# 对称的二叉树

## 题目

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

1. 定义一个对称的前序遍历，即root -> right -> left 与普通的前序遍历进行对比
2. 相同则认为树是对称的

boolean isSymmetrical(TreeNode pRoot) {
 LinkedList<Integer> scanner = new LinkedList<>();
 LinkedList<Integer> symmetricalScanner = new LinkedList<>();

 preScanner(scanner, pRoot);
 symmetricalPreScanner(symmetricalScanner, pRoot);

 return scanner.equals(symmetricalScanner);
}

/\*\*
 \* 普通的前序遍历
 \* @param res
 \* @param root
 \*/
private void preScanner(LinkedList<Integer> res, TreeNode root) {
 if (root == null) {
 res.addLast(null);
 return;
 }

 res.addLast(root.val);

 preScanner(res, root.left);
 preScanner(res, root.right);
}

/\*\*
 \* 先右再左的前序遍历
 \* @param res
 \* @param root
 \*/
private void symmetricalPreScanner(LinkedList<Integer> res, TreeNode root) {
 if (root == null) {
 res.addLast(null);
 return;
 }

 res.addLast(root.val);

 symmetricalPreScanner(res, root.right);
 symmetricalPreScanner(res, root.left);
}