Given a binary tree, return the *inorder* traversal of its nodes' values.

For example:  
Given binary tree [1,null,2,3],

1

\

2

/

3

return [1,3,2].

分析：中序遍历二叉树

/\*\*

\* 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:

vector<int> inorderTraversal(TreeNode\* root) {

std::vector<int> result;

if (root == NULL) {

return result;

}

std::vector<int> left\_vec = inorderTraversal(root->left);

for (int i = 0; i < left\_vec.size(); ++i) {

result.push\_back(left\_vec[i]);

}

result.push\_back(root->val);

std::vector<int> right\_vec = inorderTraversal(root->right);

for (int i = 0; i < right\_vec.size(); ++i) {

result.push\_back(right\_vec[i]);

}

return result;

}

};