



**程序设计课程设计报告**

题 目\_\_\_学生奖学金管理系统\_

学 院\_\_\_计算机学院\_\_\_\_\_\_\_\_\_

专 业\_\_\_计算机类\_\_\_\_\_\_\_\_\_\_\_

年级班别\_\_\_计算机2020级（12）

学 号\_\_\_3220005188\_\_\_\_\_\_\_\_\_

学生姓名\_\_\_聂芳琪\_\_\_\_\_\_\_\_\_\_\_\_\_

指导教师\_\_\_曾 颖\_\_\_\_\_\_\_\_\_\_\_\_\_\_

成 绩 优秀

|  |  |
| --- | --- |
| 程序功能完成情况 |  |
| 测试用例全面情况 |  |
| 报告格式是否与要求相符 |  |
| 报告内容是否准确全面 |  |

2021年1月

目录

[**二．课程设计内容和要求** 1](#_Toc79067373)

[一、课程设计内容： 1](#_Toc79067374)

[一、 设计内容 1](#_Toc79067375)

[二、 设计要求 1](#_Toc79067376)

[二．课程设计要求 3](#_Toc79067377)

[**三．** **需求分析** 4](#_Toc79067378)

[**3.2功能需求** 4](#_Toc79067379)

[3.2.1功能划分： 4](#_Toc79067380)

[3.2.2功能需求的描述： 4](#_Toc79067381)

[**3.4界面需求** 5](#_Toc79067382)

[**四．概要设计：在需求分析的基础上，确定系统总体框架(系统功能结构图)。** 6](#_Toc79067383)

[**4.1系统总体框架设计** 6](#_Toc79067384)

[**4.2系统流程图：** 7](#_Toc79067385)

[**4.3模块设计** 7](#_Toc79067386)

[**1）主函数模块：** 8](#_Toc79067387)

[**2）输入记录模块：** 8](#_Toc79067388)

[**3）查询记录模块：** 8](#_Toc79067389)

[**4）更新记录模块：** 9](#_Toc79067390)

[**5）统计排序模块：** 9](#_Toc79067391)

[**6）输出记录模块：** 9](#_Toc79067392)

[**7）文件操作：** 10](#_Toc79067393)

[**4.4函数功能简要概述：** 10](#_Toc79067394)

[**4.5公用数据结构设计及用法，全局变量说明** 11](#_Toc79067395)

[1）结构体 11](#_Toc79067396)

[2）全局变量 12](#_Toc79067397)

[**五、** **详细设计** 13](#_Toc79067398)

[**5.1主要功能模块的算法设计思路以及对应的功能模块的流程图（或N-S图）** 13](#_Toc79067399)

[**六、** **实验结果** 30](#_Toc79067400)

[**6.1各模块输入输出结果：** 30](#_Toc79067401)

[**6.2出错处理情况** 39](#_Toc79067402)

[**七、** **体会总结** 42](#_Toc79067403)

[**（课程设计完成了哪些功能，有什么扩展功能有什么地方需要改进，通过课程设计有何收获，调试程序的体会等）** 42](#_Toc79067404)

[1. 存在的问题及解决方法 42](#_Toc79067405)

[2. 完成情况 43](#_Toc79067406)

[3. 收获和体会 45](#_Toc79067407)

[**八、参考文献** 45](#_Toc79067408)

**一．课程设计目的**

1. 进一步掌握和利用C语言进行程设计的能力；

2. 进一步理解和运用结构化程设计的思想和方法；

3. 初步掌握开发一个小型实用系统的基本方法；

4. 学会调试一个较长程序的基本方法；

5. 学会利用流程图或N-S图表示算法；

6. 掌握书写程设计开发文档的能力（书写课程设计报告）。

**二．课程设计内容和要求**

一、课程设计内容：

一、 设计内容

1、学生个人信息示例如下:

学号 姓名 班级 高数 英语 物理 总分

2020001 张三 20 软件 1 70 90 79 239

2020202 李四 20 软件 2 60 70 55 185

2020205 陈武 20 软件 2 80 90 85 255

2、显示全部记录

3、查询功能：

（1）按名次查询某学生的基本个人成绩；

（2）按班级查询总分前 3 名学生的个人成绩；

（3）按成绩段查询，例如查询某科成绩在 90-100 的学生信息。

4、统计功能：

（1）总计每科最高分，最低分 统计最高分

（2）分班统计班级总分

5、删除功能：删除某个学生的所有信息；

6、增加功能：增加某个学生的所有信息

7、排序功能：

（1）对所有学生按照总分排序

（2）对所有学生按照学号排序

8、修改功能：修改除学号外的其他任意信息

9、保存文件，文件名student.txt。

9、复制文件到 des.txt 系统集成

10、退出系统

11、编写程序主界面，调用各功能，调用界面和各个功能的操作界面应尽可能清晰美观 !

