

Estrutura de Dados 1 – Trabalho Final

Você foi contratado para o desenvolvimento de um sistema de cadastro de funcionários para a fábrica Acme®. Para tal, seu software deverá utilizar obrigatoriamente o padrão de estruturas de dados **Lista Ligada/Sequencial Dinâmica**.

A Lista Ligada implementada em seu programa deverá conter um ponteiro especial para o primeiro elemento (ponteiro para ponteiro). Os nós da lista deverão ser compostos por duas partes: a primeira parte armazenará os dados de um funcionário e será formada por uma **struct** com os dados pertencentes a este. A segunda parte armazenará um ponteiro com o endereço para o próximo nó da lista. O último nó da lista deverá apontar para NULL, indicando assim o final da mesma.

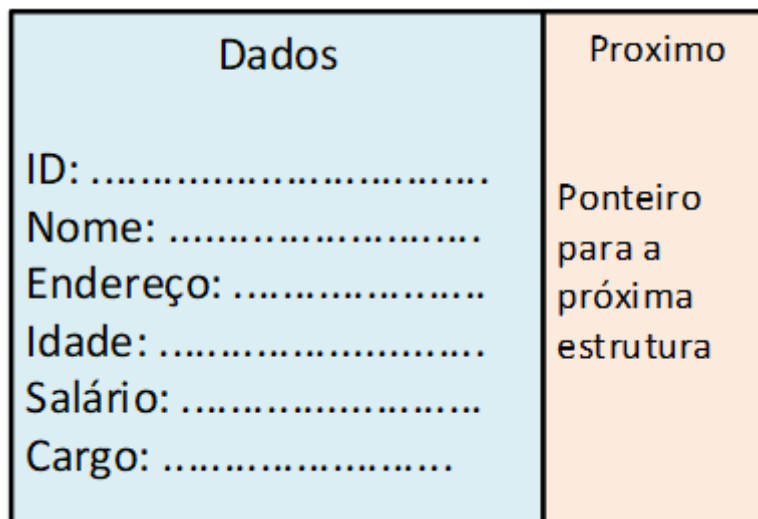


Figura 1 - Representação da estrutura (struct) nó, contendo todos os dados necessários para controle de uma entidade "Funcionário".

O projeto deverá ser montado utilizando o conceito de "Tipos Abstratos de Dados - TAD", e deverá contar com as **operações básicas necessárias** ao bom funcionamento do programa para controle de funcionários. Operações que não sejam necessárias ao funcionamento podem ser removidas. Os campos da **struct** destinados aos dados do funcionário, devem ter seus tipos definidos de forma a permitirem o correto armazenamento de todas as informações pertencentes a um funcionário. A Lista ligada que armazenará os dados dos funcionários da fábrica, deverá ser criada no momento da inicialização de seu programa, e logicamente destruída quando encerrada a aplicação.

Menu de opções

Seu software deve possuir ao menos um menu (podem ser criados quantos forem necessários), e contar com as seguintes funcionalidades:

- Adicionar funcionário;
- Excluir funcionário;
- Editar funcionário, inclusive com opção específica para aumento de salários por percentual de aumento (dissídio salarial informado pelo usuário);
- Buscar funcionário;
- Exibir funcionário ordenados por ID, com todos os seus dados;
- Exibir uma lista de funcionários por faixa salarial. Esta faixa deverá ser informada pelo usuário, com mínimo e máximo. Exemplo: de R\$ 1.000,00 à R\$ 2.000,00.
- Sair da aplicação.

Descrição das funcionalidades das opções de menu

- Adicionar funcionário:
 - O funcionário deve ser inserido na lista ordenadamente por ID;
 - Após a inclusão, a aplicação deve retornar ao menu principal.
- Excluir funcionário:
 - Antes de remover o funcionário, a aplicação irá exibir as informações do funcionário em questão, e em seguida perguntar se realmente o usuário deseja excluir o funcionário. Caso a resposta seja positiva, o funcionário deve ser excluído da lista.
 - Após a exclusão do funcionário, a aplicação deve retornar ao menu principal.
- Editar funcionário:
 - A aplicação primeiro localiza o funcionário por ID na lista, e em seguida serão exibidas as suas informações;
 - Neste ponto deve-se determinar se será uma **edição de informações** ou **reajuste salarial**;
 - Em seguida, no caso de edição, a aplicação pede a reinserção/correção dos dados e salva-os na lista;
 - Caso a opção seja “reajuste”, o mesmo será aplicado;
 - Após a edição das informações do funcionário ou reajuste salarial, a aplicação deve voltar ao menu principal.

- Buscar funcionários:
 - A aplicação primeiro localiza o funcionário por ID na lista, e em seguida serão exibidas as informações do funcionário;
 - Após exibir as informações do funcionário, a aplicação deve voltar ao menu.
- Exibir funcionários ordenado por ID:
 - A aplicação irá exibir o ID, nome, cargo e salário dos funcionários;
 - Após exibir as informações dos funcionários, a aplicação deve retornar ao menu principal.
- Exibir funcionários por faixa salarial:
 - A aplicação irá exibir o ID, nome, cargo e salário dos funcionários, selecionados entre um intervalo mínimo e máximo de salários, que deverá ser passado pelo usuário;
 - Após exibir as informações dos funcionários, a aplicação deve retornar ao menu principal.
- Sair da aplicação;
 - A aplicação será encerrada, liberando toda a memória utilizada para o processamento de seus dados.

Regras:

- Implementar o código na Linguagem C;
- Utilizar exclusivamente estruturas de dados **alocadas dinamicamente**, sendo portanto, obrigatório o conceito de lista ligada ou dinamicamente encadeada;
- No programa principal, o seu arquivo *"main()"*, você deverá somente "chamar as funções" do programa. Nenhum dado de funcionário deve ser preenchido, impresso ou editado no *"main()"*, **para isso modularize seu programa, crie funções específicas!**
- Clareza, indentação e comentários que sejam pertinentes são bem vindos;
- **Trabalhos copiados (e o Trabalho que serviu de fonte, inclusive) terão nota 0. Todos os códigos serão submetidos ao detetor de plágio, que além de verificar possíveis cópias em todos os projetos enviados, também verificará códigos oriundos da internet;**
- O trabalho será realizado no máximo em duplas;
- Use o bom senso, **não complique!** Implemente uma funcionalidade por vez, facilitando assim a correção de possíveis erros durante seu desenvolvimento.

O Trabalho final deverá ser entregue com:

- Código fonte (bem indentado e comentado);
- Arquivo executável, e todos os arquivos gerados pelo compilador (.o, .depend, .layout, .cbp);
- Relatório:
 - Explicar as decisões de implementação tomadas, uma visão geral do funcionamento do programa, comentários sobre os testes executados, etc.
 - Manual **básico** de como utilizar o programa (máximo 2 páginas);
 - Resultado dos testes, quais foram os problemas, como foram resolvidos, etc.

Todo o material deve ser entregue na plataforma Moodle, compactado (zipado) em data especificada na plataforma.

Não há a possibilidade de alteração da data de entrega.