

INF 1620 – P2 - 23/10/02	Questão 1
Nome:	
Matrícula:	Turma

Considere as seguintes declarações de uma lista encadeada para armazenar valores inteiros:

```
struct lista {
    int info;
    struct lista* prox;
};
typedef struct lista Lista;
```

Escreva uma função que receba uma lista e um valor inteiro x , e tenha como valor de retorno um ponteiro para o *último* nó que armazena um valor igual a x .

Essa função deve obedecer o protótipo:

```
Lista* ultimo_x (Lista* l, int x);
```

INF 1620 – P2 - 23/10/02	Questão 2
Nome:	
Matrícula:	Turma

Considere uma lista encadeada que armazena um conjunto de retângulos. Nessa lista, os retângulos são representados pelos valores da base e da altura. Os tipos dos dados são apresentados abaixo:

```
struct rect {
    float b;    /* valor da base */
    float h;    /* valor da altura */
};
typedef struct rect Rect;

struct lista {
    Rect info;
    struct lista *prox;
};
typedef struct lista Lista;
```

Considerando que os retângulos da lista são armazenados em ordem crescente de área, escreva uma função que insira um novo retângulo na lista (mantendo a ordenação). A função deve ter o protótipo abaixo e ter como valor de retorno o ponteiro para o primeiro nó da lista após a inserção:

```
Lista* insere (Lista* l, float b, float h);
```

INF 1620 – P2 - 23/10/02	Questão 3
Nome:	
Matrícula:	Turma

Considere a existência de um tipo abstrato `Pilha` de números inteiros, cuja interface está definida no arquivo `ilha.h` da seguinte forma:

```
typedef struct pilha Pilha;  
Pilha* cria(void);  
void push (Pilha* p, int v);  
int pop (Pilha* p);  
int vazia (Pilha* p);  
void libera (Pilha* p);
```

Sem conhecer a representação interna desse tipo abstrato `Pilha` e usando apenas as funções declaradas no arquivo `ilha.h`, implemente uma função que verifique se os valores armazenados nos dois elementos do topo da pilha são iguais. A função deve retornar 1 se os elementos forem iguais ou 0 se os elementos forem diferentes. Se não existir pelo menos dois elementos na pilha, a função deve retornar -1. Em qualquer caso, após a chamada da função a pilha deve ter os mesmos elementos existentes antes da função ser chamada. O protótipo da função deve ser:

```
int topo_iguais (Pilha* p);
```

INF 1620 – P2 - 23/10/02	Questão 4
Nome:	
Matrícula:	Turma

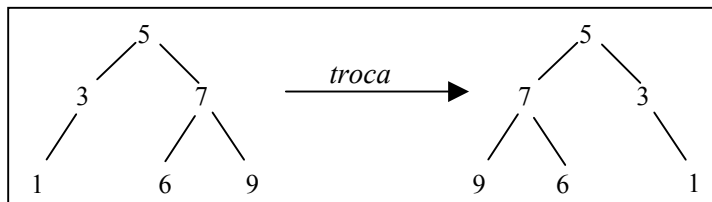
Considerando a estrutura de árvore binária definida pelo tipo abaixo:

```
struct arv {  
    int info;  
    struct arv *esq, *dir;  
};  
typedef struct arv Arv;
```

Escreva uma função que troque a sub-árvore à esquerda com a sub-árvore à direita de todos os nós da árvore. O protótipo da função é dado por:

```
void troca (Arv* a);
```

A figura abaixo ilustra o efeito dessa função.



Não separe as folhas deste caderno. Todas as folhas devem ter seu nome. Responda cada questão na folha correspondente. Use o verso se necessário.

INF 1620 – P2 - 23/10/02	Questão 5
Nome:	
Matrícula:	Turma

Suponha as seguintes declarações, relativas a um tipo “árvore genérica”, ou seja, uma árvore cujos nós têm um número não limitado de filhos, que armazena valores inteiros.

```
struct arvgen {
    int info;
    struct arvgen *prim; /* ponteiro p/ primeiro filho */
    struct arvgen *prox; /* ponteiro p/ irmão */
};
typedef struct arvgen ArvGen;
```

Escreva uma função que tenha como valor de retorno a soma dos valores inteiros armazenados em todos os *nós folhas* de uma dada árvore. O protótipo da função é dado por:

```
int soma_folhas (ArvGen* a);
```

Obs: *nó folha* é um nó que não tem filhos.

RASCUNHO

Respostas nesta folha não serão consideradas.

Não separe as folhas deste caderno.