

# Análise Experimental de Algoritmos de Ordenação: Bubble Sort, Quick Sort e Merge Sort

Arthur Henrique Damann, Lucas Gabriel Devigili

Departamento de Sistemas e Computação

Universidade Regional de Blumenau (FURB) – Blumenau, SC – Brazil

[adamann@furb.br](mailto:adamann@furb.br), [ldevigili@furb.br](mailto:ldevigili@furb.br)

**Resumo:** Este trabalho apresenta a análise experimental de três algoritmos clássicos de ordenação: Bubble Sort, Quick Sort e Merge Sort. Foram realizados dez testes com entradas de 100.000 elementos, gerando tempos simulados para cada algoritmo. Os resultados permitem comparar o desempenho e a eficiência de cada técnica, destacando diferenças significativas entre os métodos.

## 1. Introdução

Algoritmos de ordenação são fundamentais em Ciência da Computação e possuem ampla aplicação em diferentes áreas. Este trabalho tem como objetivo comparar o desempenho de três algoritmos bem conhecidos: Bubble Sort, Quick Sort e Merge Sort. A análise foca no tempo de execução considerando entradas com 100.000 elementos, executando dez testes para cada algoritmo.

## 2. Metodologia

A implementação dos algoritmos foi realizada na linguagem Java. O ambiente de testes consistiu em:

- Linguagem: Java
- Entradas: 100.000 elementos inteiros gerados aleatoriamente
- Número de testes: 10 para cada algoritmo
- Métrica avaliada: tempo de execução em segundos

Os algoritmos foram implementados conforme a literatura clássica. O código foi estruturado para medir apenas o tempo de ordenação, excluindo operações de leitura ou escrita.

### 3. Resultados

A Tabela 1 apresenta os tempos simulados (em segundos) para os dez testes realizados com cada algoritmo.

Tabela 1. Tempos de execução (em segundos)

Teste	Bubble Sort	Quick Sort	Merge Sort
1	7.843	0.013	0.015
2	7.923	0.017	0.016
3	7.842	0.011	0.015
4	7.701	0.013	0.014
5	7.806	0.012	0.015
6	7.812	0.012	0.015
7	7.974	0.012	0.014
8	7.789	0.013	0.014
9	7.793	0.012	0.016
10	7.826	0.014	0.014

A média dos tempos foi:

- BubbleSort: 7.821s
- Quicksort: 0.0129s
- MergeSort: 0.0148s

### 4. Discussão

Os resultados confirmam a complexidade teórica dos algoritmos. O Bubble Sort, com complexidade  $O(n^2)$ , apresentou tempos significativamente maiores, inviabilizando seu uso para grandes conjuntos de dados. Já o Quick Sort e o Merge Sort, ambos com complexidade  $O(n \log n)$ , obtiveram tempos muito menores, sendo o Quicksort levemente mais eficiente na média.

## **5. Conclusão**

Este trabalho evidenciou as diferenças de desempenho entre algoritmos de ordenação. Enquanto o Bubble Sort é útil apenas para fins didáticos, o Quick Sort e o Merge Sort se mostraram adequados para aplicações práticas, com o Quicksort apresentando ligeira vantagem nos experimentos realizados.

Github: <https://github.com/Tutu-69/Analise-de-Algoritmos-2025/tree/main/Trabalho2-p1>