



UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO – CAMPUS PAU DOS FERROS
BACHARELADO EM TECNOLOGIA DA INFORMAÇÃO
DISCIPLINA: LABORATÓRIO DE ALGORITMOS E ESTRUTURAS DE DADOS I

LISTA DE EXERCÍCIOS - LISTAS ENCADEADAS PRÁTICA 2

Questão 1) Crie um tipo abstrato de dados para representar uma lista duplamente encadeada de valores inteiros, implementando as operações ilustradas em sala de aula. A interface do módulo pode ser dada pelo arquivo *lista2.h* a seguir:

```
/* TAD: lista de inteiros */  
  
typedef struct lista2 Lista2;  
  
Lista2* lst2_cria(void);  
void lst2_libera(Lista2* l);  
  
Lista2* lst2_insere(Lista2* l, int v);  
Lista2* lst2_retira(Lista2* l, int v);  
  
int lst2_vazia(Lista2* l);  
Lista2* lst2_busca(Lista2* l, int v);  
void lst2_imprime(Lista2* l);
```

Questão 2) Implemente uma função que imprime as informações dos nós de uma lista circular de inteiros com encadeamento simples. Essa função deve obedecer ao protótipo:

```
void lcirc_imprime (Lista* l);
```

Questão 3) Implemente uma função que percorre uma lista circular duplamente encadeada de inteiros, imprimindo as informações de seus nós em ordem reversa. Essa função deve obedecer ao protótipo:

```
void l2circ_imprime_rev (Lista2* l);
```

Questão 4) Implemente funções para inserir e retirar um elemento de uma lista circular duplamente encadeada.

Questão 5) Considere uma lista encadeada para armazenar os registros de funcionários de uma empresa. Um funcionário é definido pela estrutura a seguir:

```
struct funcionario {  
    char depto;    /* departamento */  
    int mat;       /* matrícula */  
    char nome[81];
```

```
        float salario;  
    };  
typedef struct funcionario Funcionario;
```

Implemente uma função que retorna o valor gasto com a folha de pagamento para um dado departamento da empresa. Essa função deve obedecer ao seguinte protótipo:

```
float lstfunc_folha_pagto (ListaFunc* l, char depto);
```

Questão 6) Escreva um programa que manipula listas de objetos geométricos planos para cálculo de áreas.

$$A_{\text{ret}} = b \cdot h, \quad A_{\text{tri}} = (b \cdot h)/2, \quad A_{\text{circ}} = \pi r^2$$

- a) Defina um tipo para cada objeto a ser representado;
- b) Defina a estrutura de um nó da lista de objetos;
- c) Implemente três variações para a função que cria um nó da lista;
- d) Implemente uma função que fornece como valor de retorno a maior área entre os elementos da lista.

Sugestões de protótipos:

```
ListaHet* cria_ret (float b, float h);  
ListaHet* cria_tri (float b, float h);  
ListaHet* cria_circ (float r);  
static float ret_area (Retangulo* r);  
static float tri_area (Triangulo* t);  
static float circ_area (Circulo* c);  
static float area (ListaHet* p);  
float max_area (ListaHet* l);
```

OBS.: Os exercícios apresentados sugerem a implementação de diferentes funções. Sendo assim, você deve escrever um programa (função *main*) para testar sua implementação.