

Instituto Federal Do Norte De Minas Gerais - Campus Januária

Curso: Bacharelado Em Sistemas De Informação

Disciplina: Estrutura de Dados 2

Atividade : Trabalho Prático - Ordenação

Docente: Paulo Veloso Santos Junior

Discente: Lucas Figueiredo Câmara

Relatório sobre os resultados encontrados nos testes dos quatro algoritmos de ordenação, Quicksort, Shellsort, Mergesort e Heapsort respectivamente.

Ordenando vetor com números aleatórios de 10.000 posições.

```
O tamanho do vetor é 10000
----- Início do programa -----

----- Ordenação com | QuickSort -----
Tempo : 0.001911
  Qnt Comp: 36823 | Qnt Troca: 34462
----- Ordenação com | Shellsort -----
Tempo : 0.003032
  Qnt Comp: 75243 | Qnt Troca: 154650
----- Ordenação com | Mergesort -----
Tempo : 0.003405
  Qnt Comp: 120411 | Qnt Troca: 59054
----- Ordenação com | Heapsort -----
Tempo : 0.005697
  Qnt Comp: 57758 | Qnt Troca: 37758
```

10.000 posições	qnt Comparações	qnt Troca	Tempo
Maior	120441/ Merge	154650/ Shell	0.005697/ Heap
Menor	36823/ Quick	34462/ Quick	0.001911/ Quick

Ordenando vetor com números aleatórios de 30.000 posições.

```
O tamanho do vetor é 30000
----- Início do programa -----

----- Ordenação com | QuickSort -----

Tempo : 0.006856

  Qnt Comp: 121623 | Qnt Troca: 114647

----- Ordenação com | Shellsort -----

Tempo : 0.012184

  Qnt Comp: 255719 | Qnt Troca: 661241

----- Ordenação com | Mergesort -----

Tempo : 0.010600

  Qnt Comp: 408547 | Qnt Troca: 201724

----- Ordenação com | Heapsort -----

Tempo : 0.016383

  Qnt Comp: 173552 | Qnt Troca: 113552
```

30.000 posições	qnt Comparações	qnt Troca	Tempo
Maior	408547/ Merge	661241/ Shell	0.006856/ Quick
Menor	173552/ Heap	12552/ Heap	0.016363/ Heap

Ordenando vetor com números aleatórios de 50.000 posições.

```
O tamanho do vetor é 50000
----- Início do programa -----

----- Ordenação com | QuickSort -----
Tempo : 0.014871
  Qnt Comp: 210619 | Qnt Troca: 199082
----- Ordenação com | Shellsort -----
Tempo : 0.022107
  Qnt Comp: 455719 | Qnt Troca: 1232008
----- Ordenação com | Mergesort -----
Tempo : 0.018151
  Qnt Comp: 717996 | Qnt Troca: 354716
----- Ordenação com | Heapsort -----
Tempo : 0.026340
  Qnt Comp: 289372 | Qnt Troca: 189372
```

50.000 posições	qnt Comparações	qnt Troca	Tempo
Maior	717996/Merge	1232007/Shell	0.026340/Heap
Menor	210619/Quick	189372/Heap	0.014871/Quick

Ordenando vetor com números aleatórios de 90.000 posições.

```
O tamanho do vetor é 90000
----- Início do programa -----

----- Ordenação com | QuickSort -----

Tempo : 0.026138

  Qnt Comp: 398175 | Qnt Troca: 377135

----- Ordenação com | Shellsort -----

Tempo : 0.048881

  Qnt Comp: 857146 | Qnt Troca: 2538927

----- Ordenação com | Mergesort -----

Tempo : 0.015149

  Qnt Comp: 1369435 | Qnt Troca: 675577

----- Ordenação com | Heapsort -----

Tempo : 0.022495

  Qnt Comp: 521140 | Qnt Troca: 341140
```

90.000 posições	qnt Comparações	qnt Troca	Tempo
Maior	1369435/ Merge	2538927/ Shell	0.048881/ Shell
Menor	398175/ Quick	341140/ Heap	0.015149/ Merge

Ordenando vetor com números aleatórios de 100.000 posições.

```
O tamanho do vetor é 100000
----- Início do programa -----

----- Ordenação com | QuickSort -----

Tempo : 0.029464

  Qnt Comp: 444338 | Qnt Troca: 420889

----- Ordenação com | Shellsort -----

Tempo : 0.049521

  Qnt Comp: 967146 | Qnt Troca: 2858440

----- Ordenação com | Mergesort -----

Tempo : 0.015936

  Qnt Comp: 1536522 | Qnt Troca: 760002

----- Ordenação com | Heapsort -----

Tempo : 0.025737

  Qnt Comp: 578820 | Qnt Troca: 378820
```

100.000 posições	qnt Comparações	qnt Troca	Tempo
Maior	1536522/ Merge	420889/ Quick	0.04921/ Shell
Menor	444338/ Quick	2858440/ Shell	0.015936/ Merge

Analisando as tabelas, em relação ao desempenho dos cinco teste executados, resumo que:

- Merge e Quick tiveram a maior e menor quantidade de comparações respectivamente
- Quanto à quantidade de troca, Shell ficou duas vezes com a maior quantidade e Quick uma, na menor quantidade Quick e Shell aparece uma vez, Heap aparece três.
- Quanto ao menor tempo de execução, Quick e Merge tiveram melhor desempenho por duas vezes, e Heap apenas uma. No maior tempo de execução Shell e Heap aparecem duas vezes no referido teste, Quick apenas uma.
- Aparentemente Shell e Merge faz maior uso de processador, pois um executa um elevado número de trocas e o outro elevada quantidade de comparações
- Heap, quanto aos testes presente teve desempenho equilibrado, tendo baixa escalabilidade quanto a necessidade de trocas e comparações, e tempo mediano.