

Complexidade de Algoritmos

Projeto e Análise de Algoritmos | Trabalho Prático 01



Universidade Estadual Paulista “Julio de Mesquita Filho”

Câmpus de Presidente Prudente

2012

O Projeto

Este projeto tem como objetivo analisar o desempenho e complexidade de diversos algoritmos de ordenação, no qual foram implementados nas linguagens PHP e Java, podendo assim gerarmos uma comparação entre ambas linguagens após a realização de vários testes de execução com entradas possuindo quantidade significativa de elementos ordenados de forma aleatória, acedente (do menor para o maior) ou descendente (do maior para o menor).

As linguagens



Os algoritmos em PHP podem ser executados em tempo real na aplicação sendo executada a partir de um servidor web com suporte a linguagem, para cada algoritmo fizemos uma análise do tempo e comparação para cada tipo de entrada. Em Java, desenvolvemos uma aplicação com o mesmo objetivo de executar os algoritmos de ordenação, também com análise de tempo e entradas com diferentes formas de ordenação e quantidade de elementos, para uma análise mais concreta executamos esses algoritmos 10 vezes cada.

Ao realizarmos diversas execuções dos algoritmos percebemos uma vantagem bastante superior da linguagem Java comparado ao PHP. Iremos analisar detalhadamente esses dados na sessão "Relatórios".

