

# Universidade Federal de Viçosa - *Campus* de Rio Paranaíba - MG

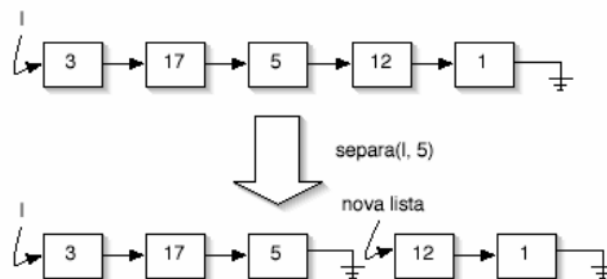
SIN 211 - Algoritmos e Estruturas de Dados  
Professor: Joelson Antônio dos Santos  
*email: joelsonn.santos@gmail.com*  
Sala PVA 233

April 10, 2018

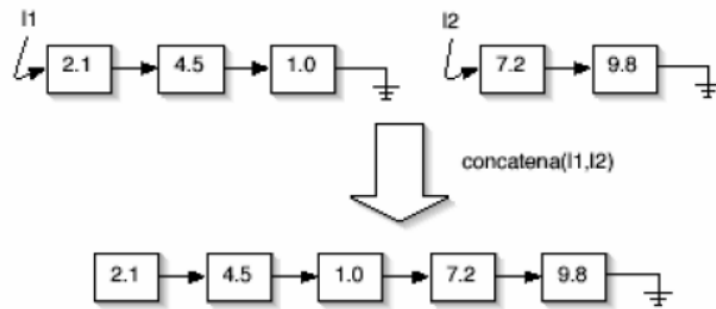
## Exercícios

1. Considerando uma *TAD lista simplesmente encadeada*, em que situações seriam apresentados o melhor e pior caso de cada operação abaixo? Justifique sua resposta para cada item apresentado.
  - (a) Inserir um elemento em uma “posição” K da lista.
  - (b) Remover um elemento em uma “posição” K da lista.
  - (c) Inserir no fim da lista.
  - (d) Inserir no início da lista.
2. Considerando uma *TAD lista simplesmente encadeada circular* (implementada com nó cabeçalho), em que situações seriam apresentados o melhor e pior caso de cada operação abaixo? Justifique sua resposta para cada item apresentado.
  - (a) Inserir no fim da lista.
  - (b) Inserir no início da lista.
  - (c) Remover no início da lista.
  - (d) Remover no fim da lista.
3. Implemente uma *TAD lista simplesmente encadeada* que manipule inteiros (campo info = **int**) e que contenha as seguintes operações:
  - Inserir em uma “posição” K da lista.
  - Remover em uma “posição” K da lista.
  - Liberar/Destruir todos os nós da lista.
  - Crie na **função principal** um menu para que o usuário possa realizar testes relacionados a cada operação criada anteriormente.
4. Explique sobre os métodos de listas auto organizadas e o método ordenado.
5. Crie funções para os métodos auto organizados de ordenação para uma **lista simplesmente encadeada**:
  - (a) Método mover para frente.
  - (b) Método de transposição.
  - (c) Método de contagem.

6. Implemente uma **TAD lista estática** que manipule inteiros (campo `info = int`) e que contenha as seguintes operações:
- Inserir no início.
  - Inserir no final.
  - Remover no início.
  - Remover no final.
  - Remover em uma posição  $K$ ;
  - Inserir em uma posição  $K$ ;
  - Crie na **função principal** um menu para que o usuário possa realizar testes relacionados a cada operação criada anteriormente.
7. Imagine que você seja um programador de uma empresa de desenvolvimento de aplicativos *android* e que seu chefe deseja que você desenvolva uma aplicação de reprodução de músicas. Para o desenvolvimento dessa aplicação, você precisa levar em conta algumas informações que devem ser consideradas no momento do cadastro de cada música na aplicação, tais como: *nome da música*, *duração em minutos* e *banda*. Adicionalmente, seu chefe lhe sugere que a TAD adequada para a implementação da aplicação seja uma **lista simplesmente encadeada circular**, pois os usuários terão a facilidade acessar, pelo fim, o início da lista. Para que isso seja possível, você deverá implementar as seguintes operações da aplicação:
- Inserir música no início.
  - Inserir música no final.
  - Remover música no início.
  - Remover música no final.
  - Mostre/imprima a lista de todas as músicas de uma única vez ao usuário.
  - Excluir todas as músicas.
  - Crie na **função principal** um menu para que o usuário possa realizar criar sua *playlist* favorita na nova aplicação de você implementou.
8. Considerando uma lista encadeada simples de valores inteiros, implemente uma função que receba como parâmetro um ponteiro para o ponteiro externo de uma lista e um valor inteiro  $n$  que indica qualquer elemento inteiro da lista, caso o mesmo exista. Assim, faça com que essa função divida a lista em duas, de tal forma que a segunda lista comece no primeiro nó/célula logo após a primeira ocorrência de  $n$  na lista original. A figura a seguir ilustra essa separação:



- A função deve retornar um ponteiro para a segunda sub-divisão da lista original, enquanto o ponteiro externo que aponta para o início da lista original deve continuar apontando para o primeiro elemento da primeira subdivisão da lista.
9. Considerando estruturas (TAD) de listas simplesmente encadeadas que armazenam valores inteiros. Implemente uma função que, dadas duas listas encadeadas ( $l1$  e  $l2$ ), concatene a lista  $l2$  no final da lista  $l1$ , conforme ilustra a figura abaixo:



- Observe que l1 e/ou l2 podem ser listas vazias.
10. Considerando estruturas (TAD) de listas simplesmente encadeadas que armazenam valores inteiros. Implemente uma função que, dadas uma lista, remova apenas os elementos que são pares.
  11. Considerando estruturas (TAD) de listas simplesmente encadeadas que armazenam valores inteiros. Implemente uma função que, dadas uma lista, imprima todos os elementos na ordem inversa.
    - Exemplo: se a lista contiver os elementos nesta ordem: (1,2,3,4,5), imprima os mesmo na função da seguinte forma: (5,4,3,2,1).