

UNIVERSIDADE FEDERAL DO MARANHÃO  
DEPARTAMENTO DE INFORMÁTICA  
II LISTA DE EXERCÍCIOS DE ESTRUTURA DE DADOS (PARTE A)  
PROFESSOR ALEXANDRE CÉSAR

- 1 Qual a vantagem de usar lista encadeada circular em relação a uma lista encadeada simples para implementar uma Fila.
  2. Desenvolva um procedimento para concatenar duas listas encadeadas apontadas por lista1 e lista2.
  - 3 Desenvolva um procedimento PTRNO PROCURA(PTRNO lista, int x) que procura por um valor x armazenado numa lista encadeada classificada circular. Ele deve retornar um ponteiro NULO, indicando que não encontrou x, ou um ponteiro com o endereço do nó onde x foi encontrado.
  - 4 Desenvolva um procedimento <int> TAMANHO( PTRNOCIRC p) que recebe um ponteiro para um nó qualquer de uma lista circular e retorna o número de nós dessa lista.
  - 5 Desenvolva um procedimento <int> TAMANHO( PTRNODUP p) que recebe um ponteiro para um nó qualquer de uma lista duplamente encadeada e retorna o número de nós dessa lista.
  - 6 Desenvolva um procedimento <int> ELIMINA( PTRNODUP p, INFO x) que elimina todas as ocorrências de x na lista referenciada por p e retorna o número de elementos eliminados.
  7. Implemente uma fila de prioridades usando uma lista circular
  8. Fazer um procedimento para exibir os elementos de uma lista na ordem inversa.
  9. Fazer um procedimento para mesclar duas listas em uma única lista ordenada.
- Seja  $L : [a_1; a_2; \dots ; a_n]$  uma lista linear. Então,  $\text{null}(L)$  é true se e só se  $L = []$ . Ademais, se  $L \neq []$ , temos que  $\text{head}(L)$  é  $a_1$  e  $\text{tail}(L)$  é  $[a_2; \dots ; a_n]$ . Usando essa notação escreva rotinas recursivas para:
10. Determinar o produto dos itens de uma lista linear. Assuma que os itens na lista sejam números reais. Por exemplo, devemos ter  $\text{prod}([5; 2; 3])$  é 30.
  11. Determinar se um certo item é membro de uma lista linear. Por exemplo, devemos ter  $\text{member}(c; [d; a; c; b])$  é true e  $\text{member}(e; [d; a; c; b])$  é false.

II Trabalho implementado: Uma agenda eletrônica armazenada em lista encadeada. A estrutura de dados completa é uma lista de listas. Cada nó da lista é um dia do mês (data: dia/mês/ano). Cada dia (nó da lista) guarda um resumo sobre aquele dia, como aniversários, eventos importantes, etc (campo string). Além disso, cada dia guarda uma referência (ponteiro) para uma outra lista com os eventos e seus respectivos horários. As seguintes opções mínimas devem ser dadas ao usuário: Inserir evento, apagar evento, buscar por evento, buscar todos eventos de um dia.

