

1. Um deque geral deve prover as seguintes operações:

procedimento *insereInicio*(**ref** D: Deque, x: <TElem>)

procedimento *insereFim*(**ref** D: Deque, x: <TElem>)

função *removeInicio*(**ref** D: Deque): <TElem>

função *removeFim*(**ref** D: Deque): <TElem>

função *buscaInicio*(**ref** D: Deque): <TElem>

função *buscaFim*(**ref** D: Deque): <TElem>

Defina o estrutura Deque e escreva os procedimentos e funções acima de forma análoga ao realizado com Pilhas e Filas. Considere o uso da alocação tanto sequencial quanto encadeada.

2. Reescreva as operações de Pilha utilizando duas Filas como estrutura de dados auxiliar para guardar os elementos. Manipule as filas por suas interfaces padrão.

3. Reescreva as operações de Fila utilizando duas Pilhas como estrutura de dados auxiliar para guardar os elementos. Manipule as pilhas por suas interfaces padrão.

4. Escreva um algoritmo que dada uma pilha P, inverta a ordem dos elementos de P. Seu algoritmo deve usar **espaço auxiliar constante** (*somente variáveis locais e nenhum vetor extra*) e:

- a) uma fila
- b) duas pilhas
- c) uma pilha

Seu algoritmo deve manipular as pilhas e filas por suas interfaces padrão (*somente métodos do TAD*).

5. Escreva um algoritmo que dada uma fila F, inverta a ordem dos elementos de F. Seu algoritmo deve usar **espaço auxiliar constante** (*somente variáveis locais e nenhum vetor extra*) e:

- a) uma pilha
- b) duas filas

Seu algoritmo deve manipular as pilhas e filas por suas interfaces padrão (*somente métodos do TAD*).

6. Criar uma variação de pilha, chamada de PilhaMin, que, além de fornecer as operações de pilha em tempo constante, define a operação

função *obtenMinimo*(**ref** P: PilhaMin): <TElem>

que retorna o elemento de P com a menor chave em tempo constante.

7. Escrever um algoritmo que converta uma expressão aritmética parentizada usando as 4 operações para a expressão correspondente em notação polonesa reversa. Ex: **Entrada:** ((A+B)*(C-(F/D))) **Saida:** AB+CFD/-*