

# 题目是这样的

## 24. 两两交换链表中的节点

已解答 

中等

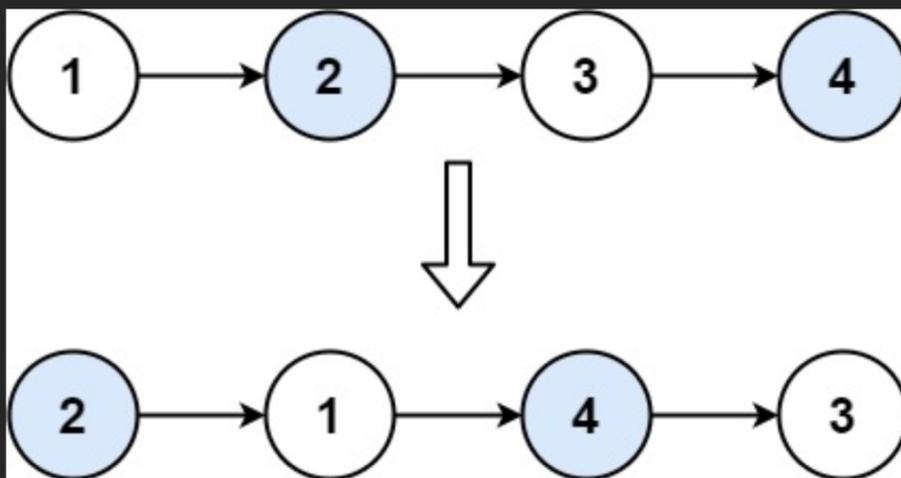
 相关标签

 相关企业

Ax

给你一个链表，两两交换其中相邻的节点，并返回交换后链表的头节点。你必须在不修改节点内部的值的情况下完成本题（即，只能进行节点交换）。

示例 1：



两两节点交换，比较基础，直接写代码

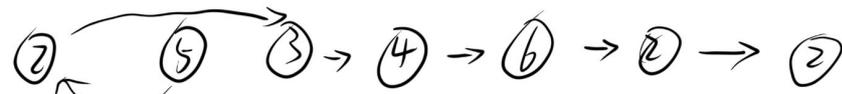
```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode() : val(0), next(nullptr) {}
 *     ListNode(int x) : val(x),
next(nullptr) {}
 *     ListNode(int x, ListNode *next) :
val(x), next(next) {}
};
```

```

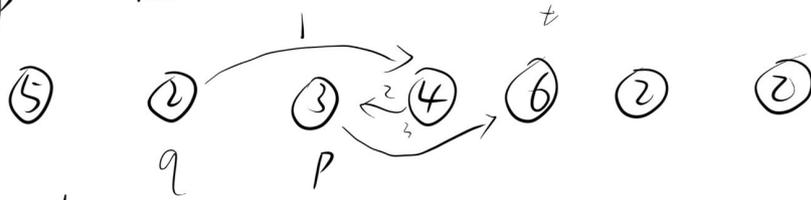
    \* };
    */ //题目给的结构体
class Solution {
public:
    • ListNode* swapPairs(ListNode* head) {
    •     if(head==nullptr||head->next==nullptr)return head;//一个或没有节点直接返回,剩下的就至少有两个节点了
    •     ListNode *p=head,*q=head->next,*t;
    •     p->next=q->next;
    •     q->next=p;
    •     head=q; //这三句是先给前面两个特殊的节点交换一下
    •     while(p->next&&p->next->next)//查看后面两个节点是否存在,只有一个或没有不执行
    •     {
    •         q=p;
    •         p=p->next;
    •         t=p->next->next;
    •         q->next=p->next;
    •         p->next->next=p;
    •         p->next=t;
    •     }
    •     return head;//由于改变了结构,返回新链表
    • }
};

```

要看懂链表最好结合图来理解,我不会画太惊艳的图,循环里的语句是我像下面图那样推出来的,可以照我这样画图一步步执行语句理解



head  
p  
q  
head



head

$q \rightarrow next = p \rightarrow next$

$p \rightarrow next \rightarrow next = p$

$p \rightarrow next = t$

z q      4      } p      6      t

