

```
public ListNode reverseBetween(ListNode head, int m, int n) {  
    if (m >= n || head == null) {  
        return head;  
    }  
  
    ListNode dummy = new ListNode(0);  
    dummy.next = head;  
    head = dummy;  
  
    for (int i = 1; i < m; i++) {  
        if (head == null) {  
            return null;  
        }  
        head = head.next;  
    }  
  
    ListNode preNode = head;  
    ListNode mNode = head.next;  
    ListNode nNode = mNode, postNode = mNode.next;  
    for (int i = m; i < n; i++) {  
        if (postNode == null) {  
            return null;  
        }  
        ListNode temp = postNode.next;  
        postNode.next = nNode;  
        nNode = postNode;  
        postNode = temp;  
    }  
    mNode.next = postNode;  
    preNode.next = nNode;  
  
    return dummy.next;  
}
```

# 交换相邻结点

给定一个链表，对每两个相邻的结点作交换并返回头节点。

例如：

给定 1->2->3->4，你应该返回 2->1->4->3。

你的算法应该只使用额外的常数空间。不要修改列表中的值，只有节点本身可以更改

(<https://leetcode-cn.com/problems/swap-nodes-in-pairs/description/>)