

重庆师范大学

《数据结构课程设计》

课程名称： 数据结构

题 目： 学生成绩管理系统

学 院： 计算机与信息科学学院

专业年级： 计科 192 (3+2) 班

小组组长： 何 倩

小组成员： 刘焦、杨冰玲、刘雪静、王世莉

指导教师： 张万里 职称： 讲师

2021 年 12 月 31 日

目录

1	准备工作	- 2 -
1.1	需求分析	- 2 -
1.1.1	需要实现的功能	- 2 -
1.1.2	系统性能要求	- 2 -
1.1.3	系统功能分析	- 2 -
1.1.4	系统功能模块	- 2 -
1.1.5	模块框架图	- 3 -
1.2	任务分工	- 3 -
2	内容介绍	- 4 -
2.1	问题描述	- 4 -
2.2	基本要求	- 4 -
2.3	算法思路	- 19 -
3	程序概要	- 19 -
3.1	程序模块	- 19 -
3.2	主程序	- 20 -
3.3	运行结果	- 20 -
3.3.1	录入成绩	- 20 -
3.3.2	删除成绩	- 22 -
3.3.3	修改成绩	- 23 -
3.3.4	成绩查询	- 24 -
3.3.5	成绩计算并排名	- 26 -
3.3.6	显示成绩信息	- 26 -
3.3.7	保存信息并退出系统	- 27 -
4	总结	- 27 -

学生成绩管理系统

1 准备工作

1.1 需求分析

1.1.1 需要实现的功能

1. 能对学生各科的成绩进行录入
2. 能对学生的成绩进行删除
3. 能对学生的成绩进行修改
4. 能对学生的成绩进行查询
5. 能对学的成绩计算并排名
6. 能显示学生的成绩相关信息

1.1.2 系统性能要求

1. 系统安全、可靠
2. 功能齐全
3. 操作方便、界面友好直观
4. 利于维护和扩充

1.1.3 系统功能分析

1. 信息维护：信息维护包括对学生信息的录入操作以及数据表中的信息进行浏览
2. 系统功能：可以对学生成绩数据表中的信息进行浏览
3. 系统查询：可以对学生的信息进行查询

1.1.4 系统功能模块

学生成绩管理系统：成绩录入、删除、修改、查询、计算以及排名、显示信息、保存信息并

退出系统。

1. 1. 5. 模块框架图

学生成绩管理系统主要包含六个功能下面对整体项目的模块框架图，如图 1 所示。

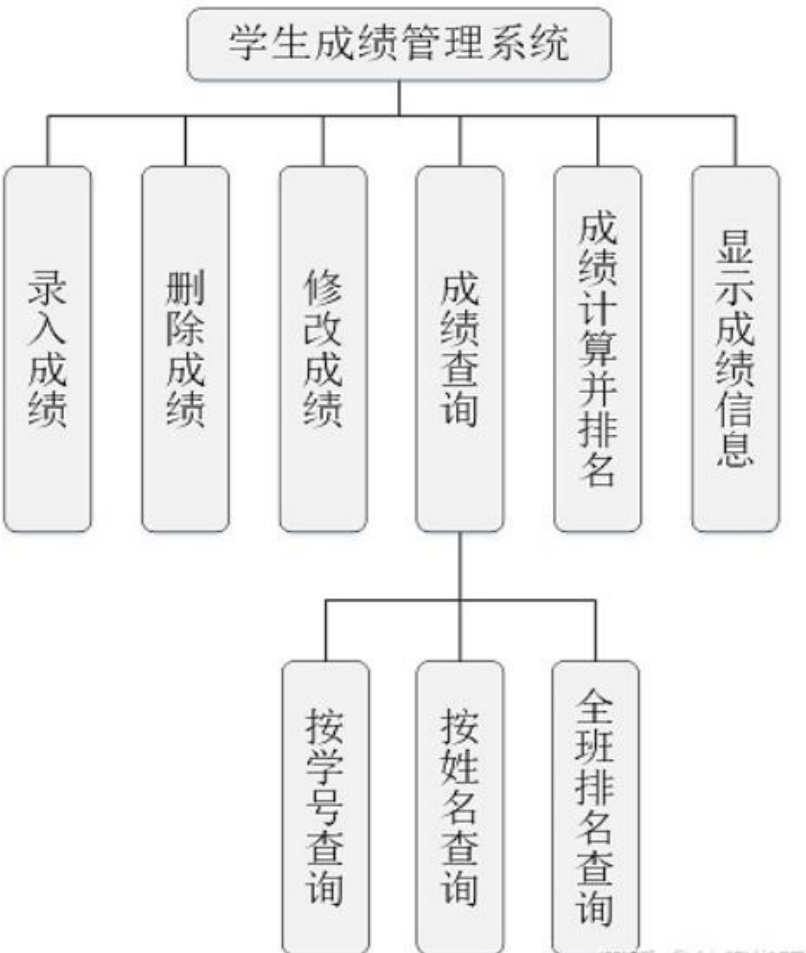


图 1.模块框架图

1. 2 任务分工

如下表：

姓名	分工
何倩	实现系统中的录入和删除功能，且参与了实训报告的撰写。
刘焦	实现了系统中的计算成绩功能，制作了 ppt.
王世莉	实现了系统中修改成绩的功能且参与了实训报告的撰写。
杨冰玲	实现了系统中以各种方式的查询成绩的功能。

刘雪静	实现了系统中以各种方式的查询成绩的功能。
-----	----------------------

表 1. 任务分工表

2 内容介绍

2.1 问题描述

实现学生成绩管理系统首先需要明确用户对学生成绩管理系统的需求，比如录入成绩、删除成绩、修改成绩、成绩查询（按学号查询、按姓名查询、全班排名查询）、成绩计算并排名、成绩的显示、保存信息并退出系统等功能的实现。

2.2 基本要求

（1）学生成绩管理系统的信息包括：录入成绩、删除成绩、修改成绩、成绩查询（按学号查询、按姓名查询、全班排名查询）、成绩计算并排名、成绩的显示等功能的实现。

（2）系统能实现的操作功能如下：

学生成绩系统：

```
32
33  switch(flag)
34
35  {
36
37  case 1: printf("请输入您要录入的人数: ");|
38
39  scanf("%d",&n);
40
41  lr(n);
42
43  count += n;break;
44
45  case 2: count=sc(count);break;
46
47  case 3: xg(count);break;
48
49  case 4: cx(count);break;
50
51  case 5: jspm(count);break;
52
53  case 6: xs(count);break;
54
55  case 0: bccj();break;
56
57  default:printf("输入错误, 请重试! \n");getch();
58
59  }
60
```

