**《计算机实践》—数据结构**

**实验一：线性表的实验**

专 业：自动化

班 级：0319101

姓 名：赵巍

学 号：031910115

日 期：2022.11.06

南京航空航天大学

目录

[**《计算机实践》—数据结构** 1](#_Toc118722131)

[实验1.1 ：单链表的插入和删除 3](#_Toc118722132)

[1. 实验目的： 3](#_Toc118722133)

[2. 实验要求： 3](#_Toc118722134)

[3. 实验主要步骤： 3](#_Toc118722135)

[4. 修改程序： 4](#_Toc118722136)

[5. 调试过程： 5](#_Toc118722137)

[实验1.2 线性表元素的区间删除 6](#_Toc118722138)

[1. 实验内容 6](#_Toc118722139)

[2.实验要求 6](#_Toc118722140)

[3. 测试用例 7](#_Toc118722141)

[4. 输入输出要求 7](#_Toc118722142)

[5. 选作内容 7](#_Toc118722143)

[6. 程序调试 8](#_Toc118722144)

[附录 10](#_Toc118722145)

## 实验1.1 ：单链表的插入和删除

### 1. 实验目的：

了解和掌握线性表的逻辑结构和链式存储结构，

掌握单链表的基本算法及相关的时间性能分析。

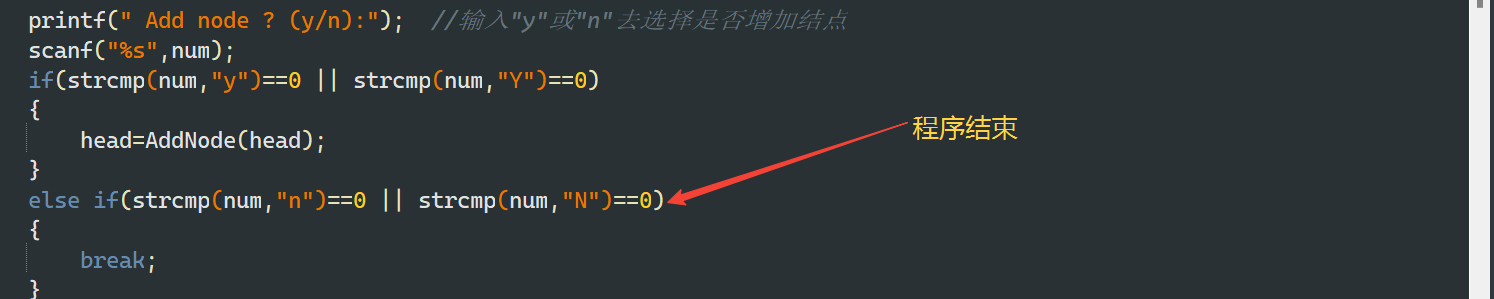
### 2. 实验要求：

建立一个数据域定义为字符串的单链表，在链表中不允许有重复的字符串；

根据输入的字符串，先找到相应的结点，后删除之。

### 3. 实验主要步骤：

分析、理解给出的示例程序并加以改进。(AddDelSingle\_L\_Lists\_Student.c)