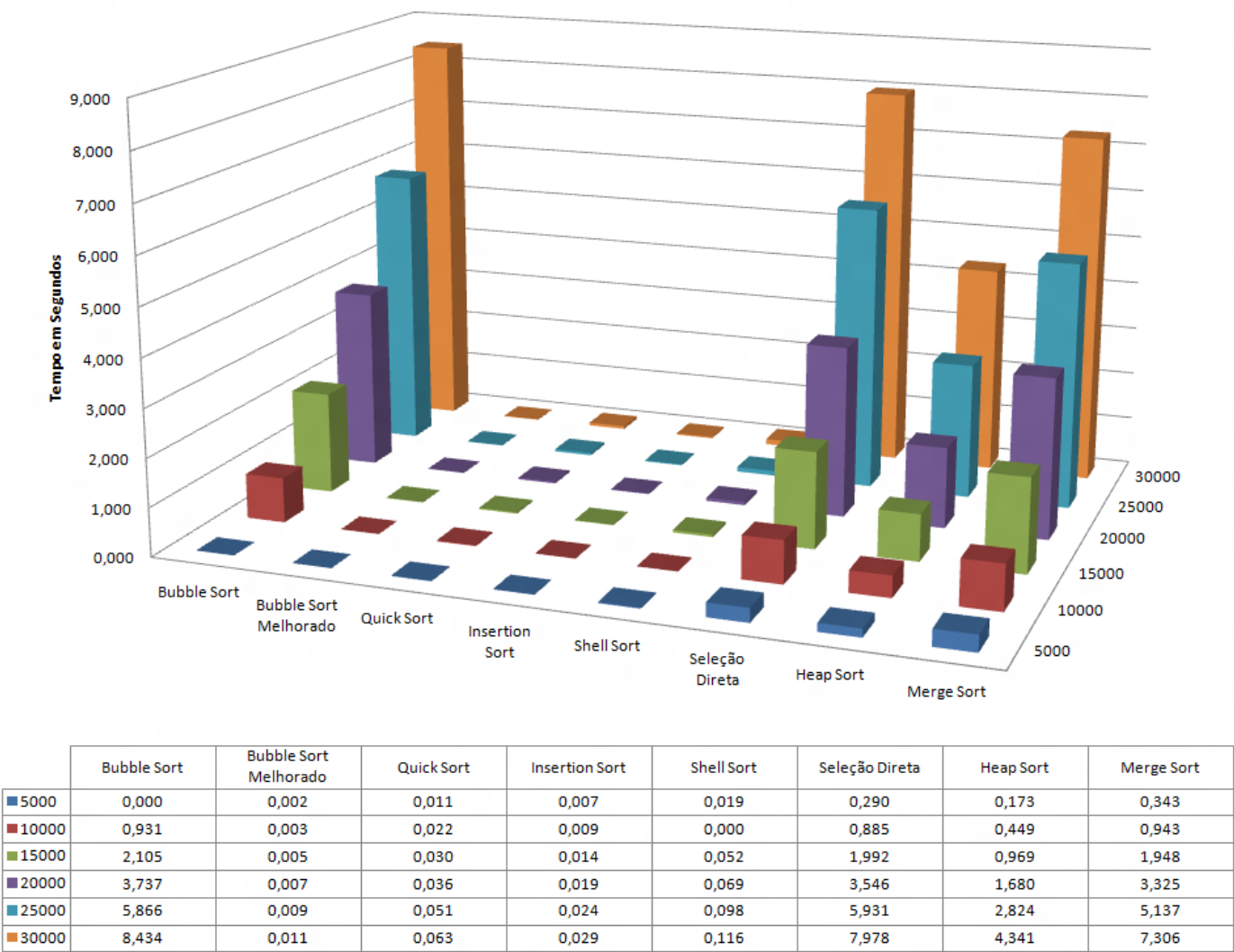
Relatórios

Os gráficos a seguir apresentam os dados de execução dos algoritmos nas duas linguagens utilizadas. Alguns detalhes devem ser considerados:

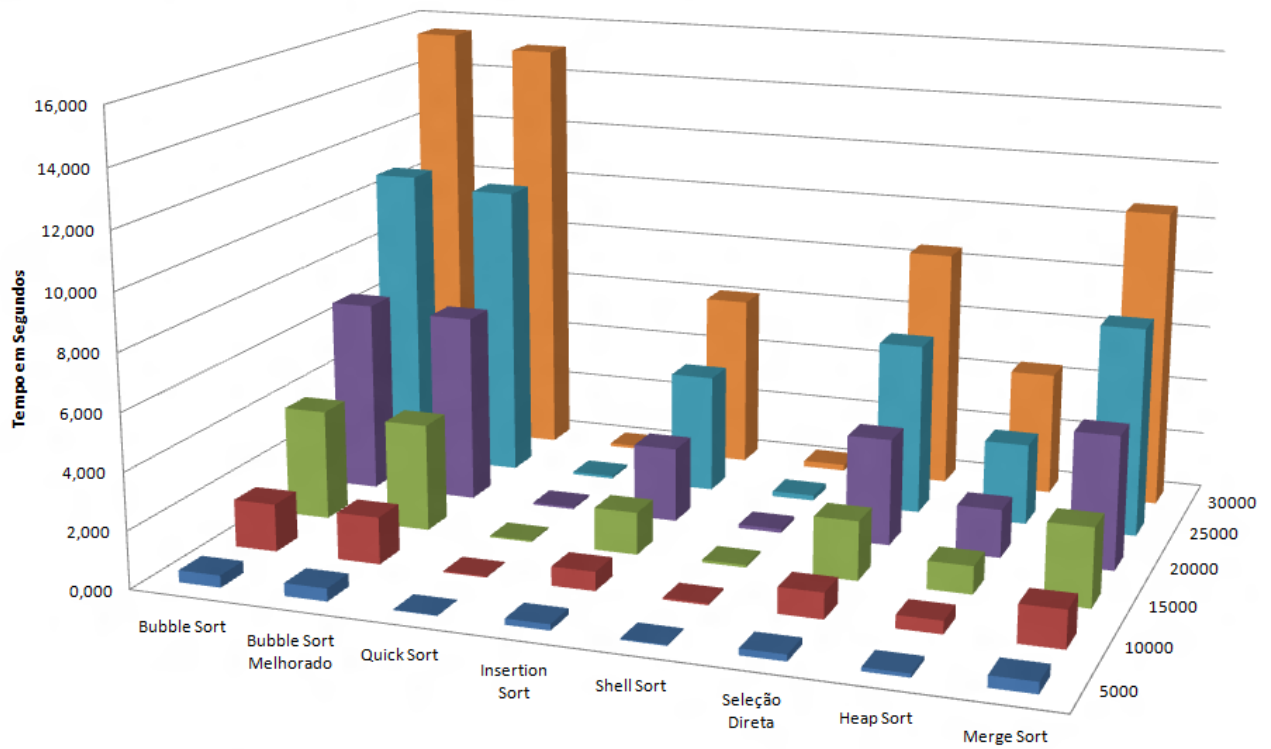
- Todos os algoritmos foram executados 10 vezes cada em ambas as linguagens, portanto os dados apresentados são uma média de tempo dessas execuções. Este método foi utilizado para garantirmos dados mais concretos;
- A quantidade de entrada de elementos são as mesmas para ambas as linguagens;
- Os algoritmos foram executados sempre no mesmo hardware;
- O cálculo de tempo foi feito apenas no início e fim do algoritmo de ordenação, demais rotinas foram desconsideradas;

