

# Algoritmos de ordenação

## Comparativos de métodos de ordenações Tempo(s) x Tamanho dos vetores

- Implementar os algoritmos:  
Bubble sort, insertion sort, selection sort, merge sort e quick sort.
- Executar para vetores de tamanho:  
10.000, 50.000, 100.000, 500.000, 1.000.000, 5.000.000, 10.000.000.
- Gerar um gráfico comparativo de tempo em segundos.
- Executar 20 vezes.

### Tabela Comparativa geral:

Após as medições, foram feitas as médias de tempo de execução para cada tamanho de vetor e chegamos aos dados a seguir.

Comparativos Ordenações Tamanho x tempo(segundos)					
Tamanho vetor	Buble Sort	Insertion Sort	Selection Sort	Merge Sort	Quick Sort
	Tempo(s)	Tempo(s)	Tempo(s)	Tempo(s)	Tempo(s)
10.000	0,57285	0,164	0,257	0,01	0
50.000	13,5489	4,1125	6,2316	0,035	0,01
100.000	54,73835	17,257	25,0476	0,064	0,0175
500.000	1385,286	416,11	715,2713	0,494	0,08455
1.000.000	5512,873	1680,509	3135,144	0,9814	0,1945
5.000.000	*	*	*	5,07	1,77
10.000.000	*	*	*	10,3015	5,244

(\*) Medição desconsiderada nos casos em que o tempo de espera da ordenação ficou inviável.

### Gráfico comparativo:

Através dos dados foi feito o gráfico Tempo(s) x Tamanho do vetor e se observou que os métodos elementares funcionam para casos de vetores pequenos já vetores muito grandes sua complexidade se torna  $O(n^2)$  enquanto dos algoritmos Merge e Quick a complexidade fica em  $O(n \log n)$ .

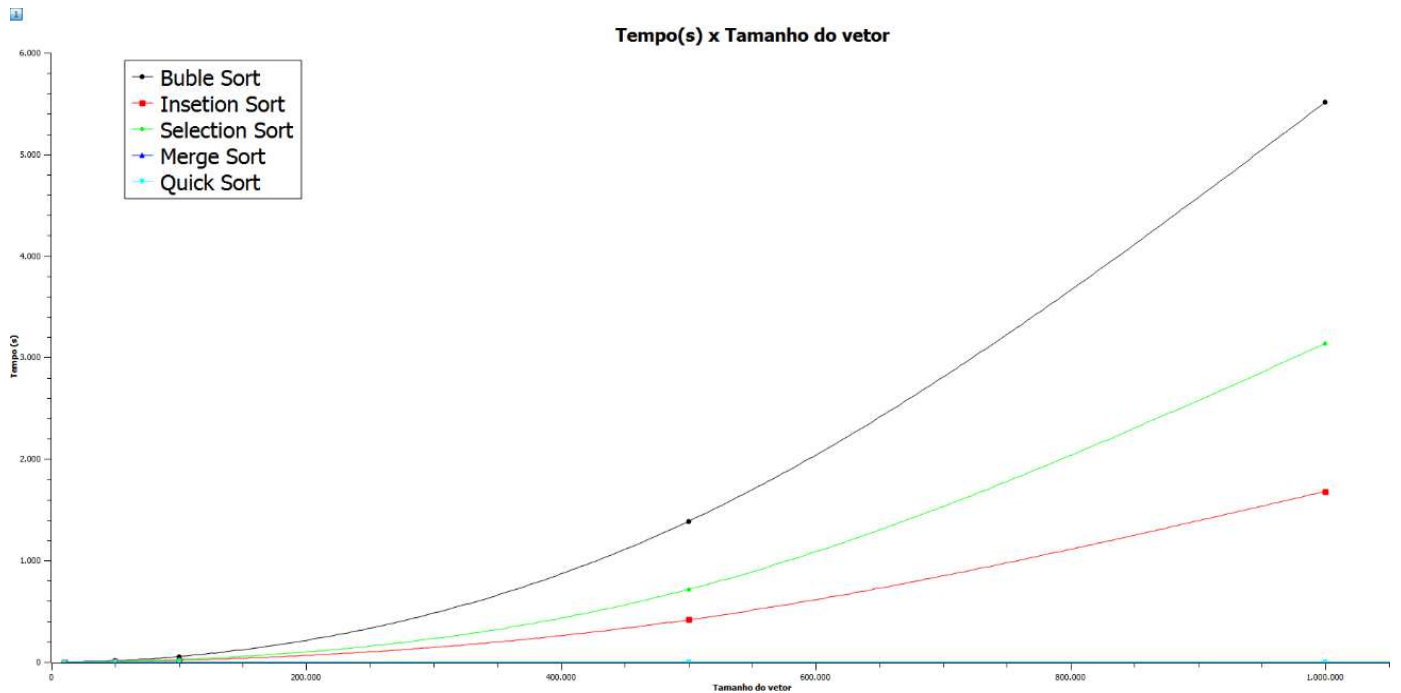


Figure 1: gráfico tempo(s) x tamanho do vetor para vetores de aproximadamente até 1 milhão de elementos

