

UNIFEI	<p style="text-align: center;">Universidade Federal de Itajubá</p> <p style="text-align: center;">Instituto de Engenharia de Sistemas e Tecnologias da Informação-IESTI</p>
5º Laboratório de Estrutura de Dados Prof^a. Thatyana de Faria Piola Seraphim	

Esta atividade de laboratório da disciplina de ECOP12 deverá ser postado como tarefa no SIGAA até o prazo definido no SIGAA. O aluno que não postar a atividade, ficará com falta no laboratório.

Crie um arquivo chamado **ecop12-labo5.c**, que será utilizado para implementar as funções para o laboratório de lista dinâmica duplamente encadeada.

No período universitário, criam-se vários amigos e por isso você decidiu implementar uma agenda para não perder seus contatos. Para que sempre possa inserir novos amigos essa agenda será implementada por uma lista dinâmica duplamente encadeada. Para manter a organização da estrutura, essa lista será sempre organizada em função da ordem lexicográfica do nome dos seus contatos.

a) (1 ponto) Declare duas constantes: *TAM_NOME* e *TAM_TELEFONE*. Inicialize ambas com os valores 20 e 10, respectivamente.

b) (1 ponto) Inclua a definição da estrutura *noContato*:

```
typedef struct no{
    char nome[TAM_NOME];
    char telefone[TAM_TELEFONE];
    struct no *prox;
    struct no *ant;
} noContato;
```

Cada elemento do tipo *noContato* armazenará as informações de um contato da sua agenda. As informações que serão mantidas para cada contato serão: seu nome e seu número de telefone. Além disso, ele terá dois ponteiros: *prox* e *ant*. O ponteiro *prox* armazena o endereço do próximo contato da agenda (se existir), ou contém o valor NULL se for o último contato da agenda. Já o ponteiro *ant* armazena o endereço do contato anterior da agenda (se existir), ou contém o valor NULL se for o primeiro elemento da agenda.

c) (2 pontos) Crie uma função chamada **insereContatoNaAgenda**, que realizará a inserção de um novo contato em uma lista duplamente encadeada (agenda). A inserção do novo aluno sempre será feita de acordo com a ordem lexicográfica do nome do novo contato, mantendo assim os contatos da agenda em ordem crescente em relação aos nomes deles. Para realizar a **comparação entre strings**, utilizar a função `strcmp` disponível na biblioteca *string.h*. Essa função receberá como parâmetro, um ponteiro para o novo contato a ser inserido. Esse parâmetro refere-se ao contato já alocado dinamicamente antes da chamada da função **insereContatoNaAgenda**. A função deverá retornar verdadeiro se o novo contato a ser inserido não constar na agenda, ou seja, se for realmente um novo contato. Caso o contato passado como parâmetro já estiver na agenda, a função retornará falso.

d) (1 ponto) Faça uma função para impressão das informações de todos os contatos que já estão na agenda. A função deverá ser denominada **imprimeAgenda**. Ela não retornará nada (void) e receberá um único parâmetro: um ponteiro para o primeiro elemento da sua agenda de contatos, ou seja, do tipo *noContato**. Essa função é responsável por imprimir, na ordem crescente dos nomes dos contatos, os dados da lista duplamente encadeada recebida como parâmetro pela função. Ou seja, para cada elemento da lista deverão ser impressos o nome e o telefone do contato correspondente.

e) (2 pontos) Faça uma função chamada **removeContatoDaAgenda**. Esta função é responsável pela remoção de um contato de sua agenda. Ela recebe como parâmetro o telefone do contato a ser removido. A função **removeContatoDaAgenda** deverá retornar verdadeiro se o telefone informado for encontrado na agenda e removido com sucesso. Caso o contato passado como parâmetro não estiver na agenda, a função retornará falso.

f) () Faça uma função chamada **destroiContato**. Esta função é responsável pela liberação de toda a memória usada durante a execução do programa. A função não retorna nada e não recebe nenhum parâmetro. Ela deverá ser chamada no programa principal.

g) (2 pontos) Crie a função principal do programa (**main**) e nela inclua a definição de um menu com as seguintes opções para o usuário:

```
Digite uma das opcoes abaixo:
>> 1 para inserir um contato na agenda
>> 2 para remover um contato na agenda
>> 3 para listar todos os contatos da agenda
>> 4 para sair da agenda
```

Para cada opção do menu faça a chamada da função correspondente.

h) Faça o teste da aplicação, solicitando a inserção dos seguintes nomes, nesta ordem:

1. nome: Zé Silva – tel: 1111-1111
2. nome: Ana Maria – tel: 2222-2222
3. nome: Ana Julia – tel: 3333-3333
4. nome: Bernardo Soares – tel: 4444-4444
5. nome: Ana Julia – tel: 3333-3333 **(novamente)**
6. nome: Vagner Santos – tel: 5555-5555

i) Por fim, solicite a remoção dos seguintes nomes:

1. Zé Silva
2. Ana Julia
3. Vagner Santos
4. Paulo Abreu

j) Imprima a lista resultante

k) Faça a chamada da função destroiContato.