import algorithm.TreeNode;

public class SubtreeofAnotherTree {

public boolean isSubtree(TreeNode s, TreeNode t) {

if(s == null || t == null)

return false;

if(isSameTree(s,t)){

return true;

}

else

{

boolean left = isSubtree(s.left,t);

boolean right = isSubtree(s.right,t);

return left || right;

}

}

public boolean isSameTree(TreeNode node1, TreeNode node2){

if(node1==null && node2 == null)

return true;

if(node1 == null || node2 == null)

return false;

if(node1.val != node2.val){

return false;

}

boolean left = isSameTree(node1.left,node2.left);

boolean right = isSameTree(node1.right,node2.right);

return left & right;

}

}