

Lista de exercícios

1. Escreva um algoritmo que recebe o nó raiz de duas árvores $t1$ e $t2$ e retorna 1 se a soma dos valores dos nós de $t1$ for maior que a soma dos valores dos nós de $t2$, 0 se forem iguais e -1 de outro modo. Não pode usar memória auxiliar (por exemplo converter a árvore para um vetor.). Se quiser pode considerar a seguinte estrutura para TNode :

```
typedef struct tnode {  
    int data;  
    struct tnode *l, *r;  
} TNode;  
int ComparaArvores ( TNode *t1, TNode *t2);
```

2. Considerando os algoritmos vistos em sala de aula, desenhe a árvore binária de busca resultante da inserção dos seguintes elementos (nesta ordem): 20, 25, 10, 5, 12, 22 e 23. Remova a raiz da árvore. Quais elementos podem ser utilizados para substituir a raiz?
3. Desenhe a árvore binária que tem os percursos descritos abaixo.
Pré-ordem: C E A D H K J B M F L G I
In-ordem: D A K H E C B J L F M G I
Desenhe uma outra árvore que tenha o mesmo percurso em pré-ordem.
Desenhe o percurso em pós-ordem das duas árvores.

4. Desenhe a árvore binária que apresenta os percursos descritos abaixo.
Pré-ordem: B M K A C J L Z N F W D Y E
In-ordem: A C K M L J Z B F N D Y W E



5. Desenhe a árvore binária que apresenta os percursos descritos abaixo.
Pré-ordem: J L G N D M E A F B I H C
In-ordem: N G L M E D J A B I F H C

6. Escreva uma função recursiva que conta o número de nós em uma árvore binária que possuem pelo menos um filho.
7. Seja $dh(t) = |h_e - h_d|$ o módulo da diferença entre as alturas das sub-árvores esquerda h_e e direita h_d de uma árvore binária t , sendo $dh(t) = 0$ quando $t == \text{NULL}$. A função $dh_max(t)$ deve calcular o valor máximo de dh para a árvore t . Implemente o código da função recursiva auxiliar $aux_dh_max(t)$, que, em um mesmo percurso, calcula a altura da árvore t (armazenando o resultado em $*h$) e retorna o dh máximo encontrado. Você pode utilizar as funções $\max(a,b)$ (retorna o máximo entre a e b) e $\text{abs}(a)$ (retorna o valor absoluto de a).
8. Escreva um algoritmo que recebe a raiz de uma árvore binária e um número h e retorna o número de nós de uma árvore binária que tem altura maior que h .
`int NumNosAlturaMaiorH (TNode *t1, int h);`