trabalho apresentado na disciplina Estrutura de Dados como
nota complementar a primeira Avaliação

Saquarema - RJ

2025

Resumo

As Árvores AVL são uma estrutura de dados balanceada utilizada para otimizar operações de busca. Sua principal característica é o auto-balanceamento, garantindo que a diferença de altura entre as subárvores de qualquer nó não ultrapasse 1. Para manter esse equilíbrio, elas utilizam rotações simples e duplas. A inserção e a remoção de nós seguem regras semelhantes às das árvores de busca binária (BSTs), mas adicionam verificações para manter o balanceamento. Isso torna as AVL ideais para aplicações como bancos de dados, indexação de arquivos e sistemas de busca.

Palavras-chave: Árvore AVL, Auto-balanceamento, Inserção e remoção

Sumário

Lista de Figuras

1 - Introdução	p. 5
1.1 Fator de Balanceamento	p. 5
2 - Rotações Simples e duplas	p. 6
2.1 Rotação Simples à Direita	p. 6
2.2 Rotação Simples à Esquerda	p. 6
2.3 Rotação Dupla à Direita	p. 7
2.4 Rotação Dupla à Esquerda	p. 7
3 - Inserção e Remoção Em Árvores AVL	p. 8
3.1 Inserção Padrão	p. 8
3.2 Remoção Em Árvores AVL	p. 8
3.2.1 Nó Folha	p. 8
3.2.2 Nó com Um Filho	p. 9
3.2.3 Nó com Dois Filhos	p. 9
4 - Características Adicionais Importantes	p. 10
4.1 Vantagens e Desvantagens	p. 10
5 - Conclusão	p. 11
Referências	p. 12

Lista de Figuras

1	Árvore AVL balanceada.	p. 5
2	Rotação Simples à Direita.	p. 6
3	Rotação Dupla à Direita.	p. 7
4	Inserção Padrão de um nó.	p. 8

Capítulo 1 - Introdução

As Árvores AVL, nomeadas em homenagem a seus criadores Adelson-Velsky e Landis, são uma estrutura de dados fundamental na ciência da computação. Elas representam uma forma eficiente de organizar dados, garantindo operações de busca, inserção e remoção em tempo logarítmico. O principal diferencial das Árvores AVL em relação às árvores de busca binária tradicionais é sua capacidade de auto-balanceamento, que assegura que a diferença de altura entre as subárvores de qualquer nó não exceda 1. Essa característica é crucial para manter a eficiência nas operações, especialmente em aplicações que requerem um desempenho elevado e um acesso rápido a dados. Neste texto, exploraremos as propriedades, operações e a importância das Árvores AVL, assim como suas vantagens e desvantagens.

1.1 Fator de Balanceamento

O fator de balanceamento (FB) de um nó é calculado como a altura da subárvore esquerda menos a altura da subárvore direita. Um fator de balanceamento de 0 indica uma árvore perfeitamente balanceada, enquanto valores de +1 ou -1 indicam um leve desbalanceamento. Para manter o balanceamento, as árvores AVL utilizam rotações e operações que reestruturam a árvore para restaurar a propriedade de balanceamento. Na Figura 1 podemos observar um exemplo de uma árvore AVL balanceada.

Exemplo visual de uma árvore AVL balanceada:

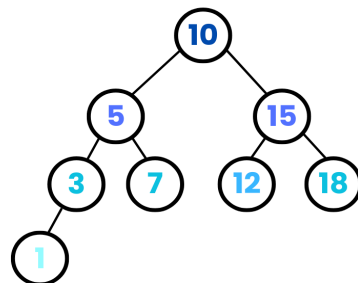


Figura 1: Árvore AVL balanceada.

Capítulo 2 - Rotações Simples e duplas

As Árvores AVL utilizam operações de rotação para rebalancear após inserções ou exclusões. Elas são realizadas para corrigir o desequilíbrio entre os nós da árvore ou rebalancear a Árvore por completo. Os tipos de rotação são: Rotação simples à direita, Rotação simples à esquerda, Rotação dupla à direita e Rotação à dupla à esquerda.

