

## Lista 04 – Estrutura de Dados - Pilha

E ai pessoal, como vão?

A lista 5 está um pouco extensa, porém tem vários exercícios fáceis de pilha. Em si, somente as 7 últimas questões vão dar um pouquinho mais de trabalho.

Leia 3 vezes cada exercício antes de fazer, se ainda assim não entender, me mande sua dúvida no chat e vá para o próximo exercício.

Todo esforço traz resultados, cedo ou tarde. Abraço pessoal!!

- 1) O que é e como funciona uma estrutura do tipo Pilha? Em que situações ela é utilizada.
- 2) O que significa alocação sequencial de memória para um conjunto de elementos?
- 3) O que significa alocação estática de memória para um conjunto de elementos?
- 4) O que significa alocação dinâmica de memória para um conjunto de elementos?
- 5) Implemente no método main(...), um programa que leia 20 números de um vetor (tendo valores positivos, negativos, pares e ímpares). Crie três pilhas (P1, P2 e P3), estáticas de tamanho 20 e proceda, para cada um deles, como segue:
  - se o número for par (seja positivo ou negativo), insira-o na pilha P1;
  - se o número lido for ímpar(seja positivo ou negativo), insira na pilha P2;
  - se tiver valor positivo na pilha P2, retirar da pilha P2 e inserir na pilha P3;
  - se tiver negativo na pilha P1, retirar da pilha P1 e inserir na pilha P3.
  - Ao final, imprima cada uma das pilhas P1, P2 e P3
- 6) Analise as afirmações a seguir a respeito de pilhas:
  - I - Novos elementos entram, no conjunto, exclusivamente, no topo da pilha.
  - II - O único elemento que pode sair da pilha em um dado momento, é o elemento do topo.
  - III - As Pilhas são conhecidas como LIFO (last in, first out), isto é, o último a entrar é o último a sair.
  - IV - As operações Push e Pop são respectivamente para desempilhar e empilhar

Estão corretas as afirmações:

- a. I e II.
- b. I e III.
- c. II e III.
- d. I, II e III.
- e. III e IV

Para os próximos dois exercícios, considere as seguintes rotinas como disponíveis.

Pilha\_Vazia(P): retorna True se a pilha P estiver vazia e False se estiver cheia;

Pilha\_Cheia(P): retorna True se a pilha P estiver cheia e False se estiver vazia;  
Empilha(P,'x'): insere o elemento x no topo da pilha P;  
Desempilha(P): remove o 1º elemento da pilha P e retorna o conteúdo como valor da função;  
Imprime(P) – imprime a pilha P.

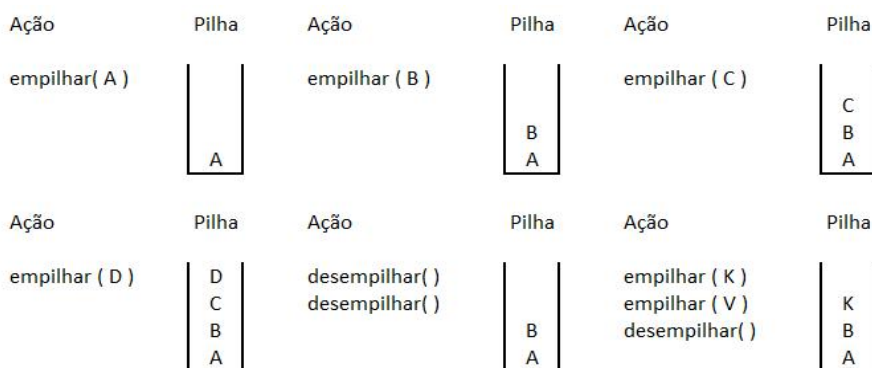
7) Desenhe a evolução da pilha (de tamanho máximo de 8 elementos) e mostre o que será impresso no vídeo, considerando a execução da sequência de instruções abaixo. Leia as instruções por linha e não por coluna.

Empilha(P,'b'), pilha\_cheia(P), Empilha(P,'s'); Empilha(P,'a'), Empilha(P,'c'),  
Desempilha(P), Desempilha(P), Empilha(P,'h'), Empilha(P,'d'), Empilha(P,'f');  
Desempilha(P), Empilha(P,'n'), Empilha(P,'b'), pilha\_cheia(P), Desempilha(P)  
Empilha(P,'m'), Desempilha(P), Desempilha(P), Desempilha(P), pilha\_vazia(P)  
Imprime(P)

8) Desenhe a evolução da pilha (de tamanho máximo de 6 elementos) e mostre o que será impresso na tela, considerando a execução da sequência de instruções abaixo. Leia as instruções por linha e não por coluna

Empilha(P,'b'), pilha\_cheia(P), Empilha(P,'s'); Empilha(P,'a'), Empilha(P,'c'),  
Empilha(P,'h'), Empilha(P,'d'), pilha\_cheia(P), Desempilha(P), Empilha(P,'n'),  
Empilha(P,'b'), pilha\_cheia(P), Desempilha(P), Empilha(P,'m'), Desempilha(P),  
Desempilha(P), Desempilha(P), Imprime(P), pilha\_vazia(P)

9) Observe o desenho abaixo(por linha e não por coluna) e transcreva para código as ações efetuadas. No final exiba a pilha.



10) Dado uma pilha que armazene números, escreva uma função que forneça o maior, o menor e a média aritmética dos elementos da pilha. Utilize somente as operações Push e Pop.

11. Desenvolva uma função para inverter a posição dos elementos de uma pilha P. Utilize somente as operações

Push e Pop.

12. Desenvolva uma função para testar se uma pilha P1 tem mais elementos que uma pilha P2. Utilize somente as operações Push e Pop.

13. Desenvolva uma função para testar se duas pilhas P1 e P2 são iguais no tamanho e nos elementos que possuem. Peça para o usuário popular as pilhas. Utilize somente as operações Push e Pop.

14. Faça um programa que inverte uma string usando recursão.

15. Escreva um programa que determina número usando recursão. Por exemplo: se o usuário inserir o número 250, a saída esperada deve ser : “o número digitado tem 3 dígitos”.

16. Faça uma rotina recursiva para calcular a somatória de todos os número de 1 a N (N será lido do teclado). Por exemplo: se N é igual a 5, a somatória dos números será igual a 15, ou seja,  $1+2+3+4+5 = 15$ .