

UNIVERSIDADE DO ESTADO DE SANTA CATARINA - UDESC
TECNÓLOGO EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

GABRIEL OTÁVIO DE BARROS

IGOR JOSÉ DA COSTA MOTA

LORENZO GADOTTI MURARA BECKER MACHADO

OTÁVIO LUIZ MEDEIROS NANDI

**ANÁLISE DA COMPLEXIDADE ALGORÍTMICA NAS OPERAÇÕES DE ADIÇÃO
E REMOÇÃO EM ÁRVORES AVL, RUBRO-NEGRA E B**

JOINVILLE

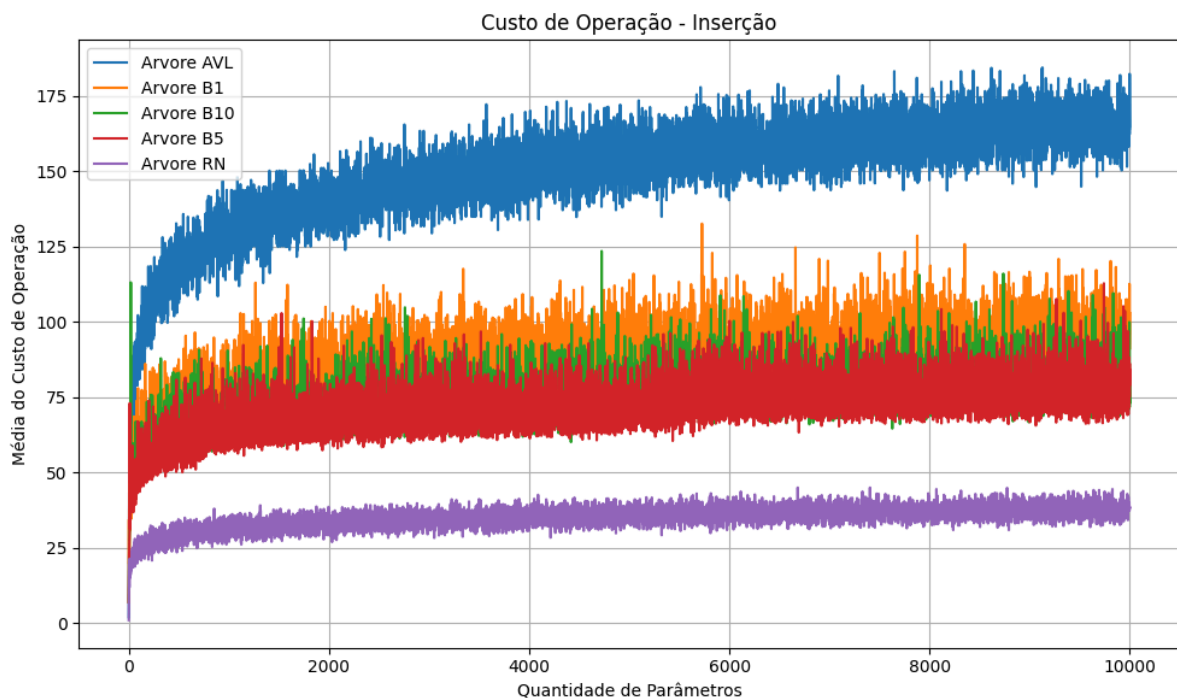
2023

1. OBJETIVO DO TRABALHO

O objetivo deste trabalho consiste em analisar a complexidade algorítmica das operações de adição e remoção de nós (considerando o balanceamento) em árvores AVL, rubro-negra e B. No caso da árvore B, devem ser considerados os parâmetros de ordem da árvore igual a 1, 5 e 10. A análise deve ser realizada considerando a geração de um conjunto de dados (chaves) com tamanho variando entre 1 e 10000. As chaves devem ser geradas reproduzindo um caso médio, isto é, utilizando chaves aleatórias. Para geração das chaves aleatórias, sugere-se o uso da função rand em C considerando uma quantidade de amostra de pelo menos 10 conjuntos para validade estatística, calculando a média dos resultados obtidos.

O resultado final do experimento deve ser apresentado em um relatório contendo dois gráficos de linha (um para a operação de adição e outro para a operação de remoção), onde o eixo X representa o tamanho dos conjuntos de dados (1 a 10000) e o eixo Y representa o esforço computacional das operações considerando eventuais balanceamentos. Cada gráfico deve apresentar 5 linhas, as quais representam as respectivas operações para cada estrutura de dados avaliada (árvore AVL, rubro-negra e as três variações da árvore B). O relatório também deve conter uma discussão final sobre os resultados obtidos. Além do relatório, os códigos-fonte utilizados nos experimentos devem ser entregues também.

