## 题目描述

## 请实现一个函数，用来判断一颗二叉树是不是对称的。注意，如果一个二叉树同此二叉树的镜像是同样的，定义其为对称的

## 解题思路

\*思路：首先根节点以及其左右子树，左子树的左子树和右子树的右子树相同

\* 左子树的右子树和右子树的左子树相同即可，采用递归

\* 非递归也可，采用栈或队列存取各级子树根节点

算法描述：

class Solution {

public:

bool isSymmetrical(TreeNode\* pRoot)

{

if (pRoot == NULL)

return true;

return isSymmetrical(pRoot->left,pRoot->right);

}

bool isSymmetrical(TreeNode\* t1, TreeNode\* t2)

{

if(t1 == NULL && t2 == NULL)

return true;

if(t1 == NULL || t2 == NULL)

return false;

if(t1->val != t2->val)

return false;

return isSymmetrical(t1->left,t2->right) && isSymmetrical(t1->right, t2->left);

}

};