图 2.学生成绩系统图

学生成绩系统主流程图：

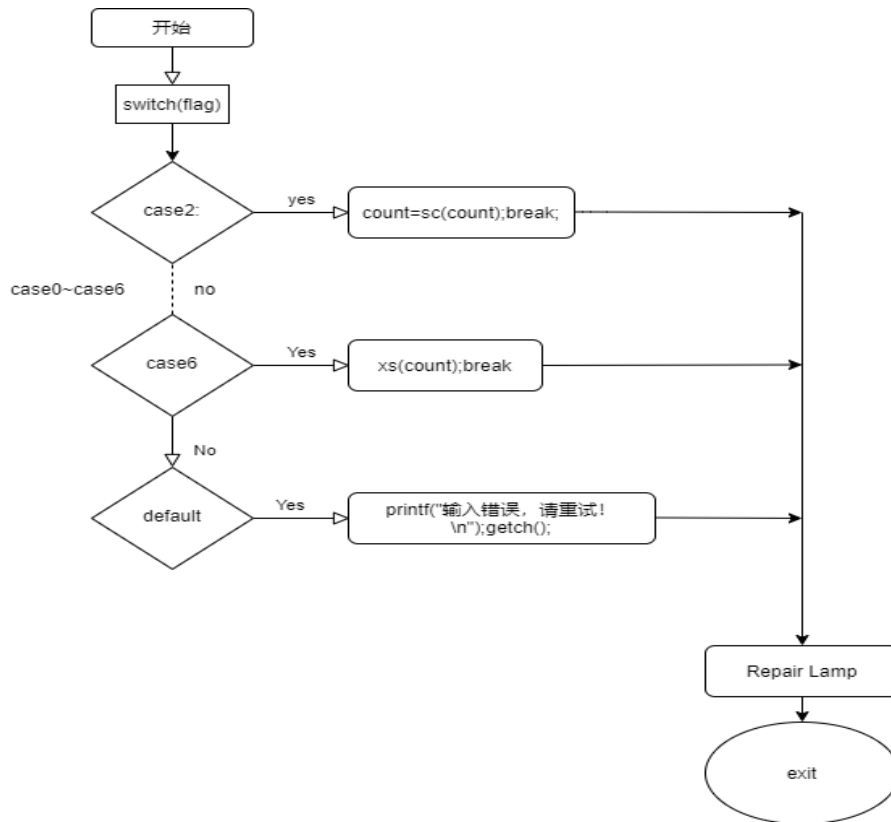


图 3.学生成绩系统主流程图

- ① 录入成绩： 将学生成绩管理系统的相关数据录入。

```

207 //1录入成绩
208
209 void lr(int n)
210
211 {
212
213     struct score stu[100];
214
215     FILE *fp;
216
217     fp=fopen("cj.txt","a+");
218
219     if(fp==NULL)
220
221     {
222
223         printf("打开文件失败,请重试! \n");
224
225         getch();
226
227         fclose(fp);
228
229         exit(0);
230
231     }
232

```

图 4.成绩录入代码图

```

for(i=0;i<n;i++)
{
printf("----第%d个学生----\n", i+1);
printf("学号:");
scanf("%s",&stu[i].snum);
printf("姓名: ");
scanf("%s",&stu[i].sname);
printf("C++成绩: ");
scanf("%d",&stu[i].chinese);
printf("JAVA成绩: ");
scanf("%d",&stu[i].math);
printf("Python成绩: ");
scanf("%d",&stu[i].english);
stu[i].sumcj=0;
stu[i].rank=0;
// 输入成绩

```

图 5.成绩录入代码图

流程图:

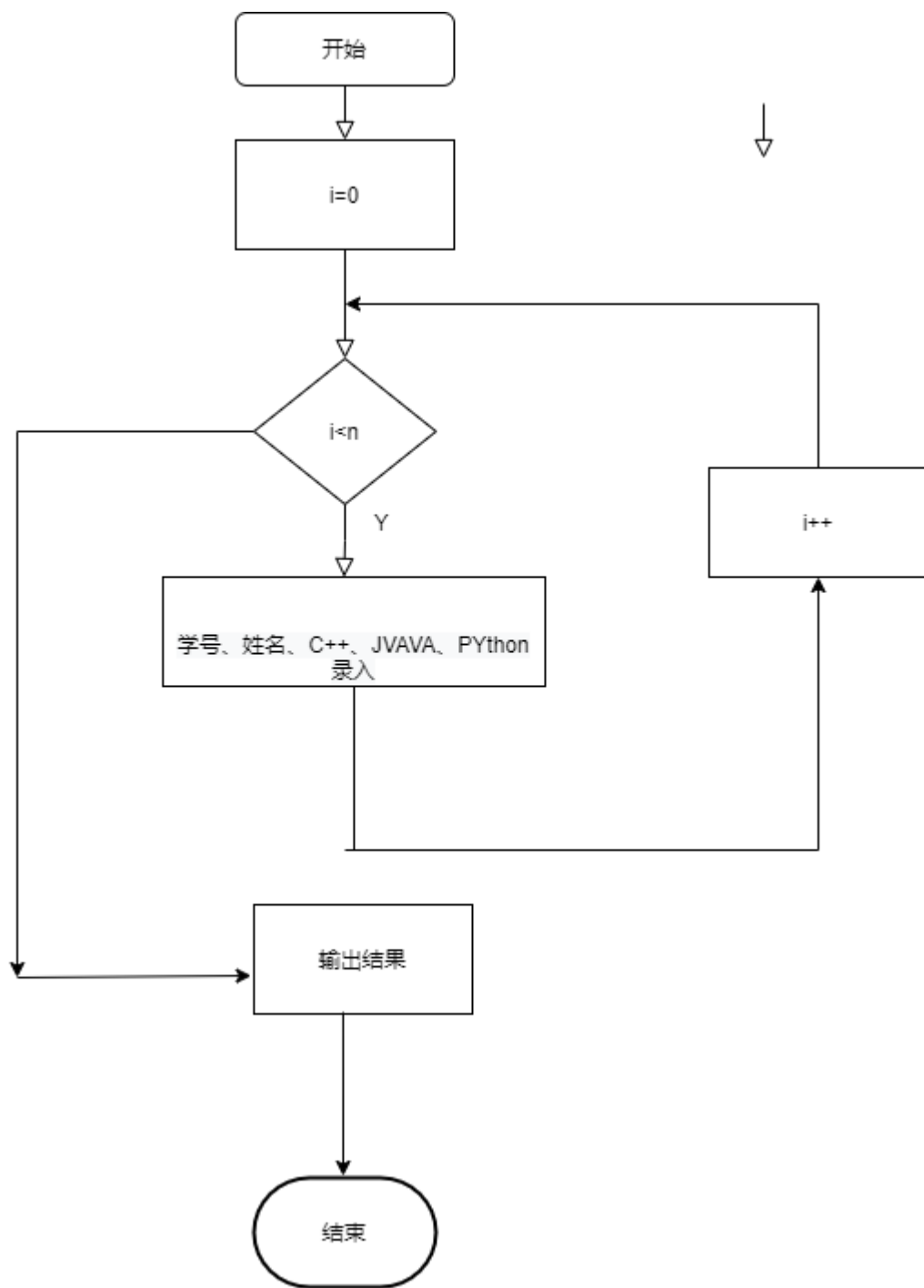


图 6.成绩录入流程图

② 删除成绩：对相关信息进行删除处理（根据学号进行查询然后做出删除处理）。

```

//2删除成绩

int sc(int n)
{
    struct score stu[100];
    struct score sub[99]; //删除后的结构体数组
    char stnum[9]; //学号
    int find=0;
    printf("请输入你要删除的学生学号:");
    scanf("%s", &stnum);
    FILE *fp;
    fp=fopen("cj.txt", "r");