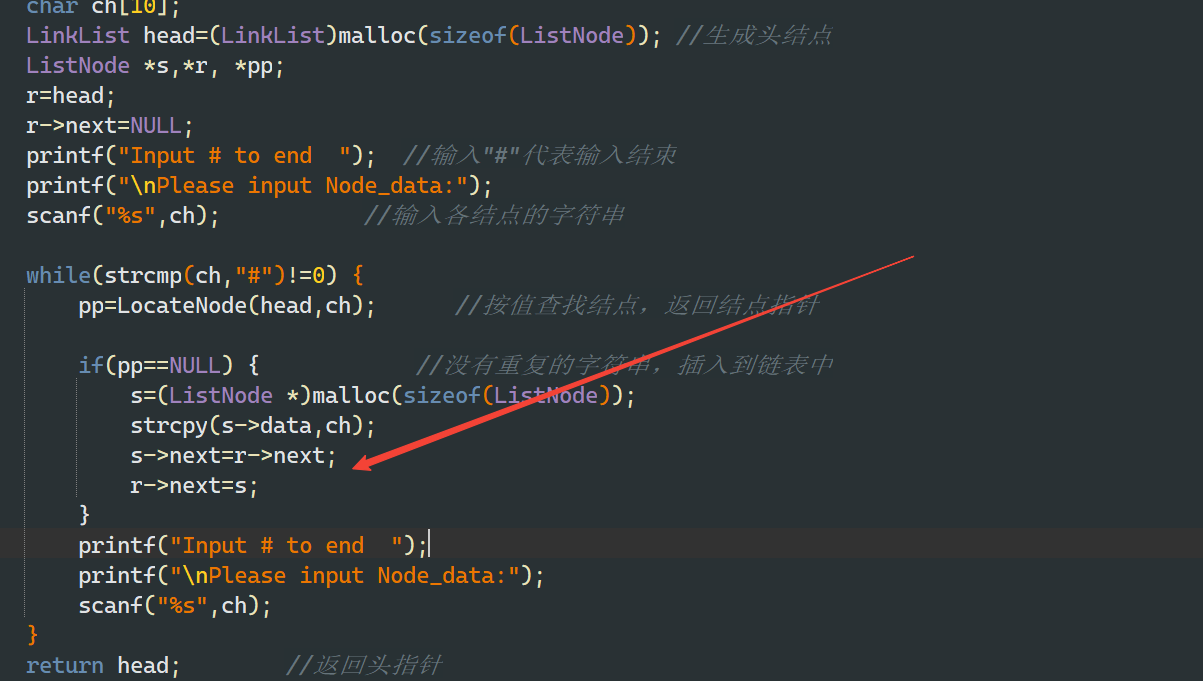


调试程序，并设计输入数据（如：bat，cat，eat，fat，hat，jat，lat，mat，#），测试程序的如下功能：不允许重复字符串的插入；根据输入的字符串，找到相应的结点并删除。

