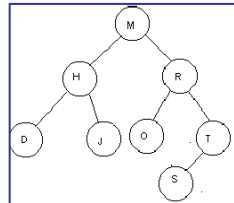


## Lista PPPPP2 – Exercícios de revisão para a prova P2

- Recursividade,
- Árvore Binária - conceitos
- Árvore Binária de Busca – Conceitos (inserção, remoção, busca, máx., min),
- Lista Ordenada
- Tabela Hashing - conceitos

1) Dada a árvore, indique:

- Os nós folha
- Grau da árvore
- Altura da árvore
- Os filhos de R



2) Dada uma árvore de Busca Binária (BST) inicialmente vazia, insira e desenhe os seguintes elementos e nessa ordem: 39, 18, 57, 4, 14, 30, 48, 7, 3, 15, 24, 16. Após, escreva os percursos;

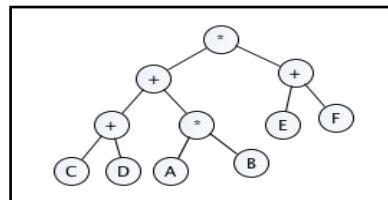
- Pré-ordem
- In-ordem
- Pós-ordem

3) Considerando as expressões aritméticas:

- $3 / 5 + 2 + (3 * 5)$
- $2 * 7 + 2 / 1 * 5 * 1 / 2$
- $A / B / C / D + E / F$
- $2 + 5 * 4 - 2 * 2 + 2$

Mostre as árvores binárias que as representam e as notações correspondentes (prefixa e posfixa) em cada caso.

4) Considerando a árvore binária, mostrada na figura abaixo, escreva os percursos na pré-ordem, em ordem e pós-ordem, respectivamente.



5) Considerando o modelo matemático, abaixo, construa a função C++ correspondente e a pilha de recursão para calcular o valor de  $mt(2,5)$ .

$$mt(a, b) = \begin{cases} a, & \text{se } b = 1 \\ mt(a, b - 1) + 1, & \text{se } b > 1 \end{cases}$$

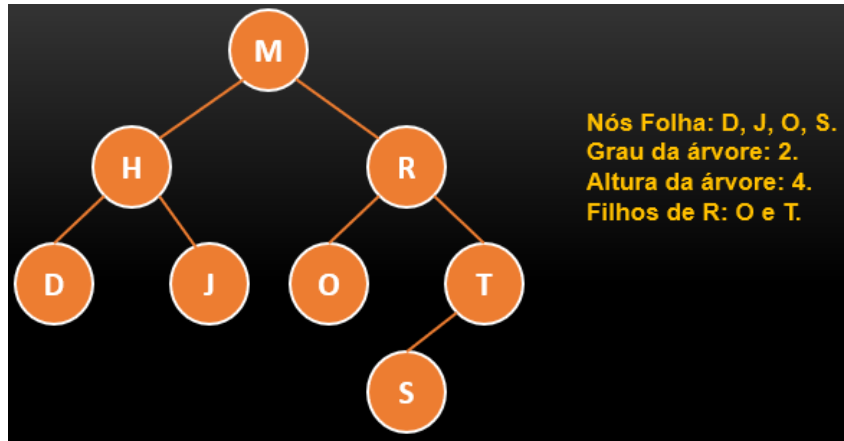
6) Considerando os números inteiros, abaixo, construa a árvore AVL, inicialmente vazia, nos seguintes casos:

a) 1, 5, 7, 8, 15, 18, 20, 23, 32, 45 remove(8), remove(23)

