

4ª lista de exercícios

- 1) Discorra sobre a importância e utilidade dos estudos dos métodos de ordenação interna.
- 2) O limite inferior do problema de ordenação por comparação de elementos é $O(n \log n)$. Demonstre essa afirmação.
- 3) Quais as principais diferenças entre os algoritmos InsertionSort (inserção simples) e ShellSort?
- 4) Esquematize o funcionamento do método MergeSort.
- 5) Implemente em C o algoritmo MergeSort e faça a respectiva análise assintótica pela (i) equação de recorrência (com prova por indução), pela (ii) árvore de recorrência e pelo (iii) método mestre.
- 6) Esquematize o funcionamento do método QuickSort.
- 7) Utilizando o seguinte vetor como entrada:

3	7	6	10	5	11	1	8	4	9
---	---	---	----	---	----	---	---	---	---

Execute passo a passo e manualmente o algoritmo Quicksort recursivo. Mostre o estado atual do vetor e das variáveis auxiliares em cada passo do algoritmo.

- 8) O vetor $v = (25, 17, 20, 5, 11, 19, 13)$ satisfaz a propriedade *heap*. Realize *graficamente* três passos do algoritmo HeapSort, mostrando como ficará o vetor após cada passo.
- 9) Sob quais circunstâncias é vantajoso utilizar cada um dos seguintes métodos de ordenação? Baseie-se nas características de todos os métodos de ordenação comparativos dados em aula.
 - a) InsertionSort (inserção simples)
 - b) BubbleSort
 - c) QuickSort

- 10) Classifique cada método de ordenação dado em aula como "estável" ou "instável":

<i>InsertionSort</i> :	() Estável	() Instável
<i>SelectionSort</i> :	() Estável	() Instável
<i>BubbleSort</i> :	() Estável	() Instável
<i>HeapSort</i> :	() Estável	() Instável
<i>MergeSort</i> :	() Estável	() Instável
<i>QuickSort</i> :	() Estável	() Instável
<i>ShellSort</i> :	() Estável	() Instável

- 11) Compare os métodos de ordenação em termos de complexidade de tempo e de espaço.