

Pedro Wilian Palumbo Bevilacqua

**Balanceamento de Árvores Binárias de Pesquisa:  
Uma Análise Comparativa entre Estruturas AVL,  
Rubro-Negras e B-Trees**

São Caetano do Sul

2025

Pedro Wilian Palumbo Bevilacqua

# **Balanceamento de Árvores Binárias de Pesquisa: Uma Análise Comparativa entre Estruturas AVL, Rubro-Negras e B-Trees**

Artigo apresentado como atividade para a disciplina de Tópicos Avançados em Estruturas de Dados, do curso de Engenharia da Computação do Instituto Mauá de Tecnologia.

Instituto Mauá de Tecnologia  
Escola de Engenharia  
Curso de Engenharia da Computação

Orientador: Prof. Dr. Robson Calvetti

São Caetano do Sul  
2025

# Resumo

O balanceamento de árvores binárias de pesquisa constitui um elemento essencial para manter a eficiência de operações fundamentais, como busca, inserção e remoção. Este trabalho desenvolve uma análise comparativa de três das principais técnicas de balanceamento: Árvores AVL, Rubro-Negras e B-Trees. A investigação percorre desde os fundamentos conceituais até as propriedades estruturais e os mecanismos de reorganização, estabelecendo relações entre essas características e os diferentes contextos de aplicação em sistemas computacionais contemporâneos.

O propósito é oferecer uma visão crítica e abrangente que auxilie profissionais e pesquisadores a compreender as potencialidades e limitações de cada abordagem, bem como a reconhecer os cenários em que cada uma se mostra mais apropriada — seja em ambientes que exigem consultas rápidas e consistentes, em sistemas sujeitos a intensas operações de atualização ou em aplicações que dependem fortemente de memória secundária, como bancos de dados e sistemas de arquivos.

**Palavras-chave:** Estruturas de Dados; Árvores Balanceadas; AVL; Rubro-Negra; B-Tree.

# Abstract

Balancing binary search trees is a key element for preserving the efficiency of core operations such as search, insertion, and deletion. This article presents a comparative analysis of three major balancing strategies: AVL Trees, Red-Black Trees, and B-Trees. The discussion ranges from conceptual foundations to structural properties and reorganization mechanisms, linking these aspects to different application scenarios in contemporary computing systems. The goal is to provide a critical and comprehensive perspective that enables researchers and practitioners to grasp the strengths and weaknesses of each structure, while identifying the contexts in which they are most effective—whether in environments that demand fast and consistent queries, in systems with frequent update operations, or in applications heavily dependent on secondary storage, such as databases and file management systems.

**Keywords:** Data Structures; Balanced Trees; AVL; Red-Black; B-Tree.

# Sumário

<b>0.1</b>	<b>Introdução . . . . .</b>	<b>5</b>
<b>0.2</b>	<b>Desenvolvimento . . . . .</b>	<b>5</b>
0.2.1	Árvore AVL . . . . .	5
0.2.2	Árvore Rubro-Negra . . . . .	5
0.2.3	Árvore B-Tree . . . . .	6
<b>0.3</b>	<b>Conclusão . . . . .</b>	<b>6</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>7</b>

## 0.1 Introdução

As árvores binárias de pesquisa (ABP) figuram entre as estruturas mais relevantes da ciência da computação, oferecendo uma forma organizada e eficiente de armazenar e recuperar informações. Em condições ideais, permitem que operações como busca, inserção e remoção sejam realizadas em tempo assintótico  $\mathcal{O}(\log n)$  [1].

No entanto, essa eficiência depende diretamente da forma como os elementos são inseridos. Em situações desfavoráveis, a árvore pode se tornar desbalanceada, aproximando-se de uma lista encadeada e elevando o custo das operações para  $\mathcal{O}(n)$  [2].

Com o intuito de mitigar esse problema, surgem as chamadas *árvores balanceadas*, projetadas para limitar o crescimento da altura da estrutura e, assim, preservar o desempenho teórico em cenários práticos. Este artigo tem como objetivo comparar três das técnicas de balanceamento mais difundidas: as Árvores AVL, as Árvores Rubro-Negras e as B-Trees, discutindo suas propriedades, vantagens, limitações e contextos de aplicação mais adequados.