2. OPERAÇÕES DE INSERÇÃO



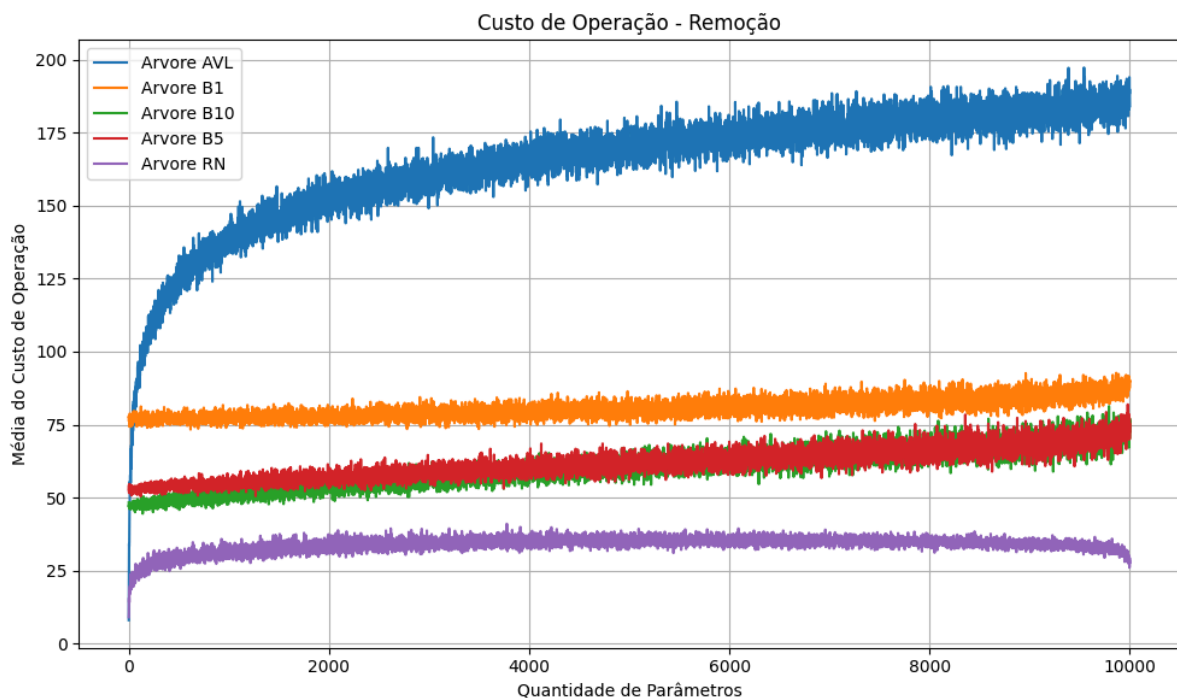
O gráfico compara o custo médio de inserção em diferentes tipos de árvores: AVL, Rubro-negra (RN) e Árvores B (ordens B1, B5 e B10). Todas essas estruturas apresentam complexidade logarítmica ($O(\log n)$) para inserção devido à necessidade de manter balanceamento.

Resumo:

- **Árvore AVL:** Apresenta o maior custo de inserção, pois realiza balanceamento estrito a cada operação, o que aumenta o número de rotações.
- **Árvore RN:** Tem um custo menor que a AVL, já que seu balanceamento é menos rigoroso, mantendo eficiência sem excesso de ajustes.
- **Árvores B:** O custo diminui com o aumento da ordem ($B10 < B5 < B1$), pois árvores de ordem maior armazenam mais chaves por nó, reduzindo a profundidade da árvore e o número de operações de balanceamento.

Embora todas tenham desempenho logarítmico, as Árvores B de maior ordem são mais eficientes para inserções, enquanto AVL e RN priorizam balanceamento para buscas rápidas.

3. OPERAÇÕES DE REMOÇÃO



O gráfico compara o custo médio de remoção em diferentes tipos de árvores: AVL, Rubro-negra (RN) e Árvores B (ordens B1, B5 e B10). Assim como na inserção, todas mantêm complexidade logarítmica ($O(\log n)$) devido à necessidade de balanceamento após a remoção.

Resumo:

- **Árvore AVL:** Apresenta o maior custo de remoção, pois realiza balanceamento estrito a cada operação, envolvendo várias rotações e ajustes para manter o fator de balanceamento.
- **Árvore RN:** O custo é menor que na AVL, já que o balanceamento é menos rigoroso, reduzindo a frequência de rotações necessárias.
- **Árvores B:** O custo de remoção diminui conforme a ordem aumenta ($B10 < B5 < B1$), pois árvores de maior ordem têm menor profundidade, resultando em menos operações para balanceamento e redistribuição.

Embora todas operem em ($O(\log n)$), as Árvores B de ordens maiores demonstram eficiência superior para remoções, enquanto AVL e RN priorizam balanceamento para manter um bom desempenho em operações futuras.

4. CONCLUSÃO

Conforme o número de itens inseridos ou removidos nas árvores se aproxima de 10.000, as diferenças no custo médio de operação ficam mais evidentes:

- **Árvore AVL:** Mantém o maior custo tanto para inserções quanto para remoções. Isso ocorre porque seu balanceamento estrito exige múltiplas rotações e ajustes a cada operação. À medida que a árvore cresce, essas operações aumentam em frequência, refletindo o alto custo observado nos gráficos.
- **Árvore Rubro-negra (RN):** Apesar de também realizar balanceamento, a Árvore RN é menos rigorosa que a AVL, o que reduz o número de rotações necessárias. Isso resulta em um custo menor e mais estável, mesmo com 10.000 itens, sendo uma boa opção quando buscas e alterações rápidas são necessárias sem custos operacionais elevados.
- **Árvores B (B1, B5 e B10):** O custo médio das operações nas Árvores B é significativamente menor, especialmente em ordens mais altas (B10 e B5). À medida que o número de itens cresce, essas árvores continuam eficientes devido à menor profundidade relativa, já que cada nó pode armazenar múltiplas chaves, reduzindo o número de operações de balanceamento. Entre as árvores B, a ordem B10 apresenta o melhor desempenho.

Sendo assim, conforme o número de operações aumenta para valores próximos a 10.000, a diferença no desempenho entre as estruturas é amplificada. Árvores AVL são ideais para cenários que exigem alto rigor no balanceamento, mas apresentam maior custo operacional. Árvores RN equilibram eficiência com simplicidade no balanceamento. Já Árvores B, especialmente em ordens altas, são as mais eficientes para operações em grandes volumes de dados, sendo a melhor escolha quando o objetivo é minimizar o custo de inserções e remoções.