**Problem Title:** Symmetric Tree

**Problem Description:** Oliver is working on a symmetrical design for a project and needs to check if a given binary tree is symmetric around its center. Given the root of a binary tree, check whether it is a mirror of itself (i.e., symmetric around its center).

**Input Format:**

* The first line contains the values of all the nodes in the binary tree in a pre-order format where true suggests the node exists and false suggests it is NULL.

**Output Format:**

* Print true if the binary tree is symmetric, otherwise, print false.

**Constraints:**

* The number of nodes in the tree is in the range [0, 2000].
* -1000 <= Node.val <= 1000

**Examples:**

Example 1:

Input:

1 true 2 true 3 false false true 4 false false true 2 true 4 false false true 3 false false

Output:

true

Explanation:

The binary tree is:

1

/ \

2 2

/ \ / \

3 4 4 3

The binary tree is symmetric.

Example 2:

Input:

1 true 2 true 3 false false true 4 false false true 2 true 4 false false true 5 false false

Output:

false

Explanation:

The binary tree is:

1

/ \

2 2

/ \ / \

3 4 4 5

The binary tree is not symmetric.

Example 3:

Input:

1 true 2 true 3 false false false true 2 true 3 false false false

Output:

false

**Notes:**

* A binary tree is symmetric if the left subtree is a mirror reflection of the right subtree.

**Test Case 1:**

Input:

1 true 2 true 3 false false true 4 false false true 2 true 4 false false true 3 false false

Output:

true

Explanation:

The binary tree is:

1

/ \

2 2

/ \ / \

3 4 4 3

The binary tree is symmetric.

**Test Case 2:**

Input:

1 true 2 true 3 false false true 4 false false true 2 true 4 false false true 5 false false

Output:

false

Explanation:

The binary tree is:

1

/ \

2 2

/ \ / \

3 4 4 5

The binary tree is not symmetric.

**Test Case 3:**

Input:

1 true 2 true 3 false false false true 2 true 3 false false false

Output:

false

Explanation:

The binary tree is:

1

/ \

2 2

/ \

3 3

The binary tree is not symmetric.

**Test Case 4:**

Input:

1 true 2 false true 3 false false true 2 false true 3 false false

Output:

true

Explanation:

The binary tree is:

1

/ \

2 2

\ \

3 3

The binary tree is symmetric.

**Test Case 5:**

Input:

1 true 2 true 3 true 4 false false false false true 2 true 3 true 4 false false false false

Output:

false

Explanation:

The binary tree is:

1

/ \

2 2

/ /

3 3

/ /

4 4

The binary tree is not symmetric.

**Sample Test Case 1:**

Input:

15 true 10 true 7 true 5 false false true 8 false false true 10 false false true 7 true 5 false false true 16 false false

Output: false

Explanation: The binary tree is:

        15

       /  \

     10    10

    / \      / \

   7   7  7   7

  / \     / \

 5   8   8   16

15

/ \

10 7

/ \ / \

7 8 10 7

/ \ / \

5 16 5 16

This tree is non symmetric.

**Sample Test Case 2:**

Input: 15 true 10 true 7 true 5 false false true 8 false false true 10 false false true 7 true 5 false false true 15 false false

Output: true

Explanation: The binary tree is:

         15

        /  \

     10     10

    /  \    /  \

   7   7  7   7

  / \     / \

 5   8   8   5

This tree is symmetric.

**Constraints:**

The number of nodes in the tree is in the range `[1, 1000]`.

100 <= Node.val<= 100`

**Extra Test Cases:**

Input: 15 true 10 true 7 true 5 false false true 8 false false true 10 false false true 7 true 5 false false true 16 false false

Output: false

Input: 15 true 10 true 7 true 5 false false true 8 false false true 10 false false true 7 true 5 false false true 15 false false

Output: true

Input: root = 0 false false

Output: true

Input: root = 1 1 false false true 1 false false

Output: true

Input: root = 1 false false

Output: true