

10、两个链表A和B，奇数存放在A，偶数存放在B

24计算机考研成员一战成硕！



题目描述：

将带头结点的单链表A分解为两个带头结点的单链表A和B，使得A表中含有原表中值为奇数的元素，而B表中含有原表中值为偶数的元素

🕒 倒计时

1、知识点及难度



题解人：酒客

难度：简单

知识点：双指针

2、算法题

思路

这段代码是一个函数，函数名为divide，它接受一个链表A作为参数，并返回一个新的链表B。

在函数内部，首先创建一个新的链表B，用变量Bfrost表示B链表的当前节点。

然后，用变量Afrost表示A链表的当前节点，通过while循环遍历A链表。

在循环中，首先将Afrost的下一个节点保存在变量temp中。

然后，判断temp是否为空并且temp的值是否为偶数。如果是，则执行下面的操作

- 将Afrost的下一个节点指向temp的下一个节点，即将temp从A链表中删除。

- 将temp的下一个节点指向空，即将temp作为新链表B的最后一个节点。
- 将Bfrost的下一个节点指向temp，即将temp插入到B链表中。
- 更新Bfrost为B链表的最后一个节点。

最后，将Afrost指向A链表的下一个节点，继续循环直到遍历完整个A链表。

最后，返回链表B。

1.

基本实现-C++

```
1 LinkedList divide(LinkedList A){
2     ListNode* B = new ListNode();
3     LinkedList Afrost = A;
4     LinkedList Bfrost = B;
5     while (Afrost){
6         LinkedList temp = Afrost->next;
7         if (temp != nullptr && temp->val % 2 == 0){
8             Afrost->next = temp->next;
9             temp->next = nullptr;
10            Bfrost->next = temp;
11            Bfrost = Bfrost->next;
12        }
13        Afrost = Afrost->next;
14    }
15    return B;
16 }
```

基本实现C

```
1 struct ListNode* divide(struct ListNode* A) {
2     struct ListNode* B = (struct ListNode*)malloc(sizeof(struct ListNode));
3     struct ListNode* Afrost = A;
4     struct ListNode* Bfrost = B;
5     while (Afrost) {
6         struct ListNode* temp = Afrost->next;
7         if (temp != NULL && temp->val % 2 == 0) {
8             Afrost->next = temp->next;
9             temp->next = NULL;
10            Bfrost->next = temp;
11            Bfrost = Bfrost->next;
12        }
13    }
```

```
13         Afrost = Afrost->next;
14     }
15     return B;
16 }
```

3、总结



总结栏

蓝蓝B站首页：[蓝蓝希望你上岸呀B站首页](#)

蓝蓝公众号：[算法训练营9分计划](#)

蓝蓝知识星球介绍：[📖 关于知识星球的权益](#)

如何在星球打卡记录：

- 计算机考研数据结构算法专项day[1/60]:
- 学习内容：最好能发出自己写的图片
- 遇到的问题：如果无就不用写了
- 小结：这部分一周写一次即可。