

## Lista 3

### Listas encadeadas

1. Considerando uma lista encadeada de inteiros do tipo `Lista`, faça uma função que retorne o número de nós da lista que possuem o campo `info` com valores maiores que  $n$ . Essa função deverá obedecer ao protótipo:  
`int maioresQue(Lista *l, int n);`
2. Faça uma função que insira elementos no final da lista.  
`Lista* inserirFinal(Lista *p, int elem);`
3. Faça uma função que retorne o ponteiro para o último nó de uma lista de inteiros.  
`Lista* ultimo(Lista *l);`
4. Faça uma função que receba duas listas de inteiros e retorne a lista resultante da concatenação das duas listas recebidas como parâmetro.  
`Lista* concatena(Lista *l1, Lista *l2);`
5. Faça uma função que receba como parâmetros uma lista e um valor inteiro  $n$ , retire da lista todas as ocorrências de  $n$  e retorne a lista resultante.  
`Lista* retiraElemento(Lista *l, int n);`
6. Faça uma função para testar se duas listas passadas como parâmetros são iguais.  
`int iguais(Lista *l1, Lista *l2);`
7. Faça um TAD para criar listas de pessoas que contenham nome e idade. O TAD deverá conter as seguintes operações:
  - a) Estrutura para o tipo (`struct`)
  - b) Criar uma nova lista
  - c) Inserir novo contato
  - d) Remover um contato
  - e) Mostrar lista de contatos
  - f) Buscar e mostrar um contato