```

图 7.删除成绩代码图

删除时文件为空或者不存在流程图：

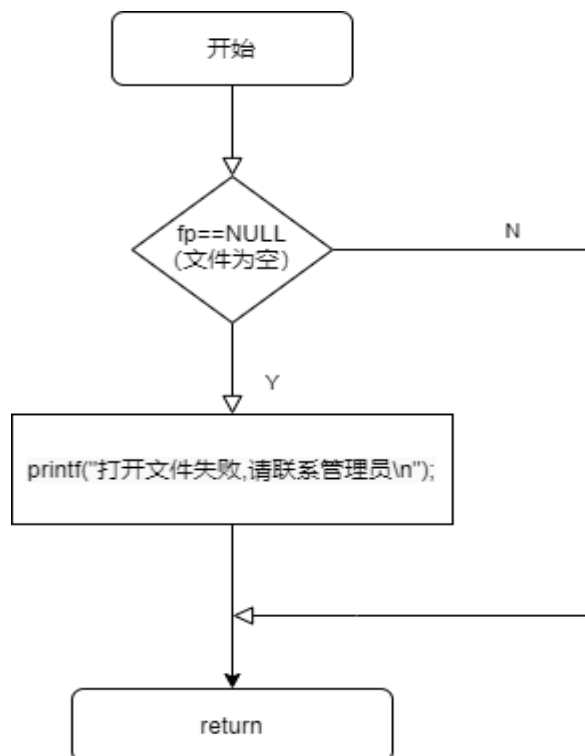


图 8.删除时文件为空或者不存在流程图

文件存在执行删除操作流程图：

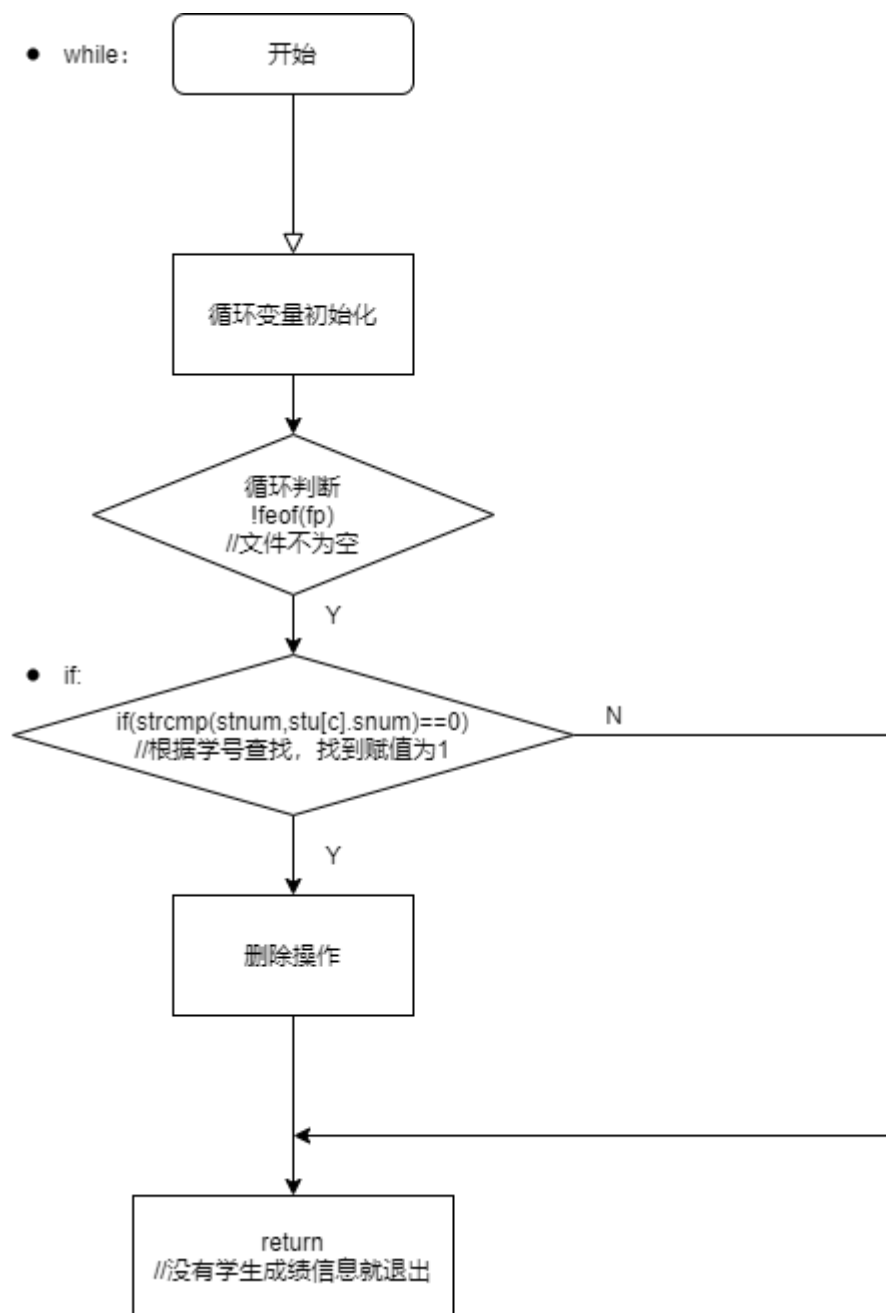


图 9.文件存在执行删除操作流程图

③ 修改成绩： 输入要修改的学生学号然后对学生的 C ++成绩、JAVA 成绩、Python 成绩进行修改。

```
//3 修改成绩  
  
int xg(int n)  
{  
    struct score stu[100];  
    FILE *fp;  
    char stnum[9];  
    int find=0;  
    printf("请输入要修改的学生学号:");  
    scanf("%s",stnum);  
    fp=fopen("cj.txt","r");  
    if(fp==NULL)  
    {
```

