**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:

Solution(){

ios::sync\_with\_stdio(false);

std::cin.tie(nullptr);

std::cout.tie(nullptr);

}

int dfs(TreeNode\* root, int& ans){

if(!root)

return 0;

int l=dfs(root->left,ans);

int r=dfs(root->right,ans);

l=l>0?l:0;

r=r>0?r:0;

ans=max(ans,l+r+root->val);

return max(l,r)+root->val;

}

int maxPathSum(TreeNode\* root) {

int ans=INT\_MIN;

dfs(root,ans);

return ans;

}

};

**TIME COMPLEXITY: O(N)**

**SPACE COMPLEXITY: O(1)**