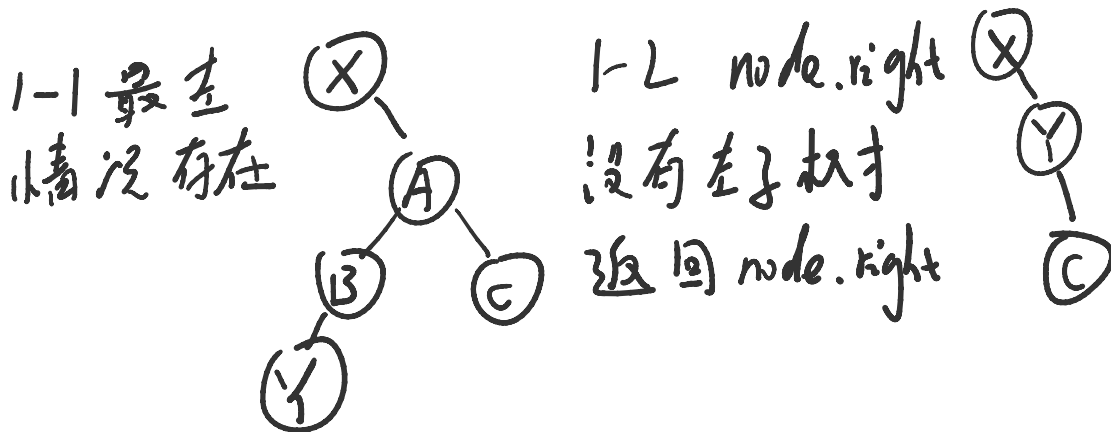


因为有指向父节点的指针, 可以通过这个
求出后继结点.

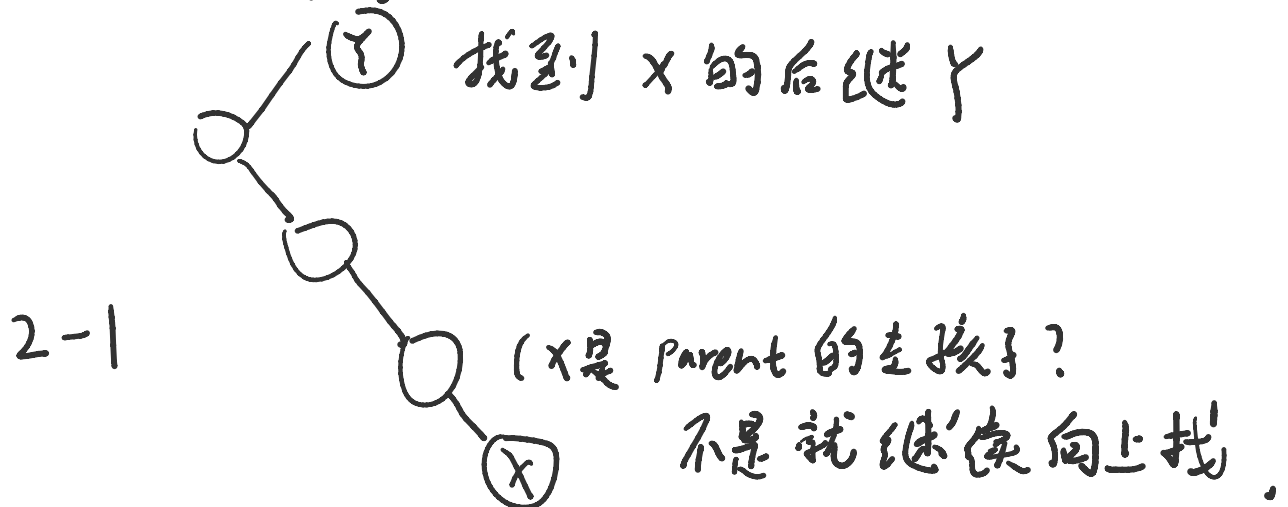
X到Y的距离为K, 时间复杂度为 $O(K)$

情况①:

X有右树的时候, X的后继节点为右树上的最左节点



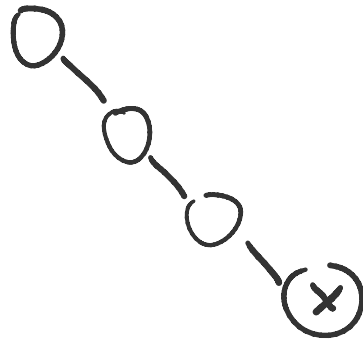
情况②: X没有右树 → 往上找



X是Y左树最后打印的节点, 所以X之后

X是(左树取后序遍历的最后一个), 所以X是Y
是Y

2-2



X为树上最右节点,
找不到后继节点!

代码:

```
*
* @Description 求一个节点在二叉树中序遍历中的后继节点 (这个节点后面一个)
* @Author
*/
public class SuccessorNode {
    public static Node getSuccessorNode(Node head, Node x){
        if(head == null){
            return null;
        }
        if(x.right != null){
            return getMostLeft(x.right); //查找x右树上的最左节点
        }else{
            Node parent = x.parent;
            while(parent != null && parent.right == x){
                //包括情况2-2, parent为null即x为最右节点
                x = parent;
                parent = x.parent;
            }
            return parent;
        }
    }
}

public static Node getMostLeft(Node node){
    Node cur = node;
    while(cur.left != null){
        cur = cur.left;
    }
    return cur; //包括了情况1-2, 没有左子树直接返回head.right
}
```