图 10.修改成绩代码图

修改成绩流程图与删除成绩流程图类似，如下：

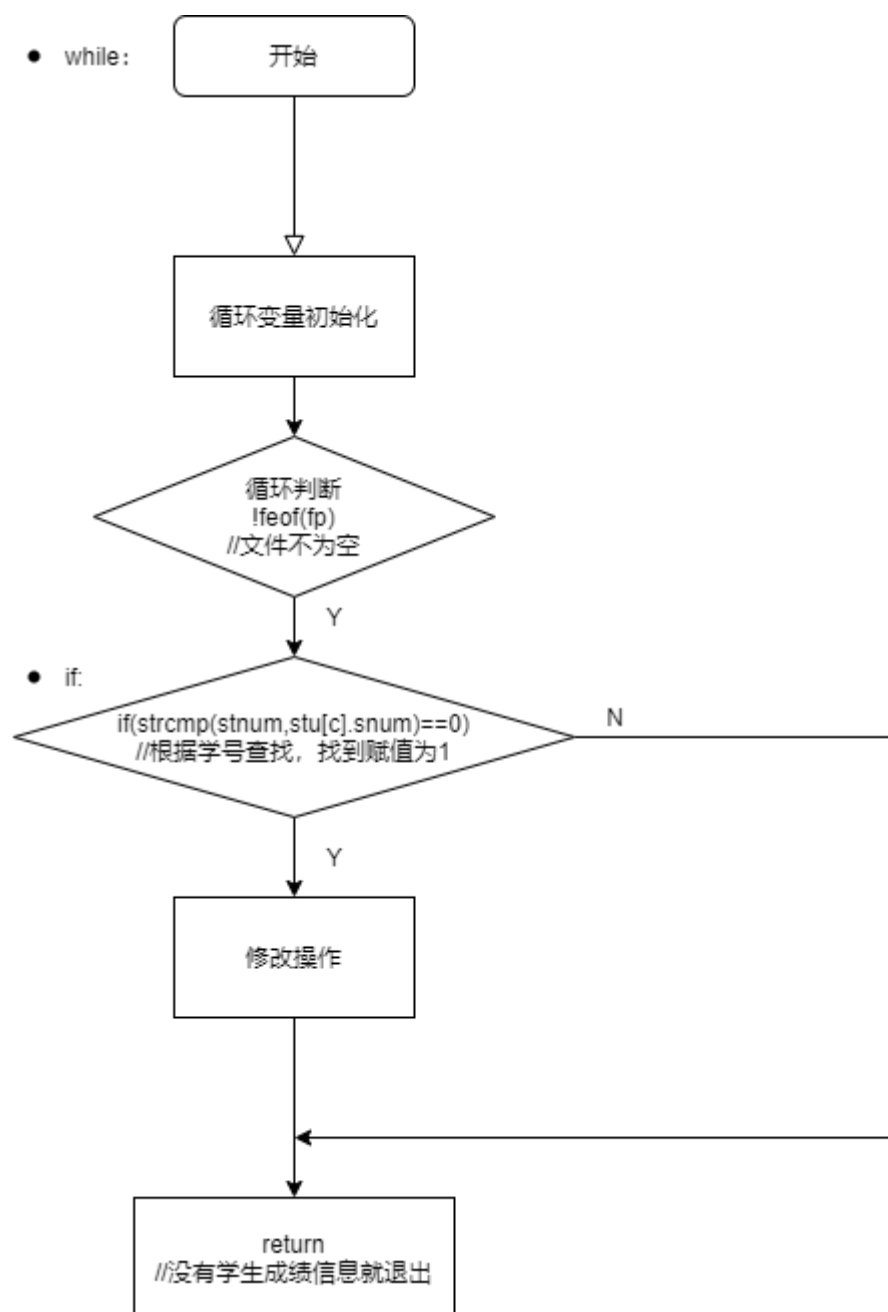


图 11.修改成绩流程图

④成绩查询： 成绩查询可以根据按学号查询、按姓名查询、全班排名查询三种方式进行查询。

```

//4成绩查询
void cx(int n) //n全班人数
{
    int k;
    do
    {
        showcsmenu();
        scanf("%d",&k);
        switch(k)
        {
            case 1:xhcx(n);break;//学号查询
            case 2:xmccx(n);break;//姓名查询
            case 3:allrankcx(n);break;//全班排名查询
            case 0:break;
            default:printf("输入错误,请重新选择\n");getch();
        }
    } while(k != 0);
}

```

图 12.查询成绩代码图

学生成绩信息查询流程图：

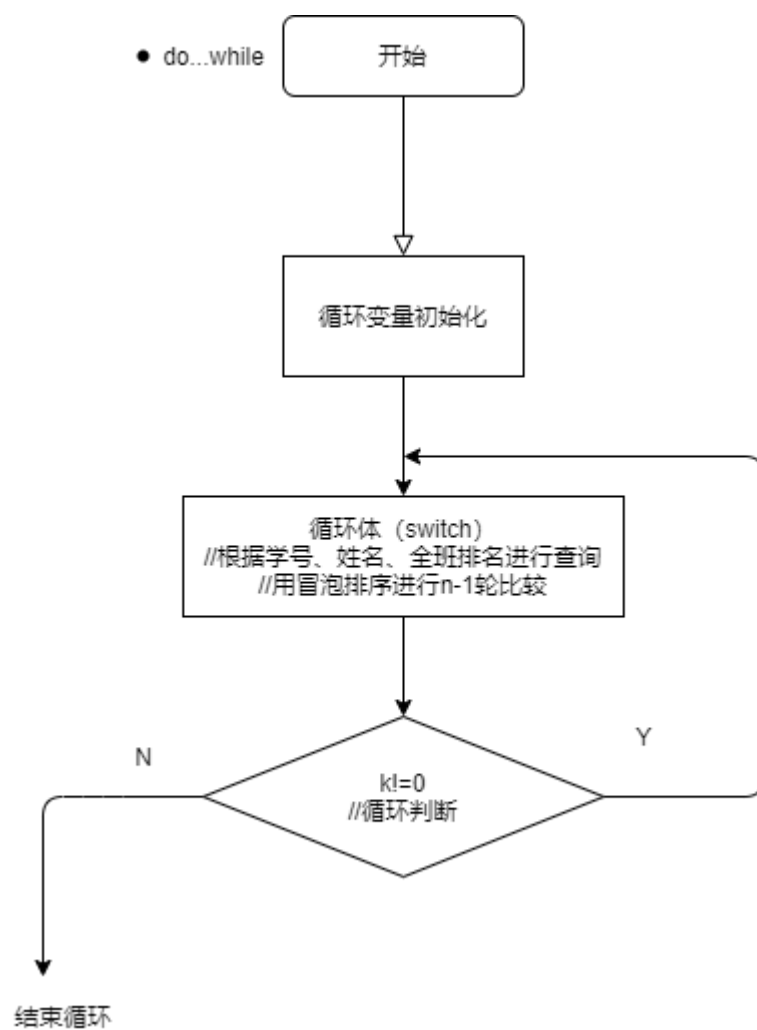


图 13.学生成绩信息查询流程图

⑤ 成绩计算并排名：对成绩进行总分计算和排名

```

905 //计算排名并存储
906
907 for(i=0;i<n;i++)
908 {
909
910     temp=1; //第1名
911
912     for(j=0;j<n;j++){
913
914         if(stu[i].sumcj<stu[j].sumcj) temp++; //如果较小, 排名下降1名
915
916     }
917
918     stu[i].rank=temp;
919
920 }
921
922 //更新记录
923
924 fp=fopen("cj.txt","w");
925
926 for(i=0;i<n;i++)
927 {
928
929     fprintf(fp,"%s %s %d %d %d %d\n",stu[i].snum,stu[i].sname,stu[i].chinese,stu[i].math,stu[i].english,stu[i].sumcj,stu[i].rank);
930
931 }
932
933 fclose(fp);
934
935 printf("数据重新计算和排名成功,请按任意键返回主菜单。 \n");
936
937 getch();
938
939 return;
940
941 }
942
943