## 0.2 Desenvolvimento

### 0.2.1 Árvore AVL

Propostas por Adelson-Velsky e Landis em 1962, as Árvores AVL foram a primeira estrutura autobalanceada da literatura. Seu princípio básico consiste em garantir que a diferença de altura entre as subárvores esquerda e direita de qualquer nó não ultrapasse uma unidade. Quando essa condição é violada, a árvore é reequilibrada por meio de rotações — simples ou duplas — classificadas como LL, RR, LR e RL [3].

Esse rigor garante excelente desempenho em operações de busca, mantendo a árvore próxima do ideal binário. Por outro lado, inserções e remoções podem demandar múltiplas rotações, aumentando o custo de atualização da estrutura. Por essa razão, as Árvores AVL são especialmente indicadas para cenários em que predominam consultas frequentes, como sistemas de indexação e mecanismos de recuperação de informação.

### 0.2.2 Árvore Rubro-Negra

As Árvores Rubro-Negras (*Red-Black Trees*) oferecem uma alternativa mais flexível, conciliando custo de manutenção e altura da árvore. Nessa abordagem, cada nó recebe uma cor (vermelho ou preto) e deve obedecer a regras que asseguram um equilíbrio relativo na distribuição dos nós. Caso essas regras sejam violadas, o balanceamento é restaurado por meio de rotações e recolorações locais [4].

Embora possam gerar árvores um pouco mais altas que as AVL, as Rubro-Negras exigem menos rotações durante inserções e remoções, o que as torna mais eficientes em

cenários de atualizações constantes. Essa característica explica sua adoção em bibliotecas padrão de linguagens de programação, como o `std::map` em C++ e o `TreeMap` em Java.

### 0.2.3 Árvore B-Tree

As B-Trees foram concebidas para cenários em que o acesso a armazenamento externo representa o principal gargalo de desempenho, como em bancos de dados e sistemas de arquivos. Diferentemente das árvores binárias convencionais, cada nó pode armazenar várias chaves e múltiplos ponteiros, aumentando o fator de ramificação e reduzindo a quantidade de acessos a disco [5].

Em uma B-Tree de ordem  $m$ , cada nó interno pode conter entre  $\lceil m/2 \rceil$  e  $m$  filhos, armazenando até  $m - 1$  chaves. Além disso, todas as folhas localizam-se no mesmo nível hierárquico. Esse desenho garante operações clássicas em complexidade  $\mathcal{O}(\log_m n)$ , tornando as B-Trees a estrutura predominante em sistemas gerenciadores de bancos de dados e em arquiteturas modernas de armazenamento.

## 0.3 Conclusão

A análise das diferentes técnicas de balanceamento evidencia que não existe uma solução universal, mas sim estruturas que se adequam melhor a determinados contextos. As Árvores AVL destacam-se pela eficiência em consultas, sendo indicadas para sistemas em que buscas superam largamente as atualizações. As Árvores Rubro-Negras, ao reduzirem o custo de manutenção estrutural, são recomendadas em cenários dinâmicos, com inserções e remoções frequentes. Já as B-Trees configuram-se como escolha natural para aplicações que lidam com grandes volumes de dados persistentes, onde a minimização de acessos a disco é fator determinante.

Assim, compreender em profundidade as propriedades e particularidades de cada técnica possibilita ao profissional de computação tomar decisões mais fundamentadas, alinhando as características da estrutura à natureza e às demandas do problema em questão.

## Referências

- [1] CORMEN, T. H. et al. *Algoritmos: Teoria e Prática*. Rio de Janeiro: Campus, 2002.
- [2] ZIVIANI, N. *Projeto de Algoritmos: Com Implementações em Pascal e C*. São Paulo: Thomson, 2004.
- [3] FEOFILOFF, P. *Minicurso de Análise de Algoritmos*. Universidade de São Paulo, 2010.
- [4] GOODRICH, M. T.; TAMASSIA, R. *Algorithm Design and Applications*. Hoboken: Wiley, 2015.
- [5] SKIENA, S. S. *The Algorithm Design Manual*. 2. ed. New York: Springer, 2008.