Ficha de comparação de algoritmos de ordenação 2022/1

|  |  |
| --- | --- |
| *Nome do(a) aluno(a):* | Matheus Barbosa Ferreira |

## Especificação do equipamento

|  |  |
| --- | --- |
| **Sistema Operacional** | Windows 10 |
| **Processador** | I5 10° |
| **Arquitetura (32 ou 64 bits)** | 64 bits |
| **Memória RAM Total** | 8 RAM |
| **Versão do Python** | 3.10 |

Para que a análise de eficiência (abaixo) tenha resultados comparáveis, todos os testes precisam ser realizados no mesmo equipamento (descrito acima).

## Análise de eficiência

Anotar, no quadro abaixo, o tempo e a memória dispendidos para a execução de cada algoritmo trabalhando sobre diferentes tamanhos de amostra.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Tamanho da amostra | | | | | | | |
| Algoritmo | **10.000** | | **25.000** | | **50.000** | | **100.000** | |
| **Bubble Sort** | *Tempo* |  | *Tempo* |  | *Tempo* |  | *Tempo* |  |
| *Memória* |  | *Memória* |  | *Memória* |  | *Memória* |  |
| **Selection Sort** | *Tempo* |  | *Tempo* |  | *Tempo* |  | *Tempo* |  |
| *Memória* |  | *Memória* |  | *Memória* |  | *Memória* |  |
| **Merge Sort RECURSIVO** | *Tempo* | **198.01974296569824ms** | *Tempo* | **503.56173515319824ms** | *Tempo* | 1273.484230041504ms | *Tempo* | 2609.7888946533203ms |
| *Memória* | **0.24070167541503906MB** | *Memória* | **0.5978260040283203MB** | *Memória* | 1.1947269439697266MB | *Memória* | 2.2983264923095703MB |
| **Merge Sort ITERATIVO** | *Tempo* |  | *Tempo* |  | *Tempo* |  | *Tempo* |  |
| *Memória* |  | *Memória* |  | *Memória* |  | *Memória* |  |
| **Quick Sort RECURSIVO** | *Tempo* |  | *Tempo* |  | *Tempo* |  | *Tempo* |  |
| *Memória* |  | *Memória* |  | *Memória* |  | *Memória* |  |
| **Quick Sort ITERATIVO** | *Tempo* |  | *Tempo* |  | *Tempo* |  | *Tempo* |  |
| *Memória* |  | *Memória* |  | *Memória* |  | *Memória* |  |
| **MELHOR RESULTADO** | *Tempo* |  | *Tempo* |  | *Tempo* |  | *Tempo* |  |
| *Memória* |  | *Memória* |  | *Memória* |  | *Memória* |  |

## Passadas, comparações e trocas

Anotar, no quadro abaixo, o tempo e a memória dispendidos para a execução de cada algoritmo trabalhando sobre diferentes tamanhos de amostra.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Tamanho da amostra | | | | | | | |
| Algoritmo | **10.000** | | **25.000** | | **50.000** | | **100.000** | |
| **Bubble Sort** | *Passadas* |  | *Passadas* |  | *Passadas* |  | *Passadas* |  |
| *Comparações* |  | *Comparações* |  | *Comparações* |  | *Comparações* |  |
| *Trocas* |  | *Trocas* |  | *Trocas* |  | *Trocas* |  |
| **Selection Sort** | *Passadas* |  | *Passadas* |  | *Passadas* |  | *Passadas* |  |
| *Comparações* |  | *Comparações* |  | *Comparações* |  | *Comparações* |  |
| *Trocas* |  | *Trocas* |  | *Trocas* |  | *Trocas* |  |
| **Quick Sort RECURSIVO** | *Passadas* |  | *Passadas* |  | *Passadas* |  | *Passadas* |  |
| *Comparações* |  | *Comparações* |  | *Comparações* |  | *Comparações* |  |
| *Trocas* |  | *Trocas* |  | *Trocas* |  | *Trocas* |  |
| **Quick Sort ITERATIVO** | *Passadas* |  | *Passadas* |  | *Passadas* |  | *Passadas* |  |
| *Comparações* |  | *Comparações* |  | *Comparações* |  | *Comparações* |  |
| *Trocas* |  | *Trocas* |  | *Trocas* |  | *Trocas* |  |
| **MELHOR RESULTADO** | *Passadas* |  | *Passadas* |  | *Passadas* |  | *Passadas* |  |
| *Comparações* |  | *Comparações* |  | *Comparações* |  | *Comparações* |  |
| *Trocas* |  | *Trocas* |  | *Trocas* |  | *Trocas* |  |