```

图 14.成绩计算代码图

```

//计算排名并存储

for(i=0;i<n;i++)

{

temp=1; //第1名

for(j=0;j<n;j++){

if(stu[i].sumcj<stu[j].sumcj) temp++; //如果较小, 排名下降1名

}

stu[i].rank=temp;

}

```

图 15.成绩排名代码图

计算排名存储流程图:

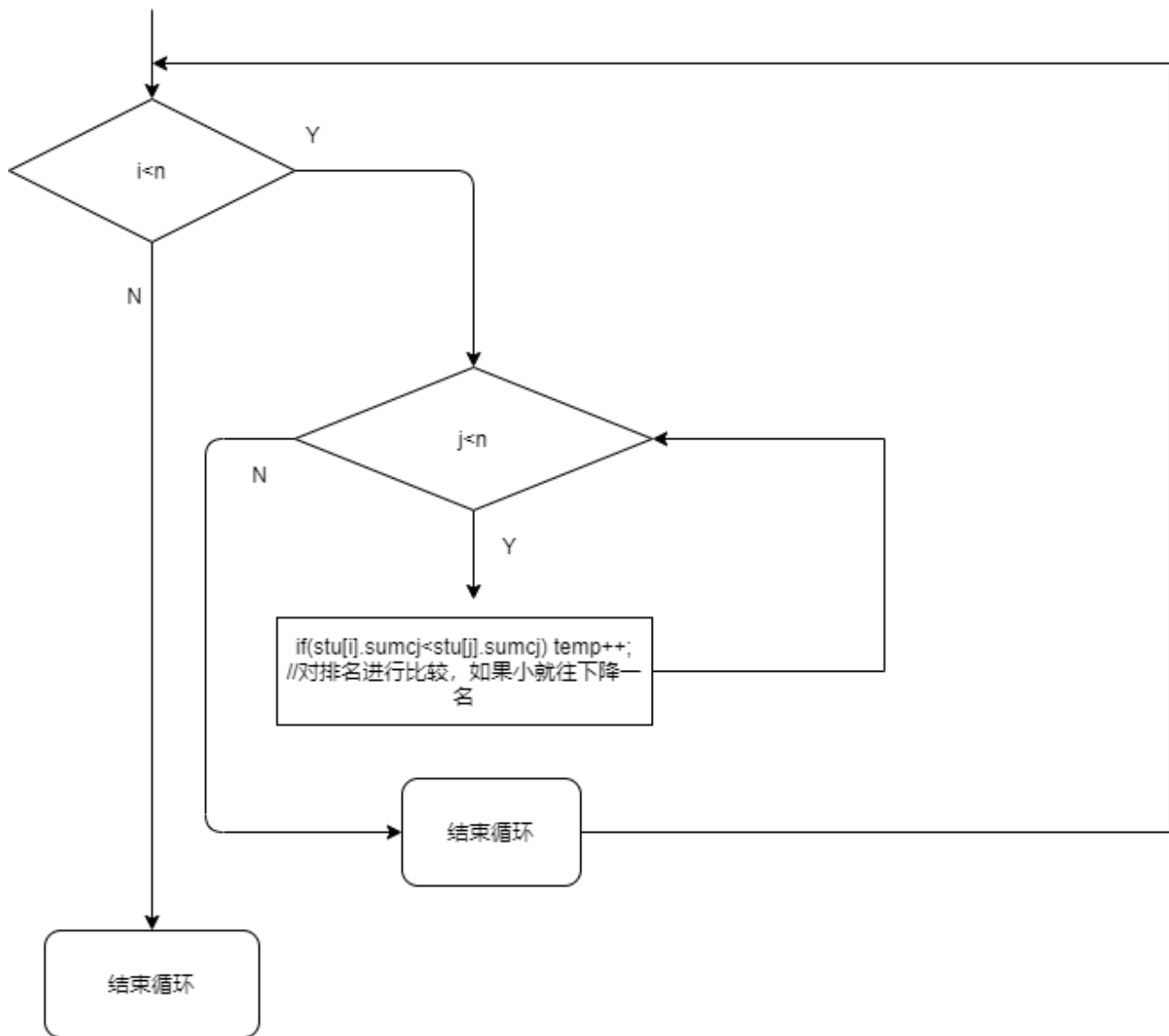


图 16.计算排名存储流程图

⑥显示成绩信息：对学生的姓名、学号、成绩等相关信息检查查询。

```

945 //6显示成绩信息
946
947 void xs(int n)
948
949 {
950
951     struct score stu[100];
952
953     int i;
954
955     printf("成绩信息如下: \n");
956
957     FILE *fp;
958
959     fp=fopen("cj.txt","r");
960
961     if(fp==NULL)
962
963     {
964
965         printf("打开文件失败,请重试! \n");
966
967         getch();
968
969         fclose(fp);
970
971         exit(0);
972
973     }
974
975     printf("学号\t姓名\tC++\tJAVA\tPython\t总分\t名次\n");
976

```

图 17.显示成绩信息代码图

```

89 if(n!=0){//人数不为0
90
91     for(i=0;i<c;i++)
92
93     {
94
95         printf("%s\t%s\t%d\t%d\t%d\t%d\t%d\t%d\n",stu[i].snum,stu[i].sname,stu[i].chinese,stu[i].math,stu[i].english,stu[i].sumcj,stu[i].rank);
96
97     }
98
99 }
00
01 fclose(fp);
02
03 printf("显示完毕 ,请按任意键返回\n");
04
05 getch();
06
07 return;
08
09 }

