Ficha de comparação de algoritmos de ordenação

|  |  |
| --- | --- |
| *Nome do(a) aluno(a):* | Victor Vendrera Vieira |

## Especificação do equipamento

|  |  |
| --- | --- |
| **Sistema Operacional** | Windows 10 |
| **Processador** | Core i7 |
| **Arquitetura (32 ou 64 bits)** | 64bits |
| **Memória RAM Total** | 24 |
| **Versão do Node.js** | 14.16.1 |

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 | **8.500** | | **35.000** | | **140.000** | | **282.000** | |
| **Bubble Sort** | *Tempo* | **613.633ms** | *Tempo* | **9.623s** | *Tempo* | 5:24.873 (m:ss.mmm) | *Tempo* | 25:10.683 (m:ss.mmm) |
| *Memória* | **4.785** | *Memória* | **11.789** | *Memória* | 38.213 | *Memória* | 75.520 |
| **Selection Sort** | *Tempo* | **284.845ms** | *Tempo* | **4.587s** | *Tempo* | 2:19.386 (m:ss.mmm) | *Tempo* | 11:18.912 (m:ss.mmm) |
| *Memória* | **5.503** | *Memória* | **12.446** | *Memória* | 47.179 | *Memória* | 85.203 |
| **Merge Sort** | *Tempo* | **13.871ms** | *Tempo* | **32.724ms** | *Tempo* | 87.212ms | *Tempo* | 178.128ms |
| *Memória* | 7.003 | *Memória* | 14.561 | *Memória* | 57.282 | *Memória* | 104.864 |
| **Quick Sort** | *Tempo* | 21.414 | *Tempo* | 32.078ms | *Tempo* | 96.104ms | *Tempo* | 202.218ms |
| *Memória* | 6.568 | *Memória* | 12.640 | *Memória* | 44.702 | *Memória* | 77.418 |
| **MELHOR RESULTADO** | *Tempo* | Merge sort 13.871ms | *Tempo* | Quick sort 32.078ms | *Tempo* | Merge sort 87.212ms | *Tempo* | Merge sort 178.128ms |
| *Memória* | Bubble sort 4.785 | *Memória* | Bubble sort 11789 | *Memória* | Bubble sort 38213 | *Memória* | Bubble sort 75520 |