

Atualizado!

Construir um programa para cada exercício, incluindo as bibliotecas de pilha, fila, listas ligadas o que for necessário para cada um. Desenvolver uma ou mais funções que necessite para cada um e construir a função `main` para testá-las, também para cada um deles.

**1) (pilhas)** Considere a estrutura pilha implementada sobre vetores e listas ligadas estudada em aulas. Considere agora o problema das Torres de Hanoi que não pode ter um valor maior empilhado sobre um valor menor.

Desenvolver uma função que recebe um elemento e faz o seu empilhamento, de acordo com a regra. Não manipule os dados internos da pilha, utilize somente as funções disponíveis. Você pode ter pilhas auxiliares nas suas funções para executar a tarefa.

Observação: fazer para as duas implementações.

**2) (filas)** Considere a seguinte situação: tem uma pessoa na primeira posição da fila, mas ela precisa esperar seu colega que foi buscar uma mercadoria que esqueceu. Desenvolva uma função `passa_vez`, que implementa a situação descrita.

**3) (recursão)** Uma função que encontra o maior valor em um vetor pode ser escrita da seguinte maneira:

```
int maximo (int v[], int n) {
    int i, m = v[0];
    for (i=1; i<n; i++)
        if (v[i] > m)
            m = v[i];
    return m;
}
```

Note que foi utilizado um comando de repetição para fazer a busca. Escreva a função `maximo_rec` que executa a mesma tarefa, utilizando recursão.

**4) (listas ligadas)** Desenvolver uma função que recebe uma lista e remove todos os elementos pares dela.