Given a tree node, find its next node based on in-order transversal

1. If current node has right sub node, return its further left leave node
2. Go find its parent
   1. If its parent’s left is itself, return its parent
   2. Else this node may be located in right subtree of left tree, going up to root to find next value
   3. If still cannot find, return None, this suggests current node is the right most node and no more node is available to transverse.

# Definition for a binary tree node.

# class TreeNode(object):

# def \_\_init\_\_(self, x):

# self.val = x

# self.left = None

# self.right = None

# self.father = None

class Solution(object):

def inorderSuccessor(self, p):

"""

:type q: TreeNode

:rtype: TreeNode

"""

if p.right:

temp = p.right

while temp.left:

temp = temp.left

return temp

father = p.father

while father and father.left != p :

p = father

father = p.father

return father

<https://www.acwing.com/problem/content/31/>

No.19