

Curso: Análise e Desenvolvimento de Sistemas

Disciplina: Estrutura de Dados

Professora: Jaqueline Brigladori Pugliesi

Alunos: Helio Barolo Junior e Thales Vinicius de Souza Silva

Trabalho - 1º Bimestre

Utilizando os métodos de ordenação apresentados em sala, o processo para analisar os conjuntos de objetos e rearranjar na ordem específica foi considerado uma forma eficiente de ser feito a ordenação de dados em vetores. Contudo, visto que o método apresentado de seleção direta, inserção direta, bubblesort e quicksort teve variadas entradas e tempo de respostas distintos o método que consideramos mais eficiente foi o “quicksort” tendo uma diferença significativa em comparação aos demais métodos utilizados. Abaixo apresentamos os códigos utilizados em cada método:

Comparando o código apresentado acima e realizado uma medição, temos os seguintes dados para um vetor de 100, 1.000, 10.000 e 100.000 dados em milissegundos:

Método	100	1.000	10.000	100.000
Inserção Direta	1 ms	5 ms	47 ms	3.775 ms
Seleção Direta	1 ms	5 ms	176 ms	17.127 ms
Bubblesort	2 ms	6 ms	178 ms	20.842 ms
Quicksort	0 ms	3 ms	9 ms	59 ms

Explicando um pouco de cada método, iniciamos com o método “seleção direta” onde a ideia é que cada passagem pelo vetor, será selecionado o menor elemento e colocado este elemento a mais esquerda possível. É um algoritmo de ordenação simples e recomendado para pequenos conjuntos. As vantagens desse processo é que ele é viável para arquivos pequenos e com registros muito grandes, contudo o método se torna estável e os registros iguais nem sempre vão manter a mesma posição relativa de antes do início da ordenação.

O método de “inserção direta” considera os elementos do vetor um a um e inseri em seu lugar entre os outros elementos tratados (exemplo, ordenar cartas de jogo), de grosso modo o termo ‘inserção’ implica em arranjar um novo espaço, movendo os números maiores para a posição a direita e os elementos a esquerda do índice correto ficarão ordenados mas não necessariamente na posição final pois ainda podem ser deslocados para a direita e dar lugar a algum elemento menor que for encontrado posteriormente. É um método a ser utilizado quando o arquivo já está quase ordenado e bom para caso desejar adicionar alguns poucos itens a um arquivo já ordenado, além de ser mais estável pois mantém os registros iguais na mesma posição relativa aquele item.

O “bubblesort” faz um trabalho complicado onde realiza múltiplas passagens pelos dados e realiza troca de dois elementos adjacentes por vez e que estejam fora de ordem, até que não haja mais trocas a serem feitas. Muito fácil de implementar, mas consideravelmente mais lento que os métodos anteriores (vide tabela), podendo ser considerado adequado para pequenos arquivos e vetores já ordenados e ruim caso o registro seja muito grande e tendo um vetor de entrada em ordem reversa. É um método extremamente lento pois só faz comparações entre posições adjacentes e menos eficiente entre os métodos simples.

Por último o melhor método apresentado “quicksort” dividi o problema de forma que fará a ordenação de um conjunto de X itens em dois problemas menores onde a ordenação desses será de forma independente e os resultados serão combinados para produzir a solução final. Consideramos o melhor, pois realiza a ordenação de forma mais rápida em comparação aos demais métodos e também de grande variedade de situações. A parte onde se necessita maior atenção é a escolha de um ‘pivô’ para que seja feito a divisão da partição, onde os dados terão os valores menores que o ‘pivô’ direcionados para a esquerda e os maiores para a direita.