二、 设计要求

已知有多名学生（至少 10 名）的成绩信息的文件 student.txt（该文件自行建立），要求利用自定义函数实现成绩录入、统计、查询、删除、排序和退出等

功能。具体要求如下：

A．要求显示如下界面

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1-------------- 录入

2-------------- 显示全部记录

3-------------- 查询

4--------------统计

5-------------- 删除

6-------------- 增加

7--------------排序

8--------------保存文件

9--------------复制文件

10-------------退出

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

通过选择 1-10 来确定要执行哪一步操作。

B．若选 1，则输入表中相关信息（格式见表所示），输入学号为0输入结束,

并保存数据保存到文件 student.txt 中。

C．若选择 2，按指定格式输出记录

D． 若选 3，则出现如下界面

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1----------按名次查询某学生的个人成绩

2----------按班级查询班级前 3 名学生的个人成绩

3----------按成绩段查询某学生的个人成绩

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

通过选择 1-3 来确定要做哪一个操作,有符合的记录输出字段（学号 姓名 班级 高数 英语 物理）的内容到屏幕，若无，则输出相关提示信息。

E． 若选 4，则选择

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1-------------- 总计

2-------------- 分班统计

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

若选择 2，输出：班号（学号前 5 位为班号），总分, 以追加的方式把内容保存到文件 student. txt 中。

F．若选 5，把删除后剩下的数据保存到文件 student1.txt，若无，则输出相关提示信息。

G．若选 6，把增加后的数据保存到文件 student2.txt，若无，则输出相关提示

信息。

H. 若选 7，则出现如下界面：

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1----------对所有学生按名次排序

2----------对所有学生按学号排序

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

通过选择 1-2 来确定要执行哪一步操作,有符合的记录输出到屏幕，若无则输出相关提示信息

二．课程设计要求

1.只能使用 C 语言，工具采用 TC2.0 或 3.0,源程序要有适当的 注释，使程序容易阅读,所有程序要求使用多文件多函数编程方式

2.要有用户界面。要求至少采用文本菜单界面；鼓励采用图形菜单界面，使用文件保存数据，鼓励自行增加新功能。

3.写课程设计报告（要求正规打印，A4 幅面，双面打印），内容 包括：

•封面（以附件格式为主）

•课程设计题目、内容、要求。

• 总体设计（包含几大功能模块），画出系统功能模块结构图

• 公用数据结构设计及用法说明

•详细设计（各功能模块的具体实现，画出相关流程图）

• 各函数模块的功能介绍，程序结构（画流程图），数据结构设计描述，参数说明等。（画流程图推荐使用 Microsoft Visio 工具

• 试验结果（包括输入数据和输出结果）

• 体会，总结（课程设计完成了哪些功能，有什么扩展功能有什么地方需要改进，通过课程设计有何收获，调试程序的体会等）

• 参考文献

•附录：程序清单

4. 格式：正文字体 宋体 五号 行距 1.25 倍

5. 附：封面格式

（包含所有学生的资料，每个学生的资料放在以学号姓名命名的文件夹中（源程序（\*.c）,目录为：1、应用程序；2、所需文件；3、课程设计报告文档电子档案；4、使用手册）。（计 20 级 03 班 XXXXX 号张三）

