这个代码里面的创建树这个函数还可以，



最好不要用malloc那是c语言里面的东西

Tree的广度优先，要用到queue队列，详见102题

#include <stdio.h>

#include <iostream>

#include <math.h>

#include <algorithm>

using namespace std;

struct TreeNode {

int val;

TreeNode \*left;

TreeNode \*right;

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

};

TreeNode\* CreateTree(TreeNode \*root);

void PrintTree(TreeNode \*root);

int Balance(TreeNode \*root);

void main() {

// TreeNode \*root = (TreeNode\*)malloc(sizeof(struct TreeNode));

TreeNode \*root = NULL;//由于CreateTree需要root作为参数，那么root必须被初始化。重要！！！！

root = CreateTree(root);

//PrintTree(root);

int result = Balance(root);

cout << endl;

if (result == -1)

cout << -1;

}

int Balance(TreeNode \*root) {

if (root == NULL)

return 0;

int ld = Balance(root->left);

int rd = Balance(root->right);

if (ld == -1 || rd == -1 || abs(ld - rd) > 1) {

return -1;

}

return max(ld,rd) + 1;

}

TreeNode\* CreateTree(TreeNode \*root){

int temp;

cin >> temp;

if (temp == 0) {

root = NULL;

return root;

}

else{

//root = (TreeNode\*)malloc(sizeof(struct TreeNode));

root = new TreeNode(-1);

root->val = temp;

root->left = CreateTree(root->left);

root->right = CreateTree(root->right);

}

return root;

}

void PrintTree(TreeNode \*root) {

if(root){

cout << root->val;

PrintTree(root->left);

PrintTree(root->right);

}

}