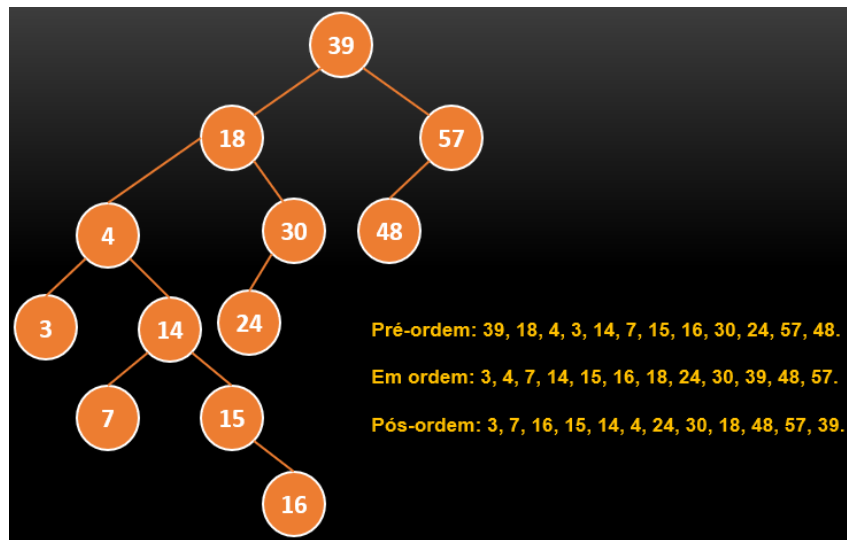
b) 3, 5, 19, 2, 1, 27, 10, 22, 35, 25, 26, 24, 50, 32, remove (19); remove(10), remove(32)

7) Insira em uma Hash Table de largura 3, os seguintes números inteiros: 10, 2, 15, 5, 9, 19, 6, 11, 14, 17, 13, 24, 1, 4, 20, 18, 7, 12, 21, 22 e verifique se a função de espalhamento  $(x \% 3)$  é ótima.

1)

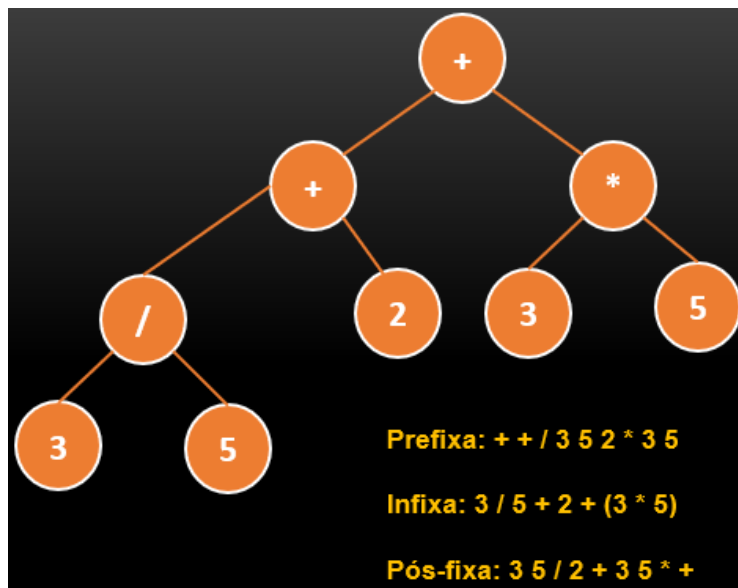


2)

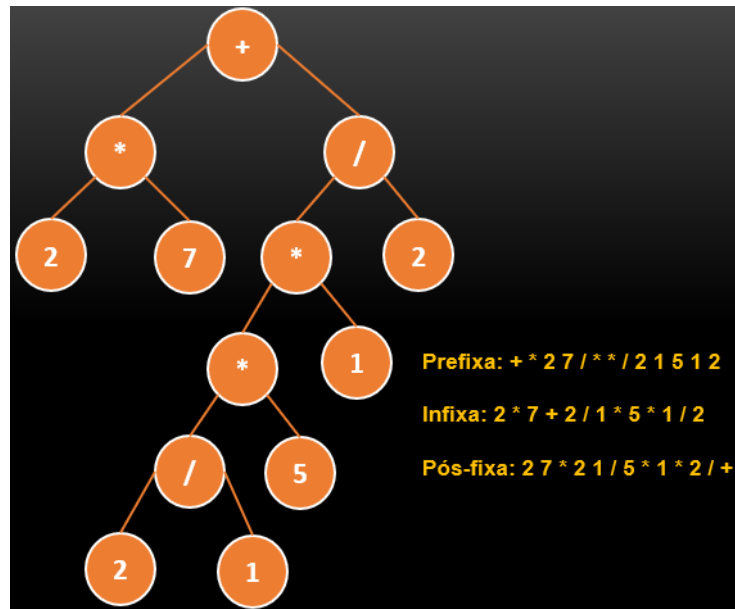


3)