1. **需求分析**

**3.1编写目的 背景 任务概述**  
3.1.1编写需求分析目的:进一步方便使用者，以及让用户更好的理解学生奖学金管理系统。  
　　 说明：  
　　a．软件系统的名称：学生奖学金管理系统  
　　b．本项目的任务 提出者：曾颖

开发者：聂芳琪

用户及实现该软件的平台：学校管理员 学生 DEV C++

3.3.2目标  
意图：方便学校奖学金相关事项的管理

应用目标：解决实际问题，减少繁杂重复的人力劳动

说明：本软件产品是一项独立的软件，而且全部内容自含。  
3.3.3用户的特点  
　　a.本软件的最终用户的特点：学校教师或学校奖学金管理工作人员,学生

b.操作人员、维护人员的教育水平和技术专长：能够使用c语言

**3.2功能需求**

3.2.1功能划分：

学生奖学金管理系统可分为七个部分：图形化界面模块，输入记录，查询记录，更新记录，统计排序，输出记录，文件操作。

3.2.2功能需求的描述：

1）显示欢迎界面

2）进入开始界面后，用户选择“管理员”、“学生”两个选项。

3）用户选择“学生”，则进入学生界面，可以进行的操作是个人信息查询（输入个人学号）

4）用户选择“管理员”，则进入管理员登陆界面，输入密码，正确后可以进行如下操作：

·对学生的信息进行录入，录入基本信息。

·信息总览：此时总览的是原始录入信息，没有经过任何处理。

·查询记录：可以按照名次查询，按照班级查询，按照分数段查询，但首先要进行的操作是排好序，排序后可以知道名次。

·统计信息：分为总计和统计班级总分

·删除信息：先输入想删除学生信息的学号，在查找到此学生的信息，并删除，再保存文件，在信息总览里面确认是否删除。

·修改信息：先输入想修改学生信息的学号，注意，此时系统设置是：学生学号不可以修改。

·排序：有两种选择，一种是按照学号排序一种是按照总分排序。

·保存信息：保存每一次更新后的信息。

·复制文件

·退出系统

**3.3性能需求**  
（1）精度  
　　对该软件的输入、输出数据精度的要求:任意实数，最多保留六位小数  
（2）灵活性  
　　说明对该软件的灵活性的要求，即当需求发生某些变化时，该软件对这些变化的适应能力，如：  
　　没有数据时，输入信息查询等，会打印提示，并返回主菜单，提示信息录入。查询操作时会有温馨小提示。  
（4）其他专门要求

a.可扩展性需求：系统设计要求能够体现扩展性要求，采用模块化的思想，以适应将来功能扩展的需求。

b.系统安全保密性需求：系统有严格的权限管理功能，功能模块的进入（如管理员和学生具有不同的访问权限）。

**3.4界面需求**

3.4.1页面内容：课程信息内容准确，术语和行文格式统一，规范，明确，栏目，菜单设计和布局合理，传递的信息准确，及时。

3.4.2导航结构：页面有明确的导航指示，且便于理解，方便用户使用。

3.4.3艺术风格：界面形象清新悦目，布局合理，字号大小适宜，字体选择合理，前后一致，美观大方；色彩和谐自然，以黑色为背景色，字体显示以蓝色，金色为主调，提醒删除界面采用红色字体，起强调作用。

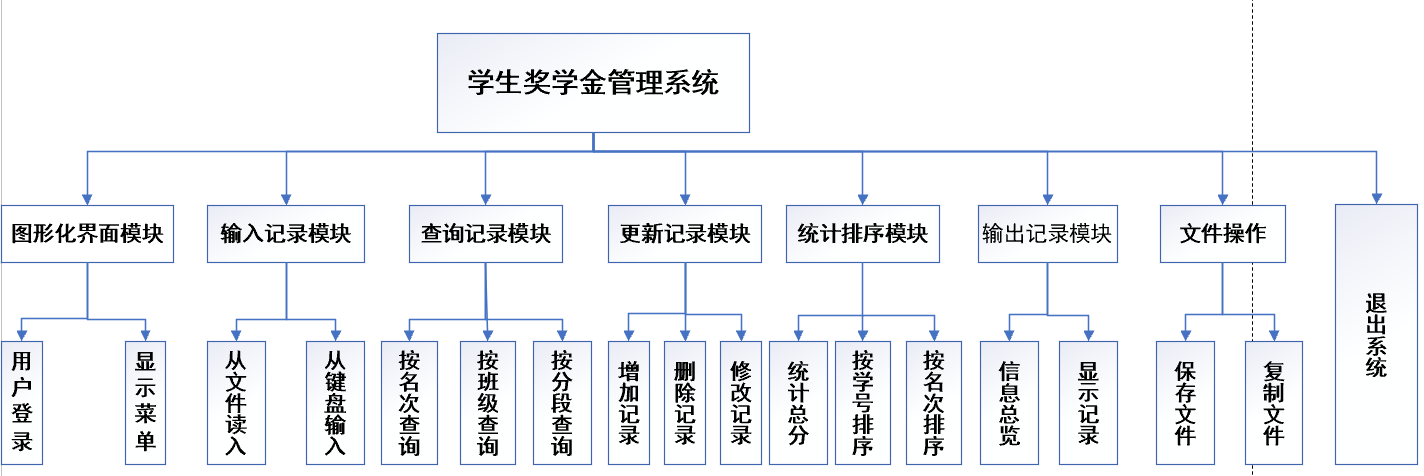
**四．概要设计**

**在需求分析的基础上，确定系统总体框架(系统功能结构图)**

**4.1系统总体框架设计**

学生奖学金管理系统以图形化界面的形式实现，由如下七大功能模块组成，图形化界面模块，输入记录模块，查询记录模块，更新记录模块，统计排序模块，输出记录模块，文件操作

系统功能模块结构图：



**4.2系统流程图：**



**4.3模块设计**

* 函数名称：
* 功能说明：
* 数据结构说明：
* 参数说明：

**1）主函数模块：**

定义一个结构体类型的指针head，调用函数creat（），把返回的头结点给head，然后执行登陆函数，登陆函数会返回一个数值，在while中确认是否执行，如果返回0，说明身份是学生，则不执行while，如果返回1，说明身份是管理员，则执行主循环和主菜单，进行按键判断。

