

Trabalho 2 – Algoritmos e Programação: Estruturas de Dados

## Análise Comparativa dos Métodos de Ordenação

**Objetivo:** O objetivo deste trabalho é realizar uma análise comparativa *de tempo* dos principais métodos de ordenação apresentados em aula. Os métodos a serem estudados são: Bubble Sort, Insertion Sort, Selection Sort, Heap Sort, Shell Sort, Merge Sort e Quick Sort.

**Modalidade:** Grupos de 1 a 3 participantes.

### Instruções

1. Os estudantes devem implementar os algoritmos de ordenação mencionados.
2. Para cada método de ordenação, os estudantes devem realizar testes de desempenho em diferentes cenários. Os testes devem ser conduzidos em arrays de diferentes tamanhos e em quatro cenários distintos:
  - o Array ordenado em ordem crescente, sem valores repetidos;
  - o Array ordenado em ordem decrescente, sem valores repetidos;
  - o Array aleatória sem valores repetidos; e
  - o Array aleatória com valores repetidos.
3. Após a geração dos cenários (em memória ou em arquivo), deverá ser analisado para cada método de ordenação e tamanho “n” do array (primeira coluna da Tabela 1), observando que os métodos devem receber o mesmo array original. O tempo de processamento deverá ser computado em nanossegundos.
4. Os resultados dos testes devem ser registrados em tabelas para cada cenário, contendo os tempos de execução de cada método de ordenação para diferentes tamanhos de array.

### Estrutura da Tabela:

Tabela 1. Tempos computados para um cenário

Tamanho do Array (n)	Bubble Sort	Insertion Sort	Selection Sort	Heap Sort	Shell Sort	Merge Sort	Quick Sort
128							
256							
512							
1024							
2048							
4096							
...							
65536							

5. Deverá ser realizado 10 execuções armazenando o tempo de processamento para cada cenário, método e tamanho de array. A partir daí, calcula-se a média destas 10 execuções. Feito isso, calcula-

se com base nestas 10 execuções a variância da amostra e, consequentemente, o desvio padrão. Com base no desvio padrão, será desconsiderado os tempos destas 10 execuções que estejam fora da média  $\pm$  desvio padrão. Por fim, calcula-se o tempo médio somente dos valores que estiverem dentro desta faixa. Vamos ao exemplo?

Suponha que após 10 execuções para o cenário, método de ordenação e tamanho do array, geraram os seguintes tempos em nanossegundos: 27633, 16974, 30396, 24081, 23686, 13816, 23291, 21317, 39081 e 25264. Abaixo, mostram-se os passos para chegarmos à média final.

#### a) Cálculo da média

A fórmula para calcular a média ( $\bar{x}$ ) de um conjunto de valores é a seguinte:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$$

Onde:

- $\bar{x}$  é a média.
- $n$  é o número de elementos no conjunto.
- $x_i$  são os valores individuais do conjunto.

Para os valores fornecidos:

$$\begin{aligned}\bar{x} &= \frac{27633 + 16974 + 30396 + 24081 + 23686 + 13816 + 23291 + 21317 + 39081 + 25264}{10} \\ &= \frac{245539}{10} \\ &= 24553.9\end{aligned}$$

#### b) Cálculo da variância

A fórmula para calcular a variância  $s^2$  de um conjunto de valores é a seguinte:

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$$

Onde:

- $s^2$  é a variância.
- $n$  é o número de elementos no conjunto.
- $x_i$  são os valores individuais do conjunto.
- $\bar{x}$  é a média.

Para os valores fornecidos:

$$\begin{aligned}s^2 &= \frac{(27633 - 24553.9)^2 + (16974 - 24553.9)^2 + \dots + (25264 - 24553.9)^2}{9} \\ s^2 &= \frac{440958568.9}{9} = 48995396.54\end{aligned}$$

#### c) Cálculo do Desvio Padrão

O desvio padrão  $s$  é a raiz quadrada da variância:

$$s = \sqrt{s^2}$$

Para os valores fornecidos:

$$s = \sqrt{48995396.54} \approx 6999.67$$

Agora, com base na média e no desvio padrão, podemos identificar quais valores estão dentro do intervalo  $\bar{x} \pm s$  e calcular a média somente desses valores. Vou prosseguir com esse cálculo.

**Intervalo (média  $\pm$  desvio padrão):** 24553.9  $\pm$  6999.67

Valores dentro do intervalo: 27633, 30396, 24081, 23686, 23291, 21317, 25264

**Média somente dos valores dentro do intervalo:**

$$\frac{27633+30396+24081+23686+23291+21317+25264}{7} = \frac{175668}{7} = 25095.42$$

Portanto, a média somente dos valores dentro do intervalo é aproximadamente **25095 nanossegundos**.

6. Deverão ser entregues, via github (ou outro repositório de código):
  - Os códigos fontes (algoritmos de ordenação + geração dos testes)
  - Num arquivo .md, as tabelas com os resultados + informações pertinentes

**BOM TRABALHO!** 😊

Dica: lembre-se que um problema grande pode ser decomposto em problemas menores.