Ficha de comparação de algoritmos de ordenação 2023/1

|  |  |
| --- | --- |
| *Nome do(a) aluno(a):* | João Pedro Borges Araújo |

## Especificação do equipamento

|  |  |
| --- | --- |
| **Sistema Operacional** | Windows 11 |
| **Processador** | Core i5-4790 |
| **Arquitetura (32 ou 64 bits)** | x64 |
| **Memória RAM Total** | 16gb DDR3 |
| **Versão do Python** | 3.11.1 |

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* | **110s** ≅ 1min e 50s | *Tempo* | 709s ≅ 11min e 49s | *Tempo* | 2903s ≅ 48min e 23s | *Tempo* | 11.782s ≅ 3h16min |
| *Memória* | **10,223 MB** | *Memória* | **24,248 MB** | *Memória* | 48,246 MB | *Memória* | 96,256 MB |
| **Selection Sort** | *Tempo* |  | *Tempo* |  | *Tempo* |  | *Tempo* |  |
| *Memória* |  | *Memória* |  | *Memória* |  | *Memória* |  |
| **Merge Sort RECURSIVO** | *Tempo* |  | *Tempo* |  | *Tempo* |  | *Tempo* |  |
| *Memória* |  | *Memória* |  | *Memória* |  | *Memória* |  |
| **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* | **9.942** | *Passadas* | **24.675** | *Passadas* | 49.353 | *Passadas* | 99.259 |
| *Comparações* | **99.410.058** | *Comparações* | **616.850.325** | *Comparações* | 2.467.600.647 | *Comparações* | 9.925.800.741 |
| *Trocas* | **25.297.553** | *Trocas* | **157.485.392** | *Trocas* | 625.828.205 | *Trocas* | 2.498.177.562 |
| **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* |  |

## Considerações finais

Descreva, no quadro abaixo, suas considerações acerca do resultado da comparação dos algoritmos de ordenação.