Given a binary tree, check whether it is a mirror of itself (ie, symmetric around its center).

For example, this binary tree is symmetric:

1

/ \

2 2

/ \ / \

3 4 4 3

But the following is not:

1

/ \

2 2

\ \

3 3

**Note:**  
Bonus points if you could solve it both recursively and iteratively.

递归：

深度优先遍历，如果两棵子树都是空则返回True

如果左子树和右子树都不为空，是否满足条件：

1. left.val == right.val
2. 递归(left.left,right.right)、递归（left.right,right.left）

如果都是True，则返回True

如果此时还在执行，则说明一个为空一个不为空，返回False