Gráficos | Tabelas - Execuções em PHP

- Entrada Ascendente

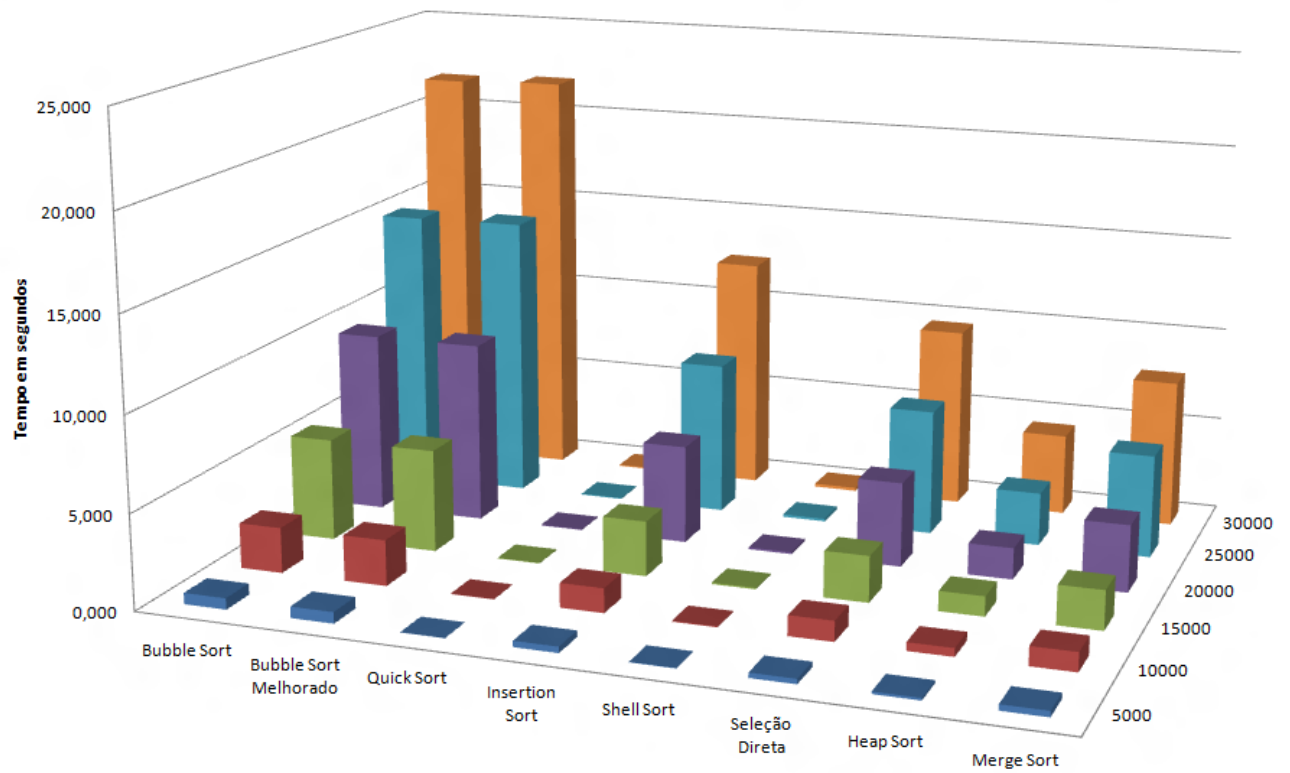


- Entrada Aleatória



	Bubble Sort	Bubble Sort Melhorado	Quick Sort	Insertion Sort	Shell Sort	Seleção Direta	Heap Sort	Merge Sort
5000	0,418	0,446	0,020	0,212	0,033	0,228	0,131	0,397
10000	1,670	1,645	0,044	0,665	0,062	0,907	0,455	1,311
15000	3,876	3,769	0,052	1,488	0,090	2,066	0,978	2,742
20000	6,791	6,641	0,072	2,623	0,130	3,753	1,698	4,687
25000	10,785	10,434	0,092	4,202	0,184	6,062	2,866	7,379
30000	15,518	15,086	0,109	6,131	0,217	8,492	4,390	10,585

