513. Find Bottom Left Tree Value

Description SubmissionsSolutions

- Total Accepted: **9944**
- Total Submissions: **17871**
- Difficulty: **Medium**
- Contributors: **abhijeet17**

Given a binary tree, find the leftmost value in the last row of the tree.

Example 1:

```
Input:


    2
   / \
  1   3


Output:
1
```

Example 2:

```
Input:


       1
      / \
     2   3
    /   / \
   4   5   6
      /
     7
```

```
Output:

7
```

Note: You may assume the tree (i.e., the given root node) is not **NULL**.

Subscribe to see which companies asked this question.

```
/**
 * Definition for a binary tree node.
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode(int x) : val(x), left(NULL), right(NULL) {}
 * };
 */
class Solution {
public:
    int findBottomLeftValue(TreeNode* root) {
        bool cont = true;
        queue<TreeNode*> q;
        int ret;
        int sz = 1;
        q.push(root);
        while(q.size() && cont)
        {
            ret = q.front()->val;
            cont = false;
            while(sz--)
            {
                TreeNode *n = q.front();
                q.pop();
                if((n->left || n->right) && !cont)
                    cont = true;
                if(n->left)
                    q.push(n->left);
                if(n->right)
                    q.push(n->right);
            }
            sz = q.size();
        }
```

```
        return ret;
    }
};
```