

Lista de exercício - PILHA

1. Observe o trecho de código abaixo:

```
...  
int stack[10]; Inicia a pilha com tamanho 10.  
int item1 = 1;  
int item2 = 0;  
int item3 = 4;  
push(stack, item2); Empilha o valor = 0.  
push(stack, item1); Empilha o valor = 1.  
push(stack, item1 + item3); Empilha o valor = 5.  
item2 = peek(stack); Espia se há o item 2.  
push(stack, item3 * item3); Empilha o valor 16.  
push(stack, item2); Empilha o valor 0.  
push(stack, 3); Empilha o valor 3.  
item2 = peek(stack); Espia se há o item 2.  
pop(stack); Desempilha.  
  
while (!isEmpty(stack)) { Se a pilha não estiver vazia.  
    item1 = pop(stack, top); Desempilha do topo  
    cout << item1; Mostra o item desempilhado.  
}...
```

Stack é uma estrutura do tipo LIFO (última a entrar, primeira a sair) onde são aplicadas as operações usuais push() e pop(). Existe ainda a operação peek() que retorna o elemento no topo da pilha sem removê-lo. A seqüência de valores impressa pelo programa é:

- (A) 5, 16, 1
- (B) 4, 16, 4, 0, 1
- (C) 5, 8, 5, 1, 0
- ☒ (D) 5, 16, 5, 1, 0
- (E) 4, 0, 4, 4, 1

2.

```
...
Stack pilha;
...
push(pilha, 3);
push(pilha, 4);
pop(pilha);
push(pilha, 5);
push(pilha, 6);
pop(pilha);
push(pilha, 7);
push(pilha, 8);
while (true) {
    cout << pop(pilha ) << " ";
}...
```

A saída produzida pelo programa acima será:

- ☒ A) 8 7 5 3;
- ☐ B) 5 6 7 8;
- ☐ C) 3 4 5 6 7 8;
- ☐ D) 8 7 6 5 4 3;
- ☐ E) 8 7 6 5.

3. Dada a sequência de operações abaixo:

Push(POLICIAL_CIVIL, HARDWARE) *Empilha o valor = Hardware.*
Push(POLICIAL_CIVIL, SOFTWARE) *Empilha o valor = Software.*
POP(POLICIAL_CIVIL) *Desempilha o valor do topo.*
Push(POLICIAL_CIVIL, INTERNET) *Empilha o valor = Internet.*
Peek(POLICIAL_CIVIL) *Olha o valor do topo.*
Push(POLICIAL_CIVIL, SEGURANCA) *Empilha o valor = segurança.*
POP(POLICIAL_CIVIL) *Desempilha o valor do topo.*
Push(POLICIAL_CIVIL, Peek(POLICIAL_CIVIL)) *empilha o que está no topo*
Push(POLICIAL_CIVIL, POP(POLICIAL_CIVIL)) *empilha o ultimo valor desempilhado*
Push(POLICIAL_CIVIL, REDES) *empilha o valor redes*
POP(POLICIAL_CIVIL) *desempilha.*
Push(POLICIAL_CIVIL, Peek(POLICIAL_CIVIL)) *empilha o que esta no topo.*

Considere a estrutura de dados PILHA, inicialmente vazia, suportando três operações básicas. Após a execução da última operação Push(POLICIAL_CIVIL, PEEK(POLICIAL_CIVIL)), o elemento de topo da pilha será igual a:

- (A) REDES.
- (B) INTERNET.
- (C) SOFTWARE.
- ☒ (D) SEGURANCA.
- (E) CRIPTOGRAFIA.

4. Considere uma área de armazenamento de 5 nodos. Sobre esta área será montada uma pilha de nome PILHA. Inicialmente ela encontra-se vazia. Se possível preencha os gabaritos abaixo mostrando o comportamento da pilha durante as seguintes operações:

1. Inclusão de um nodo com valor VERMELHO ✓
2. Inclusão de um nodo com valor VERDE -
3. Inclusão de um nodo com valor AMARELO ✓
4. Inclusão de um nodo com valor BRANCO ✓
5. Inclusão de um nodo com valor PRETO ✓
6. Retirada de um nodo ✓
7. Retirada de um nodo
8. Inclusão de um nodo com valor ROSA
9. Retirada de um nodo
10. Retirada de um nodo
11. Retirada de um nodo
12. Inclusão de um nodo com valor CINZA
13. Retirada de um nodo
14. Retirada de um nodo
15. Retirada de um nodo

1

4	
3	
2	
1	
0	Vermelho

2

4	
3	
2	
1	verde
0	Vermelho

3

4	
3	
2	amarelo
1	verde
0	vermelho

4

4	
3	Branco
2	amarelo
1	verde
0	Vermelho

5

4	Preto
3	Branco
2	amarelo
1	verde
0	vermelho

6

4	
3	Branco
2	amarelo
1	verde
0	vermelho

7

4	
3	
2	
1	
0	

8

4	
3	
2	
1	
0	

9

4	
3	
2	
1	
0	

10

4	
3	
2	
1	
0	

11

4	
3	
2	
1	
0	

12

4	
3	
2	
1	
0	

13

4	
3	
2	
1	
0	

14

4	
3	
2	
1	
0	

15

4	
3	
2	
1	
0	