



04、删除下标 $i \sim j$ ($i \leq j$) 的所有元素

24计算机考研成员一战成硕!

🔄 题目描述：
从给定顺序表L中删除下标 $i \sim j$ ($i \leq j$) 的所有元素，假定 i, j 都是合法的

🔄

🔄 算法题解

思路

把顺序表给定的范围后面的数移到给定的区间

假设：

给定顺序表{0, 1, 2, 3, 4}

给定范围：1 - 2

(把3和4挪到前面给定的范围，再把顺序表长度缩小)

输出的顺序表{0, 3, 4}

这段代码是用来操作一个顺序表的数据结构。首先，定义了一个结构体 `SqList` 用来表示顺序表，它包含一个整型数组 `arr` 用来存储数据，数组的大小由 `Maxsize` 定义，`length` 表示当前顺序表中的元素个数。

接下来，定义了一些函数：

1. `init_Array` 函数用于初始化顺序表。它通过输入从终端读入 `length`，然后依次读入 `length` 个整数，并将它们存储到顺序表的数组 `arr` 中。

2. `del_arr` 函数用于删除顺序表中指定位置的元素。它接受三个参数：`array` 是要操作的顺序表，`i` 是要删除元素的起始位置，`j` 是要删除元素的结束位置，`len` 是当前顺序表中的元素个数。

在函数内部，定义了两个变量 `p` 和 `q`，分别初始化为 `j+1` 和 `i`。然后，使用循环将 `array.arr[p]` 的值复制到 `array.arr[q]`，并分别递增 `p` 和 `q`，实现了删除元素的操作。最后，更新顺序表的 `length`，删除的元素个数为 $(j - i) + 1$ 。

3. `printArray` 函数用于打印顺序表的元素。它接受两个参数：`array` 是要打印的顺序表，`length` 是顺序表中的元素个数。在函数内部，使用循环遍历顺序表的数组 `arr`，并打印其中的元素。

在 `main` 函数中，首先定义了一个测试用的数组 `test`，然后计算了该数组的长度。接着，定义了一个顺序表 `array`，并调用 `init_Array` 函数对其进行初始化。然后，调用 `del_arr` 函数删除顺序表中的元素，起始位置为 1，结束位置也为 1，最后一个参数为顺序表当前的元素个数。最后，调用 `printArray` 函数打印剩余的元素。

```
1 C版本实现
2
3 #include <stdio.h>
4
5 #define Maxsize 100
6 //结构体定义
7 typedef struct Sqlist {
8     int arr[Maxsize];
9     int length;
10 } Array;
11
12 //参数为指针 可以理解为数组
13 //初始化数组
14 void init_Array(Array* array) {
15     scanf("%d", &(array->length));
16     for (int i = 0; i < array->length; i++) {
17         scanf("%d", &(array->arr[i]));
18     }
19 }
20
21 void del_arr(Array* array, int i, int j) {
22     int p = j + 1;
23     int q = i;
24     while (p < array->length) {
25         array->arr[q] = array->arr[p];
26         q++;
27         p++;
28     }
29     array->length = array->length - (j - i) - 1;
30 }
31
32 void printArray(Array* array) {
33     for (int i = 0; i < array->length; i++) {
34         printf("%d ", array->arr[i]);
35     }
36 }
37
38 int main() {
39     int test[] = { 1, 2, 3, 4, 5 };
40     int len = sizeof(test) / sizeof(test[0]);
41     // 也可以直接这样初始化顺序表
```

```

42 // Array array = { {1,2,3,4,5},5 };
43 Array array;
44 array.length = len;
45 for (int i = 0; i < len; i++) {
46     array.arr[i] = test[i];
47 }
48 // 初始化顺序表
49 // init_Array(&array);
50
51 del_arr(&array, 1, 1);
52 printArray(&array);
53
54 return 0;
55 }

```

```

1 C++版本实现
2 typedef struct Sqlist {
3     int arr[Maxsize];
4     int length;
5 }Array;
6 void init_Array(Array &array)
7 {
8     cin >> array.length;
9     for( int i = 0; i < array.length; i++ ) {
10         cin >> array.arr[i];
11     }
12
13 }
14
15 void del_arr(Array & array,int i ,int j,int len) {
16     int q = i;
17     int p = j + 1;
18     while( p < len ) {
19         array.arr[q++] = array.arr[p++];
20     }
21     array.length = len - ( j - i ) - 1;
22 }
23 void printArray(Array& array , int length) {
24     for( int i = 0; i < array.length; i++ ) {
25         cout << array.arr[i];
26     }
27 }
28 int main() {
29     int test[] = { 1,2,3,4,5 };
30     int len = sizeof(test) / sizeof(test[0]);
31     //也可以直接这样初始化顺序表
32     //Array array = { {1,2,3,4,5},5 };
33     Array array;
34     init_Array(array);
35     del_arr(array,1,1,5);
36     printArray(array , array.length);
37 }

```



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。

蓝蓝算法训练营