

## 实验四实验报告

1. 编程语言：C++
2. 操作系统：Windows
3. 功能：实现线性表在顺序存储结构上的插入、删除运算；实现线性表在链接存储结构的插入、删除运算。

4. 顺序结构：如果线性表已满，则插入失败。若不满，则将插入位置以及其后元素往后移一位，再将插入的元素赋值给线性表的对应位置。

链接存储结构：则将该位置前一个元素的 `next` 指针指向插入元素的结点，将插入元素的结点的 `next` 指针指向下一位置的结点。

顺序结构删除元素则将该位置以后的元素都往前移一位。

链表删除元素则将前一个结点的 `next` 指针指向删除位置后一个结点，然后删除该位置的结点。

### 5. 实验展示：

顺序：



```
命令提示符 - a.exe
array list:
1. insert
2. remove
3. show
0. quit
_
```

插入：

```
命令提示符 - a
array list:
1. insert
2. remove
3. show
0. quit
1
please input element:
c
please input position:
1
insert success!
请按任意键继续. . .
```

连续插入后:

```
命令提示符 - a
array list:
1. insert
2. remove
3. show
0. quit
3
c o m p
请按任意键继续. . .
```

删除第三个:

```
命令提示符 - a
array list:
1. insert
2. remove
3. show
0. quit
2
please input position:
3
remove success!
请按任意键继续. . .
```

剩余:



```
命令提示符 - a
array list:
1. insert
2. remove
3. show
0. quit
3
c o p
请按任意键继续...
```

链式:



```
命令提示符 - a.exe
link list:
1. insert
2. remove
3. show
0. quit
1

```

连续插入:



```
命令提示符 - a.exe
link list:
1. insert
2. remove
3. show
0. quit
3
c o m p
请按任意键继续...
```

删除第二个:



```
命令提示符 - a.exe
link list:
1. insert
2. remove
3. show
0. quit
2
please input position:
2
remove success!
请按任意键继续. . .
```

剩余:



```
命令提示符 - a.exe
link list:
1. insert
2. remove
3. show
0. quit
3
c m p
请按任意键继续. . .
```

6. 实验感想：顺序存储和链表存储结构是数据结构的基础内容。然而平日的编程中，指针使用率非常少，导致对链表操作不甚熟悉