

Universidade Federal de Ouro Preto
Bacharelado em Ciência da Computação

Comparativo de Algoritmos de Ordenação

Estrutura de Dados I

Fabio Henrique Alves Fernandes

Professora Amanda Nascimento

Ouro Preto, 18 de outubro de 2019

O bubble sort, ou ordenação por flutuação (literalmente "por bolha"), é um algoritmo de ordenação dos mais simples. A ideia é percorrer o vetor diversas vezes, e a cada passagem fazer flutuar para o topo o maior elemento da sequência. Essa movimentação lembra a forma como as bolhas em um tanque de água procuram seu próprio nível, e disso vem o nome do algoritmo.

Insertion Sort, ou ordenação por inserção, é o algoritmo de ordenação que, dado uma estrutura (array, lista) constrói uma matriz final com um elemento de cada vez, uma inserção por vez. Assim como algoritmos de ordenação quadrática, é bastante eficiente para problemas com pequenas entradas, sendo o mais eficiente entre os algoritmos desta ordem de classificação.

A ordenação por seleção (do inglês, selection sort) é um algoritmo de ordenação baseado em se passar sempre o menor valor do vetor para a primeira posição (ou o maior dependendo da ordem requerida), depois o de segundo menor valor para a segunda posição, e assim é feito sucessivamente com os $n-1$ elementos restantes, até os últimos dois elementos.

O merge sort, ou ordenação por mistura, é um exemplo de algoritmo de ordenação por comparação do tipo dividir-para-conquistar. Sua ideia básica consiste em Dividir (o problema em vários subproblemas e resolver esses subproblemas através da recursividade) e Conquistar (após todos os subproblemas terem sido resolvidos ocorre a conquista que é a união das resoluções dos subproblemas). Como o algoritmo Merge Sort usa a recursividade, há um alto consumo de memória e tempo de execução, tornando esta técnica não muito eficiente em alguns problemas.

Método	Complexidade de Tempo
Bubble Sort	$O(n^2)$
Insertion Sort	$O(n^2)$
Selection Sort	$O(n^2)$
Merge Sort	$O(n \log n)$

Depois de implementados os 4 métodos de ordenação, foi feito um teste com um milhão de elementos, com a finalidade de medir o tempo gasto por cada um durante o processo. Foi usada uma entrada de 3 vetores, um com números aleatórios, um feito de forma crescentes e o último de forma decrescente. Todos os vetores passaram por todos os métodos de ordenação devidamente implementados, e depois foi usado um algoritmo de cronometragem, para que possa ser medido o tempo de execução. O tempo é dado em segundos.

Bubble Sort

Vetor Aleatório	Vetor Crescente	Vetor Decrescente
5156 segundos	0 segundos	5400 segundos

Insertion Sort

Vetor Aleatório	Vetor Crescente	Vetor Decrescente
1871 segundos	1 segundo	3831 segundos

Selection Sort

Vetor Aleatório	Vetor Crescente	Vetor Decrescente
2012 segundos	2352 segundos	5486 segundos

Merge Sort

Vetor Aleatório	Vetor Crescente	Vetor Decrescente
0 segundos	0 segundos	0 segundos

Logo após os testes, foi possível constatar que o Merge Sort, por conta de sua complexidade de tempo, foi o algoritmo mais rápido de todos os outros implementados. Ele levou milésimos de segundos para ordenar todos os vetores, diferente dos outros, que levaram minutos ou então horas durante a execução.