2.1 Rotação Simples à Direita

Uma rotação simples à direita é aplicada quando o fator de balanceamento de um nó é $+2$ e seu filho esquerdo tem um fator de balanceamento de $+1$, 0 ou -1 . A rotação corrige o desbalanceamento movendo o filho esquerdo do nó original para a posição do nó original. Em seguida, o nó original se torna o filho direito do seu antigo filho esquerdo, preservando a ordem da árvore AVL. Na Figura 2 podemos observar uma rotação simples à direita.

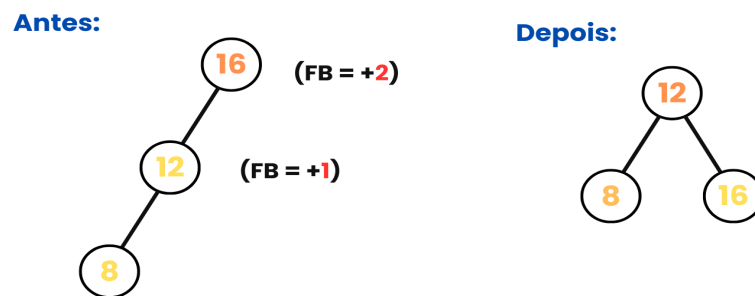


Figura 2: Rotação Simples à Direita.

2.2 Rotação Simples à Esquerda

Uma rotação simples à esquerda é usada quando o fator de balanceamento (FB) de um nó é -2 e seu filho direito tem um fator de balanceamento de -1 , 0 ou $+1$. A rotação

corrige o desbalanceamento movendo o filho direito para a posição do nó original. O nó original, então, se torna o filho esquerdo do antigo filho direito, preservando a estrutura da árvore AVL.

2.3 Rotação Dupla à Direita

Uma rotação dupla à direita é aplicada quando o fator de balanceamento de um nó é $+2$ e seu filho esquerdo tem um fator de balanceamento de -1 . Essa operação é realizada em duas etapas: Primeiro, uma rotação simples à esquerda é realizada no filho esquerdo do nó desbalanceado. Em seguida, uma rotação simples à direita é aplicada ao próprio nó desbalanceado, restaurando o equilíbrio da árvore AVL. Na Figura 3 temos um exemplo de uma rotação dupla à direita.

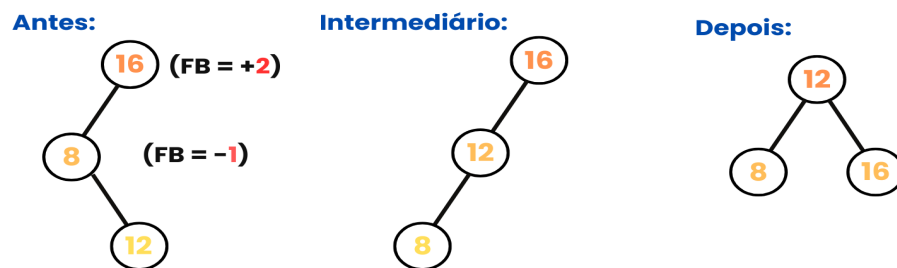


Figura 3: Rotação Dupla à Direita.

2.4 Rotação Dupla à Esquerda

A rotação dupla à esquerda é o inverso da rotação dupla à direita. Ela é aplicada quando o fator de balanceamento de um nó é -2 e seu filho direito tem um fator de balanceamento de $+1$. Essa rotação envolve duas etapas: Primeiro, realiza-se uma rotação simples à direita no filho direito do nó desbalanceado. Em seguida, realiza-se uma rotação simples à esquerda no próprio nó desbalanceado, restaurando o equilíbrio da árvore AVL.

Capítulo 3 - Inserção e Remoção Em Árvores AVL

3.1 Inserção Padrão

A inserção em uma árvore AVL segue inicialmente o mesmo algoritmo de inserção em uma árvore de busca binária simples. O novo nó é inserido na posição apropriada, mantendo a ordem da árvore. Após a inserção, o caminho percorrido desde a raiz até o nó inserido é verificado em busca de desequilíbrios. Se um desequilíbrio for detectado, a árvore é rebalanceada usando as rotações simples ou duplas descritas anteriormente. O rebalanceamento pode afetar os nós ancestrais do nó inserido, pois as rotações podem afetar a estrutura da árvore acima do nível de inserção.

