## 222. Count Complete Tree Nodes (need to review)

link

优化可以达到(logn) ^2, 可以根据右子树的高度来判断右边是不是完全二叉树.

```
public int getCount(TreeNode root){
   if(root == null) return 0;
   int leftHeight = getHeight(root.left);
   int rightHeight = getHeight(root.right);
   if(rightHeight == leftHeight){
      return (int)Math.pow(2, leftHeight) + getCount(root.right);
   }
   return (int)Math.pow(2, rightHeight) + getCount(root.left);
}
```

## bfs

```
public int helper(TreeNode root){
    if(root == null) return 0;
    Queue<TreeNode> queue = new LinkedList();
    queue.offer(root);
    int res = 0;
    while(!queue.isEmpty()){
        int size = queue.size();
        res += size;
        for(int i = 0; i < size; ++i){</pre>
            TreeNode node = queue.poll();
            if(node.left != null)
                queue.offer(node.left);
            if(node.right != null)
                queue.offer(node.right);
        }
    return res;
}
```

## dfs

```
public int getCount(TreeNode root){
```

```
if(root == null) return 0;
if(root.left == null && root.right == null)
    return 1;
int left = getCount(root.left);
int right = getCount(root.right);
return left + right + 1;
}
```