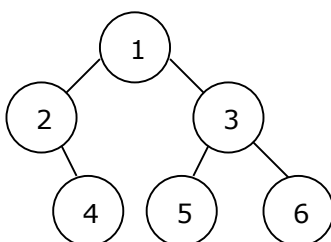


Árvores Binárias



- 1) Demonstre que uma árvore binária de altura **h** tem, no mínimo, **$h+1$** nós, e, no máximo, **$2h+1 - 1$** .
- 2) Usando as operações básicas de uma árvore binária, escreva um programa que represente a seguinte árvore:



- 3) Simule a chamada da função **preordem()** aplicada à árvore ilustrada pela questão anterior para verificar que o resultado da chamada é a impressão dos valores 1-2-4-3-5-6. Em outras palavras, acompanhe os passos (mostrando a situação da pilha e a operação executada) para todas as chamadas recursivas até o término da operação de impressão.
- 4) Adicione à sua biblioteca de árvores binárias, um método que percorra a árvore binária e determine sua altura. O protótipo da operação é definido por:

```
def altura(self)
```

- 5) Adicione à sua biblioteca de árvores binárias um método que retorne a quantidade de nós existentes na árvore. O protótipo da função pode ser dado por:

```
def count(self):
```

- 6) Adicione à sua biblioteca de árvores binárias um método que determine a quantidade de nós folhas na árvore. O protótipo da função pode ser dado por:

```
def leafs(self):
```

- 7) Adicione à sua biblioteca de árvores binárias um método que determine qual o nível em que uma chave **key** de busca é encontrada na árvore. Se não for encontrada, devolve 0 (zero). O protótipo da função pode ser dado por:

```
def getLevel(self, key):
```

- 8) Uma operação bastante útil a ser adicionada à sua biblioteca é aquela que remove os nós pertencentes a uma determinada subárvore de um relativo nó. Considerando a árvore ilustrada na questão 3, caso seja determinada a exclusão da subárvore a partir da chave "3", este nó e seus descendentes ("5" e "6") serão removidos. Observe o código da operação a seguir

```
def libera( node ):  
    if ( not vazia( node )):  
        libera(node.esq)  
        libera(node.dir)  
    return None
```

Uso da função:

```
arv = BinaryTree()  
... # posiciona o cursor no nó 3  
arv.preordem(arv)  
cursor = arv.getCursor()  
arv.libera(cursor);  
arv.preordem()
```

Perguntas:

- (a) O que será exibido após a execução da última instrução **arv.preordem()**?
 - (b) O código da operação **libera()** está fazendo o que se propõe? Caso contrário, faça os ajustes necessários para que seja possível remover a subárvore determinada por uma determinada chave.
- 9) Modifique o código que você implementou em atendimento à questão 3 para se comportar da seguinte forma:
- (a) Exiba a árvore após sua criação;
 - (b) Localize o nó com conteúdo igual a "3";
 - (c) Remova a subárvore correspondente ao elemento encontrado;
 - (d) Imprima o resultado após a remoção ("poda") da sub-árvore direita (sub-árvore cujo elemento era a informação "3").