**[Diameter of Binary Tree](https://leetcode.com/problems/diameter-of-binary-tree/)**

**public** **class** DiameterOfTree {

**public** **static** **int** *diameter* = 0;

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

TreeNode root = **new** TreeNode(10);

root.left = **new** TreeNode(5);

root.left.left = **new** TreeNode(3);

root.left.right = **new** TreeNode(7);

root.right = **new** TreeNode(15);

root.right.right = **new** TreeNode(18);

System.***out***.println(*diameterOfBinaryTree*(root));

}

**public** **static** **int** diameterOfBinaryTree(TreeNode root) {

**if**(root == **null**) {

**return** 0;

}

*helper*(root);

**return** *diameter*;

}

**public** **static** **int** helper(TreeNode root) {

**if**(root == **null**) {

**return** 0;

}

**int** left = *helper*(root.left);

**int** right = *helper*(root.right);

*diameter* = Math.*max*(*diameter*, left + right);

**return** 1 + Math.*max*(left, right);

}

}

Time Complexity : O(n) , n is no of nodes in Tree

Space Complexity : O(1), constant space