

Lista de Exercícios de Árvores e Ordenação de Dados

- 1) Mostre, passo a passo, a ordenação do vetor [27, 32, 35, 29, 12, 22, 19] utilizando o método quicksort.
Obs: Considere uma implementação do quicksort que usa o primeiro elemento como pivô.
- 2) Mostre, passo a passo, a ordenação do vetor [47, 21, 55, 39, 13, 32, 14] utilizando o método quicksort.
Obs: Considere uma implementação do quicksort que usa o primeiro elemento como pivô.
- 3) Mostre, passo a passo, a ordenação do vetor [59, 17, 42, 23, 75, 09, 34] utilizando o método quicksort.
Obs: Considere uma implementação do quicksort que usa o primeiro elemento como pivô.
- 4) Veja o algoritmo de quicksort abaixo. Implemente o método “partição” para ordenar, em **ordem decrescente**, um **vetor de inteiros**.

```
public static void quicksort (<tipo_componente> v [ ], int i, int f) {  
    int k;  
    if (f > i) { // tamanho do segmento é maior que 1?  
        k = partição (v, i, f);  
        quicksort (v,i,k-1); // ordenar segmento S1  
        quicksort (v, k+1, f); // ordenar segmento S3  
    }  
}
```

- 5) Implemente o algoritmo de quicksort para ordenar, em **ordem crescente**, um **vetor de String**.
- 6) Implemente o algoritmo de quicksort para ordenar, em **ordem decrescente de notas**, um vetor de Aluno (conforme descrição da classe Aluno abaixo)

```
public class Aluno {  
    private String matr;  
    private String nome;  
    private double nota;  
    private int faltas;  
  
    ...  
    ...  
}
```