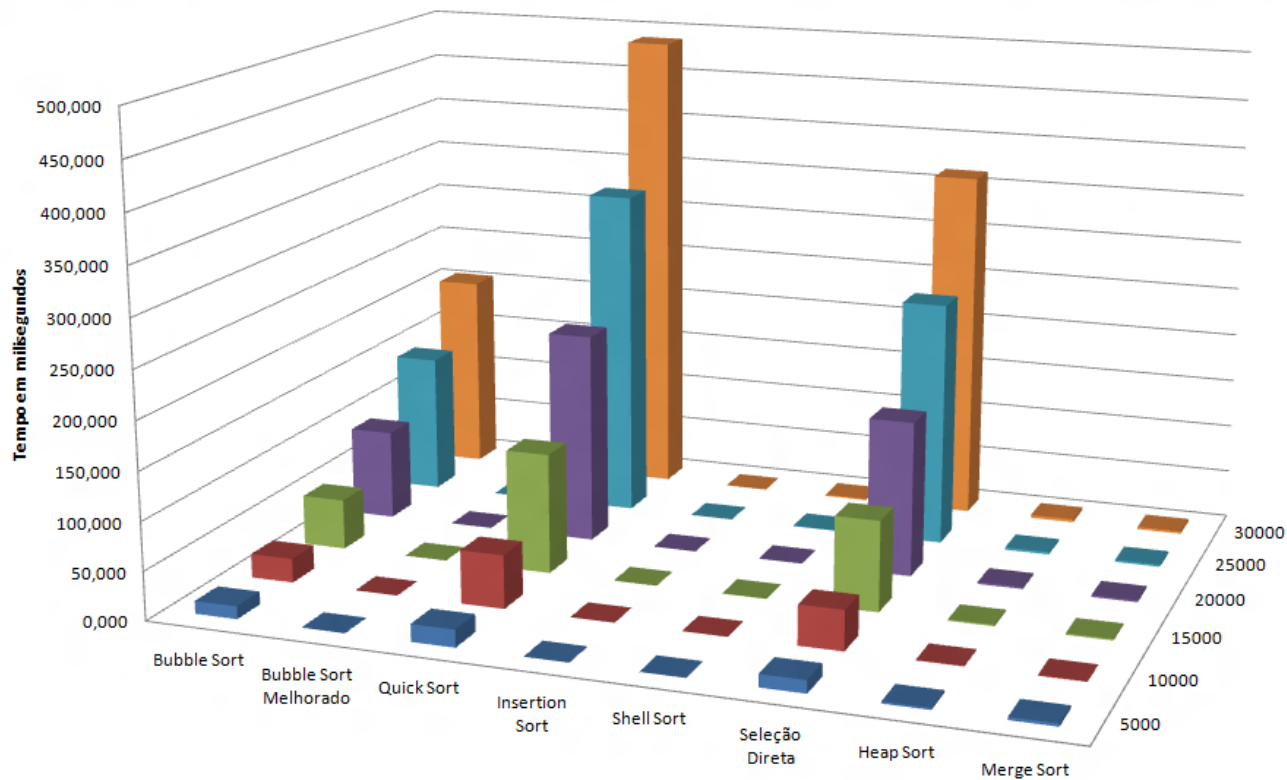
- Entrada Descendente



	Bubble Sort	Bubble Sort Melhorado	Quick Sort	Insertion Sort	Shell Sort	Seleção Direta	Heap Sort	Merge Sort
5000	0,591	0,595	0,011	0,325	0,027	0,266	0,127	0,311
10000	2,400	2,415	0,022	1,288	0,046	1,067	0,443	1,002
15000	5,413	5,433	0,031	2,912	0,077	2,441	1,052	2,073
20000	9,601	9,602	0,038	5,195	0,103	4,469	1,664	3,538
25000	14,953	15,021	0,052	8,082	0,139	6,664	2,817	5,445
30000	21,718	21,866	0,065	12,381	0,167	9,558	4,323	7,810

