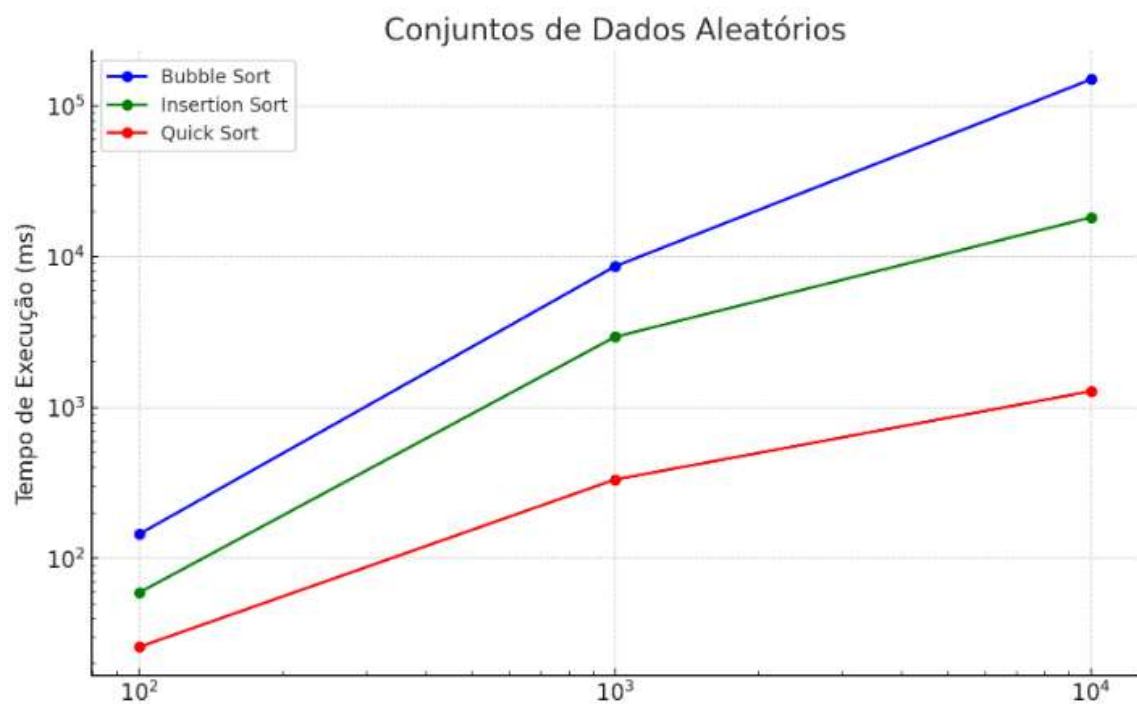


RELATÓRIO DE ANÁLISE DE ALGORITMOS DE ORDENAÇÃO

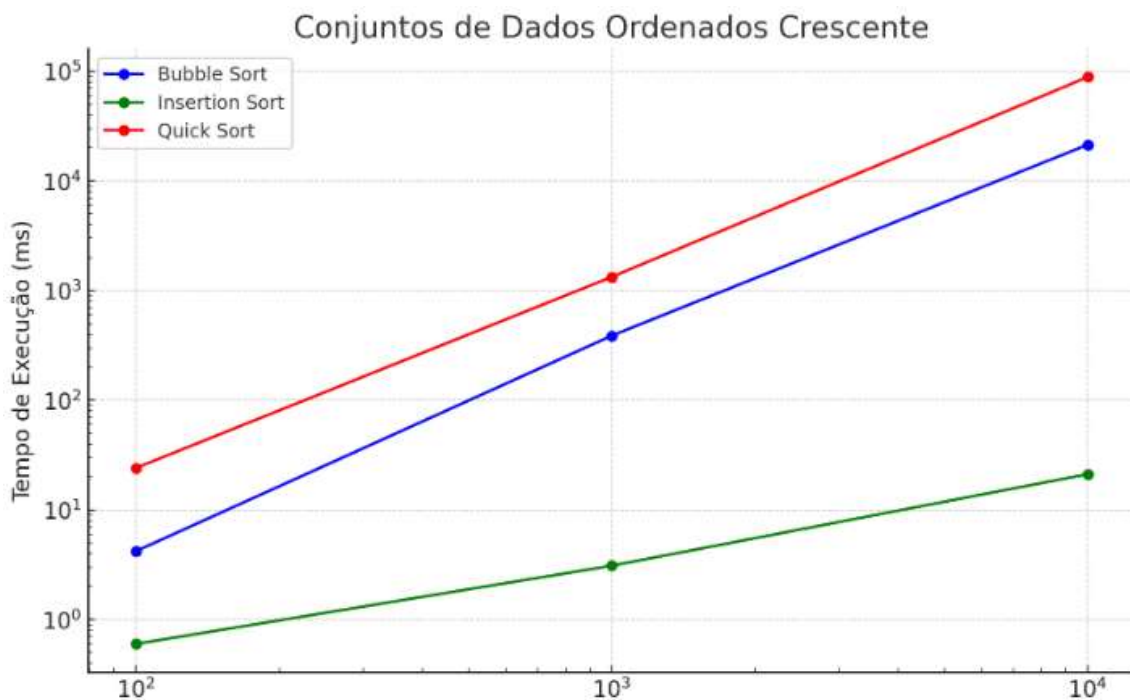
1. Conjuntos de Dados Aleatórios

| Tamanho | Bubble Sort | Insertion Sort | Quick Sort |
|---------|----------------|----------------|--------------|
| 100 | 145,100 ns | 59,400 ns | 25,900 ns |
| 1,000 | 8,665,500 ns | 2,950,400 ns | 334,100 ns |
| 10,000 | 150,431,300 ns | 18,288,900 ns | 1,286,100 ns |



