

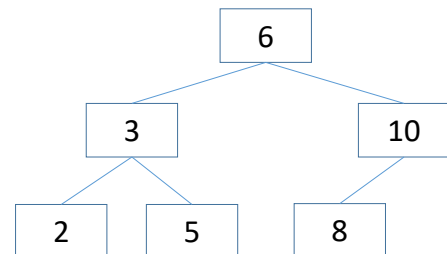
1–[15%] Considere a seguinte função:

```
int f(int array[]){
    int n=array.length;
    int S=0;
    while(S+1<n){
        for(int i=S+1;i<n;i++)
            if(array[i]>array[S])
                { int tmp=array[i];
                  array[i]=array[S];
                  array[S]=tmp;
                }
        S++;
    }
}
```

Indique, justificando, qual é a ordem de complexidade do método no pior caso, em função de **n**.

2 – [15%] Construa a árvore de *Huffman* para a string “PIPOCAPICANTE”, e reescreva a string usando a codificação correspondente.

3 – [20%] Considere a seguinte Splay Tree apresentada na figura.



- Indique qual o último valor que foi introduzido ou acedido na árvore.
- Indique qual é a árvore resultante após introdução do número “12”, explicando devidamente os vários passos.

4- [20%] Indique, justificando, qual o resultado da inserção da sequência {5,2,6,4} numa árvore *red-black* vazia (apresente e justifique os vários passos intermédios. (identifique em cada nodo a sua cor usando as letras “R” e “B”).

5 – [15%] Indique qual o resultado da conversão da expressão  $1*(2+3)*4$  para notação *postfix*. Apresente os vários passos intermédios. Assuma que a multiplicação é associativa à esquerda, ou seja  $a*b*c=(a*b)*c$

6 – [15%] Considere uma *hash table* que utiliza a função de *hash*  $H(X)=2*X-1$ . Indique dois valores para X que gerem colisões, sabendo que a dimensão da tabela é 50.