/\*\*

 \* Definition for a binary tree node.

 \* struct TreeNode {

 \*     int val;

 \*     struct TreeNode \*left;

 \*     struct TreeNode \*right;

 \* };

 \*/

void compareTree(struct TreeNode\* p, struct TreeNode\* q,bool \*ans){

    if( p != NULL && q != NULL){//如果比較該節點不為NULL就比較數字

        //printf("%d %d\n",p->val,q->val);

        if(p->val != q->val){

            \*ans=false;

            return;

        }

    }else if( p == NULL && q == NULL){//如果都是NULL就停止

        return;

    }else{//如果有一個有NULL另外一個有值就將ans 改為false

        \*ans=false;

        return;

    }

    if( p->left != NULL && q->left  != NULL){//先一起找看看左邊，如果都不為NULL就往左邊探看看

        compareTree( p->left , q->left ,ans);

    }else if( p->left == NULL && q->left == NULL){//如果左邊都是NULL就不做任何動作

    }else{//如果任一邊有值另外一邊是NULL就將ans改為false

         \*ans=false;

    }

    if( p->right != NULL && q->right != NULL){//一起找看看右邊，如果都不為NULL就往左邊探看看

        compareTree( p->right , q->right ,ans);

    }else if( p->right == NULL && q->right == NULL){//如果右邊都是NULL就不做任何動作

    }else{//如果任一邊有值另外一邊是NULL就將ans改為false

        \*ans=false;

    }

}

bool isSameTree(struct TreeNode\* p, struct TreeNode\* q){

    bool ans=true;

    compareTree(p,q,&ans);

    return ans;

}