



南昌大学实验报告

学生姓名： 丁俊 学 号： 8003119100 专业班级： 信安 193 班

实验类型： ☒ 验证 ☐ 综合 ☐ 设计 ☐ 创新 实验日期： 3.11 实验成绩：

一、实验项目名称

不带头结点的单链表

二、实验目的

利用已经给定的头文件实现删除第一个节点，以及插入数据节点操作。

三、实验任务

- 1、根据 `slinklist.h` 给出的头文件编写函数 `linklist delx(linklist head,datatype x)` 删除不带头节点单链表 `head` 中的一个值为 `x` 的结点；
- 2、假设单链表是升序排列的，设计算法函数 `linklist insert(linklist head,datatype x)`，将值为 `x` 的结点插入到链表 `head` 中，并爆出链表有序性，分别插入到表头、表中和表尾 3 种情况；
- 3、编写算法函数 `linklist delallx(linklist head,int x)`，删除不带头结点单链表 `head` 种所有值为 `x` 的结点。

四、主要仪器设备及耗材

Dec++5.15 windows10

五、 实验步骤

代码如下：

1、删除链表 head 种第一个值为 x 的结点

```
#include "slinklist.h"
```

```
linklist delx(linklist head, datatype x) {
```

```
    linklist p, pre;
```

```
    p = head;
```

```
    // 第一个结点就是 x
```

```
    if (p->info == x) {
```

```
        head = head->next;
```

```
        return head; // 提前退出返回 head
```

```
    }
```

```
    while (p && p->info != x) {
```

```
        pre = p;
```

```
        p = p->next;
```

```
    }
```

```
    if (p) {
```

```
        if (!pre)
```

```
            head = head->next;
```

```
        else
```

```
            pre->next = p->next;
```

```
        free(p);
```

```

    }

    return head;
}

int main() {
    datatype x;
    linklist head;
    head = creatbyqueue();
    print(head);
    printf("输入要删除的值: ");
    scanf("%d", &x);
    head = delx(head, x);
    print(head);
    delList(head);

    return 0;
}

```

2、保持链表有序性，并分别在头、中、尾部进行插入

```

#include "slinklist.h"

linklist insert(linklist head, datatype x) {
    linklist p, q, pre; // q 是要插入的节点
    q = (linklist)malloc(sizeof(node));
    q->info = x;

```

```
p = head;

// 插入头节点

if (p->info >= x) {

    head = q;

    q->next = p;

    return head;

}

pre = p, p = p->next;

while (p && x >= p->info) { // 找到第一个大于等于的数

    pre = p;

    p = p->next;

}

// 插入中间,插入到 p 后面

if (pre->next) {

    q->next = pre->next;

    pre->next = q;

}

// 最后一个点

else if (pre->next == NULL) {

    pre->next = q;

    q->next = NULL;

}
```

```

    return head;
}

int main() {
    datatype x;
    linklist head;
    printf("输入一组升序排列的整数: \n");
    head = creatbyqueue();
    print(head);
    printf("请输入要插入的值:");
    scanf("%d", &x);
    head = insert(head, x);
    print(head);
    delList(head);
    return 0;
}

```

3、删除链表中所有值为 x 的结点

```

#include "slinklist.h"

linklist delallx(linklist head, int x) {
    linklist p = head, p2 = head;
    while (p != NULL) {
        if (p->info == x) { // 判断头结点
            if (p == head)

```

```

        head = p->next;

    else {

        p2->next = p->next;

        free(p);

        p = p2;

    }

}

p2 = p;

p = p->next;

}

return head;

}

int main() {

    datatype x;

    linklist head;

    head = creatbyqueue();

    print(head);

    printf("请输入要删除的值:\n");

    scanf("%d", &x);

    head = delallx(head, x);

    print(head);

    delList(head);

```

```
    return 0;
}
```

六、实验数据以及处理结果

1、删除链表第一个值为 x 的结点

删除头结点时:

```
D:\gitworkspace\fresh\algorithm\简单递归\slinklist1.exe
请输入若干整数序列:
1 2 3 4 7 8 91 10
)
list is:
1 2 3 4 7 8 91 10
输入要删除的值: 1
list is:
2 3 4 7 8 91 10
-----
Process exited after 11.22 seconds with return value 0
请按任意键继续. . .
```

删除中间结点时:

```
D:\gitworkspace\fresh\algorithm\简单递归\slinklist1.exe
请输入若干整数序列:
1 2 5 6 7 2 0
)
list is:
1 2 5 6 7 2
输入要删除的值: 2
list is:
1 5 6 7 2
-----
Process exited after 8.904 seconds with return value 0
请按任意键继续. . .
```

删除尾结点:

```
D:\gitworkspace\fresh\algorithm\简单递归\linklist1.exe
请输入若干整数序列:
1 2 3 6 7 1 9 4 0
List is:
1 2 3 6 7 1 9 4
请输入要删除的值: 4
List is:
1 2 3 6 7 1 9
-----
Process exited after 11.66 seconds with return value 0
请按任意键继续. . .
```

2、 保持链表有序性，并分别在头、中、尾部进行插入

中间插入

```
D:\gitworkspace\fresh\algorithm\简单递归\linklist3.exe
输入一组升序排列的整数:
请输入若干整数序列:
1 3 4 5 6 7 9 10 0
List is:
1 3 4 5 6 7 9 10
请输入要插入的值:2
List is:
1 2 3 4 5 6 7 9 10
-----
Process exited after 10.4 seconds with return value 0
请按任意键继续. . .
```

头部插入

```
D:\gitworkspace\fresh\algorithm\简单递归\linklist3.exe
输入一组升序排列的整数:
请输入若干整数序列:
2 3 4 6 8 9 0
List is:
2 3 4 6 8 9
请输入要插入的值:1
List is:
1 2 3 4 6 8 9
-----
Process exited after 7.405 seconds with return value 0
请按任意键继续. . .
```

尾部插入


```
D:\gitworkspace\fresh\algorithm\简单递归\linklist3.exe
输入一组升序排列的整数:
请输入若干整数序列:
1 3 4 5 6 7 8 10 0
List is:
1 3 4 5 6 7 8 10
请输入要插入的值:12
List is:
1 3 4 5 6 7 8 10 12

-----
Process exited after 8.964 seconds with return value 0
请按任意键继续. . .
```

3、 删除链表所有值为 x 的结点

包含头部

```
请输入若干整数序列:
1 2 1 1 4 5 6 1 1 9 10 0
List is:
1 2 1 1 4 5 6 1 1 9
10
请输入要删除的值:
1
List is:
2 4 5 6 9 10

-----
Process exited after 10.11 seconds with return value 0
请按任意键继续. . .
```

不含头部

```
D:\gitworkspace\fresh\algorithm\简单递归\slinklist2.exe
请输入若干整数序列:
1 3 4 6 8 9 2 8 2 9 20 102 1 2 1 2
0
List is:
1 3 4 6 8 9 2 8 2 9
20 102 1 2 1 2
请输入要删除的值:
2
List is:
1 3 4 6 8 9 8 9 20 102
1 1

-----
Process exited after 15.99 seconds with return value 0
请按任意键继续. . .
```

七、思考讨论题或体会或对改进实验的建议

通过这次实验我学会了不带头结点的链表数据插入和删除等操作，掌握了数据结构中指针的用法和扩展。

八、参考资料

·
数据结构 C 语言版