

Suporte para GitHub:

Carregando arquivos pelo navegador: https://drive.google.com/open?id=1Klf0HCJcB_4Q5B7efMXrs_YYDXGwH89U
Tutorial Basico GitHub com Eclipse e EGit Usando Chave SSH: <http://www.youtube.com/watch?v=ffBSazTSGZw>
Usando Github com Github Desktop em Projetos Eclipse: <http://www.youtube.com/watch?v=EgHljYyS4U>
Usando Github com SSH no Terminal Linux com chave gerada no Eclipse: <http://www.youtube.com/watch?v=0s699q5Sja4>
Usando Github com SSH no Terminal Windows com chave gerada no Eclipse: <http://www.youtube.com/watch?v=DaydwPB2WSI>

Gerando links compartilháveis:

<https://drive.google.com/file/d/1cyoxa5W67MY5xDM6gCYpkle1GU3QCKa>

Vídeos Suporte:

Quick Sort:

Teste de Mesa: <https://youtu.be/UhYIZBdPzp0>

Java: <https://youtu.be/rdIAICrcxXA>

Para todos os exercícios, quando solicitado teste de mesa, carregar a solução para um drive compartilhado e quando solicitado desenvolvimento, definir o que se pede e aplicar o código em Java e carregar a solução no Github.

1. Considere o seguinte vetor:

74	20	74	87	81	16	25	99	44	58
----	----	----	----	----	----	----	----	----	----

Fazer:

a) O teste de mesa para a aplicação de ordenação por Quick Sort. A demonstração deve deixar claro a ordem na qual cada elemento é fixado na ordenação;

2. Considere o seguinte vetor:

44	43	42	41	40	39	38
----	----	----	----	----	----	----

Fazer:

a) O teste de mesa para a aplicação de ordenação por Quick Sort. A demonstração deve deixar claro a ordem na qual cada elemento é fixado na ordenação;

3. Considere o seguinte vetor:

31	32	33	34	99	98	97	96
----	----	----	----	----	----	----	----

Fazer:

a) O teste de mesa para a aplicação de ordenação por Quick Sort. A demonstração deve deixar claro a ordem na qual cada elemento é fixado na ordenação;

4. Criar, em Java, uma Biblioteca que implemente as funcionalidades de ordenação do Quick Sort

5. Criar um projeto Java que receba as bibliotecas criadas no exercício 3 e apresente os resultados dos exercícios 1, 2 e 3.

6. Criar um projeto Java que receba as bibliotecas BubbleSort, MergeSort e QuickSort. O projeto deve prever um vetor com 1500 posições ({1499, 1498, 1497, ..., 0}) e apresentar o tempo de ordenação para cada um dos métodos.