在判断按键时，有效的输入是0~10之间的任意数值，其他的输入都会视为错误按键。若输入为0，会继续判断是否在对记录进行更新之后进行了保存操作，如果忘记了保存操作，那么最后会有友好提醒，系统会提醒管理员是否保存记录。用户会输入y或者Y，系统会进行存盘操作，最后，系统就会退出学生奖学金管理界面。

如果选择1，则调用Input（）函数执行学生信息录入操作；选择2，则调用Disp（）函数，执行查看信息函数；选择3，调用Inquire（）函数，执行查询操作；选择4，调用Tongji（）函数，执行统计操作；选择5，调用Delete（）函数，执行删除操作；选择6，调用Modify（）函数，执行修改函数；选择7，调用Sort（）函数，执行排序操作；选择8，调用Save（）函数，执行保存文件操作；选择9，调用Copy（）函数，执行复制文件操作；若输出0~9以外的数则会调用Wrong（）函数，给出出错提示。

**2）输入记录模块：**

输入记录模块主要是将数据存在单链表中。当从数据文件中输出记录时，调用fread（p,sizeof(Node),1,fp）文件读取函数，执行一次从文件中读取一条学生记录并存入指针变量所指向的结点，这个操作放在creat()函数里执行，并且直接返回头结点到主函数，即这个函数在菜单界面打开的时候就已经操作了，当管理员直接点开信息总览，如果没有存入信息，系统会提示信息为空，即单链表为空，没有任何信息可操作，此时管理员就可以选择1执行信息录入的操作。

**3）查询记录模块：**

查询模块主要实现了在单链表按照名次，班级，总分查找满足相关条件的学生信息。在查询函数中，传入的指针变量head是指向单链表的首地址，管理员根据自己的需求选择想要查询的选项。

**4）更新记录模块：**

此模块主要实现了对信息的修改，删除，增加。当然排序也算是对信息的修改，但是我把它放到和统计一块，主要的原因还是，统计与排序更像是二级操作。因为学生信息都是以单链表的结构形式储存的，所有的这些操作都是在链表中完成。

1. 增加记录

就是信息录入模块

1. 删除记录

先传入单链表的首地址，分两步走，首先按照学号或者姓名查找到某个学生的信息，在单链表中每个结点的数据域，查找相关信息。第二步，将该学生信息所在结点的前驱结点（也就前面一个结点）的指针域之后目标结点的后继结点。然后在释放目标结点的内存

1. 修改记录

修改记录主要是实现对单链表中目标结点的信息修改，两步走，首先输入要修改学生信息的学号，找到该编号的结点时，break，跳出循环。第二步，管理员可以修改除了编号的其他信息，然后再把saveflag置为1，此时并没有保存，是为了最后退出系统时可以提醒管理员是否保存更新后的信息。

**5）统计排序模块：**

统计模块没有开发太多的功能，主要是总计部分，统计了每门课程的最高分和最低分，以及总计输入信息后所有学生统计出最高分和最低分。

**6）输出记录模块：**

当把记录输出至文件的时候，调用fwrite（p,sizeof(Node),1,fp）函数，将p指针所指向的结点中的信息写到fp所指向的文件。就算是把之后更新的记录保存在文件。显示的话是直接显示链表的内容，在这里我没有显示文件存储的内容而是显示链表的内容，直接传入单链表的首地址，然后在读取单链表数据域的内容，是因为文件存储的内容，其实就是单链表的内容**。**