Na próxima imagem podemos ver mais claramente que os métodos de ordenação elementares como o Buble sort, Insertion sort e selection sort não possuem bom desempenho para vetores grandes sendo o tempo de esperar no caso do Buble sort de mais de 1,5h para ordenar um vetor de 1 milhão de elementos e podemos observar a grande diferença dos métodos de ordenação elementares com o Merge e Quick sort que possuem um desempenho excelente para casos de vetores muito grandes.

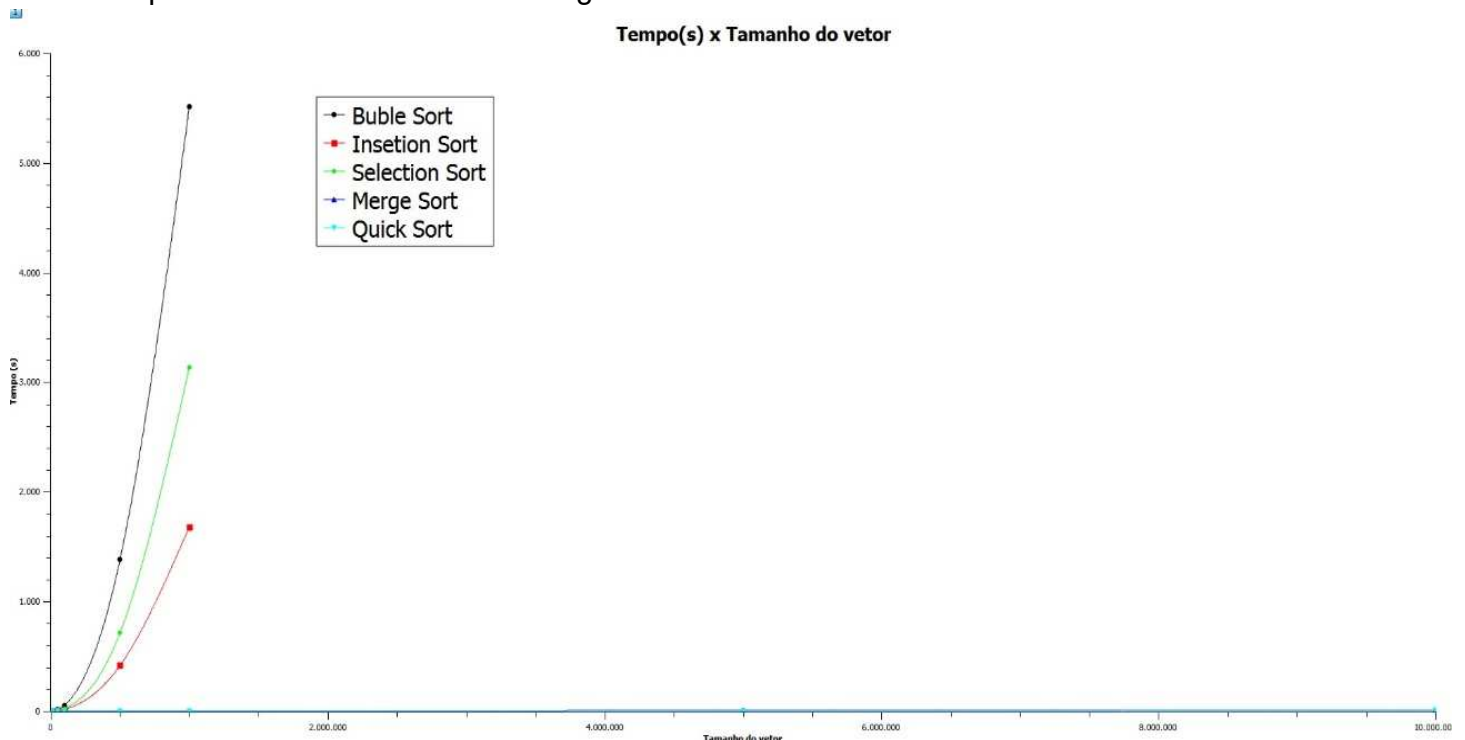


Figure 2: ordenação para até 10 milhões de elementos, para vetores muito grandes merge e quick sort possuem um desempenho excepcional e os outros métodos de ordenação se tornam inviáveis.