

Aluno(a):

1. Considere a seguinte sequência de operações:

`push(7), push(2), pop(), push(4), pop(), push(5),
push(8), pop(), pop(), push(3), push(1), pop()`.

- A soma dos elementos remanescentes em uma *fila*, inicialmente vazia, após essa sequência de operações será
- A soma dos elementos remanescentes em uma *pilha*, inicialmente vazia, após essa sequência de operações será

2. Seja *push(n)* a operação que insere um número *n* em uma *pilha*. Considere a execução desta operação, com *n* assumindo os valores 1, 2, 3, 4 e 5, nesta ordem. Em meio às inserções, pode haver remoções (*pop*). Indique a ordem das operações *push(n)* e *pop()* para que a sequência de números retirados da pilha com a operação *pop* seja:

(a) 3,4,2,5,1:

.....

(b) 1,4,5,3,2:

.....

3. Considere a seguinte sequência de operações aplicadas tanto a uma fila quanto a uma pilha: `push(5), push(7), pop(), push(1), pop(), push(6), pop()` e `push(4)`.

- I Após a realização das operações, a soma dos elementos remanescentes na pilha será menor do que a soma dos elementos remanescentes na fila.
- II Ao longo da execução das operações, a soma dos elementos armazenados na fila nunca será menor do que a soma dos elementos armazenados na pilha.
- III Ao longo da execução das operações, a soma dos elementos armazenados na pilha nunca será menor do que a soma dos elementos armazenados na fila.
- IV A diferença entre a soma dos elementos presentes na pilha e na fila nunca ultrapassa 5.

A respeito das afirmações acima, pode-se dizer que estão corretas as alternativas:

- (a) I e II (c) I e IV (e) I, II, III e IV
(b) II e IV (d) II e III

4. Considere as seguintes estruturas utilizadas para implementar uma fila:

```
typedef struct{                typedef struct{
    int dado;                    item *inicio;
    struct item *next;          item *fim;
}item;                          }fila;
```

Complete o código abaixo para implementar a função *pop* em uma fila, assumindo que o parâmetro **f* é um ponteiro para o primeiro elemento da estrutura.

```
1 int pop(fila *f){
2     int dado = f->inicio->dado;
3     _____;
4     if(f->inicio==NULL)
5         _____;
6     return dado;
7 }
```

Linha 3:

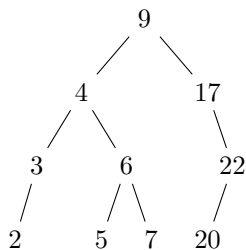
Linha 5:

5. Nas alternativas abaixo, assinale V para as verdadeiras e F para as falsas

- () Em tabelas hash livres de colisões, tem-se a garantia, independente da função hash utilizada, de que os elementos estarão dispostos na ordem em que foram inseridos. O primeiro elemento inserido ocupará a primeira posição da tabela, o segundo elemento ocupará a segunda posição e assim por diante.
- () Seja uma tabela de tamanho *m* populada com *n* elementos ($n < m$). Se colisões forem resolvidas por sondagem linear, então a inserção de um elemento requer, no mínimo, *n* comparações até encontrar a posição adequada.
- () Em tabelas livres de colisões, a busca por um elemento tem, no pior caso, complexidade igual à complexidade da busca em árvores binárias de pesquisa.
- () Na resolução de colisões por encadeamento, a busca por um elemento, no pior caso, terá complexidade semelhante à busca em listas encadeadas

6. Considere a inserção das seguintes chaves na dada ordem em uma tabela hash de tamanho 7: **{282, 44, 251, 22, 129, 266, 300}** usando o método da divisão com a função hash $h(x) = x \bmod 7$. Mostre as tabelas resultantes resolvendo as colisões por encadeamento e por sondagem linear. Responda no seguinte formato: 0:[...], 1:[...], ..., n:[...]
-
-
-
-
-
-
-
-
-
-
9. Desenhe a árvore binária de pesquisa (não balanceada) gerada pela inserção da seguinte sequência de números: 50, 30, 80, 90, 60, 95, 55, 85, 65, 87, 70, 82, 61, 89.
10. Seja uma árvore *completa*. São necessárias c comparações para constatar que a informação procurada não está armazenada nesta árvore. Informe a quantidade de nós desta árvore para os valores de c abaixo:
- (a) $c = 6$: -----
- (b) $c = 8$: -----
- (c) $c = 12$: -----

Para as questões 7 a 8, considere a árvore da figura abaixo:



7. Escreva abaixo a sequência de números que será impressa caso a árvore seja percorrida
- a) Em pré-ordem: -----
- b) Em ordem: -----
- c) Em pós-ordem: -----
8. Informe qual será a *soma* do valor das folhas da árvore após a exclusão dos nós indicados abaixo:
- 4: -----
 - 17: -----
 - 7: -----