```

图 18.显示成绩信息代码图

显示成绩信息流程图：

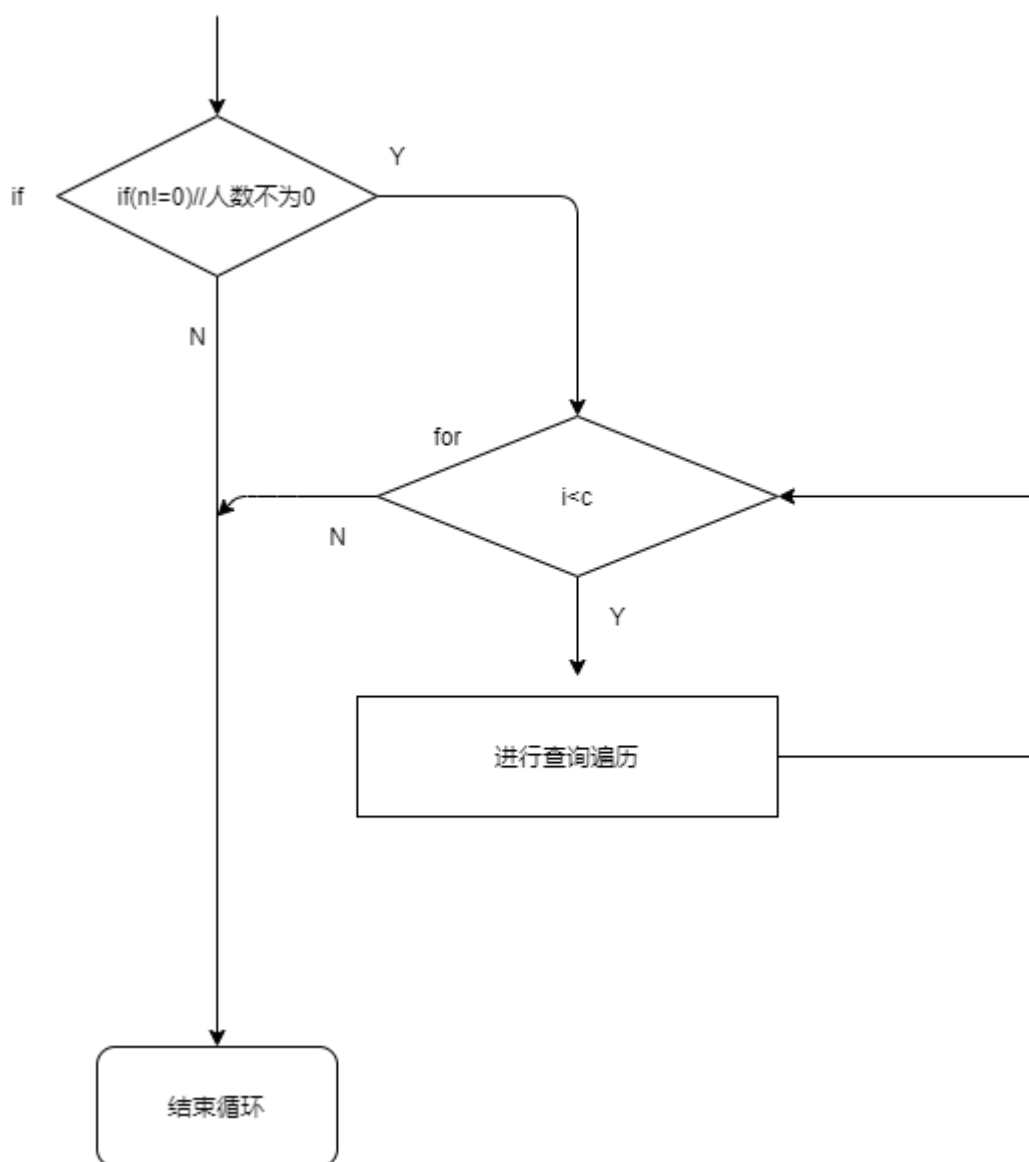


图 19.显示成绩信息流程图

⑦保存信息并退出系统：对信息进行保存，退出程序。

```

L1 //0保存信息并退出系统
L2
L3 void bccj()
L4
L5 {
L6
L7 printf("感谢您的使用, 欢迎下次使用!\n");
L8
L9 getch();
L10
L11 return;
L12
L13 }

```

日志 调试 搜索结果 关闭

图 20.保存信息并退出代码图

2.3 算法思路

用 C++进行编写，根据学生成绩管理系统的相关要求以及要实现的功能，先对相关函数进行函数声明，然后利用数组、冒泡排序算法、选择、循环、函数、指针、结构体、链表，还有包括整个框架的构思，图、栈、队列等等的运用，最终构成了整个程序。

3 程序概要

3.1 程序模块

程序的模块有以下几点

录入功能：学生成绩的录入包括：C++成绩，JAVA 成绩，Python 成绩。

删除功能：删除指定学生信息。

修改功能：修改指定学生的成绩。

查看功能：查询学生成绩的信息，包括：姓名、学号、成绩、排名、总分等。

计算功能：学生成绩的总分计算和排名。

退出功能：退出系统。

3.2 主程序

程序 1. 学生成绩管理系统程序

3.3 运行结果

3.3.1 录入成绩

将学生成绩管理系统的相关数据录入。

```

1*-----*
1 选择操作<0-6>
2 1
2 请输入您要录入的人数: 6
2 请输入全部学生的学号, 姓名, C++成绩, JAVA成绩, Python成绩;
2 ----第1个学生----
2 学号:2019001
2 姓名: 何倩
2 C++成绩: 66
2 JAVA成绩: 79
2 Python成绩: 91
2 ----第2个学生----
2 学号:2019002
2 姓名: 王世莉
2 C++成绩: 69
2 JAVA成绩: 82
2 Python成绩: 73
2 ----第3个学生----
2 学号:2019003
2 姓名: 杨冰玲
2 C++成绩: 61
2 JAVA成绩: 96
2 Python成绩: 86
2 ----第4个学生----
2 学号:2019004
2 姓名: 刘焦
2 C++成绩: 76
2 JAVA成绩: 79
2 Python成绩: 99
2 ----第5个学生----
2 学号:20190005
2 姓名: 刘雪静
2 C++成绩: 99
2 JAVA成绩: 86
2 Python成绩: 76
2 ----第6个学生----
2 学号:2019006
2 姓名: 小六
2 C++成绩: 63
2 JAVA成绩: 62
2 Python成绩: 71
2 添加学生成功, 请按任意键返回

```

图 21.学生成绩录入图



图 22.学生成绩录入图

3.3.2 删除成绩

对相关信息进行删除处理。



图 23.删除成绩操作图

删除后:

