

# Primeira versão do projeto da disciplina

## Comparação entre os algoritmos de ordenação elementar

---

---

## 1. Introdução

Este relatório corresponde ao relato dos resultados obtidos no projeto da disciplina de LEDA que tem como objetivo criar um comparativo de performance entre os algoritmos de ordenação elementares: selection sort, insertion sort, counting sort, heap sort, merge sort e quick sort. Para isso, uma coleção de dados foi processada por cada algoritmo, e foram coletados dados sobre a execução de cada ordenação para quantificar sua performance.

## 2. Descrição geral sobre o método utilizado

O conjunto de dados extraído a partir dos dados tabulares foi submetido a ordenação em diferentes condições, no qual o uso de memória e tempo de execução foram medidos. Para simular o pior caso, os dados foram ordenados de forma decrescente; para simular o melhor caso, os dados foram ordenados de forma crescente antes de serem processados por cada algoritmo, e para o caso médio não houve nenhum tratamento prévio antes do processamento (Descrever como os testes foram feitos )

A ferramenta foi implementada na linguagem *Java*, na qual cada algoritmo executa a ordenação de *arrays* de objetos derivados da interface *Comparable*. Cada algoritmo de ordenação modifica o *array* inicial e retorna um segundo array contendo os índices modificados de cada elemento. Com esses dados é possível reconstruir o conjunto de dados original a partir dos índices sem carregar o arquivo inteiro em memória. (Descrever a implementação da sua ferramenta)

## Descrição geral do ambiente de testes

---

---

A máquina utilizada como ambiente de testes possui processador AMD A12-9720p (4 núcleos, 4 *threads*), 8GB de memória do tipo DDR4 em sistema operacional Windows 10 versão 22H2 de 64 bits. A versão do JDK utilizada foi 1.8.0\_341.

### **3. Resultados e Análise**