

Lista de exercício - PILHA

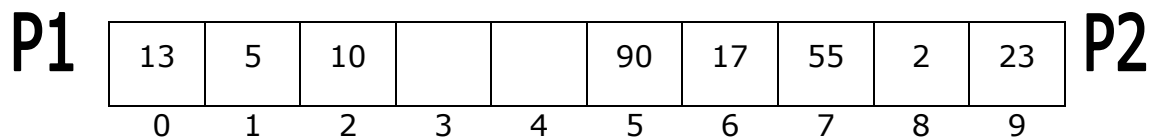
1. Faça um programa que manipule uma pilha A, usando um registro (struct) para armazenar os dados, onde as informações de topo, base e os valores da pilha são partes do registro. O programa deve permitir as ações básicas (empilhar, desempilhar, consultar topo e mostrar pilha). Além disso, as seguintes regras devem ser observadas:
 - Não permitir a inclusão (empilhar) de valores duplicados;
 - Ao remover (desempilhar) um elemento da pilha A, ele deve ser armazenado em um vetor de 20 posições, sendo possível remover somente se tiver espaço para incluir no vetor. No menu deve ter uma opção para mostrar os dados do vetor.
2. Faça uma função para retornar o número de elementos da pilha que possuem valor ímpar.
3. Faça uma função para retornar o número de elementos da pilha que possuem valor par.
4. Desenvolva uma função para testar se duas pilhas P1 e P2 são iguais.
5. Faça um programa que cadastre em uma estrutura do tipo pilha vários números, sendo no máximo 6. Ao remover um número desta estrutura o mesmo deve ser empilhado em outra pilha, conforme o critério:
 - a. se o número for par na pilha dos **pares**,
 - b. se for ímpar na pilha dos **ímpares**.

No menu de opções deve ter uma alternativa para zerar as três pilhas, mostrar pilha inicial, a dos pares e a dos ímpares.

6. Faça um programa que usa uma pilha para armazenar o código das cidades que fazem parte de um caminho.
 - Inicialmente o usuário deve informar o número de cidades;
 - Depois informar o código de cada cidade e incluir na pilha;
 - Criar uma funcionalidade de busca de um código na pilha: para fazer a busca deve usar uma pilha auxiliar; a cada item verificado na pilha principal que for diferente do código

procurado, deve incluir na pilha auxiliar; após encontrar o valor, deve retornar os valores da pilha auxiliar para a pilha principal.

7. Faça um programa que implemente duas pilhas em um único vetor, que deve receber no máximo 10 elementos. A pilha 1 (p1) deve ser implementada da esquerda para a direita, já a pilha 2 (p2) deve ser implementada da direita para a esquerda, conforme apresentado na figura abaixo:



Analisando a imagem acima entende-se que tanto a pilha 1 quanto a pilha 2 poderão usufruir dos 10 espaços disponíveis no vetor, desde que uma não sobreponha a outra, no exemplo:

P1 = 13, 5, 10

P2 = 23, 2, 55, 17, 90

Ainda existem dois espaços no vetor que podem ser usados por qualquer uma das pilhas.

Cabe a você definir as regras neste tipo de situação, tais como:

- Quando as pilhas estão vazias?
- Quando as pilhas estão cheias?

Implemente as rotinas necessárias possibilitando em cada uma das pilhas: inserir, excluir e listar os respectivos elementos.

8. Questão Enade

Uma estrutura de dados do tipo pilha pode ser usada em um algoritmo que permite imprimir uma palavra de forma invertida. Exemplo: FELICIDADE deve ser impresso como EDADICILEF.

Utilizando as variáveis declaradas abaixo:

```
pilha[1..50]: caractere;
```

```
i, topo: inteiro;
```

```
palavra: string;
```

Em pseudocódigo, faça o que se pede nos itens a seguir.

- a) Desenvolva a rotina *push* que inclui um elemento na pilha. (valor: 3,0 pontos)
- b) Desenvolva a rotina *pop* que retira um elemento da pilha. (valor: 3,0 pontos)
- c) Desenvolva a rotina que leia a palavra e, usando a pilha, a imprima de forma invertida. (valor: 4,0 pontos)

Caso necessite de mais espaço, divida as linhas em duas ou três colunas.