

1. Implemente uma função que tenha como valor de retorno o comprimento de uma lista encadeada, isto é, que calcule o número de nós de uma lista.
2. Considere listas encadeadas de valores inteiros e implemente uma função para retornar o número de nós da lista que possuem o campo info com valores maiores que n.
3. Implemente uma função que receba duas listas encadeadas de valores inteiros e retorne a lista resultante da concatenação das duas listas recebidas como parâmetros, isto é, após a concatenação, o último elemento da primeira lista deve apontar para o primeiro elemento da segunda lista.
4. Considere listas de valores inteiros e implemente uma função que receba como parâmetro uma lista encadeada e um valor inteiro n, retire da lista todas as ocorrências de n e retorne a lista resultante.
5. Considere listas de valores inteiros e implemente uma função que receba como parâmetro uma lista encadeada e um valor inteiro n e divida a lista em duas, de forma à segunda lista começar no primeiro nó logo após na primeira ocorrência de n na lista original.
6. A função deve retornar um ponteiro para a segunda subdivisão da lista original, enquanto l deve continuar apontando para o primeiro elemento da primeira subdivisão da lista.
7. Implemente uma função que construa uma nova lista com a intercalação dos nós de outras duas listas passadas como parâmetros. Essa função deve retornar a lista resultante.
8. Considere listas que armazenam inteiros e implemente uma função para testar se duas listas passadas como parâmetros são iguais.
9. Elabore uma função que insira elementos numa lista simplesmente encadeada ordenada de maneira decrescente.
10. Faça uma função para inserir elementos numa lista circular simplesmente encadeada.