

5月4日报

本日学习内容

- 1. 每日算法题
- 2. 看完第五章内容

今日算法题

题目1:[两数相加](#)

2. 两数相加

已解答

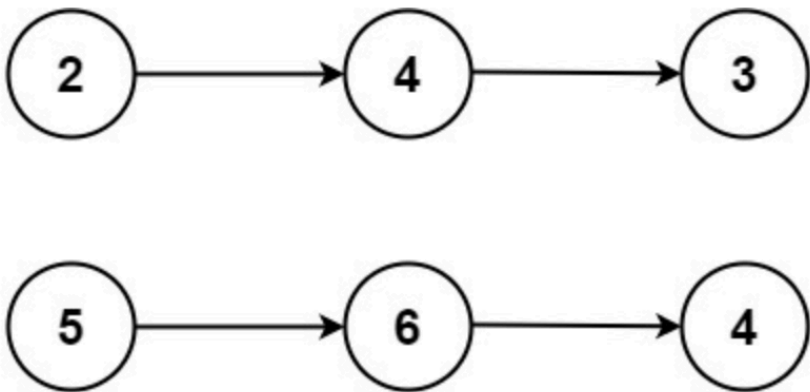
中等 🏷 相关标签 🔒 相关企业 A+

给你两个 **非空** 的链表，表示两个非负的整数。它们每位数字都是按照 **逆序** 的方式存储的，并且每个节点只能存储 **一位** 数字。

请你将两个数相加，并以相同形式返回一个表示和的链表。

你可以假设除了数字 0 之外，这两个数都不会以 0 开头。

示例 1:



```
struct ListNode* addTwoNumbers(struct ListNode* l1, struct ListNode* l2) {  
    struct ListNode* dum = (struct ListNode *)malloc(sizeof(struct ListNode));  
    struct ListNode* cur = dum;  
    int val = 0;
```

```
while (l1 != NULL || l2 != NULL || val != 0) {
    if (l1 != NULL) {
        val += l1->val;
        l1 = l1->next;
    }
    if (l2 != NULL) {
        val += l2->val;
        l2 = l2->next;
    }
    cur->next = malloc(sizeof(struct ListNode));
    cur = cur->next;
    cur->val = val % 10;
    val /= 10;
}
cur->next = NULL;
return dum->next;
}
```

题目二： [旋转链表](#)

61. 旋转链表

已解答

中等

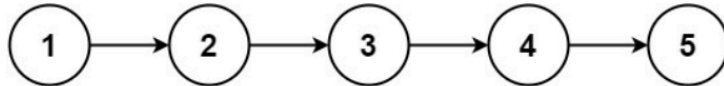
🏷 相关标签

🔒 相关企业

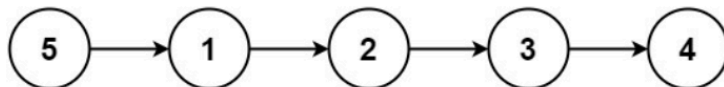
Aa

给你一个链表的头节点 `head`，旋转链表，将链表每个节点向右移动 `k` 个位置。

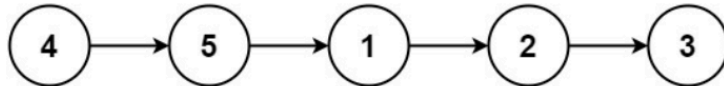
示例 1:



rotate 1



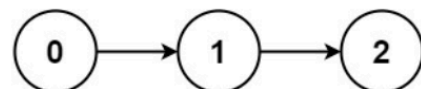
rotate 2



输入: `head = [1,2,3,4,5]`, `k = 2`

输出: `[4,5,1,2,3]`

示例 2:



```
struct ListNode* rotateRight(struct ListNode* head, int k) {
    if (head == NULL || head->next == NULL) {
        return head;
    }
    int n = 1;
    struct ListNode* p = head;
    while (p->next != NULL) {
        n++;
        p = p->next;
    }
    k = k % n;
    if (k == 0) {
        return head;
    }
    struct ListNode* new_tail = head;
    for (int i = 0; i < n - k - 1; i++) {
        new_tail = new_tail->next;
    }
}
```

```
struct ListNode* new_head = new_tail->next;
new_tail->next = NULL;
p->next = head;
return new_head;
}
```

本日遇到的问题

1. 对super与self关键字具体应用不够熟练

明日学习计划

1. 每日两道题
2. 复习第五章内容
3. 完成周报