

Lista de Exercícios de Estruturas de Dados e seus Algoritmos

Dada a seguinte representação de uma árvore binária:

```
typedef struct ab{
    int info;
    struct ab *esq, *dir;
}TAB;
```

Escreva as seguintes funções:

(Q1) cópia de uma árvore: **TAB* copia (TAB *a);**

(Q2) espelho de uma árvore (o que está a esquerda na árvore original, estará a direita no espelho, e vice-versa): **TAB* espelho (TAB *a);**

(Q3) maior elemento da árvore: **TAB* maior(TAB *a);**

(Q4) uma função que, dadas duas árvores deste tipo, testa se estas árvores são iguais. A função retorna um se elas são iguais e zero, caso contrário. A função deve obedecer ao seguinte protótipo: **int igual (TAB* a1, TAB* a2);**

(Q5) se esta estrutura **TAB** tivesse um campo **cor (int cor)**, defina uma função em C que, ao receber uma árvore binária “sem cor” e totalmente balanceada (isto é, a distância da raiz a qualquer folha da árvore é sempre a mesma), retorne esta árvore com os nós coloridos somente de vermelho e preto, sendo que o nó pai NUNCA pode ter a mesma cor de seus filhos. A função deve possuir o seguinte protótipo: **void colore (TAB* arv);**

(Q6) descubra a quantidade de nós internos: **int ni (TAB *a);**

Dada a seguinte representação de uma árvore binária de busca (ABB):

```
typedef struct ab{
    int info;
    struct ab *esq, *dir;
}TABB;
```

Escreva as seguintes funções:

(Q7) maior elemento da árvore: **TABB* maior(TAB *a);**

(Q8) menor elemento da árvore: **TABB* menor(TAB *a);**

(Q9) uma função em C que, dada uma árvore binária de busca qualquer, retire todos os elementos ímpares da árvore original. A função deve ter o seguinte protótipo: **TABB* retira_impares(TABB* a);**

(Q10) uma função em C que, dada uma árvore binária de busca qualquer, retorne, num vetor, todos os elementos menores que **N**. A função deve ter o seguinte protótipo: **int*mN(TABB*a, int N);**

(Q11) Reescreva as operações básicas da ABB para poder armazenar as figuras geométricas quadrado, triângulo, retângulo e trapézio. A área da figura será usada para definir os elementos pertencentes a sub-árvore esquerda e a sub-árvore direita. A impressão será em ordem simétrica e, além da área, devem ser informadas as dimensões da figura.