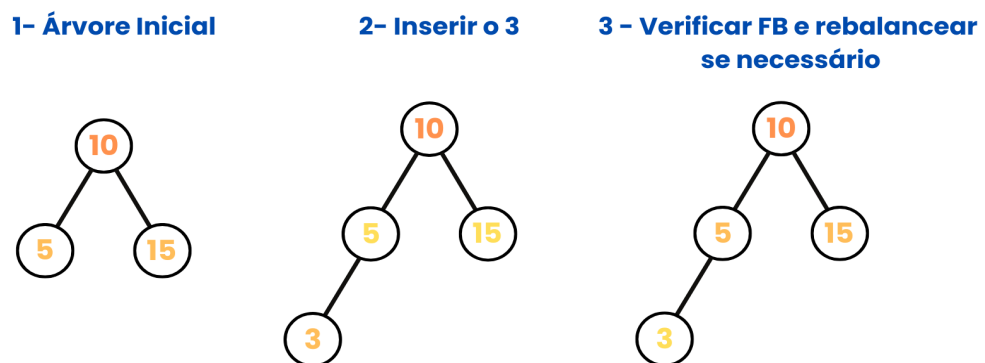


Figura 4: Inserção Padrão de um nó.

3.2 Remoção Em Árvores AVL

3.2.1 Nó Folha

A remoção de um nó folha é simples, pois ele não possui filhos. O nó é removido e os ponteiros do nó pai são atualizados para remover a referência ao nó removido.

3.2.2 Nó com Um Filho

A remoção de um nó com um filho é realizada substituindo o nó removido por seu filho único e atualizando os ponteiros para manter a estrutura da árvore.

3.2.3 Nó com Dois Filhos

A remoção de um nó com dois filhos é mais complexa. O nó removido é substituído pelo seu sucessor em ordem (o menor nó na subárvore direita) e a árvore é rebalanceada para preservar a estrutura AVL.

Capítulo 4 - Características Adicionais Importantes

4.1 Vantagens e Desvantagens

Esse tipo de Árvore são especialmente populares em sistemas embarcados e aplicações em tempo real. Entretanto em casos onde temos que realizar muitas inserções e remoções, a árvore terá que realizar muitas rotações para se manter balanceada, gerando um custo adicional em relação a uma simples árvore binária de busca. A seguir veremos as principais vantagens e desvantagens da Árvore AVL.

Vantagens:

- Tempo de busca garantido em pelo menos $O(\log_n)$ no pior caso;
- Melhor desempenho em comparação com BSTs não balanceadas;
- Ideal para aplicações com inserções e remoções frequentes;
- Altura balanceada automaticamente;
- Eficiente para conjuntos de dados grandes;
- Boa utilização de memória.

Desvantagens:

- Implementação mais complexa do que BSTs simples;
- Custo adicional para manter o balanceamento;
- Rotações podem ser custosas em algumas situações;
- Maior overhead de memória devido ao armazenamento dos fatores de balanceamento;
- Complexidade de implementação das rotações.

Capítulo 5 - Conclusão

As Árvores AVL se destacam como uma solução eficiente para a manipulação de dados em diversas aplicações, como bancos de dados e sistemas de busca. Sua habilidade de manter um equilíbrio dinâmico entre as subárvores garante operações de busca, inserção e remoção em tempo logarítmico, o que é especialmente valioso em cenários de dados dinâmicos e frequentes alterações. Apesar de sua complexidade de implementação e do custo associado ao balanceamento, as vantagens superam as desvantagens para muitas aplicações. Assim, as Árvores AVL não apenas melhoram a eficiência na manipulação de dados, mas também estabelecem um padrão importante para o desenvolvimento de estruturas de dados em ciência da computação.

Referências

BARANAUSKAS, J. A. Árvores avl: Algoritmos e estruturas - notas de aula. *Departamento de Física e Matemática – FFCLRP-USP*, 2007.

HOROWITZ, E.; SAHNI, S. *Fundamentos de estruturas de dados*. [S.l.]: Campus, 1987.

WIRTH, N. *Algoritmos e Estruturas de Dados*. [S.l.]: Prentice/Hall do Brasil, 1989.