

Trabalho Prático: Árvores Binárias

Alcides José Cândido Neto - 92220249

17/04/2025

1. Introdução

Este trabalho tem como objetivo a implementação de uma árvore binária com as funcionalidades básicas e avançadas de manipulação de dados. A estrutura permite a inserção, remoção, pesquisa, verificação de tipos (árvore cheia, completa e estritamente binária), travessias (in-order, pre-order e post-order), cálculo do grau de um nó e o nível da árvore. O projeto foi desenvolvido utilizando a linguagem Java, com base nos princípios da programação orientada a objetos.

2. Implementação

A implementação foi dividida em três classes principais: Node, BinaryTree e Main.

2.1 Estrutura de Dados

A classe Node representa cada nó da árvore, contendo o valor inteiro e referências para seus filhos esquerdo e direito. A classe BinaryTree contém os métodos de manipulação e verificação da árvore. Já a classe Main serve como interface de linha de comando para a interação com o usuário e testes.

2.2 Funcionalidades

- Inserção e remoção: Métodos recursivos que mantêm as propriedades da árvore binária de busca.
- Busca de elemento: Retorna verdadeiro se o elemento estiver presente na árvore.
- Travessias: Permite as travessias in-order, pre-order e post-order para percorrer a árvore.
- Verificação do tipo da árvore:
 - Completa: Todos os níveis, exceto o último, estão completamente preenchidos.
 - Cheia: Todos os nós têm 0 ou 2 filhos.
 - Estrita: Todos os nós internos têm exatamente dois filhos.
- Grau de um nó: Calcula o número de filhos de um nó especificado.
- Nível da árvore: Calcula a altura da árvore.

2.3 Entrada e Saída

A entrada dos dados é realizada por meio de um terminal interativo. O usuário pode escolher operações através de um menu, inserir valores e visualizar os resultados diretamente na tela.

2.4 Decisões de Implementação

Optou-se por não permitir a inserção de valores duplicados, garantindo que a árvore seja uma árvore binária de busca válida. O menu textual foi implementado para facilitar testes manuais e simulações.

3. Testes Executados

Foram realizados testes com as seguintes operações:

- Inserção dos valores: 50, 30, 70, 20, 40, 60, 80.
- Impressões das travessias in-order, pre-order e post-order.
- Verificação da árvore quanto a ser cheia, completa e estrita.
- Remoção dos nós: 20 (folha), 30 (com um filho), 50 (com dois filhos).
- Cálculo do grau do nó 70 (esperado: 2).
- Cálculo do nível da árvore (esperado: 3).

Todos os testes foram bem-sucedidos.

4. Conclusão

Este trabalho possibilitou o aprofundamento dos conceitos de árvores binárias e a aplicação prática de estruturas de dados e recursão. A maior dificuldade foi garantir o correto funcionamento dos métodos de remoção e verificação dos tipos da árvore.

5. Repositório do Projeto

O código-fonte completo do projeto, juntamente com uma versão resumida desta documentação, está disponível no seguinte repositório público:

-

6. Bibliografia

- Jovana, M. (2019). Estruturas de Dados em Java.
- Geeks for Geeks. <https://www.geeksforgeeks.org>
- Overleaf Team. (2025). Documentação em LaTeX. <https://www.overleaf.com>