

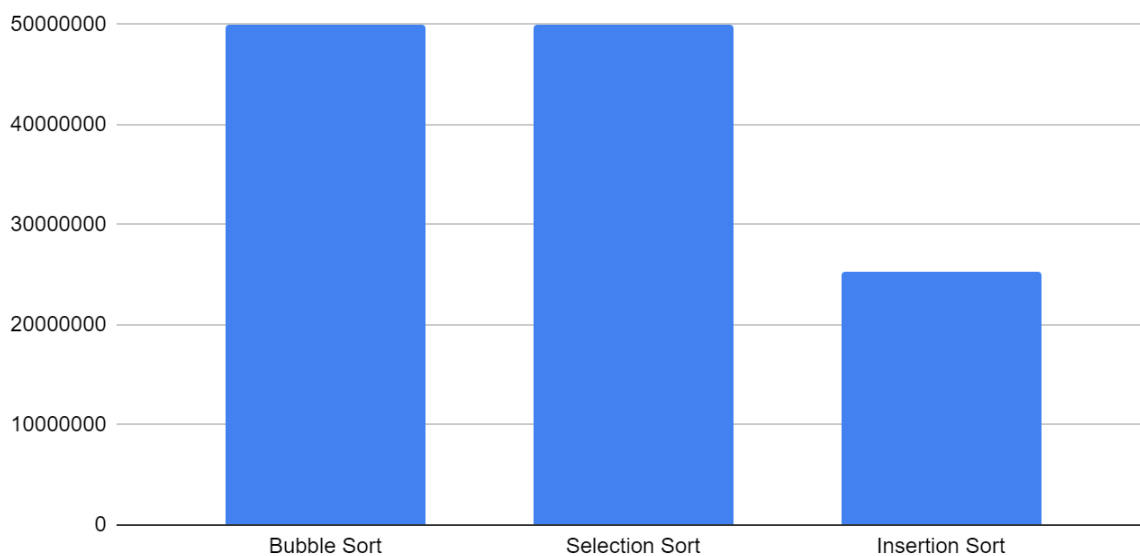
Nome: Alisson Rodrigo Carneiro da Silva
Matéria: Estrutura de dados I
Professor: Antonio oseas

Introdução:

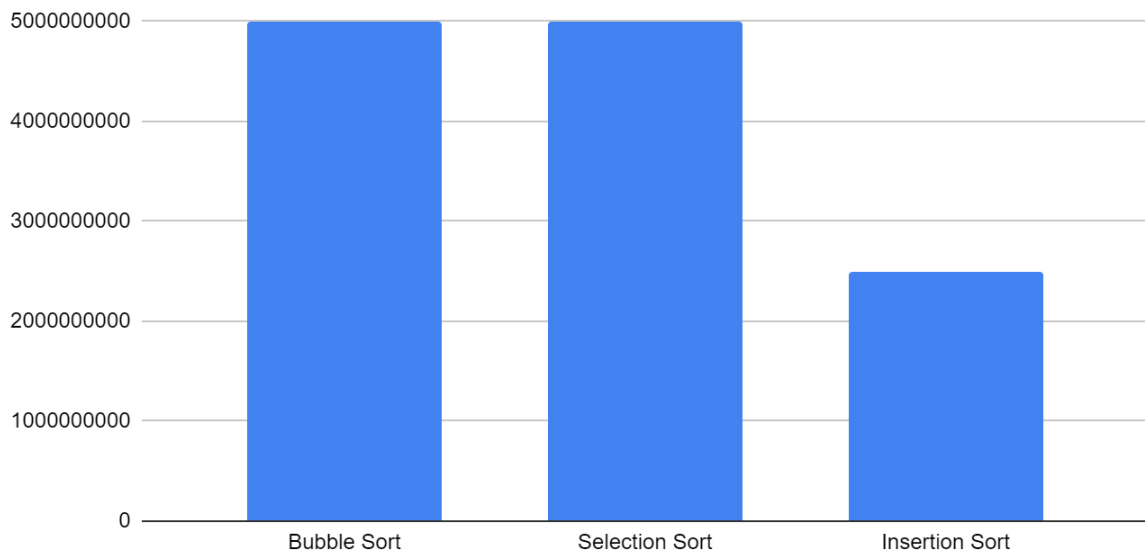
Ordenar elementos é uma tarefa essencial na computação, utilizada em uma variedade de aplicações. Nesta atividade, propomos o desenvolvimento de uma TAD (Tipo Abstrato de Dados) que implementa os três principais métodos de ordenação: Bubble Sort, Insertion Sort e Merge Sort. Utilizaremos vetores para armazenar elementos aleatórios no intervalo de 0 a 99999, em diferentes tamanhos: 10000, 100000 e 1000000 elementos.