### 4. 修改程序：

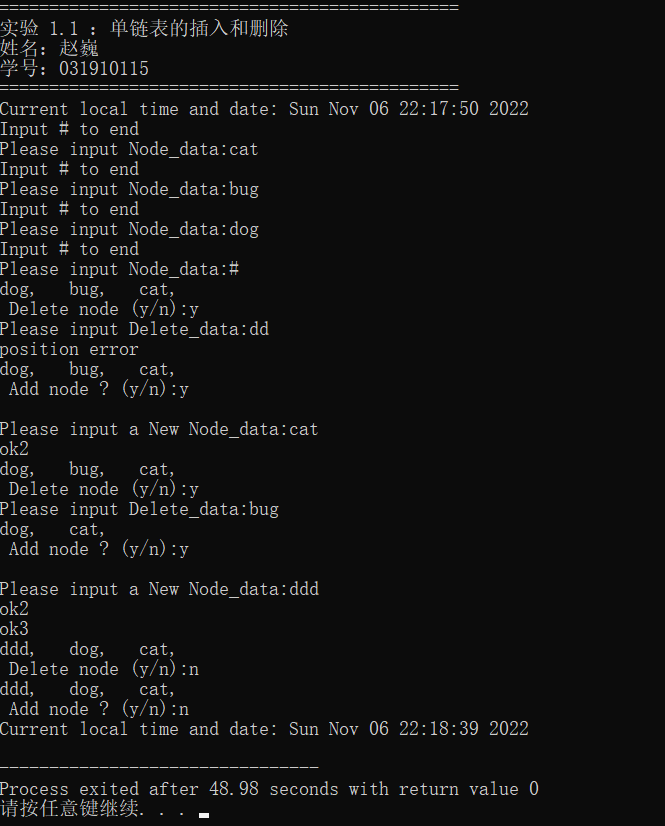
增加插入结点的功能(已提供)。

将建立链表的方法改为头插入法。



总结：头插法修改是核心把头节点的后继给新增元素的后继，然后再把头节点的后继改为新增元素地址。

### 5. 调试过程：



## 实验1.2 线性表元素的区间删除

### 1. 实验内容

创建一个线性表，然后设计一个函数删除所有值大于minD并且小于maxD的数据元素。删除后表中剩余数据元素保持原有存储结构，并且相对位置保持不变。

### 2.实验要求

（**1**）函数接口说明：

Delete(线性表类型名 \*L, ElementType minD, ElementTYpe maxD)，其中线性表数据元素设定为整数。

