```java
package binarytree;



class BinaryTree51
{
    Node root1, root2;

    /* Given two trees, return true if they are
    structurally identical */
    boolean identicalTrees(Node a, Node b)
    {
        /*1. both empty */
        if (a == null && b == null)
            return true;

        /* 2. both non-empty -> compare them */
        if (a != null && b != null)
            return (a.data == b.data
                    && identicalTrees(a.left, b.left)
                    && identicalTrees(a.right, b.right));

        /* 3. one empty, one not -> false */
        return false;
    }

    /* Driver program to test identicalTrees() function */
    public static void main(String[] args)
    {
        BinaryTree51 tree = new BinaryTree51();

        tree.root1 = new Node(1);
        tree.root1.left = new Node(2);
        tree.root1.right = new Node(3);
        tree.root1.left.left = new Node(4);
        tree.root1.left.right = new Node(5);

        tree.root2 = new Node(1);
        tree.root2.left = new Node(2);
        tree.root2.right = new Node(3);
        tree.root2.left.left = new Node(4);
        tree.root2.left.right = new Node(5);

        if (tree.identicalTrees(tree.root1, tree.root2))
            System.out.println("Both trees are identical");
        else
            System.out.println("Trees are not identical");

    }
}
```