Comparações:

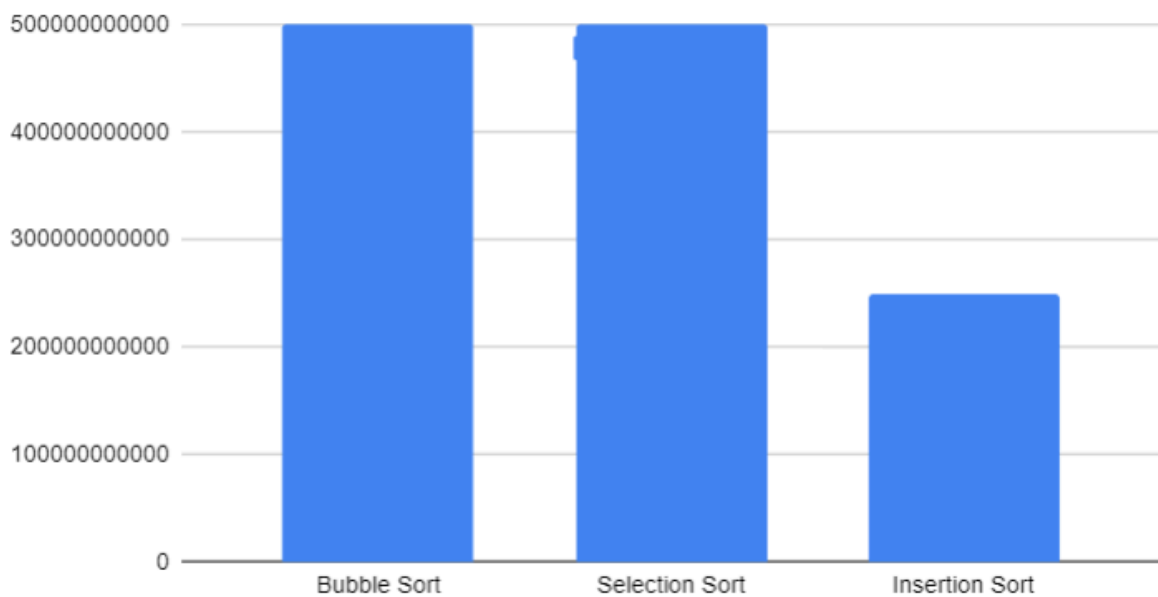
Comparações / 10.000



Comparações / 100.000



Comparações / 1000000

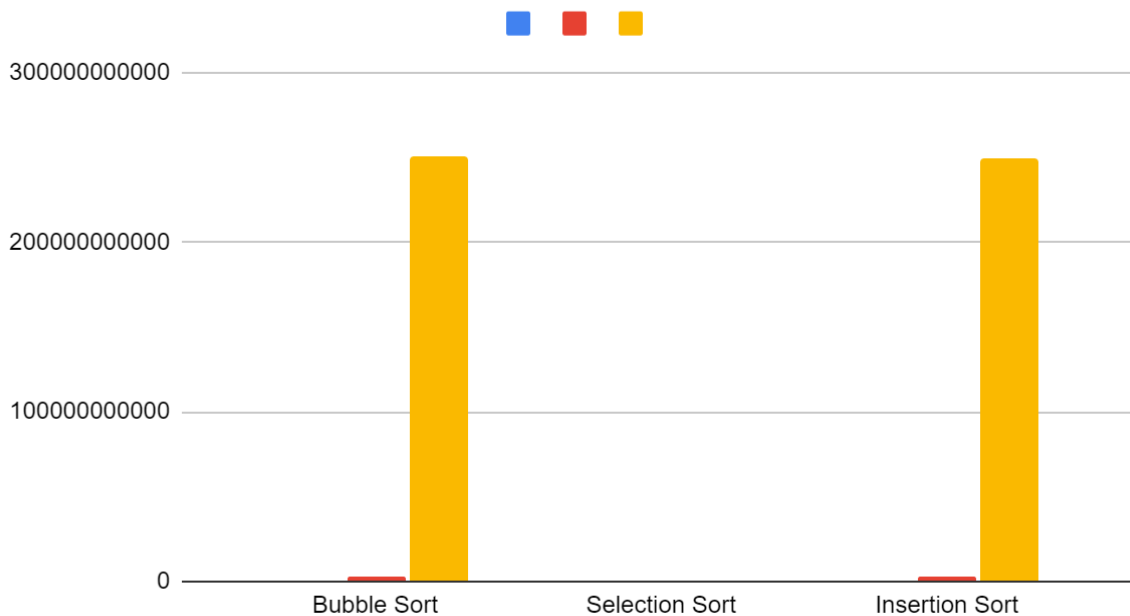


Nota-se que o desempenho dos bubble sort e selection sort são bastante parecidos, sendo mais claro, o número de comparações aumenta rapidamente de acordo com o número de dados sendo processados.