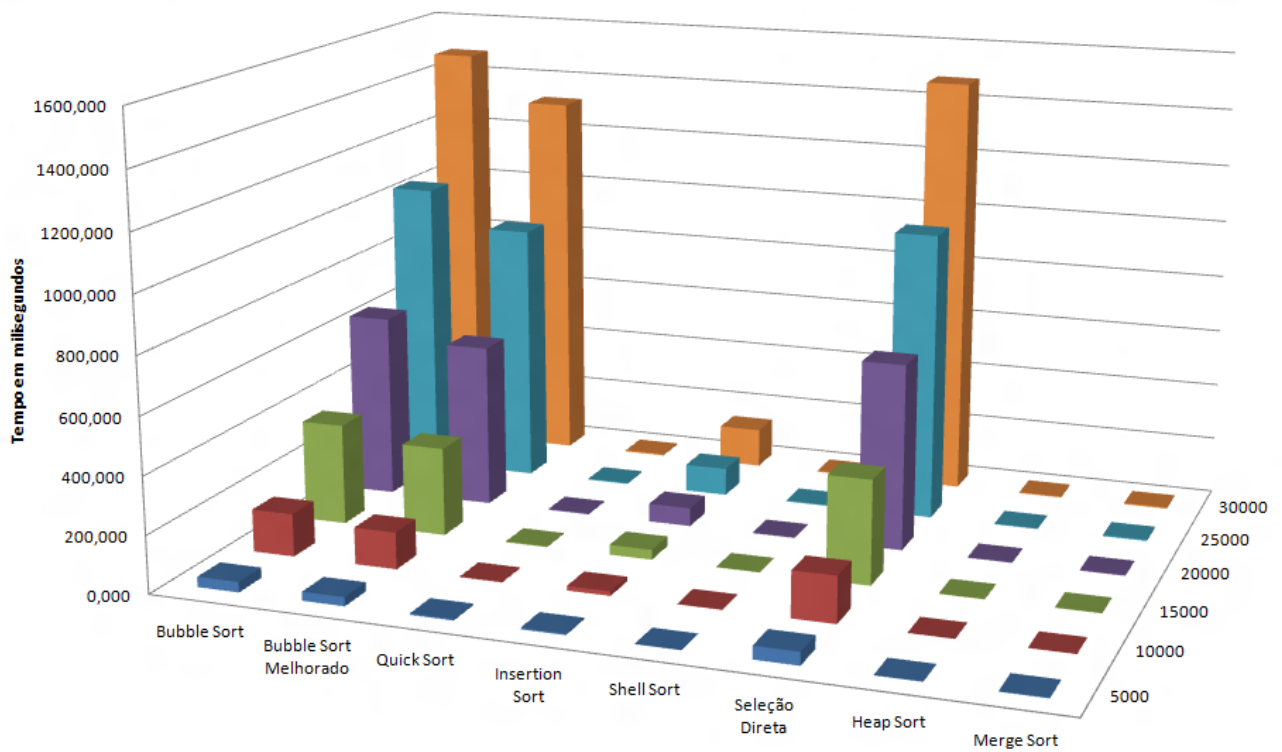
Gráficos | Tabelas - Execuções em Java

- Entrada Ascendente



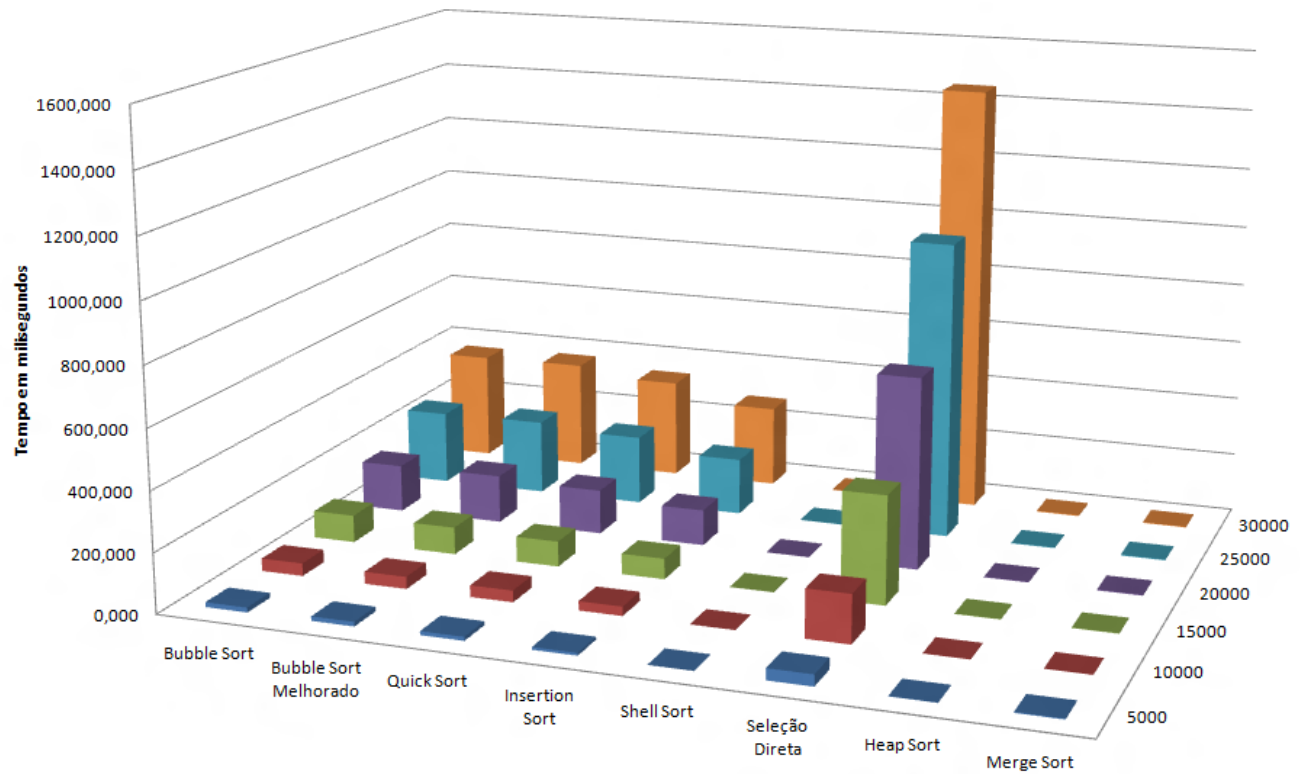
	Bubble Sort	Bubble Sort Melhorado	Quick Sort	Insertion Sort	Shell Sort	Seleção Direta	Heap Sort	Merge Sort
5000	13,100	0,200	18,000	0,000	0,000	13,000	2,000	3,000
10000	24,800	0,000	54,9	0,000	0,200	41,000	0,900	0,700
15000	52,000	0,000	124,100	0,000	0,300	93,100	1,500	1,600
20000	93,000	0,000	217,900	0,100	0,400	160,100	2,000	1,900
25000	144,700	0,000	341,500	0,000	0,600	253,400	2,500	2,200
30000	205,400	0,000	488,200	0,000	1,500	363,900	3,100	2,400

- Entrada Aleatória



	Bubble Sort	Bubble Sort Melhorado	Quick Sort	Insertion Sort	Shell Sort	Seleção Direta	Heap Sort	Merge Sort
5000	35,900	31,000	4,700	5,000	1,200	45,200	1,600	2,700
10000	149,500	128,700	2,2	15,600	1,3	161,200	1,200	1,200
15000	354,300	310,100	3,000	34,700	2,000	363,100	1,800	2,000
20000	644,000	571,400	3,500	64,500	2,700	650,700	2,800	3,300
25000	1039,600	913,200	4,200	97,300	3,600	1013,500	3,000	4,400
30000	1481,200	1318,200	5,1	137,6	4,300	1478,600	4,200	4,600

- Entrada Descendente



	Bubble Sort	Bubble Sort Melhorado	Quick Sort	Insertion Sort	Shell Sort	Seleção Direta	Heap Sort	Merge Sort
5000	16,000	15,000	13,000	9,000	0,000	38,000	1,000	3,000
10000	42,600	40,700	38,900	30,800	0,300	161,700	0,900	0,700
15000	92,200	91,600	85,300	69,300	0,500	362,000	1,400	1,600
20000	159,700	162,000	150,300	121,100	0,700	644,600	1,900	2,700
25000	248,600	252,600	234,100	191,800	1,000	1003,000	2,000	2,400
30000	364,200	368,200	336,100	275,500	1,200	1450,600	3,000	2,400