

Lista de Exercícios de Estruturas de Dados I

(Q1) Dada uma AVL que é uma folha com a informação 50:

- (a) insira os elementos {1, 64, 12, 18, 66, 38, 95, 58, 59, 70, 68, 39, 62, 7, 60, 43, 16, 67, 34, 35} nesta árvore; e
- (b) retire os elementos {50, 95, 70, 60, 35} desta árvore.

Dada a seguinte representação de uma árvore binária:

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

Escreva as seguintes funções:

(Q2) cópia de uma árvore: **TAB* copia (TAB *a);**

(Q3) espelho de uma árvore (o que está a esquerda na árvore original, estará a direita no espelho, e vice-versa): **TAB* espelho (TAB *a);**

(Q4) maior elemento da árvore, se a árvore original é ABB: **TAB* maior(TAB *a);**

(Q5) menor elemento da árvore, se a árvore original é ABB: **TAB* menor(TAB *a);**

Dada a seguinte representação de grafos:

```
typedef struct viz{
    int no, custo;
    struct viz *prox_viz;
}TV;
typedef struct grafo{
    int no;
    struct grafo *prox;
    TV *prim;
}TG;
```

Escreva as seguintes funções:

(Q6) descubra a quantidade de nós: **int nn(TG *g);**

(Q7) ache a quantidade de vértices: **int nv(TG *g);**

(Q8) se esta estrutura tivesse um campo cor (int cor) na estrutura de grafo, teste se nós vizinhos não tem a mesma cor. A função retorna um se os vizinhos não tem a mesma cor e zero, caso contrário: **int nao_tem_mesma_cor(TG *g);**

(Q9) Dada a seguinte estrutura de aluno:

```
typedef struct aluno{
    int mat;
    float cr;
}TA;
```

Escreva uma função de ordenação de um vetor: **void ordena (TA *vet, int n);**