Já o insertion sort, apresenta um desempenho um pouco melhor em termos de comparações, especialmente em conjuntos de dados maiores. é um resultado bastante significativo.

Trocas:

Trocas / 10000, 100000, 1000000



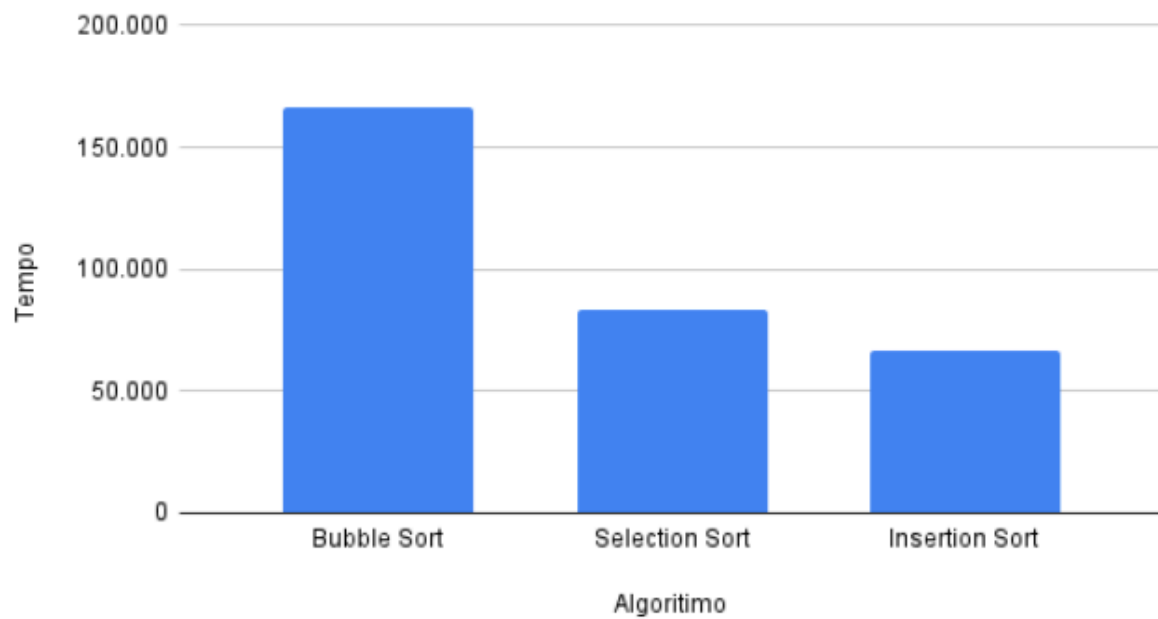
Bubble Sort: Faz muitas trocas, o que o torna menos eficiente, especialmente com muitos dados. Isso faz ele ficar mais devagar.

Selection Sort: Faz poucas trocas comparado com o Bubble Sort, o que pode fazer ele ser mais rápido, especialmente com muitos dados.

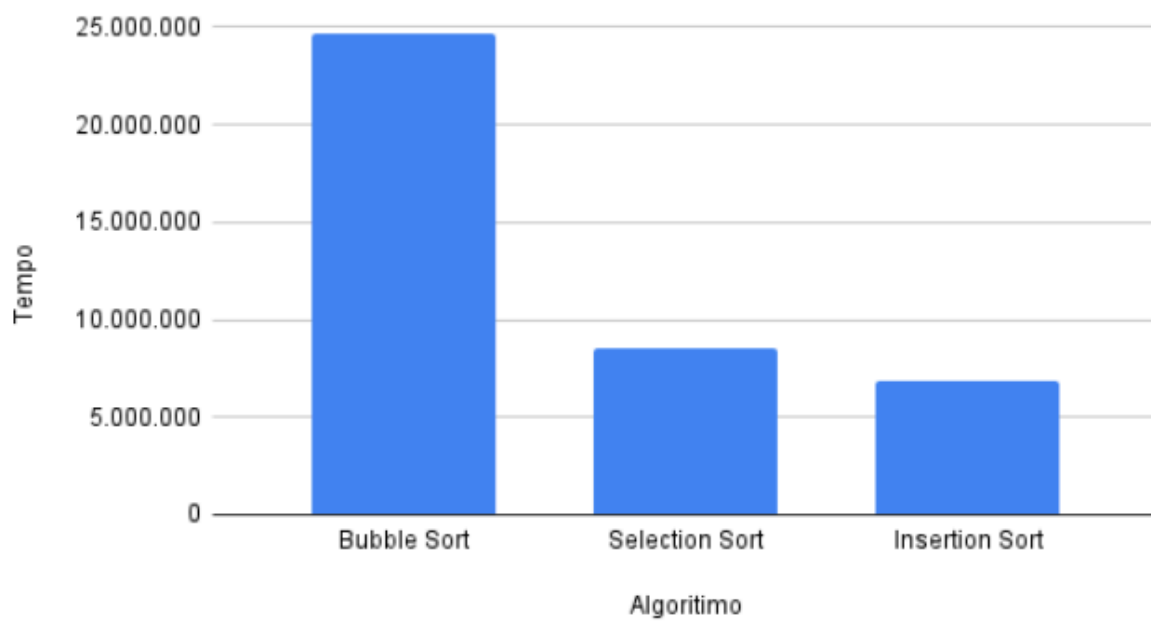
Insertion Sort: Faz quase o mesmo número de trocas que de comparações, o que quer dizer que costuma colocar os elementos onde precisam rapidinho. Por isso, é mais ou menos eficiente em trocas.

