



## Lista de Exercícios 1

### Listas simplesmente encadeadas

1. Escreva funções, em C, que recebam uma lista simplesmente encadeada ordenada, e realizem as seguintes operações:
  - a. retirar todos os elementos ímpares que são primos, e, gerar uma outra lista contendo apenas os elementos retirados da lista inicial, em ordem decrescente;
  - b. retirar todos os elementos pares da lista inicial.

**OBS.**: utilize o tipo Nodo criado em aula.

2. Escreva uma função, em C, que receba uma lista simplesmente encadeada, e retorne uma lista ordenada (crescente) sem alocar espaços para novos elementos da lista.
3. Escreva uma função, em C, que receba duas listas simplesmente encadeadas A e B, contendo informações (Código e Nome) dos pacientes do médico X e do médico Y (respectivamente), de uma determinada clínica. A clínica deseja possuir uma lista única com as informações dos pacientes desses dois médicos a partir da junção dessas duas listas. Considere que as listas A e B estão ordenadas por código de pacientes e que não existem dois (ou mais) pacientes com o mesmo código. A lista resultante deve conter todos os pacientes dos médicos X e Y ordenados por código e sem elementos repetidos.

**OBS.**: A função não deverá alocar nenhum nó, e todas as funções utilizadas na resolução deverão ser implementadas.

4. Escreva uma função, em C, para eliminar todos os elementos inteiros repetidos em uma lista simplesmente encadeada. Considere que os elementos nesta estrutura não estão ordenados.
5. Considere uma lista simplesmente encadeada que armazene números em ordem crescente. Escreva uma função, em C, que receba esta lista, e remova o nó antecessor e o sucessor a um determinado nó contendo um elemento X (inteiro).
6. Escreva uma função, em C, que receba uma lista simplesmente encadeada contendo em cada nó um inteiro, e retorne uma outra lista contendo nas

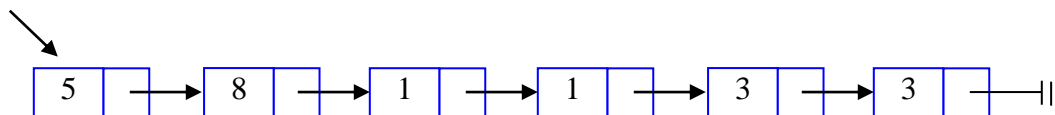


primeiras posições os elementos pares em ordem decrescente, seguidos dos elementos ímpares em ordem crescente.

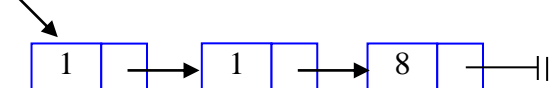
**OBS.:** A função não deverá alocar nenhum nó, e todas as funções utilizadas na resolução deverão ser implementadas.

7. Escreva uma função, em C, para somar dois números inteiros. Cada número está armazenado em uma lista simplesmente encadeada dinâmica, e cada nó da lista contém um dígito de um número e um apontador para o próximo nó. O primeiro nó de cada lista contém o dígito mais significativo (dígito mais à esquerda do número) e o último nó de cada lista contém o dígito menos significativo (dígito mais à direita do número). A função deve receber como parâmetros o endereço do início da lista que contém o primeiro número e o endereço do início da lista que contém o segundo (as duas listas já existem). A função deve retornar o endereço do início da lista que conterá o número correspondente à soma dos dois números. Veja o exemplo abaixo que soma os números 581133 e 118:

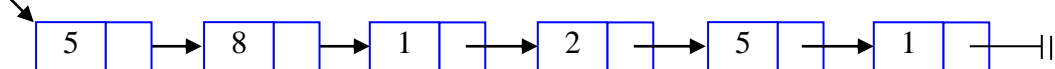
início1



início2



início3



8. Uma maneira de representar um conjunto é pela lista de seus elementos. Supondo esta representação, escreva uma função, em C, que receba duas listas simplesmente encadeadas e implemente a operação de interseção. O resultado final deve ser apresentado através de uma outra lista simplesmente encadeada (estática).

### Listas duplamente encadeadas



UNIVERSIDADE DO ESTADO DA BAHIA  
DEPARTAMENTO DE CIÊNCIAS EXATAS E DA TERRA  
SISTEMAS DE INFORMAÇÃO  
ESTRUTURA DE DADOS I

9. Escreva um programa, em C, para inserir, retirar e consultar uma lista duplamente encadeada ordenada em forma decrescente. Após cada operação, o programa deve informar a quantidade de nós, da direita para a esquerda e da esquerda para a direita. (implemente um menu de opções).
10. Considerando uma lista duplamente encadeada (dinâmica) contendo números inteiros, escreva uma função, em C, para eliminar todos os elementos múltiplos de três e retornar uma fila contendo estes elementos (ordenados).
11. Escreva uma função, em C, para transformar uma lista duplamente encadeada (dinâmica) em uma outra lista duplamente encadeada, da seguinte forma:
  - (i) o primeiro nó deve apontar para o último;
  - (ii) o último nó deve apontar para o segundo;
  - (iii) o segundo deve apontar para o penúltimo nó, e assim por diante.Neste processo de transformação nenhum novo nó da lista final deve ser alocado.
12. Considere uma lista simplesmente encadeada na qual cada nó contém as seguintes informações: matrícula (tipo inteiro) de um professor, ponteiro para o próximo nó dessa lista e um ponteiro para o primeiro nó de uma lista que contém os códigos (tipo inteiro) das disciplinas lecionadas por este professor. Implemente uma função em C para imprimir a matrícula do professor que leciona mais disciplinas.