

Terceira Prova - Julho de 2025

1. Desenhe a árvore binária de pesquisa que resulta da inserção sucessiva dos valores na seguinte ordem: **50, 25, 75, 15, 35, 65, 85, 10, 20, 30, 40, 60, 70, 80, 90**
 - a) Considerando que a visitação imprime o valor de cada nó, escreva resultado da visitação da árvore em pré-ordem, pós-ordem e ordem simétrica;
 - b) Liste os nós: a) de grau 0 na árvore; b) de grau 1 na árvore; c) de grau 2 na árvore
 - c) Desenhe a árvore após a remoção consecutiva dos seguintes valores:
 - c.1) 75
 - c.2) 25
 - c.3) 60
2. Implemente um algoritmo que recebe uma árvore binária e retorna o número de nós na árvore que somente possuem filho esquerdo.
`int abNumNosComSoFilhoEsquerdo(TNode *t)`
3. Faça um algoritmo que recebe a raiz de uma árvore binária de pesquisa, uma chave e remove o elemento identificado pela chave, somente se ele for uma folha.
`TNode *abpRemoveMaior(TNode *t, void *key, int (*cmp)(void *, void *))`
4. Definimos que uma árvore t1 é maior que uma árvore t2 se elas são topologicamente iguais e para todos os nós nas posições correspondentes o valor armazenado no nó de t1 é maior que o valor armazenado no nó de t2. Usando o conceito acima, escreva um algoritmo que recebe o nó raiz de duas árvores binárias t1 e t2 e retorna TRUE se t1 for maior que t2, e FALSE de outro modo.
Não pode usar memória auxiliar (por exemplo converter a árvore para um vetor.).
`int ComparaArvores (TNode *t1, TNode *t2, int (*getvalue) (void *));`
OBS: a função getvalue retorna o valor inteiro que está armazenado no nó da árvore.