**[Sum of Left Leaves](https://leetcode.com/problems/sum-of-left-leaves/)**

Given the root of a binary tree, return *the sum of all left leaves.*

A **leaf** is a node with no children. A **left leaf** is a leaf that is the left child of another node.

**Example 1:**



**Input:** root = [3,9,20,null,null,15,7]

**Output:** 24

**Explanation:** There are two left leaves in the binary tree, with values 9 and 15 respectively.

**Example 2:**

**Input:** root = [1]

**Output:** 0

**Constraints:**

* The number of nodes in the tree is in the range [1, 1000].
* -1000 <= Node.val <= 1000

class Solution {

public:

int sumOfLeftLeaves(TreeNode\* root) {

if (!root) {

return 0;

}

queue<pair<TreeNode\*, bool>> q; // (node, is\_left)

q.push({root, false});

int totalSum = 0;

while (!q.empty()) {

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

q.pop();

if (isLeft && !node->left && !node->right) {

totalSum += node->val;

}

if (node->left) {

q.push({node->left, true});

}

if (node->right) {

q.push({node->right, false});

}

}

return totalSum;

}

};

Link : <https://leetcode.com/problems/sum-of-left-leaves/?envType=daily-question&envId=2024-04-14>