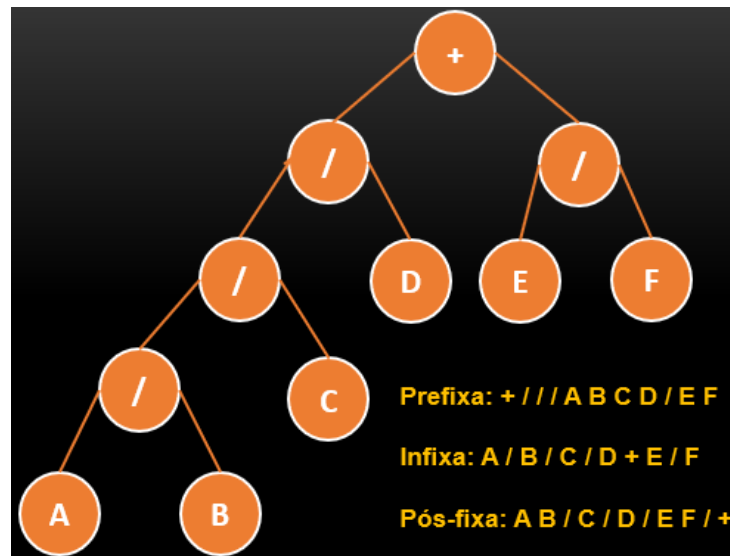
a)



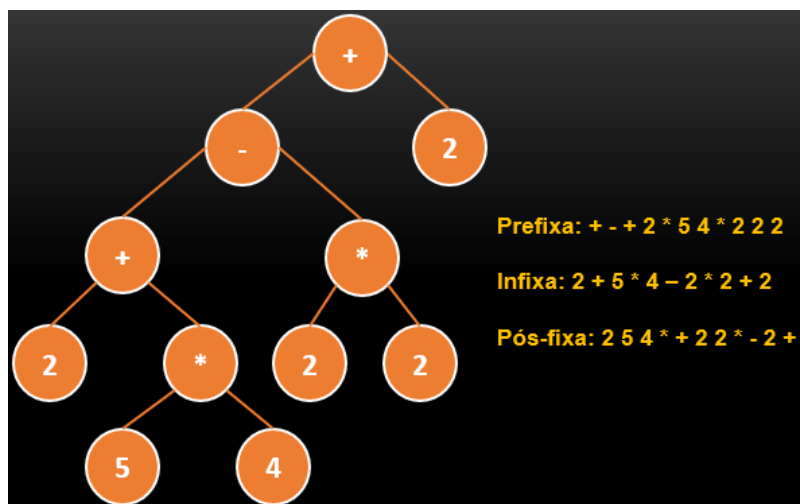
b)



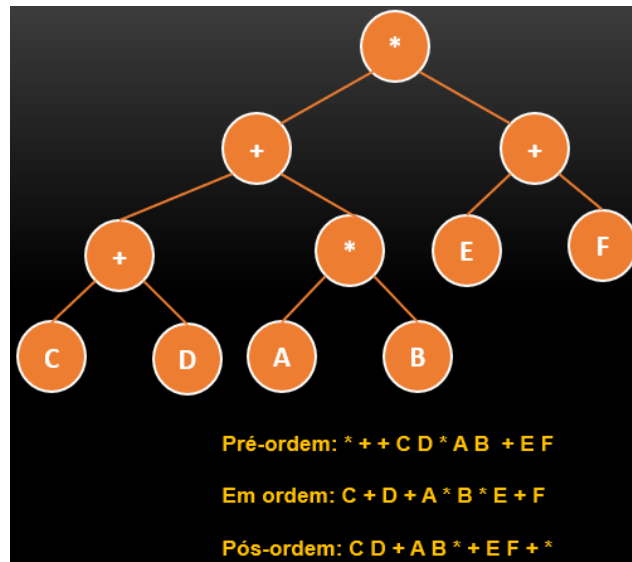
c)



d)

