Problem Link:

<https://leetcode.com/problems/minimum-depth-of-binary-tree/>

Solution:

/\*\*

\* Definition for a binary tree node.

\* struct TreeNode {

\* int val;

\* TreeNode \*left;

\* TreeNode \*right;

\* TreeNode() : val(0), left(nullptr), right(nullptr) {}

\* TreeNode(int x) : val(x), left(nullptr), right(nullptr) {}

\* TreeNode(int x, TreeNode \*left, TreeNode \*right) : val(x), left(left), right(right) {}

\* };

\*/

class Solution {

public:

int minDepth(TreeNode\* root) {

if(root == nullptr)

{

return 0;

}

queue<pair<TreeNode\*, int>> q;

q.push({root, 1});

while(!q.empty())

{

auto [node, depth] = q.front();

q.pop();

if(node->left == nullptr && node->right == nullptr)

{

return depth;

}

if(node->left)

{

q.push({node->left, depth + 1});

}

if(node->right)

{

q.push({node->right, depth + 1});

}

}

return 0;

}

};