

## Exercícios de Revisão para a Prova 1 de Alg. II

- 1) Uma seqüência de operações I, I, I, I, R, R, I, R, I, R, R, I, R, R, I, I, R, I, R, R produziu a saída E, S, T, A, C, I, O, N, O, N. Qual foi a ordem de entrada das letras?
- 2) Considerando a seqüência de entrada E, S, T, R, A, N, H, O, qual é a saída, se possível para cada seqüência de operações:
  - a) I, I, I, R, I, R, R, I, R, R, I, R,
  - b) I, I, R, R, I, R, R, I, I, R, I, I, I, R, R, R
  - c) I, I, R, R, I, R, I, I, R, I, I, I, R, R, R, R
- 3) Qual seqüência de operações permite:
  - a) Saída S, E, G, U, R, A, N, C, A para entrada S, R, E, U, C, G, A, N, A
  - b) Saída S, E, G, U, R, A, N, C, A para entrada Ç, S, R, U, G, E, A, N, A
- 4) Transforme de infixa para posfixa:
  - a)  $X / Y ^ C + D * E - A * C + D * I$
  - b)  $(A + B * (2 - 3 / (A + 2 * D) + 5)) > B \text{ or } (B + C > A)$
- 5) Analisando o código da rotina recursiva de busca binária abaixo:

```
int BS(int vet[], int low, int high, int valor){
    If (high < low) {
        return -1;
    }
    int mid = low + ((high - low) / 2);

    If (vet[mid] > valor) {
        return BS(vet, low, mid-1, valor);
    }

    If (vet[mid] < valor) {
        return BS(vet, mid+1, high, valor);
    }

    return mid;
}
```

E Sabendo que:

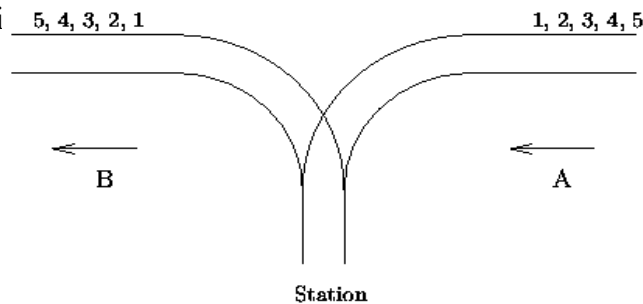
vet[10]	-94	-56	-28	0	12	17	35	41	44	55
=	0	1	2	3	4	5	6	7	8	9

low = 0 high = 9

Quais os elementos pesquisados ao procurar o número 42? Sim, o 42 não existe no vetor.

>>> BS (vet, 0, 9, 42);

- 6) Dada uma estação de trem, onde os vagões entram em um estacionamento que não oferece possibilidade de manobra, é possível que os vagões que chegam em (A), sempre na formação em ordem crescente sai



- a) 5, 4, 3, 2, 1
- b) 5, 2, 3, 4, 1
- c) 7, 4, 5, 2, 3, 6, 1
- d) 3, 5, 6, 4, 2, 1
- e) 3, 5, 4, 6, 2, 1, 7
- f) 2, 3, 5, 4, 6, 1, 7
- g) 2, 3, 5, 4, 6, 1, 9, 8, 10, 7

- a) Resposta: Yes
- b) Resposta: No
- c) Resposta:
- d) Resposta:
- e) Resposta:
- f) Resposta:
- g) Resposta:

- 7) Faça um teste de mesa sobre o seguinte programa, apresentando o resultado na tela

```
#include <iostream>
#include <stack>

using namespace std;

stack <int> pilha;

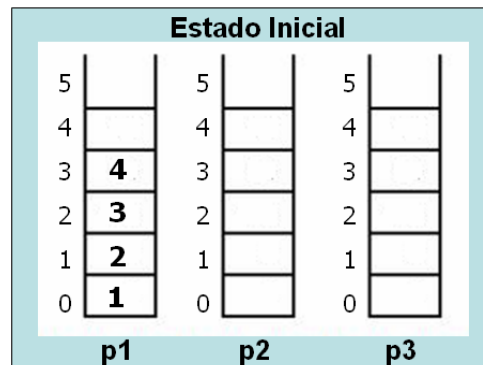
int rec (int n) {
    if (n <= 1 ) {
        return 1;
    }
    pilha.push(n);
    return rec (n-2) * n;
}

int main() {
    int i,n=5,total;
    total = rec(n);
    cout << total << endl;
    while ( pilha.size() >0) {
        total = total * pilha.top();
        pilha.pop();
    }
    cout << total << endl;
    return 0;
}
```

```
void main() {
    int x;
    cin >> x;
    cout << fatorial(x) << endl;
    return 0;
}
```

Pilha:

10) Dado o estado inicial das pilhas p1, p2 e p3 na figura abaixo, mostre (desenhe as pilhas) o estado final dessas mesmas pilhas após as operações descritas no código abaixo. Considere que p1, p2 e p3 sejam instâncias da classe **Stack (pilha com alocação seqüencial)** Caso não seja possível realizar alguma operação, escreva que não foi possível e ignore-a:



```
int temp = p1.top; p1.pop();
p2.push(temp);
p3.push(p1.top()); p1.pop()
p2.push(p1.top()); p1.pop()
temp = p1.top(); p1.pop()
p3.push(temp);
p1.push(p2.top()); p2.pop();
p3.push(p2.top()); p2.pop();
p3.push(p1.top()); p1.pop();
```

11) Uma pilha implementa o mecanismo de inserção/remoção:

- a) FIFO
- b) FIFA
- c) LIFO
- d) FFLL
- e) N.D.A.