

Algoritmos e Estruturas de Dados I

1. Uma fila F com N elementos é ordenada se os elementos quando removidos de F são e_1, \dots, e_N (nesta ordem) e tais que $e_i \leq e_{i+1}$, para todo $1 \leq i < N$. Isto é, uma fila é ordenada se quando seus elementos são todos removidos eles saem em ordem ascendente. Dada uma fila F , escreva um algoritmo que remova e reinsira elementos até que F se torne ordenada com espaço auxiliar constante acrescido de uma pilha auxiliar P (F e P devem ser manipuladas apenas através das funções enfileira/desenfileira/próximo/tamanho e empilha/desempilha/topo/tamanho, respectivamente).

2. Considere que as letras do seu nome completo, removendo-se as letras repetidas, foram usadas como elementos para serem inseridas em uma árvore binária de busca. Escreva em uma linha o seu nome completo em letra de forma, riscando todas as ocorrências de letras que se repetem, deixando apenas a primeira ocorrência de cada uma. Agora, desenhe abaixo a árvore de busca resultante da inserção das letras distintas em uma árvore binária de busca inicialmente vazia. Considere que uma letra é menor do que outra se vem antes no alfabeto. A ordem de inserção das letras é natural da leitura (da esquerda para direita).

estrutura Nó:
Elem: Inteiro
Esq, Dir: ^Nó

3. Elabore os algoritmos solicitados abaixo (definição de Nó à direita):

a. **função** BuscaArv(T : ^Nó, x : Inteiro): Lógico

```
//assume T uma árvore  
//retornar V se x é um elemento de T, ou F caso contrário
```

b. **função** MaioresInternos(T : ^Nó, x : Inteiro): Inteiro

```
//assume T uma árvore  
//retornar o número de elementos não folhas de T que são maiores que x
```

c. **função** NúmeroFolhasPares(T : ^Nó): Inteiro

```
//assume T uma árvore  
//retornar o número de folhas de T associadas a elementos pares
```

d. **função** Média(T : ^Nó): Real

```
//assume T uma árvore  
//retornar a média aritmética dos elementos associados a nós de T
```