**7）文件操作：**

复制文件，就是把student.txt复制到student1.txt，没有太多可复述的内容。

**4.4函数功能简要概述：**

void printheader1（）函数和void printheader2（）函数：用于以表格的形式显示学生信息时，打印出表头信息。无参数

int Tixing(Link head)函数：打印提醒文件数据为空时无法进行操作，参数是单链表的头结点，形参是结构体指针，实参是单链表的首地址。若数据为空返回1，数据不为空返回0

void color（int x）函数：用于控制程序中界面显示的前景色和背景色，参数时整形数据，用于颜色的设定，与系统设置有关。形参是整型数据

void HideCursor（）函数：隐藏光标，无参数

void toxy（int x，int y）函数：将光标移动到x，y处。形参是整型数据，分别代表坐标值。

void Wrong（）函数：给出按键出错提示，无参数

void menu1（）函数：管理员菜单，提示用户选择，无参数

void menu2（Link head）函数：学生菜单，且直接执行有关学生的操作，学生输入自己的学号就可以查询个人信息。形参是结构体指针，实参是保存单链表的首地址的指针变量。

void Input（Link head）函数：用于在单链表head中增加学生记录，形参是结构体指针，实参是保存单链表的首地址的指针变量。

void Disp（Link head）函数：用于显示单链表head中储存的学生记录，内容就是student结构中所定义的内容。形参是结构体指针，实参是保存单链表的首地址的指针变量。

void Search（Link head）函数：查询函数，用于单链表head中按照班级，名次，分数段查询，形参是结构体指针，实参是保存单链表的首地址的指针变量。

void Tongji（Link head）函数：统计每门课程的最高分和最低分，年级最高分年级最低分。形参是结构体指针，实参是保存单链表的首地址的指针变量。

void Delete（Link head）函数：用于在单链表中按照编号或者姓名查找满足条件的学生信息，并删除。形参是结构体指针，实参是保存单链表的首地址指针变量。

void Modify（Link head）函数：用于在单链表head中修改学生记录。形参是结构体指针，实参是保存单链表的首地址的指针变量。

void Sort（Link head）函数：用于在单链表中对学号，对（所有课程相加，再除以课程数后得到的平均分的排序）。形参是结构体指针，实参是保存单链表的首地址的指针变量。

void Save（Link head）函数：用于将单链表head中的数据保存。形参是结构体指针，实参是保存单链表的首地址的指针变量。

void Copy（Link head）函数：暂定，与修改密码有关形参是结构体指针，实参是保存单链表的首地址的指针变量。

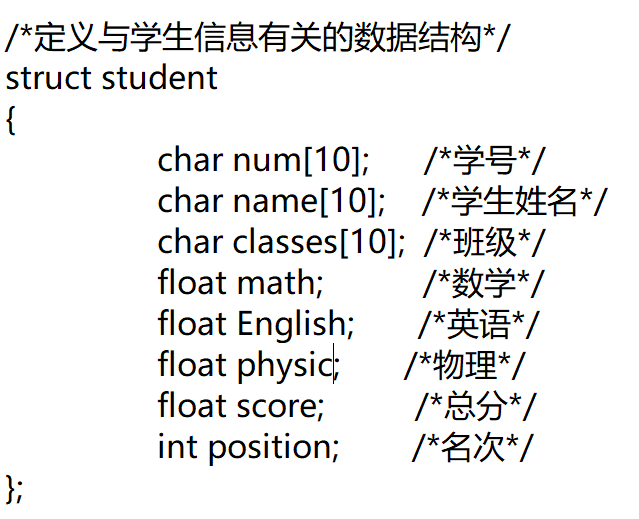
Link creat（）函数：创建链表，并传回链表的头结点到主函数。无参数。

void Quit（）函数：退出系统，无参数。

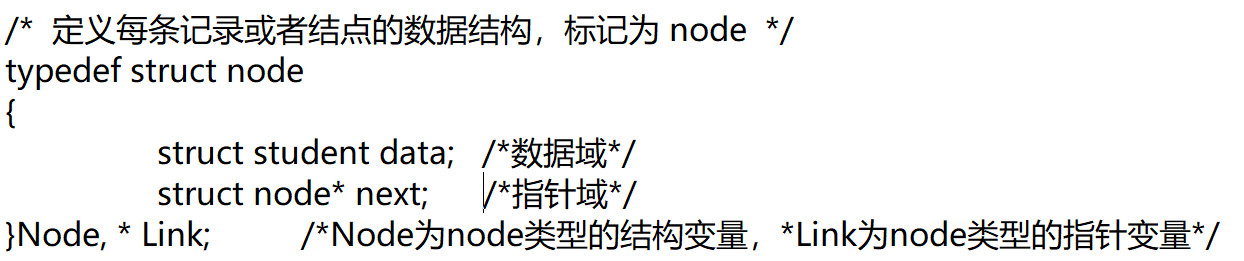
**4.5公用数据结构设计及用法，全局变量说明**

1）结构体

学生信息结构体（记作student，作为单链表的数据域）：



单链表node结构体：



data是student结构类型的数据，是单链表中的数据域，next作为单链表的指针域，用来储存后继结点的地址，Node是node类型的结构变量，Link是node类型的指针变量。