Tempo:

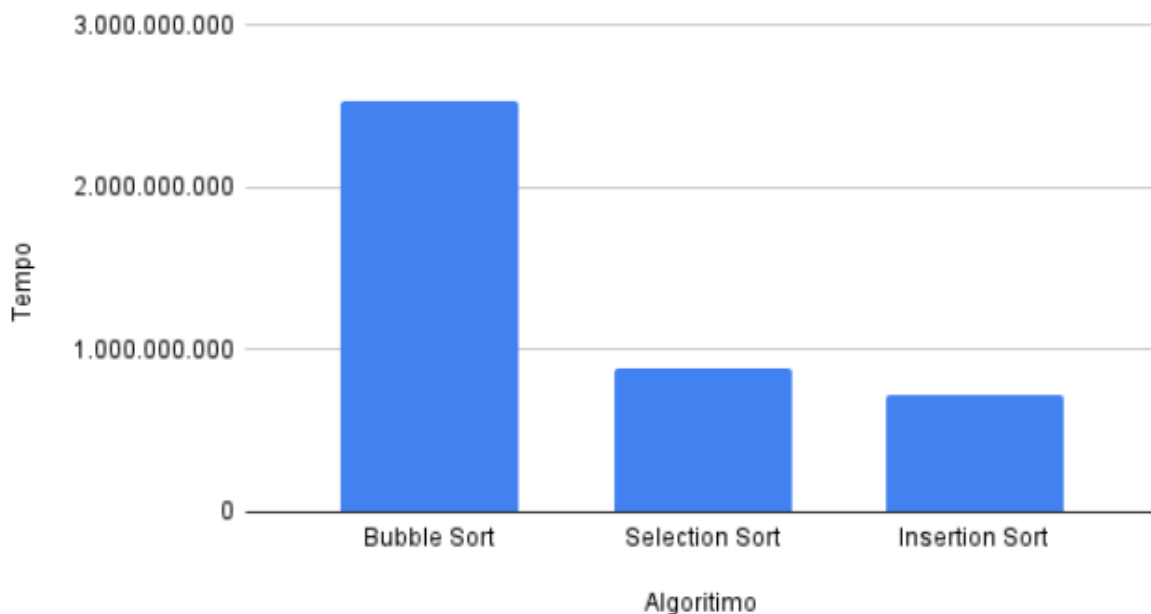
Tempo de execução com 10000



Tempo de execução com 100000



Tempo de execução com 1000000



Bubble Sort: Leva mais tempo para ordenar, especialmente com muitos dados, porque troca muitas vezes, o que o deixa mais devagar.

Selection Sort: É mais rápido que o Bubble Sort, principalmente com muitos dados, porque troca menos, fazendo ele ser mais eficiente em termos de tempo, embora seja um algoritmo meio lento.

Insertion Sort: É mais ou menos rápido, geralmente melhor que o Bubble Sort, mas pior que o Selection Sort. Ele fica no meio termo porque faz um equilíbrio entre o número de comparações e trocas.

Conclusão:

O bubble sort mostra-se simples sua implementação e de fácil compreensão, mas se torna lenta no quesito de otimização e velocidade, então a depender da sua função, pode não ser a melhor opção, o insertion apresenta um desempenho intermediário, ele é bastante equilibrado, mas o selection acaba sendo o melhor em relação as duas opções mostradas no relatório, saindo mais vantajoso sua escolha para ordenação.