

04、删除单链表L中所有值为x的结点

24计算机考研成员一战成硕！



题目描述：

删除不带头结点的单链表 L 中所有值为 x 的结点

🕒 倒计时

1、知识点及难度



题解人：酒客

难度：简单

知识点：双指针遍历链表

2、算法题

思路

首先，函数会判断链表是否为空，如果为空则直接返回，不进行任何操作。

接下来，定义两个指针 curr 和 prev，分别指向当前节点和前一个节点。初始化时，curr 指向头节点，prev 指向空指针。

然后，通过一个循环遍历链表，直到 curr 指针指向空节点。

在循环中，首先判断当前节点的数值是否等于 x。如果相等，则表示找到了要删除的节点。

接下来，创建一个临时指针 temp，指向当前节点。

然后，判断当前节点是否为头节点。如果是头节点，则将头节点指向下一个节点，否则将前一个节点的 next 指针指向下一个节点。

然后，将 curr 指针指向下一个节点，并释放临时指针 temp 所指向的节点。

如果当前节点的数值不等于 x，则将 prev 指针指向当前节点，将 curr 指针指向下一个节点，继续循环。

最终，当循环结束后，链表中所有数值为 x 的节点都被删除了。

1.

基本实现-C++

```
1
2 void deleteNodesWithValue(Node& head, int x) {
3     if (head == nullptr) {
4         return;
5     }
6
7     Node* curr = head;
8     Node* prev = nullptr;
9
10    while (curr != nullptr) {
11        if (curr->data == x) {
12            Node* temp = curr;
13
14            // 特殊处理头节点
15            if (curr == head) {
16                head = curr->next;
17            } else {
18                prev->next = curr->next;
19            }
20
21            curr = curr->next;
22            delete temp;
23        } else {
24            prev = curr;
25            curr = curr->next;
26        }
27    }
28 }
```

基本实现C

```
1 void deleteNodesWithValue(Node* head, int x) {
2
3     if (head == NULL) {
4         return;
```

```

5      }
6
7      Node* curr = head;
8      Node* prev = NULL;
9      while (curr != NULL) {
10     if (curr->data == x)
11     {
12         Node* temp = curr;
13         // 特殊处理头节点if (curr == head) {
14             head = curr->next;
15         }
16     else {
17         prev->next = curr->next;
18     }
19
20     curr = curr->next;
21     free(temp);
22 }
23     else {
24     prev = curr;
25     curr = curr->next;
26     }
27 }

```

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。