

Lista Exercício - Lista, Fila e Pilha

Questão 1: Escreva uma função que receba como parâmetro uma lista encadeada contendo apenas valores 0 e 1 e que retorne as posições inicial e final da maior seqüência S de elementos 0 dentro da lista.

Ex: Lista={0,1,1,0,0,1,0}

pini = 4 e pfim = 6 S={0,0,0}

Obs: No caso de empate em tamanho, a primeira ocorrência de S é a que deve ser retornada.

Questão 2: Construa um programa em C que retira um elemento de uma lista encadeada dinamicamente da posição pos1 e coloca na pos2.

Questão3: Considere uma lista simplesmente encadeada L1 representando uma seqüência de caracteres. Construa uma função para imprimir a seqüência de caracteres da lista L1 na ordem inversa (não é permitido o uso de listas auxiliares).

Ex: Para a lista L1={A,E,I,O,U}, a função deve imprimir "UOIEA"

Questão 4: Considere duas listas simplesmente encadeada de números inteiros L1 e L2

- (a) Crie uma lista L3 que seja a união sem repetição de L1 e L2
- (b) Crie uma lista L4 que seja a intersecção dos elementos que estão em L1
- (c) Faça uma função que receba o cabeça de uma lista e retorne a quantidade de elementos dessa lista.
- (d) Crie uma lista L5 que contenha os primeiros $n/2$ elementos de L2. Sendo n o numero de elementos de L2.

Questão 5: Implemente uma fila dinâmica em C/C++, onde cada item da fila consista em um número variável de inteiros. Defina as operações:

- a) Verifica de fila está vazia.
- b) Enfileira elemento.
- c) Retira elemento da fila.

Questão 6: Implemente uma pilha dinâmica em C/C++, onde cada item da fila consista em um número variável de inteiros. Defina as operações:

- a) Verifica de pilha está vazia.
- b) Empilha elemento.
- c) Retira elemento da pilha.

Questão 7: Considere uma pilha P vazia e uma fila F não vazia. Utilizando apenas os testes de fila e pilha vazias, as operações Enfileira(), Desenfileira(), Empilha(), Desempilha(), e uma variável aux do TipoItem, escreva uma função que inverta a ordem dos elementos da fila.

```
struct TipoItem{
    int valor;
    struct TipoItem* prox;
}
```

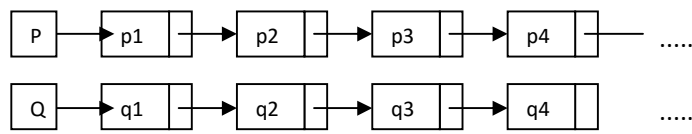
Questão 8: Um **deque** é um conjunto de itens a partir do qual podem ser eliminados e inseridos itens em ambas as extremidades. Chame as duas extremidades de um deque *esq* e *dir*. Como um deque pode ser representado como um vetor em C? Escreva quatro funções em

C: RemDir(), RemEsq(), InsDir(), InsEsq(), para remover e inserir elementos nas extremidades esquerda e direita de um **deque**. Certifique-se de que as funções funcionem corretamente para o deque vazio e detectem o estouro e o *underflow* (tentativa de remoção quando a estrutura está vazia). Quais as desvantagens dessa implementação com relação a implementação por encadeamento/alocação dinâmica?

Questão 9: Existem partes de sistemas operacionais que cuidam da ordem em que os programas devem ser executados. Por exemplo, em um sistema de computação de tempo compartilhado (*time-shared*) existe a necessidade de manter um conjunto de processos em uma **fila**, esperando para serem executados. Escreva um programa que seja capaz de ler uma série de solicitações para:

- a) Incluir novos processos na fila de processo;
 - b) Retirar da fila o processo com o maior tempo de espera;
 - c) Imprimir o conteúdo da fila de processo em determinado momento.
- Assuma que cada processo é representado por um registro composto por um número identificador do processo.

Questão 10: Partindo da situação mostrada no desenho abaixo, e sendo P e Q duas listas encadeadas dinâmicas, explique (com desenhos) o que acontece nas atribuições seguintes (obs: cada item parte da situação inicial mostrada abaixo).



- a) $P \rightarrow \text{prox} = Q \rightarrow \text{prox};$
- b) $P = Q;$
- c) $P = P \rightarrow \text{prox};$
- d) $P \rightarrow \text{dado} = (P \rightarrow \text{prox}) \rightarrow \text{dado};$
- e) $Q = \text{NULL};$

Questão 11: No desenvolvimento de um software que analisa bases de DNA, representadas pelas letras A, C, G, T, utilizou-se as estruturas de dados: pilha e fila. Considere que, se uma seqüência representa uma pilha, o topo é o elemento mais à esquerda; e se uma seqüência representa uma fila, a sua frente é o elemento mais à esquerda.

Análise o seguinte cenário: “a seqüência inicial ficou armazenada na primeira estrutura de dados na seguinte ordem: (A,G,T,C,A,G,T,T).

Cada elemento foi retirado da primeira estrutura de dados e inserido na segunda estrutura de dados, e a seqüência ficou armazenada na seguinte ordem: (T,T,G,A,C,T,G,A).

Finalmente, cada elemento foi retirado da segunda estrutura de dados e inserido na terceira estrutura de dados e a seqüência ficou armazenada na seguinte ordem: (T,T,G,A,C,T,G,A)”.

Qual a única seqüência de estruturas de dados apresentadas a seguir pode ter sido usada no cenário descrito acima? Faça o desenho das estruturas com seus respectivos elementos e apontadores.

- a) Fila - Pilha - Fila.
- b) Fila - Fila - Pilha.
- c) Fila - Pilha - Pilha.
- d) Pilha - Fila - Pilha.
- e) Pilha - Pilha - Pilha.
