

Lista 11 de Exercícios

Exercícios sobre Listas Encadeadas

1. Considere uma lista encadeada que armazena valores reais, na qual cada elemento é do tipo Elemento, descrito abaixo:

```
struct elemento {  
    float info;          /* Valor real armazenado          */  
    struct elemento *prox; /* Ponteiro para o próximo elemento    */  
};  
typedef struct elemento Elemento;
```

Escreva uma função que recebe como parâmetros o ponteiro lst, para o primeiro elemento da lista, e os valores min e max, e remove e libera todos os elementos desta lista que armazenam valores menores que min ou maiores que max. A função deve retornar o ponteiro para o início da lista, que pode ter sido modificado, e tem o seguinte protótipo:

Elemento* filtra(Elemento *lst, float min, float max);

2. Considere listas encadeadas que armazenam valores inteiros, nas quais cada elemento é do tipo Elemento, descrito abaixo:

```
struct elemento {  
    int info;            /* Valor inteiro armazenado          */  
    struct elemento *prox; /* Ponteiro para o próximo elemento    */  
};  
typedef struct elemento Elemento;
```

Escreva uma função que cria uma cópia de uma lista encadeada, ou seja, cria uma nova lista encadeada cujo conteúdo armazenado é idêntico --- tem os mesmos valores e na mesma ordem --- ao de uma lista original. A função recebe como parâmetro o ponteiro para o primeiro elemento da lista original (àquela a ser copiada) e retorna um ponteiro para a nova lista (a cópia). A lista original não deve ser alterada. O protótipo da função é:

Elemento* copia(Elemento* lst);

3. Crie uma lista encadeada onde cada elemento contém informações de um aluno. Crie funções para o gerenciamento da lista: imprimir, excluir, incluir e alterar. Aplique essas funções a partir de uma lista carregado do arquivo.