```
class ListNode:
    def __init__(self, x):
        self.val = x
        self.next = None
```

92反转链表2

pre:开始反转的前一个节点

cur:原链表当前节点

ne:原链表下一个节点

nx:新链表下一个节点,初始化为第n+1个节点

```
class Solution:
    def reverseBetween(self, head: ListNode, m: int, n: int) -> ListNode:
        tmp = ListNode(0)
        tmp.next = head
        pre = tmp
        cur = tmp


        for i in range(m):
            pre = cur
            cur = cur.next
        p = cur
        for i in range(n-m+1):
            p = p.next
        nx = p#这里的nx是更新后的nx 1→2→3→4→5中更新后4→3→2→5  那么第一次到2时 更新后2→5 nx初始化为5 cur=ne=3后nx=2
        while cur!=p:# a b c
            ne = cur.next
            # print(pre.val,cur.val,ne.val)
            cur.next = nx
            nx = cur
            cur = ne

        pre.next = nx# fro:1→2→3→4→5中的1
        return tmp.next
```