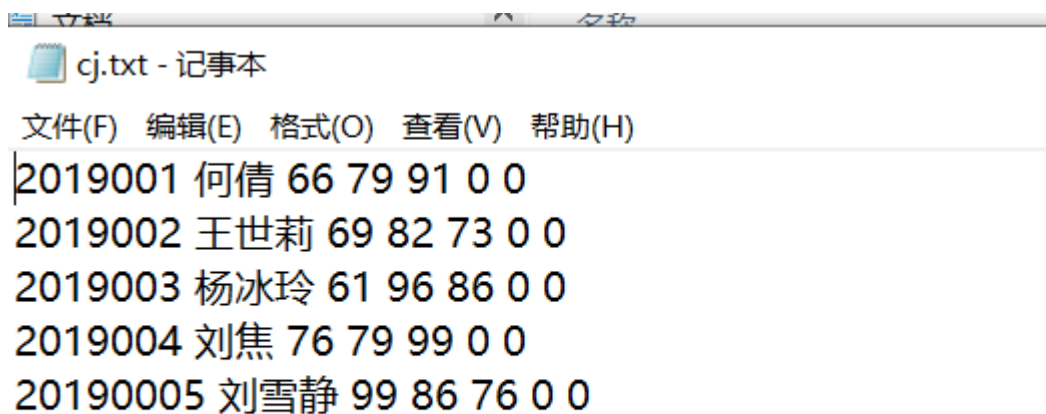


图 24.删除后显示图

3.3.3 修改成绩

输入要修改的学生学号然后对学生的数据结构成绩、网络安全成绩、操作系统成绩进行修改。

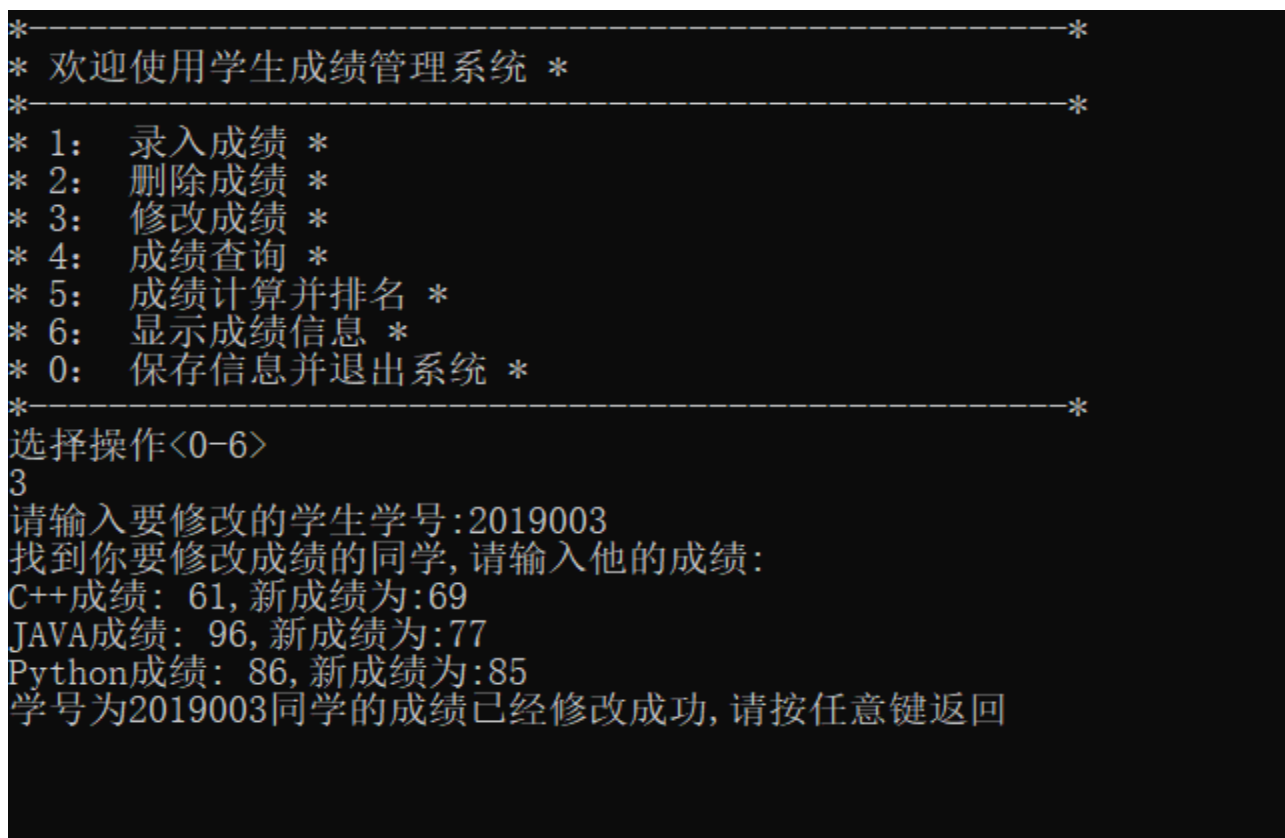


图 25.修改成绩操作图

修改后:



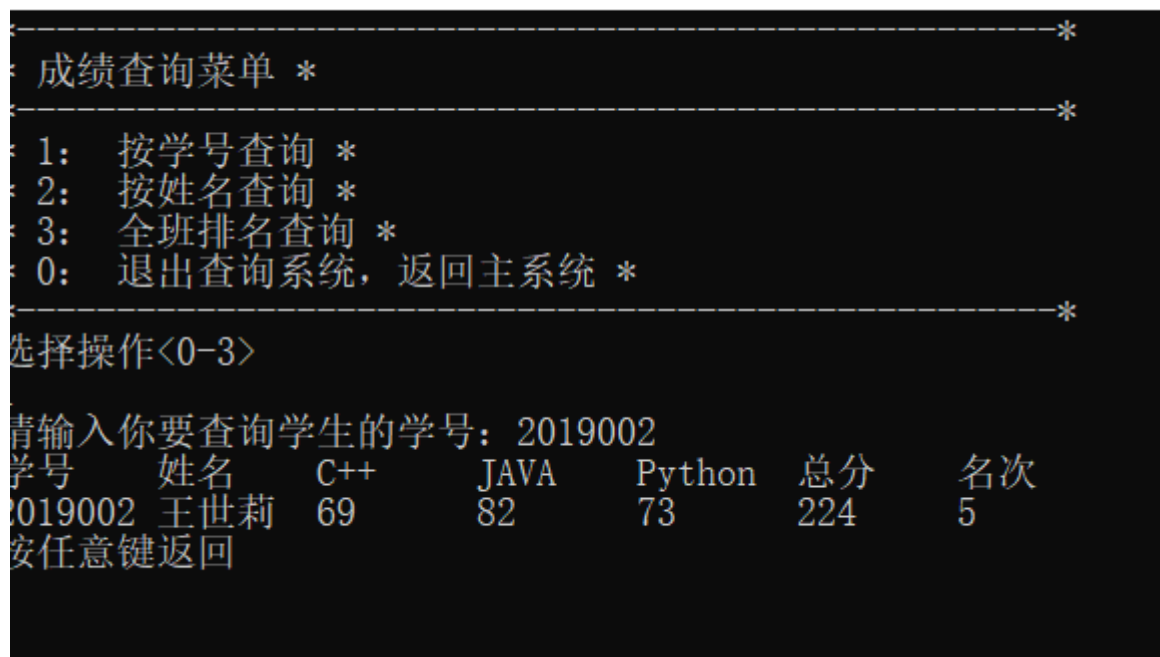
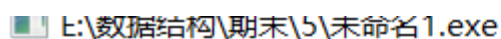
```
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
2019001 何倩 66 79 91 0 0
2019002 王世莉 69 82 73 0 0
2019003 杨冰玲 69 77 85 0 0
2019004 刘焦 76 79 99 0 0
20190005 刘雪静 99 86 76 0 0
```

