

题目：LCR 024. 反转链表

```

1  class Solution {
2  public:
3      ListNode* reverseList(ListNode* head) {
4          // //头插法
5          // ListNode* newHead = new ListNode();
6          // while(head)
7          // {
8          //     ListNode* next = head->next;
9          //     head->next = newHead->next;
10         //     newHead->next = head;
11         //     head = next;
12         // }
13         // return newHead->next;
14         // //三指针法
15         // ListNode* prev = nullptr;
16         // ListNode* curr = head;
17         // while (curr)
18         // {
19         //     ListNode* next = curr->next;
20         //     curr->next = prev;
21         //     prev = curr;
22         //     curr = next;
23         // }
24         // return prev;
25         //递归
26         if(head == nullptr || head->next == nullptr)
27         {
28             return head;
29         }
30         ListNode* newHead = reverseList(head->next);
31         head->next->next = head;
32         head->next = nullptr;
33         return newHead;
34     }
35 }
36 };

```

题目：24. 两两交换链表中的节点

```
1 class Solution {
2 public:
3     ListNode* swapPairs(ListNode* head) {
4         if(head == nullptr || head->next == nullptr)
5             {
6                 return head;
7             }
8         ListNode* newHead = swapPairs(head->next->next);
9         ListNode* next = head->next;
10        next->next = head;
11        head->next = newHead;
12        return next;
13    }
14 };
```