

**Estrutura de Dados**  
**Exercícios: *Listas com encadeamento simples (cont.)***

## 1 Instruções

1. Desenvolva os exercícios abaixo em C, utilizando como referência as informações apresentadas na aula e o material que se encontra no Moodle.
2. O objetivo desta lista de exercício é desenvolver um novo tipo *lista* utilizando *structs*, ponteiros e alocação dinâmica de memória.
3. Estenda a sua implementação da lista desenvolvida na aula passada com as funcionalidades pedidas abaixo.
4. Como de costume, crie um programa cliente para testar a sua implementação.

## 2 Exercícios

1. Faça uma função que calcula e retorna o tamanho de uma lista.
2. Faça uma função que busca um elemento numa lista. A sua função deve retornar o índice do elemento na lista ou -1 caso ele não ocorra na lista.
3. Faça uma função insere um novo elemento no final da lista.
4. Implemente a função com o cabeçalho abaixo:

```
List* insert_list(List *lst, int info, int pos);
```

que insere o elemento `info` na posição `pos` da lista. *Obs.:* se `pos ≤ 0` insira na primeira posição e se `pos ≥ size(lst)` insira no final.

5. Reescreva as funções de inserção no início e no final utilizando a função desenvolvida no exercício anterior.
6. Implemente uma função que retira um elemento de uma dada posição da lista.
7. Implemente uma função que busca e retira um elemento de uma lista. Utilize a função desenvolvida no item 2.
8. Considere uma lista de inteiros ordenada de forma não-decrescente. Crie uma função que insere um novo elemento nessa lista mantendo a ordenação.