

UFMA – CCET – Ciência da Computação- Estrutura de Dados I
Primeira Avaliação

- 1- Escreva um algoritmo que recebe uma matriz quadrada de número inteiros, com n linhas e n colunas, armazenada no vetor v, e usa o TAD Stack para inverter a ordem dos elementos da coluna k da matriz.

int InverteOrdemElementosLinhaK(int *v, int n, int k)

Exemplo: Elementos da Coluna k < 11 14 23 32 >

Elementos da Linha k invertidos < 32 23 14 11 >

- 2- Escreva um algoritmo que recebe uma fila circular e move o elemento do meio da fila para a primeira posição.

typedef struct _queue_ {

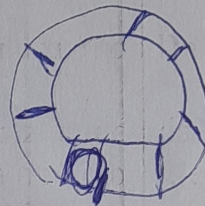
void *item;

int maxElms;

int numElms;

} Queue;

void MoveDoMeioProInicio(Queue *q);



- 3- Escreva um algoritmo que recebe uma pilha e remove todos os elementos de posição ~~ímpar~~ ^{par} a partir do topo. Considere a posição do topo como posição 1 e remova os das posições 2, 4, ... A ordem dos demais elementos da pilha deve ser mantida. Não pode usar funções do TAD Stack, e nem vetores auxiliares.

typedef struct _stack_ {

void *item;

int maxElms;

int top;

} Queue;

void RemovePares(Stack *s);

- 4- Faça um algoritmo que, usando o TAD Queue e o TAD Stack, recebe uma sequência de caracteres na qual temos letras e algarismo (um dígito) aparecendo de maneira alternada, que retorne uma sequência de caracteres na qual as letras são mantidas na sequência (ordem) original, e só depois que todas as letras aparecem os algarismos (dígitos) na ordem inversa. Exemplos:

A 1 E 5 T 7 W 8 G ----- A E T W G 8 7 5 1

3 C 9 H 4 Q 6 ----- C H Q 6 4 9 3

As letras devem ser mostradas primeiro, seguidas dos algarismos (dígitos). Não pode usar vetores auxiliares, somente chamadas às funções dos TADs.

char *ConverteString(char *str, int n)