

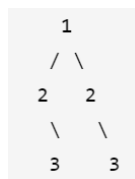
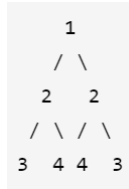
101 对称二叉树

Label: 二叉树、递归

给定一个二叉树，检查它是否是镜像对称的。

例如，二叉树 [1,2,2,3,4,4,3] 是对称的。

但是下面这个 [1,2,2,null,3,null,3] 则不是镜像对称的：



- 深度优先（递归）

```
class Solution {
    public boolean isSymmetric(TreeNode root) {
        if (root == null) {
            return false;
        }
        return symmetric(root.left, root.right);
    }

    private boolean symmetric(TreeNode p, TreeNode q) {
        if (p == null && q == null) { // 两个都为 null
            return true;
        }

        if (p == null || q == null) { // 有一个为 null
            return false;
        }

        if (p.val == q.val) { // 两个都不为 null
            return symmetric(p.left, q.right) && symmetric(p.right, q.left);
        } else {
            return false;
        }
    }
}
```

- 广度优先

```
class Solution {
    public boolean isSymmetric(TreeNode root) {
        if (root == null)
            return false;

        Queue<TreeNode> q = new LinkedList<TreeNode>();
        q.offer(root);
        q.offer(root);
        while (!q.isEmpty()) { // 这三步是广度优先的套路步骤
            // 同时弹出两个
            TreeNode u = q.poll();
            TreeNode v = q.poll();

            if (u == null && v == null) {
                continue;
            }
            if ((u == null || v == null) || (u.val != v.val)) { // 有一个为 null
                return false;
            }

            // 同时存入 两对
            q.offer(u.left);
            q.offer(v.right);

            q.offer(u.right);
            q.offer(v.left);
        }
        return true;
    }
}
```

或者 值不等