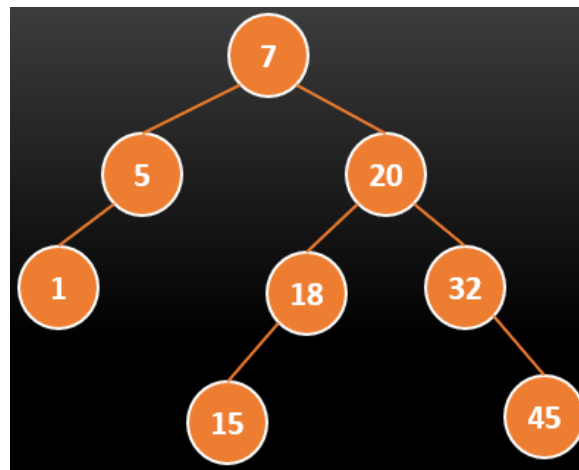


4)

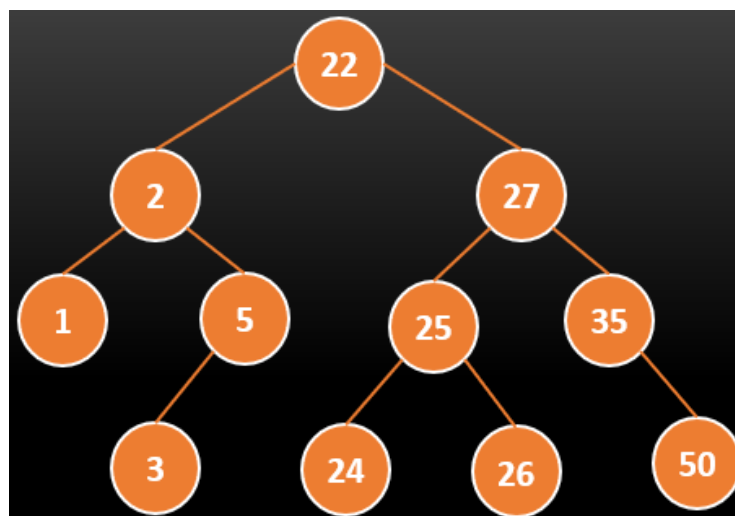


6)

a)



b)



7)

## h[0]—>6, 9, 12, 15, 18, 21, 24

## h[1] —> 1, 4, 7, 10, 13, 19, 22

## h[2] —> 2, 5, 11, 14, 17, 20

O espalhamento ( $x\%3$ ) é ótimo para esta lista.

5)

$$\mathbf{Mt}(2,5) = \mathbf{Mt}(2,4)+1 = (\mathbf{Mt}(2,3)+1)+1 = ((\mathbf{Mt}(2,2)+1)+1)+1 = (((\mathbf{Mt}(2,1)+1)+1)+1)+1 = \mathbf{Mt}(2,1) + 4 = 2 + 4 = 6$$

```
#ifndef mt
```

```
#define mt
```

```
int MT(int a, int b)
```

```
{
  if(b == 1)
  {
    return a;
  }
  else if(b > 1)
  {
    return (MT(a,(b-1)))+1;
  }
}
```

```
#endif // mt
```

```
#include <iostream>
```

```
#include "mt.h"
```

```
using namespace std;
```

```
int main()
```

```
{
    cout << "\n\n\t\t>>>>> MT(2.5) <<<<<<" << endl;
```

```
cout << "\n\n\t\t MT(2,5)= " << MT(2,5) << endl;
```

```
cout << "\n\n\t\t";
```

```
return 0;
```

}