（**2**）使用顺序存储实现实验内容；

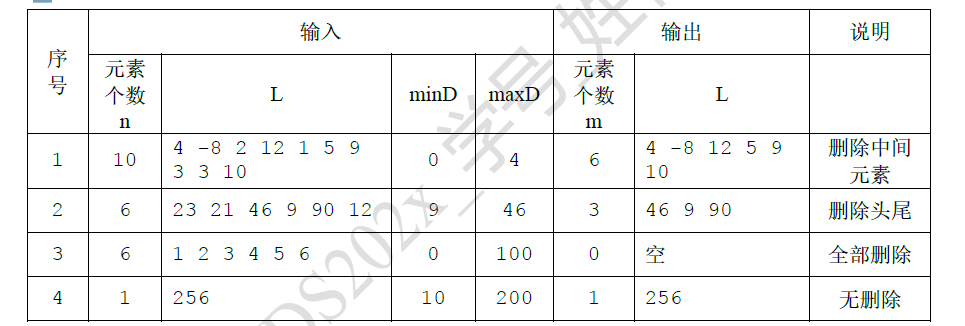
示例程序已给出

（**3**）使用链式存储实现实验内容。**（程序见附录图1，图2）**

链式存储实现实验内容时，需要注意的是删除元素的函数，free释放完内存后需要重行给判断指针赋值。



### 3. 测试用例



### 4. 输入输出要求

输入数据：通过合适的方式输入元素个数n和元素的值，创建线性表；然后输入minD和maxD的值（minD<=maxD）；

输出数据：首先输出线性表元素个数n，然后输出初始序列；换行后输出区间删除操作以后的元素个数m和正确序列。

### 5. 选作内容

尝试分析两种存储结构中，区间删除操作的时间复杂度和空间复杂度。

##### 链式存储：

* 时间复杂度O(1)：

