

## Lista de Exercícios 9 - Lista Simplesmente Encadeada com Descritor

1. A empresa XYZ possui uma lista com as informações de todos os funcionários. Para cada funcionário, as informações de código, nome, idade e salário são registradas. Com base nessas informações, crie um programa em C++ para gerenciar a lista de funcionários como segue:
  - a. Implemente uma lista simplesmente encadeada, projetada para armazenar os dados dos funcionários da empresa.
  - b. Crie um descritor para a lista contendo:
    - Ponteiro para o primeiro nó da lista
    - Ponteiro para o último nó da lista
    - Quantidade de nós
  - c. Implemente as seguintes funções:
    - Função para ler um funcionário no início da lista;
    - Função para remover um funcionário do início da lista;
    - Função para inserir funcionários no início da lista;
    - Função para inserir funcionários no final da lista:
      - Utilize o descritor para o último nó da lista, ou seja, não é necessário percorrer a lista para realizar a inserção.
    - Função para ler um funcionário no final da lista:
      - Utilize o descritor para o último nó da lista, ou seja, não é necessário percorrer a lista para realizar a leitura.
    - Função para imprimir os nós da lista:
      - Imprimir os dados de cada nó (código, nome, idade e salário).
      - Implemente via sobrecarga do operador <<, de forma que seja possível imprimir a lista com o comando *cout*.
    - Função para imprimir a média dos salários pagos pela empresa.
    - Função para imprimir o nome de todos os funcionários que possuem salário menor que a média.
    - Função para ordenar a lista de funcionários em ordem alfabética:
      - **Não é permitido** o uso de vetores ou estruturas de dados auxiliares. Altere apenas os valores dos nós, com base na comparação de *strings*. Não há necessidade de alterar os valores dos ponteiros, somente a variável *dado*.
      - Exemplo de como comparar duas string:

```
No* n;  
If (n!= NULL && n->prox != NULL && n->dado.nome > n->prox->dado.nome)  
{  
    //fazer a troca  
}
```
2. Crie as seguintes **funções** para o exercício anterior:
  - a) Inverter lista -> para implementar, use uma lista auxiliar e as funções insere fim, insere início, remove fim, remove início.
  - b) **Mostar** a lista ordenada pela idade dos funcionários: para implementar, use uma lista auxiliar para armazenar os funcionários ordenados e, em seguida, mostre a lista auxiliar.
  - c) **Unir duas listas**
  - d) **Intersecção de duas listas**
3. A empresa Carros PF precisa de um programa para gerenciar o cadastro de veículos. Para tanto, a empresa mantém uma lista de veículos, sendo que cada veículo possui os seguintes dados: placa, ano, fabricante, modelo.

Ao iniciar o sistema, a leitura do arquivo texto “cadastro.txt” é realizada. Exemplo de arquivo texto:

```
abc1256# 2000# volkswagen# fusca#  
xyz5566# 1975# willys# rural#  
klw6598# 2015# fiat# uno#
```

Cada linha do arquivo armazena os dados de um veículo, separados pelo delimitador “#” e estruturado da seguinte forma: placa (string), ano (número inteiro), fabricante (string) e modelo (string).

A cada linha do arquivo texto lida, o sistema cria um registro de um veículo e realiza a inserção **no início da lista** de veículos.

Após a leitura do arquivo texto e a criação da lista contendo os dados dos veículos, o sistema exibe o seguinte menu de operações:

- a) Mostrar lista de veículos;
- b) Ordenar lista de veículos: o sistema solicita ao usuário o tipo de exibição da lista, se deve ser ordenada pelo ano (ordem crescente) ou pelo modelo (ordem alfabética);
- c) Remover um veículo da lista: o sistema solicita ao usuário a placa e, em seguida, o veículo que possui a placa informada é removido da lista.
- d) Remover veículos com base no fabricante e no ano: o sistema solicita ao usuário o nome do fabricante e, em seguida, todos os veículos cujo o ano de fabricação é superior a 2000 e pertencem ao fabricante informado devem ser removidos da lista.
- e) Filtrar veículo: o usuário informa o nome do fabricante e o sistema mostra a placa dos veículos que são do fabricante informado.

Agora você deve escrever um programa que implementa o sistema para controlar tudo que foi descrito. Então mãos à obra! Analise o problema com atenção e veja quais as funcionalidades que devem ser implementadas.