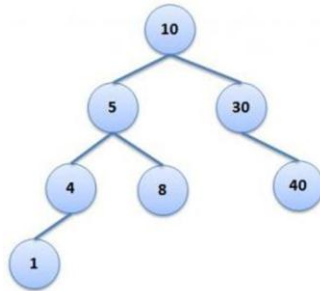


NUMBER OF NODES IN A BINARY TREE



Write a function which will return the total number of nodes in a Binary tree. For example, the Binary tree on the right has 7 nodes.

Solution:

Binary Tree problems can be easily solved using recursion.

To calculate the total number of nodes we can traverse the entire tree and increment the counter for each Node. Just add a static variable count in the traversal and return it.

The below code does exactly that (just that we are not using a static variable, rather we are returning the size).

1 **Algorithm:**

If tree is empty

return 0

Else

return 1 + Number_of_Nodes_In_Left_subtree + Number_of_Nodes_In_Right_subtree

Code:

```
1  /** Returns number of nodes in Binary Tree.
2  *
3  * root: pointer to root node of tree
4  */
5  unsigned int countNodes(Node* root)
6  {
7      if (NULL == root)
8          return 0;
9      else
10         return ( 1 + countNodes(root->lptr) + countNodes(root->rptr) );
11 }
```

The structure of Node of the Binary Tree is

```
1  struct Node
2  {
3      int data;
4      Node* lptr; // ptr to Left subtree
5      Node* rptr; // ptr to Right subtree
6  };
```

Source: <http://www.ritambhara.in/number-of-nodes-in-a-binary-tree/>