

Aula de exercícios- Listas Ligadas I (20/10/2014)

Para todos os exercícios, defina o TAD, suas operações, e faça um pequeno programa que utiliza e testa todas as funções implementadas. Não é necessário fazer entrada de dados via teclado. Faça um programa de teste determinístico.

Ao final da aula, envie todos os códigos feitos em um arquivo zip, com o título

“Aula 20/10 - [Nº Usp]” , em que [NºUsp] é o seu número usp

- 1) Defina as operações para o TAD para uma lista circular de inteiros com ponteiro para o primeiro nó como cabeçalho. O nó cabeçalho armazena o número de elementos da lista. Utilize a seguinte estrutura:

```
typedef struct type_node{
    struct type_node * prox;
    int v;
}Node;

typedef struct type_circular_list{
    Node *first;
    Int nElementos;
}CircularList;

void aloca(CircularList **L);
void libera(CircularList *L);
void insereOrdenado(CircularList *L, int v, int *erro);
Node* busca(CircularList *L, int v); // retorna o endereço do nó que contém o valor v
```

- 2) Defina as operações para o TAD “lista circular duplamente encadeada”.

```
typedef struct type_node{
    struct type_node * prox;
    struct type_node * prev;
    int v;
}Node;
typedef struct type type_list{
    Node *first;
    Node *last;
    Int nElementos;
}List;

void aloca(List **L);
void libera(List *L);
void insereComeco(List *L, int v, int *erro);
void insereFim(List *L, int v, int *erro);
int removeComeco(List *L, int *erro);
int removeFim(List *L, int *erro);
```