

# 实验一：线性结构及其应用

## 一. 实验目的

掌握链表的存储结构和基本操作，重点巩固和体会链表的插入、合并及应用。

## 二. 实验内容

3 个班级一起上数据结构课程，期末考试后，需要分别录入各班每个人的成绩，并按成绩降序存储。这个成绩管理系统需要的功能有：

(1) 班级成绩录入——链表的建立

操作 1：初始化空表

操作 2：插入一个人的成绩

操作 3：查找/定位，获取链表中第 index 个节点的值。如果索引无效，则返回-1。

操作 4：删除一个人的成绩，如果索引 index 有效，则删除链表中的第 index 个节点。

(2) 输出——链表的遍历 L1; L2; L3

(3) 综合排名——链表的合并 L1+L2+L3→L

## 三. 实验要求

- 1) 基于参考代码 main.c 操作；
- 2) 用链表实现成绩管理系统所有的功能；
- 3) 每个班的成绩分别存在一个链表中，即一共需要三个链表；
- 4) 成绩需要降序存储。
- 5) 要求在实验课上完成用链表实现成绩管理系统的 (1) 和 (2) (见实验内容)，在课下完成其他功能并撰写实验报告

## 四. 参考代码

```
#include <stdio.h>

typedef struct node{
    int performance;
    struct node *next;
}MyLinkedList;

/** Initialize your data structure here. */
MyLinkedList* myLinkedListCreate() {

}

/** 查询第 index 个学生的成绩 */
int myLinkedListGet(MyLinkedList* obj, int index) {

}

/** 按照降序插入学生的成绩 */
void myLinkedListAdd(MyLinkedList* obj, int performance) {

}

/** 删除第 index 个学生 */
void myLinkedListDelete(MyLinkedList* obj, int index) {

}

/** 输出该班级的成绩 */
void myLinkedListOutput(MyLinkedList* obj) {

}

/** 将所有班级的成绩合并降序输出 */
void merge(MyLinkedList* obj[]) {

}

int main(){
    MyLinkedList* obj[3];
```

```

while (1){
    printf("1.insert 2.delete 3.search 4.output 5.merge\n");
    int op;
    scanf("%d", &op);
    int class;
    int index;
    int performance;
    switch(op){
        case 1:
            printf("input class\n");
            scanf("%d", &class);
            printf("input performance\n");
            scanf("%d", &performance);
            myLinkedListAdd(obj[class-1], performance);
            break;
        case 2:
            printf("input class\n");
            scanf("%d", &class);
            printf("input index\n");
            scanf("%d", &index);
            myLinkedListDelete(obj[class-1], index);
            break;
        case 3:
            printf("input class\n");
            scanf("%d", &class);
            printf("input index\n");
            scanf("%d", &index);
            printf("%d\n", myLinkedListGet(obj[class-1], index));
            break;
        case 4:
            myLinkedListOutput(obj[0]);
            myLinkedListOutput(obj[1]);
            myLinkedListOutput(obj[2]);
            break;
        case 5:
            merge(obj);
            break;
        default:

```

```

        return 0;
    }
    printf("#####\n");
}
}

```

## 五. 范例说明

创建班级输入范例:

```

input class
1
input performance
80
input class
2
input performance
70
input class
1
input performance
90
...

```

班级成绩输出范例:

```

1.insert 2.delete 3.search 4.output 5.merge
4
90 80 78 77 75...
88 87 65 64 63...
97 95 76 65 54...

```

查询

```

1.insert 2.delete 3.search 4.output 5.merge
3
input class
2
input index
4
64

```

删除

```

1.insert 2.delete 3.search 4.output 5.merge
2
input class

```

3

input index

4

### 实验报告提交说明:

1. 电子版实验报告提交的截止时间为: 2019 年 4 月 20 日晚上 10 点之前;
2. 电子版实验报告提交到该邮箱: [hitsz\\_ds\\_2019@163.com](mailto:hitsz_ds_2019@163.com);
3. 请把电子版实验报告及源代码打包成一个压缩包, 命名格式如下:
  - a) 实验报告: “学号\_姓名\_实验 1”
  - b) 压缩包: “学号\_姓名\_实验 1”
  - c) 邮件标题: “学号\_姓名\_实验 1”
4. 纸质版实验报告请在 4 月 21 日晚上 9 点之前提交到 G701。