2）全局变量

int saveflag=0；用于确定是否存盘，每一次对信息修改后，将saveflag置于1，当管理员退出系统时如果没有执行保存操作，那么系统会提示，是否存盘，存盘后又将saveflag置于0。

* 1. **详细设计**

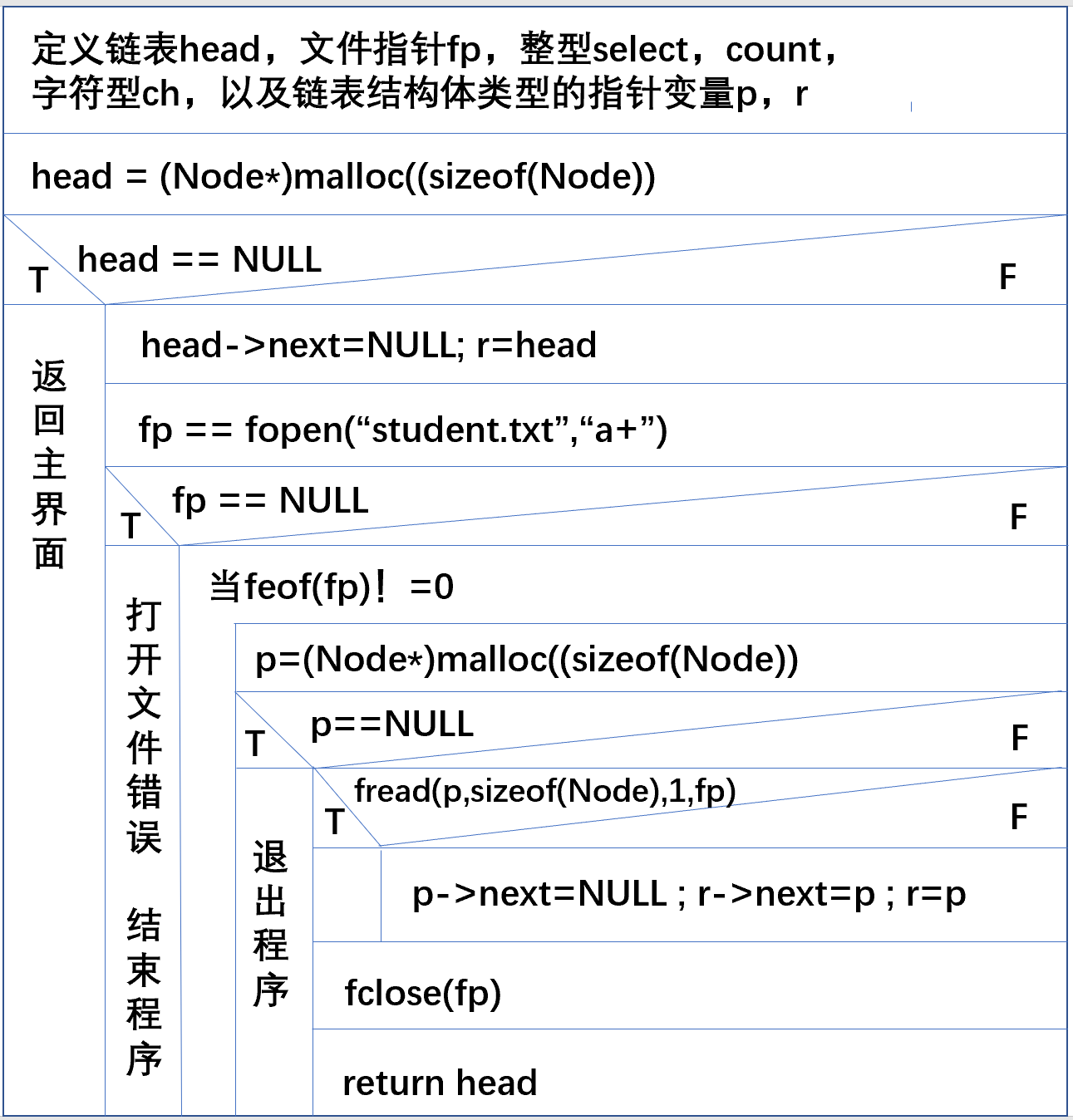
**5.1主要功能模块的算法设计思路以及对应的功能模块的流程图（或N-S图）**

1)创建链表:

算法设计思路:

1. 先定义一个链表Link head，定义一个文件指针fp，定义记录指针变量Node\*p，\*r，定义一个字符型ch用于保存管理员输入的“y , Y , n , N”。
2. head开辟一个node结构体大小的内存空间，如果没有申请到，就打印提示信息并返回主界面。先是head->next=NULL;这是链表为空时的情况。
3. 用追加的方式打开一个二进制文件，可读可写，若此文件不存在，会创建此文件。如果文件指针为空，则打印提示信息，直接结束程序。进入while(!feof(fp))检测是否到达文件末尾，p开辟一个新的空间，如果没有申请成功会打印提示信息并退出系统。用if(fread(p,sizeof(Node),1,fp)==1)来判断，若成功则返回成功读取的对象个数，若出现错误或到达文件末尾，则可能小于1。在if语句中，p->next=NULL,即代表每次只会读取一条，读完就结束，r一开始是指向头结点，在if语句中，直接r->next=p,新增入了一个结点，然后r往后移动一位，指向当前链表的尾结点。继续参与循环直到读取结束。，关闭文件。
4. 主要是实现了链表的创建，如果文件没有数据就是空链表，如果有数据，则把数据存入链表中。

N-S图:



2) 信息录入：

算法设计和思路：

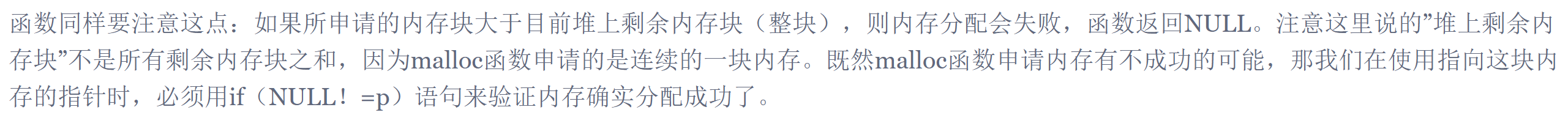
1.进入奖学金管理系统，如果数据为空，将从单链表的头部开始，增加学生记录的结点，否则就将该学生的记录结点添加在链表的尾部。

2.Input（）函数的形参是一个单链表的首地址，定义三个用于添加操作的临时的结构体指针变量，p是用来临时开辟空间，s是用来记住头结点的后驱结点地址（因为我把数据放到头结点的后一位置来储存），r是用来移动到单链表的最后一位。

3.定义字符型变量ch和flag，前者用于保存管理员输入的“y，Y，n，N”，后者用于判定是否重复输入学号。

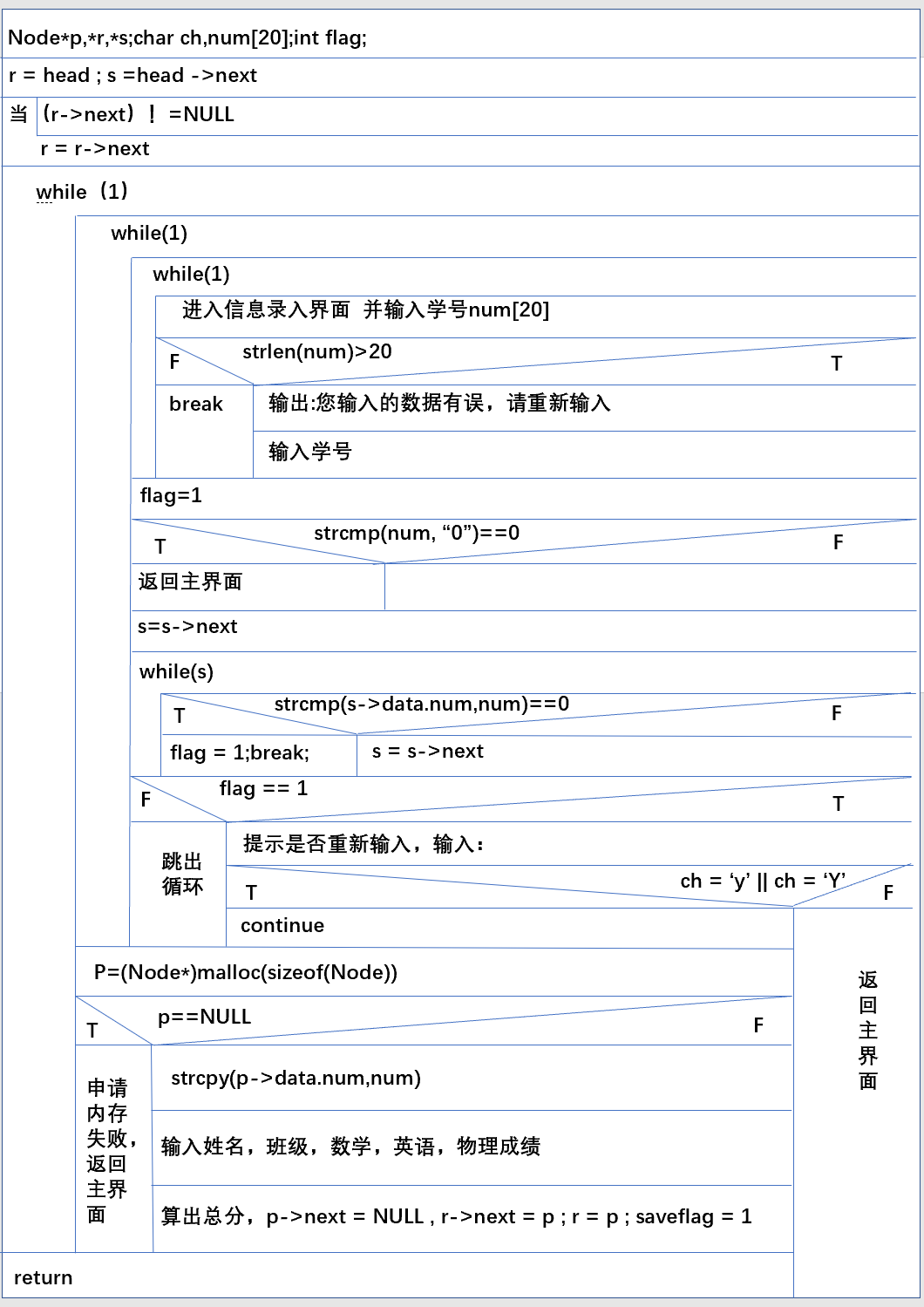
4.r指向头结点，s指向头结点的后一位，用while（r->next）循环使得r移到原来链表的最末尾，准备添加记录。

