Documentação dos Algoritmos de Ordenação

Tópicos Avançados em Programação

Jâime Cassol Stefanello

Restinga Sêca, Recanto Maestro – RS

07/04/2016

Sumário

**1 – Como Funciona3**

**1 – Insert Sort4**

**2 – Selection Sort5**

**3 – Quick Sort6**

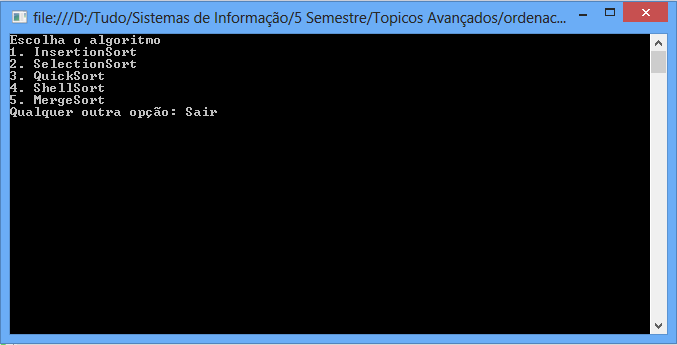
**4 – Shell Sort7**

**5 – Merge Sort8**

1 – Como Funciona

Neste projeto está implementado 5 (cinco) algoritmos de ordenação.

Tela Inicial do Projeto:

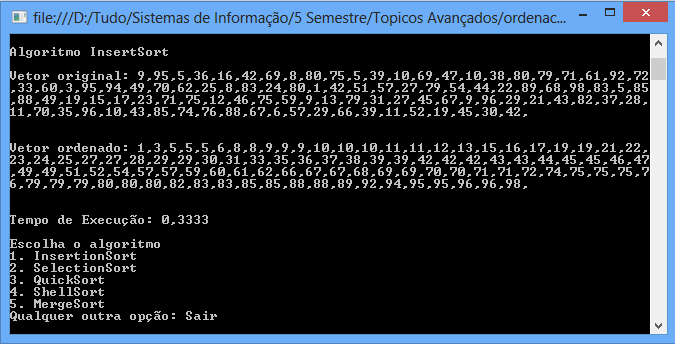


Nesta tela o usuário deve escolher qual algoritmo deseja executar, utilizando de acordo com o descrito abaixo.

* Insertion Sort – o usuário digita o número 1 e tecla ENTER.
* Selection Sort – o usuário digita o número 2 e tecla ENTER.
* Quick Sort – o usuário digita o número 3 e tecla ENTER.
* Shell Sort – o usuário digita o número 4 e tecla ENTER.
* Merge Sort – o usuário digita o número 5 e tecla ENTER.

Ao fazer isso será gerada uma tela, contendo o vetor com 100 posições antes de aplicar a ordenação escolhida, depois da ordenação e também o tempo de execução.

Exemplo utilizando o algoritmo Insertion Sort:

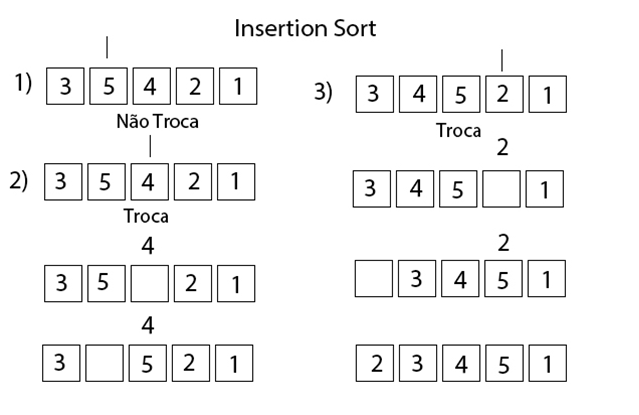


2 – Insertion Sort

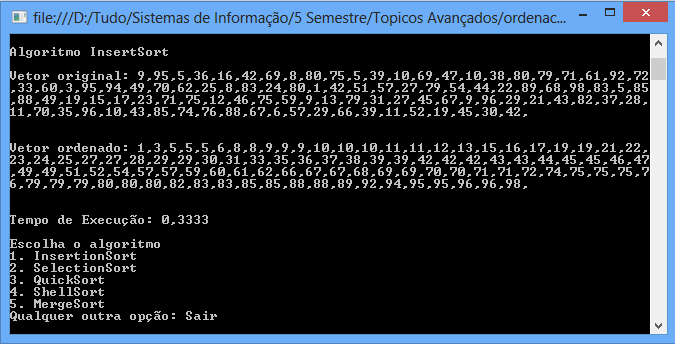
Considera que o primeiro elemento está ordenado (ou seja, na posição correta).  
 A partir do segundo elemento, insere os demais elementos na posição apropriada entre aqueles já ordenados.  
 O elemento é inserido na posição adequada movendo-se todos os elementos maiores para posição seguinte do vetor.

Mais interessante que o Bubble Sort para popular um vetor.

Abaixo Exemplo do funcionamento:



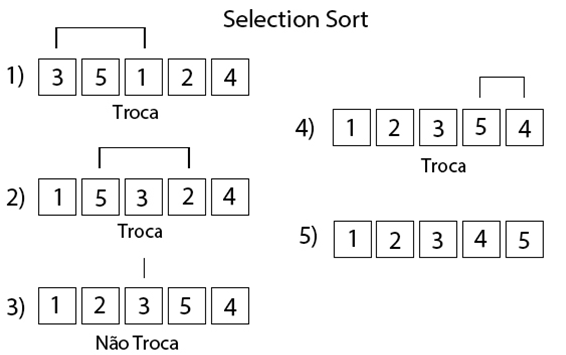
Abaixo a tela do projeto com a amostra do resultado deste algoritmo:



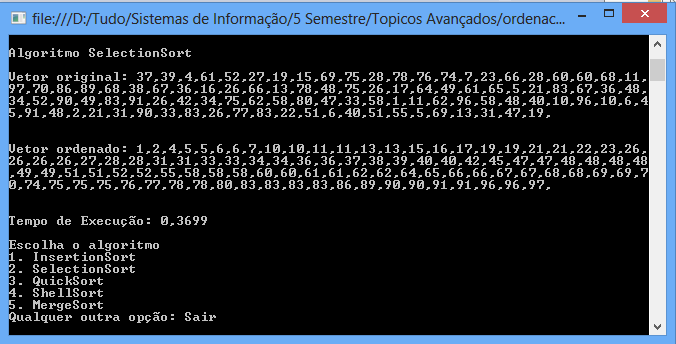
3 – Selection Sort

Consiste em encontrar o menor número por pesquisa sequencial. Encontrando o menor número, essa é permutada com a que ocupa a posição inicial do vetor, que fica então reduzido a um elemento.  
O processo é repetido para o restante do vetor, sucessivamente, até que todos os números tenham sido selecionados e colocados em suas posições definitivas.

Abaixo Exemplo do funcionamento:



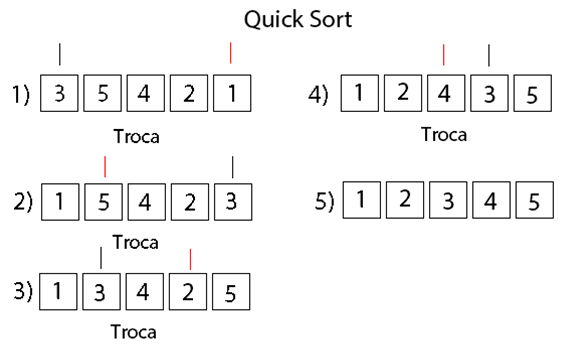
Abaixo a tela do projeto com a amostra do resultado deste algoritmo:



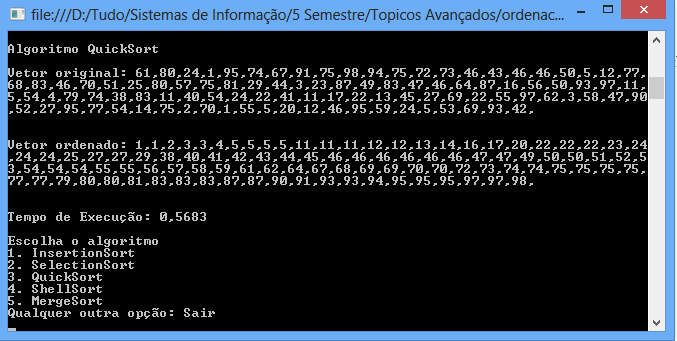
4 – Quick Sort

O Quicksort é o algoritmo mais eficiente na ordenação por comparação. Nele se escolhe um elemento chamado de pivô, a partir disto é organizada a lista para que todos os números anteriores a ele sejam menores que ele, e todos os números posteriores a ele sejam maiores que ele. Ao final desse processo o número pivô já está em sua posição final. Os dois grupos desordenados recursivamente sofreram o mesmo processo até que a lista esteja ordenada.

Abaixo Exemplo do funcionamento:



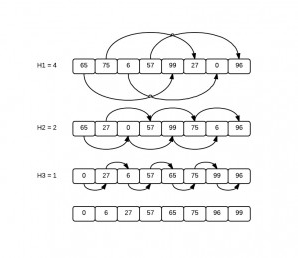
Abaixo a tela do projeto com a amostra do resultado deste algoritmo:



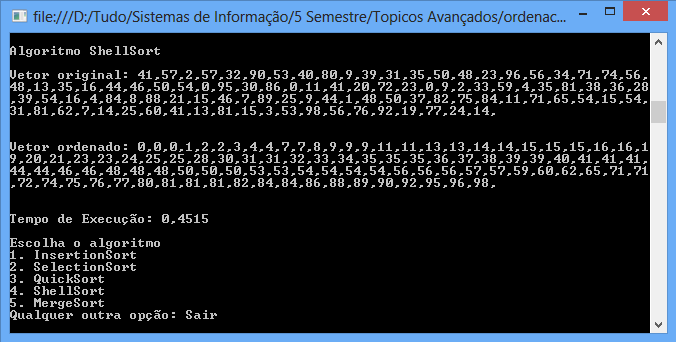
5 – Shell Sort

É um método de ordenação por inserção criado por Donald Shell que basicamente divide a lista a ser ordenada em grupos menores e aplica o método de ordenação por inserção.

Abaixo Exemplo do funcionamento:



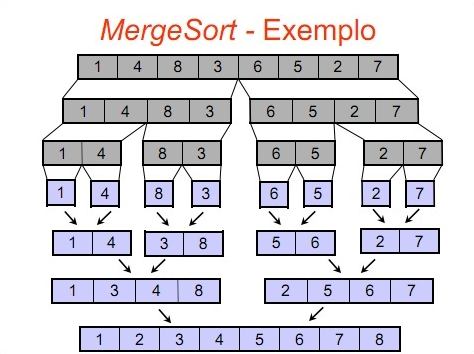
Abaixo a tela do projeto com a amostra do resultado deste algoritmo:



6 – Merge Sort

A idéia básica do Merge Sort é criar uma sequência ordenada a partir de duas outras também ordenadas. Para isso, o algoritmo Merge Sort divide a sequência original em pares de dados, agrupa estes pares na ordem desejada; depois as agrupa as sequências de pares já ordenados, formando uma nova sequência ordenada de quatro elementos, e assim por diante, até ter toda a sequência ordenada.

Abaixo Exemplo do funcionamento:



Abaixo a tela do projeto com a amostra do resultado deste algoritmo:

