

# Respostas – Atividade de Laboratório: Árvores Binárias

Gabriel Reboli, Gustavo Gazinnelli

9 de setembro de 2025

## Pergunta 1: O que muda na estrutura da árvore quando inserimos os valores em ordem crescente?

Absolutamente **nada** muda na estrutura (no formato) da árvore.

O método de inserção implementado, `insert_level_order`, preenche a árvore estritamente por nível, da esquerda para a direita, ocupando o primeiro espaço que encontrar. Esse algoritmo **ignora completamente o valor** do dado que está sendo inserido; sua lógica se baseia apenas na topologia da árvore.

Isso significa que a sequência de inserção `[1, 2, 3, 4, 5, 6]` resulta em uma árvore com uma estrutura **idêntica** à que seria gerada pelas sequências `[6, 5, 4, 3, 2, 1]` ou `[4, 1, 5, 6, 2, 3]`. O que muda são os valores armazenados em cada nó, mas o "desenho" da árvore, suas conexões e sua classificação estrutural permanecem os mesmos.

## Pergunta 2: Por que a árvore resultante não é perfeita?

A árvore resultante da inserção dos valores `[1, 2, 3, 4, 5, 6]` não é perfeita porque a definição de uma **árvore perfeita** exige que **todos os seus níveis estejam completamente preenchidos**.

Ao analisar a estrutura da árvore gerada, observamos o seguinte preenchimento dos níveis:

- **Nível 0:** Contém o nó 1. (Capacidade:  $2^0 = 1$ , Ocupado: 1. **OK**)
- **Nível 1:** Contém os nós 2 e 3. (Capacidade:  $2^1 = 2$ , Ocupado: 2. **OK**)
- **Nível 2:** Contém os nós 4, 5 e 6. (Capacidade:  $2^2 = 4$ , Ocupado: 3. **INCOMPLETO**)

Como o último nível (Nível 2) não está com sua capacidade máxima preenchida, a condição para que a árvore seja perfeita não é satisfeita.

Apesar de não ser perfeita, a árvore é classificada como **completa**, pois todos os níveis, exceto o último, estão cheios, e os nós do último nível estão alocados o mais à esquerda possível.