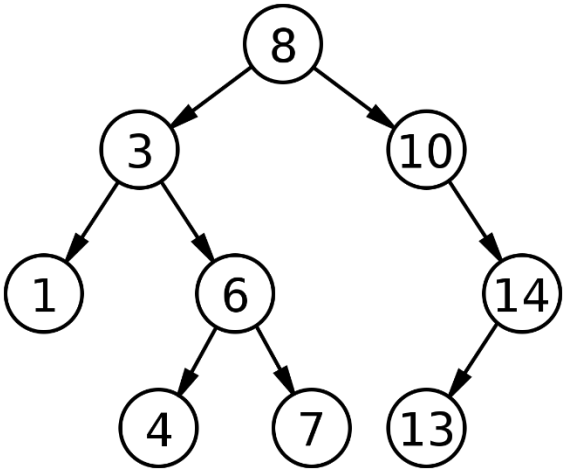
Practice problems - Tree

1.



(a) Write a function ***sumAtLevel*** that accept an integer (level) and calculates the sum of all the nodes (values) in a given level.

Function Signature: ***int sumOfLevel(level);***

For example: In the given BST, ***sumOfLevel(2);*** will return 21. Because, the sum of the values of the nodes at level 2 is = 1 + 6 + 14 = 21.

(b) Write a function ***minDepth*** that the minimum depth of that tree.

Function Signature: ***int minDepth();***

For example: In the given BST, ***minDepth()*** will return 2.

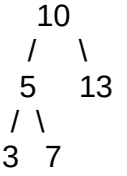
2. Write a function that will print the level of a binary search tree. (Hint use a queue)

```
      10
     /   \
    5    13
   / \     \
  3   7    14
```

The output of the function will be:

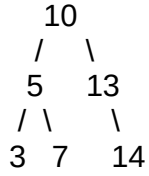
10
5 13
3 7 14

3. Write a function that will print the largest element in each level of a binary search tree.



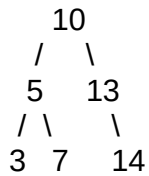
The output of the given tree when the function is applied:
10
13
7

4. Write a function to find the median value of a given binary search tree.



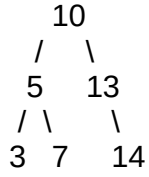
The median value of the above binary tree is 8.5

5. Check whether the leaf nodes of a BST is even or odd.



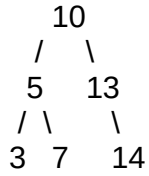
The output of the BST above is [odd, odd, even].

6. Find the predecessor and successor of 10 in the following BST.



The predecessor of 10 is 7 and the successor is 13.

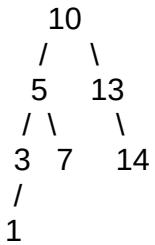
7. Write a function code to print the leaf nodes of a BST.



The output of the function is [3,7,14].

8. Write a code to find height and depth of a particular node of BST.

9. Traverse a BST and show in the number of Descendants of each node.



For example, the descendants of node 5 is 3, 7 and 1. So the number of Descendants of the node 5 is 3.