

Relatório complexidade algorítmica das operações de adição e remoção de nós  
Após mensurar a complexidade algorítmica nas operações de inserção e remoção nas árvores AVL, rubro-negras e B, podemos concluir que todas essas estruturas de dados oferecem desempenho eficiente para essas operações.

Tanto as árvores AVL quanto as árvores rubro-negras têm complexidade  $O(\log n)$  para inserção e remoção, o que significa que o tempo necessário para executar essas operações aumenta de forma logarítmica com o número de elementos na árvore. Essas estruturas de dados são particularmente adequadas quando se deseja manter um equilíbrio preciso da árvore e garantir tempos de busca, inserção e remoção consistentes em diferentes cenários.

Por outro lado, as árvores B têm complexidade  $O(\log_m n)$  para inserção e remoção, onde  $m$  é a ordem da árvore (no trabalho foi considerado os parâmetros de ordem igual a 1, 5 e 10).

Árvores AVL e Rubro-Negras: São mais adequadas quando o equilíbrio preciso da árvore é crucial para garantir um desempenho consistente. Essas árvores são indicadas para aplicações que exigem um alto grau de busca, inserção e remoção de elementos com tempos de execução previsíveis e garantidos em  $O(\log n)$ . A árvore rubro negra pode ser considerada mais complexa que a AVL visto que ainda trata a cor do atributo porém não resulta em um consumo significativamente maior de memória.

Árvores B: São mais adequadas quando o armazenamento de grandes conjuntos de dados em memória secundária é uma prioridade. Como nosso algoritmo trabalha exclusivamente na memória principal e não envolve acesso a disco, a utilização de uma árvore B pode não trazer benefícios significativos em relação às árvores AVL ou rubro-negras.

Gráfico do Caso Médio da Adição

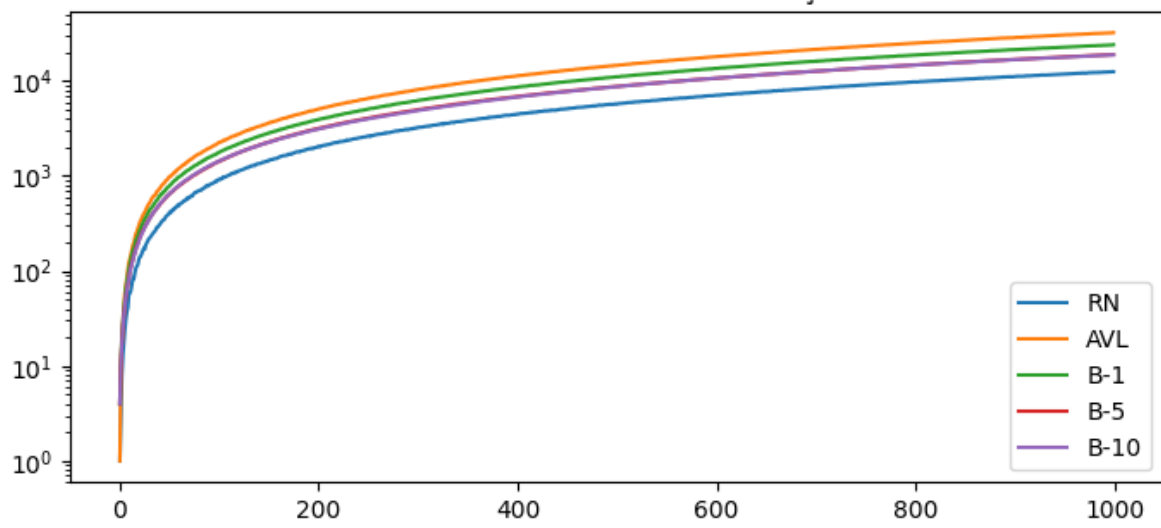


Gráfico do Caso Médio da Remoção

