

1) Considere uma pilha P vazia e uma fila F não vazia. Utilizando apenas os testes de fila e pilha vazias, as operações push e pop das duas estruturas, e uma variável aux do Tipoltem, escreva uma função que inverta a ordem dos elementos da fila.

2) Para um dado número inteiro $n > 1$, o menor inteiro $d > 1$ que divide n é chamado de fator primo. É possível determinar a fatoraçoão prima de n achando-se o fator primo d e substituindo n pelo quociente n / d , repetindo essa operação até que n seja igual a 1. Utilizando um das estruturas de dados vistos em sala (Pilha ou Fila) para auxiliá-lo na manipulação de dados, implemente uma função que compute a fatoraçoão prima de um número imprimindo os seus fatores em ordem decrescente. Por exemplo, para $n=3960$, deverá ser impresso $11 * 5 * 3 * 3 * 2 * 2 * 2$. Justifique a escolha da estrutura de dados utilizado.

3) Existem partes de sistemas operacionais que cuidam da ordem em que os programas devem ser executados. Por exemplo, em um sistema de computação de tempocompartilhadao ("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 lista de processo em determinado momento.

Assuma que cada processo é representado por um registro composto por um número identificador do processo.

4) Como você implementaria uma fila de pilhas? Uma pilha de filas? Uma fila de filas? Escreva rotinas apra implementar as operações corretas para cada uma destas estruturas de dados.

5) Suponha que o Beco do Pirão (Praça Tiradentes, Ouro Preto), durante a noite, seja usado como um estacionamento que guarda até 10 carros. Os carros entram pela Praça Tiradentes (PT) e saem pela Rua Barão de Camargos (RBC) (obs: fato fictício gerado a partir de informações extraídas de maps.google.com). Se chegar um cliente para retirar um carro que não esteja estacionado na primeira da RBC, todos os carros entre o carro do cliente e a RBC serão deslocados para fora do estacionamento, o carro do cliente sairá do estacionamento e os outros carros voltarão a entrar pela PT na mesma ordem que saíram pela RBC. Observe que sempre que um carro deixa o estacionamento, todos os carros entre ele e a PT serão deslocados até o começo da RBC de modo que, o tempo inteiro, todos os espaços

vazios estão na entrada do estacionamento, ou seja na entrada pela PT. Escreva um programa que leia um grupo de linhas de entrada. Cada linha contém um 'C', de chegada, e um 'P' de partida, além de um número de placa de licenciamento. Presume-se que os carros chegarão e partirão na ordem especificada pela entrada. O programa deve imprimir uma mensagem cada vez que um carro chegar ou partir. Quando um carro chegar, a mensagem deverá especificar se existe ou não vaga para o carro dentro do estacionamento. Se não existir vaga, o carro esperará pela vaga ou até que uma linha de partida seja lida para o carro. Quando houver espaço disponível, outra mensagem deverá ser impressa. Quando um carro partir, a mensagem deverá incluir o número de vezes que o carro foi deslocado dentro do estacionamento, incluindo a própria partida, mas não a chegada. Esse número será 0 se o carro for embora a partir da linha de espera.