图 26.修改后显示图

3.3.4 成绩查询

成绩查询可以根据按学号查询、按姓名查询、全班排名查询三种方式进行查询。

a. 按学号查询



```
-----*
成绩查询菜单 *
-----*
1: 按学号查询 *
2: 按姓名查询 *
3: 全班排名查询 *
0: 退出查询系统, 返回主系统 *
-----*
选择操作<0-3>

请输入你要查询学生的学号: 2019002
学号 姓名 C++ JAVA Python 总分 名次
2019002 王世莉 69 82 73 224 5
按任意键返回
```

图 27.按学号查询成绩图

b. 按姓名查询

E:\数据结构\期末\5\未命名1.exe

```
*-----*
* 成绩查询菜单 *
*-----*
* 1: 按学号查询 *
* 2: 按姓名查询 *
* 3: 全班排名查询 *
* 0: 退出查询系统, 返回主系统 *
*-----*
选择操作<0-3>
2
请输入你要查询学生的姓名: 何倩
学号    姓名    C++    JAVA    Python    总分    名次
2019001 何倩    66     79     91     236     3
按任意键返回
```

图 28.按姓名查询成绩图

c. 全班排名查询(排名查询前要先进行分数计算并排名)

E:\数据结构\期末\5\未命名1.exe

```
*-----*
* 成绩查询菜单 *
*-----*
* 1: 按学号查询 *
* 2: 按姓名查询 *
* 3: 全班排名查询 *
* 0: 退出查询系统, 返回主系统 *
*-----*
选择操作<0-3>
3
学号    姓名    C++    JAVA    Python    总分    名次
2019002 王世莉  69     82     73     224     5
2019003 杨冰玲  69     77     85     231     4
2019001 何倩    66     79     91     236     3
2019004 刘焦    76     79     99     254     2
2019005 刘雪静  99     86     76     261     1
按任意键返回
```

图 29.按排名查询成绩图

3.3.5 成绩计算并排名

对成绩进行总分计算和排名

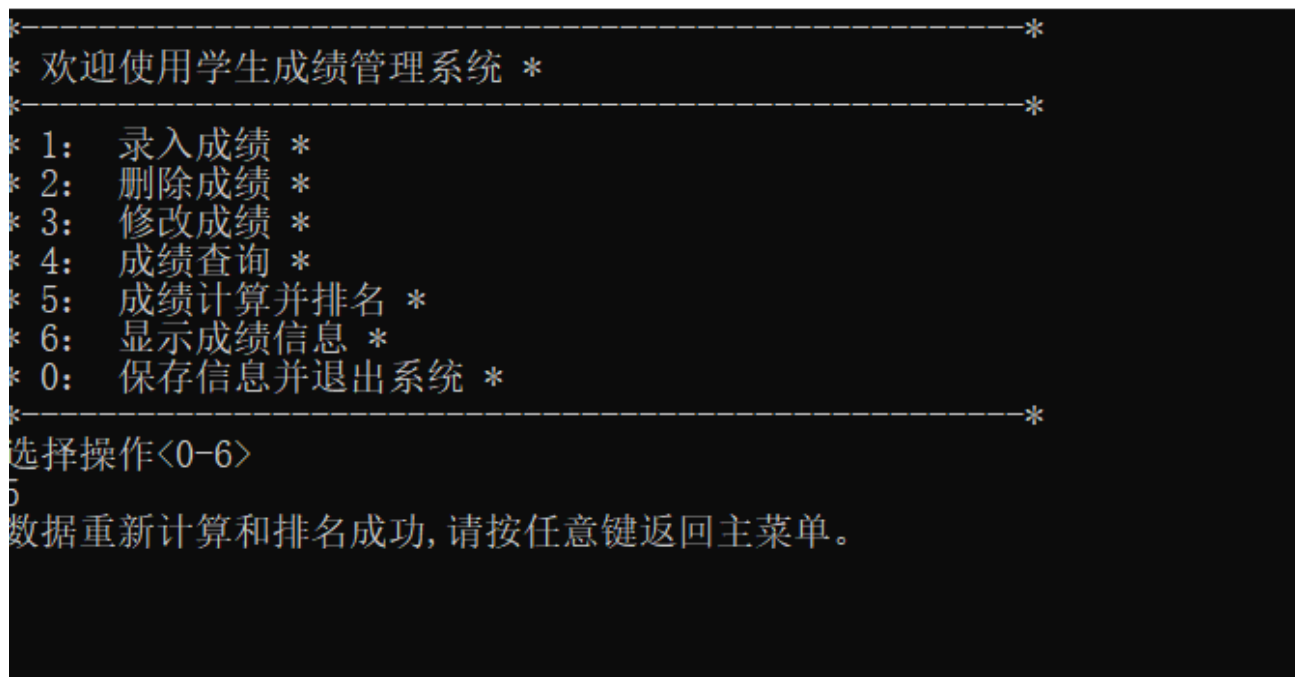


图 30.计算总分并排名图

3.3.6 显示成绩信息

对学生的姓名、学号、成绩等相关信息检查查询。

```
-----*
欢迎使用学生成绩管理系统 *
-----*
1:  录入成绩 *
2:  删除成绩 *
3:  修改成绩 *
4:  成绩查询 *
5:  成绩计算并排名 *
6:  显示成绩信息 *
0:  保存信息并退出系统 *
-----*

选择操作<0-6>

成绩信息如下:
学号  姓名  C++  JAVA  Python  总分  名次
019001 何倩  66   79   91     236   3
019002 王世莉 69   82   73     224   5
019003 杨冰玲 69   77   85     231   4
019004 刘焦   76   79   99     254   2
019005 刘雪静 99   86   76     261   1
显示完毕,请按任意键返回
```

图 31.学生信息显示图

3.3.7 保存信息并退出系统

对之前修改和加入的学生信息进行保存，退出程序，并且相关内容会存储到链接的记事本中，这样也有利于后期的查看。

4 总结

通过这次实验我们更好的掌握和巩固数据结构的相关知识，C++语言编写的相关知识和技巧，特别是数组、冒泡排序算法、选择、循环、函数、指针、结构体、链表等功能。通过该课程的设计与操作实践，我们小组最终选用了学生成绩管理系统的项目进行实施。要对学生数据管理系统的学生成绩录入、学生成绩的删除、学生成绩的修改、学生成绩的查询（按学号查询、按姓名查询、全班排名查询）、学生成绩的计算和排名、学生成绩信息的显示这些

功能的实现最终构成一个比较完整的学生成绩管理系统。