删除数据通过遍历找到，只需要把删除节点的后继赋给前一个节点的后继，然后free释放此空间就可以了。

* 空间复杂度O(n)：

每次增加元素都需要malloc从堆区申请内存

顺序存储：

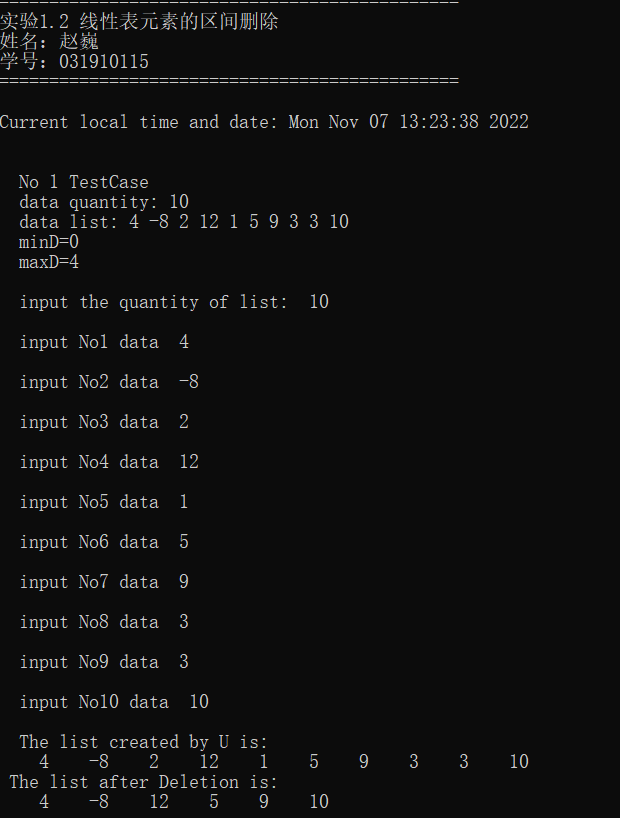
* 时间复杂度O(n)：

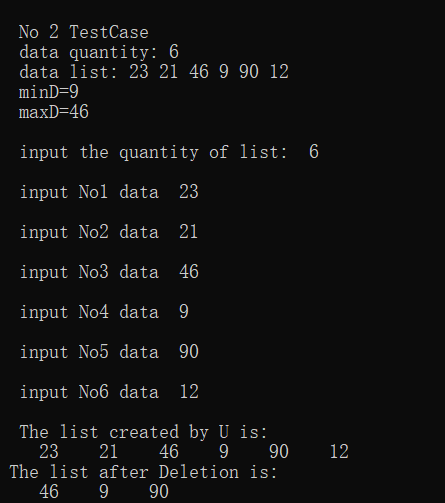
删除要在找到元素后，把后面的元素全部向前移动

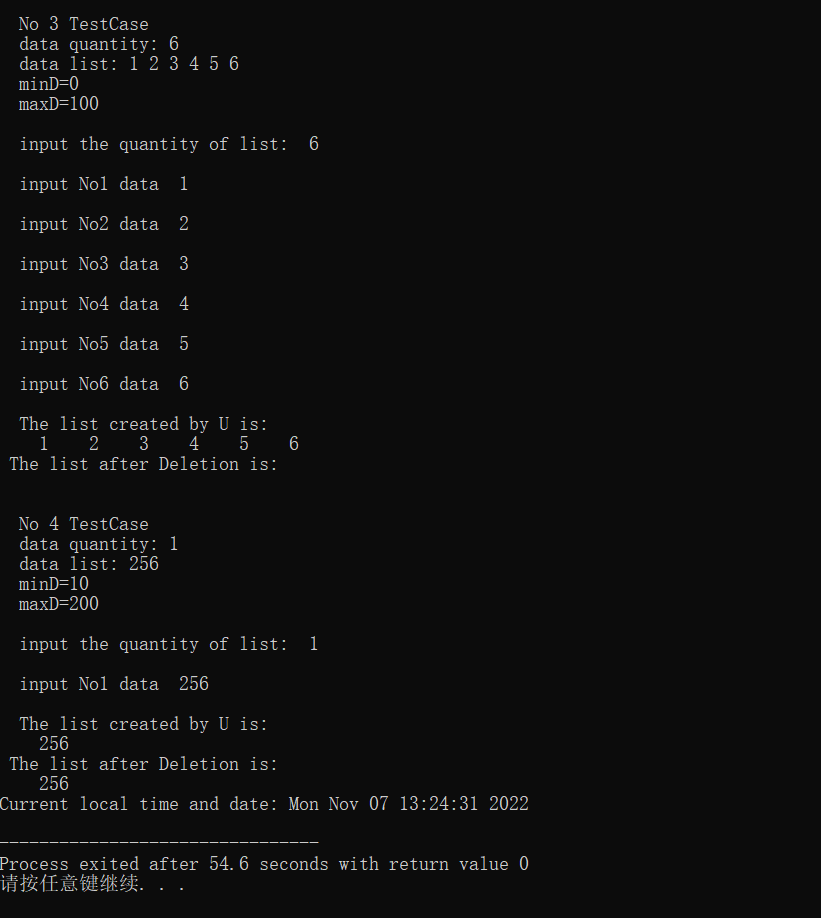
* 空间复杂度O(1)：

只要数组定义的够大，添加数据就不需要申请内存

### 6. 程序调试



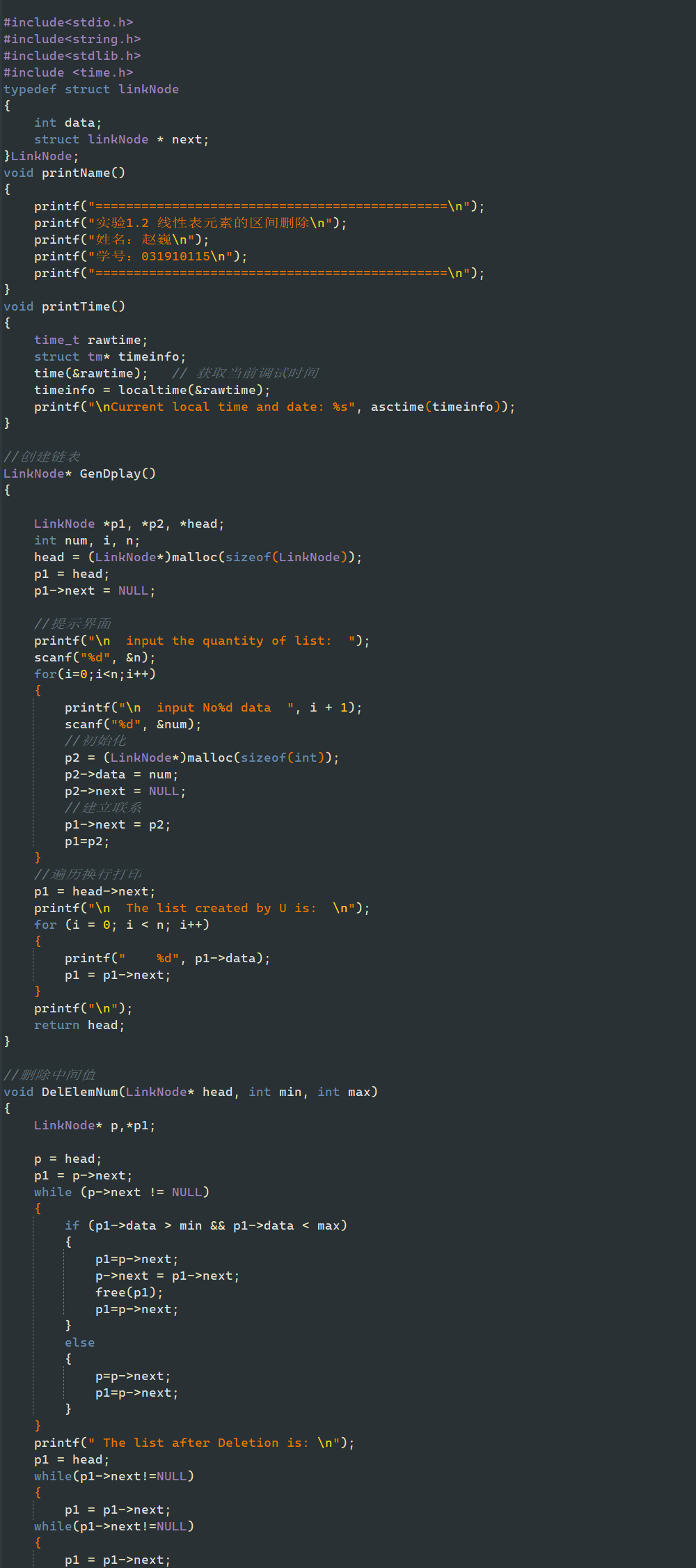




# 附录



图表 1



图表 2