

Trabalho de Ordenação

Insertion Sort - É um método que ordena os elementos da esquerda para a direita, e os elementos que já foram ordenados são posicionados à esquerda para dar continuidade ao processo de ordenação. Também pode ser considerado um método de ordenação estável, já que a ordem relativa dos itens iguais não é alterada durante a ordenação.

Bubble Sort - É um método de ordenação que funciona através de flutuação (Bolha) onde o vetor será percorrido diversas vezes, é a cada passagem o maior vetor da sequência será “flutuado” para o topo, até que todo o vetor tenha sido ordenado. É um vetor estável e relativamente lento mesmo dentre os considerados mais simples.

Selection Sort - É um método que funciona através de seleção de itens, onde o menor item é colocado na primeira posição, o segundo menor na segunda posição e assim sucessivamente. É não pode ser considerado um método estável.

Estes três primeiros métodos de ordenação são considerados métodos simples, sendo mais recomendados a pequenos vetores já que são programas pequenos e de fácil compreensão e relativamente mais lentos.

Quick Sort - É um método que ordena os elementos através de uma divisão de sua lista de entrada em sub-listas a partir de um elemento fixo (Pivô), para em sequência realizar o mesmo procedimentos nas sub-listas criadas consequentemente até que sobre apenas uma lista unitária. Esse método é considerado o mais rápido para amplas variedades de situações e não pode ser considerado um método estável.

Merge Sort - É um método que funciona através da divisão do problema (vetor) em pequenas partes e depois serão juntados novamente (Merge) após ordenação. É considerado um método de ordenação estável.

Estes dois últimos métodos de ordenação são considerados métodos eficientes, são projetados para trabalhar em conjunto com uma quantidade maior de dados, já que são mais complexos e requerem uma quantidade menor de comparações, sua velocidade de processamento aumenta consideravelmente.

Mesmo com algumas diferenças na performance dos elementos citados acima, todos são considerados elementos de ordenação interna, ou seja, todos os elementos cabem na memória principal e podem ser imediatamente acessados, o que impacta positivamente na performance ao se executar a ordenação.

Na implementação do programa foram utilizadas as seguintes estruturas de dados :

Vetor (array) - Listas - Matrizes - Classe Random

Já os métodos utilizados foram os métodos de ordenação requisitados pelo próprio trabalho.

Bubble

Selection

Insertion

Quick

Merge

Vetor A

Inicio: 992
Fim: 992
Tempo : 0

VetorB

Inicio: 992
Fim: 992
Tempo : 0

Vetor C

Inicio: 992
Fim: 992
Tempo : 0

Bubble

Selection

Insertion

Quick

Merge

Vetor A

Inicio: 760
Fim: 760
Tempo : 0

VetorB

Inicio: 760
Fim: 760
Tempo : 0

Vetor C

Inicio: 760
Fim: 760
Tempo : 0

Bubble

Selection

Insertion

Quick

Merge

Vetor A

Inicio: 153
Fim: 154
Tempo : 1

VetorB

Inicio: 154
Fim: 156
Tempo : 2

Vetor C

Inicio: 156
Fim: 157
Tempo : 1

Bubble

Selection

Insertion

Quick

Merge

Vetor A

Inicio: 930
Fim: 930
Tempo : 0

VetorB

Inicio: 930
Fim: 930
Tempo : 0

Vetor C

Inicio: 930
Fim: 930
Tempo : 0

Quanto a implementação dos vetores de ordenação, a minha principal dificuldade foi encontrada ao implementar o método de Merge, já que o vetor apresenta uma maior complexidade em sua implementação.