



GOVERNO DO ESTADO DO RIO DE JANEIRO
SECRETARIA DE ESTADO DE CIÊNCIA E TECNOLOGIA
FUNDAÇÃO DE APOIO À ESCOLA TÉCNICA – FAETEC



FACULDADE DE EDUCAÇÃO TECNOLÓGICA DO ESTADO DO RIO DE JANEIRO

PROVA

| x | AV1 | AV2 | AVS | AVF |
|--|-----|--|--------------|-------------------------|
| Professor: <i>Leonardo Soares Vianna</i> | | Disciplina: <i>Estruturas de Dados</i> | | Data: <i>07/05/2015</i> |
| Aluno: | | Matrícula: | Nº: | Turma: |
| Nota | | Visto: | Nota revista | Nota por extenso |
| | | | | Visto: |

Questão 01 [2,0 pontos]:

Pilhas e filas, em suas implementações clássicas, não oferecem as operações de *remoção*, *alteração* e *busca* de determinado elemento.

Porém, suponha que a aplicação sobre a qual você está trabalhando necessite que estas funções sejam desenvolvidas, sempre respeitando as restrições impostas quanto à manipulação de tais estruturas.

Pede-se a descrição destes algoritmos nas duas estruturas.

Questão 02 [2,0 pontos]:

Listas duplamente encadeadas consistem em um tipo especial de estrutura de dados no qual cada nó possui um ponteiro para o elemento *anterior*, além dos campos já existentes na lista encadeada (o *valor* armazenado e o ponteiro para o *próximo* nó). Similarmente ao ponteiro *prox*, o *ant* permite que a lista seja percorrida, porém na ordem inversa (do último ao primeiro elemento), até chegar ao valor *NULL*.

Considere, então, uma *fila* implementada através de uma lista duplamente encadeada. Pede-se a descrição dos algoritmos de inserção e remoção de elementos nesta fila, supondo duas estratégias de implementação: inserção no início e inserção no final da fila. Além disso, estas duas estratégias devem ser comparadas e esta comparação deve incluir a análise de complexidade dos algoritmos.

Observação: as soluções propostas devem buscar algoritmos de "baixo custo".

Questão 03 [2,0 pontos]:

Para os problemas apresentados a seguir, pede-se a análise de complexidade das soluções propostas e o levantamento dos fatos que fazem com que uma seja mais eficiente do que a outra (caso apresentem custos distintos):

- a) Buscar um elemento em determinada coluna de uma matriz quadrada de ordem n .

Solução 1: o algoritmo percorre a matriz e sempre que encontrar um elemento que pertença à coluna desejada, compara-o com o valor sendo buscado.

Solução 2: o algoritmo percorre a coluna desejada e compara seus elementos com o valor sendo buscado.

- b) Remover todos os elementos de uma lista encadeada.

Solução 1: o algoritmo, de maneira iterativa, apaga um elemento a cada iteração até que a lista se torne vazia.

Solução 2: o algoritmo executa diversas chamadas recursivas, sempre informando o endereço do próximo elemento da lista, até chegar ao final da estrutura. Em seguida, cada chamada pendente apaga o elemento para o qual aponta, anulando (*NULL*) este ponteiro.

Questão 04 [2,0 pontos]:

Para cada lista encadeada apresentada a seguir, implementar uma função que retorne a quantidade de elementos distintos na estrutura:

- a) Lista sem ordenação;
b) Lista ordenada.

Questão 05 [2,0 pontos]:

Considere uma lista encadeada contendo valores numéricos. Desenvolver uma função que ordene seus elementos, crescentemente.