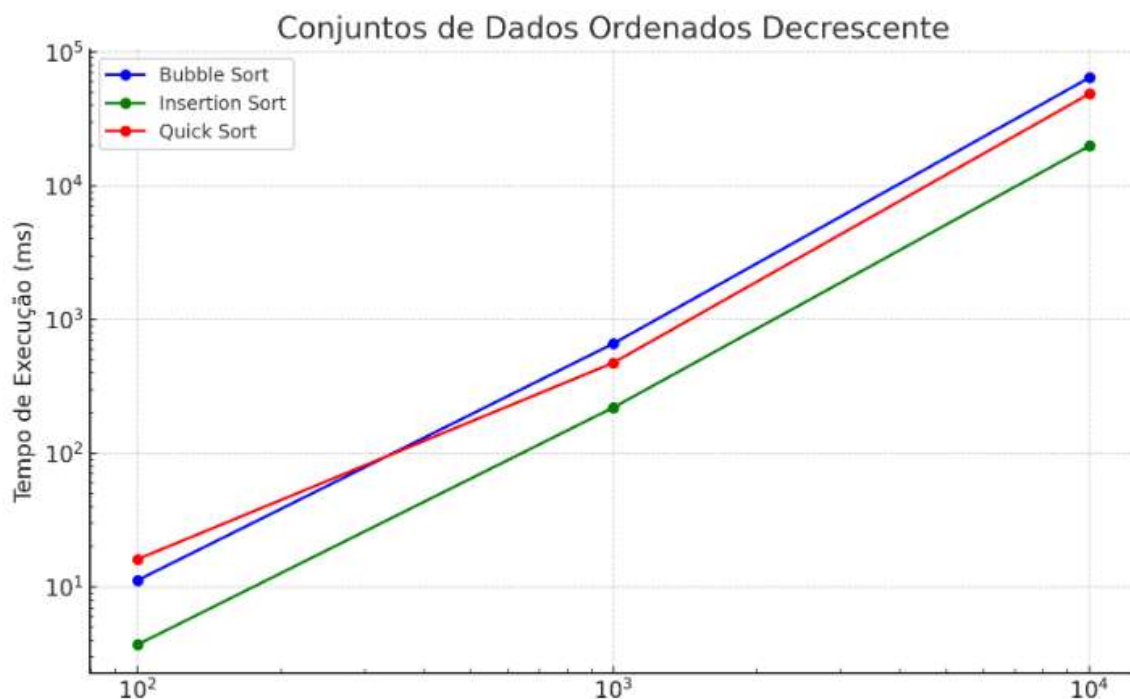
2. Conjuntos de Dados Ordenados Crescente

| Tamanho | Bubble Sort | Insertion Sort | Quick Sort |
|---------|---------------|----------------|---------------|
| 100 | 4,200 ns | 600 ns | 24,100 ns |
| 1,000 | 388,000 ns | 3,100 ns | 1,333,100 ns |
| 10,000 | 21,401,700 ns | 21,200 ns | 88,679,200 ns |



3. Conjuntos de Dados Ordenados Decrescente

| Tamanho | Bubble Sort | Insertion Sort | Quick Sort |
|---------|---------------|----------------|---------------|
| 100 | 11,200 ns | 3,700 ns | 16,100 ns |
| 1,000 | 660,500 ns | 219,700 ns | 474,700 ns |
| 10,000 | 64,320,000 ns | 19,855,100 ns | 48,703,600 ns |



Conjuntos de Dados Aleatórios

Bubble Sort é o mais lento em todos os tamanhos devido à sua complexidade $O(n^2)$.

Insertion Sort mostra uma melhora significativa, especialmente com tamanhos menores, mas também possui uma complexidade $O(n^2)$ que limita seu desempenho para grandes volumes.

Quick Sort é o mais rápido dos três, devido à sua complexidade média $O(n \log n)$, especialmente para grandes volumes como 10,000 elementos.

Conjuntos de Dados Ordenados Crescente

Insertion Sort é extremamente rápido para dados já ordenados (quase $O(n)$), pois ele apenas verifica se os elementos já estão em ordem.

Bubble Sort ainda é lento, mas apresenta uma melhoria porque ele pode interromper mais cedo se detectar que os dados estão ordenados.

Quick Sort pode ter pior desempenho com dados já ordenados, especialmente se o pivô escolhido for ineficaz, como o último elemento, levando a uma complexidade $O(n^2)$.

Conjuntos de Dados Ordenados Decrescente

Bubble Sort e Insertion Sort são afetados pela ordenação reversa, mas Insertion Sort se sai melhor com tamanhos menores.

Quick Sort é mais rápido que Bubble Sort, mas mais lento que Insertion Sort devido ao aumento nas comparações e trocas para restaurar a ordem.