5.再次进入while（1）大循环，这里实现了一次可以实现录入多条信息，想停止录入就输入0停止。又进入一个while（1）的循环，这里有友好的报错提示，先展示界面，输入学号，如果输入数大于结构体成员定义的范围，就打印“重新输入的提示”。否则就跳出循环，再把flag置于0，先判断学号是否为0，是0就返回主界面；使s=head->next;当s不是空指针的时候，用于查询该编号是否存在，如果存在就把flag置于1，然后结束循环，否则s移动到下一位。循环结束后，判断flag是否为1，如果是1，就提示用户：“该编号已经存在，是否重新输入”输入是，那么执行continue，继续循环，输入否则返回主界面。当上述两种情况都没有遇到，就跳出内部while（1）的循环，p用于申请内存空间，如果申请失败，则打印提示信息，并返回主界面（这里是学习malloc函数时，来自网上一位大佬的提醒）如下：



再分别输入student结构体设置的信息，再p->next=NULL;表示这是链表的尾部结点，r->next=p,r指向原链表的最后一个，现在r->next = p;增加结点，r=p 表示r又移动到了当前链表的最后一个结点，最后把saveflag=1，提示存盘。

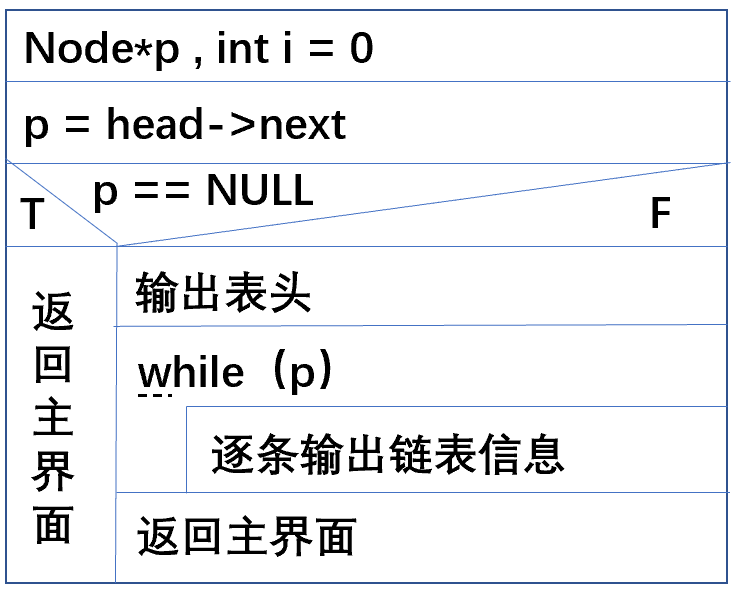
N-S图：

****

3)信息总览：

算法设计和思路：

传入链表，定义一个Node类型的指针变量，指向头结点的后一位，如果链表是空的则打印提示信息，输出表格头部，while（p）实现信息输出，由于使用了API函数，为了界面美观，先设置i=10，在循环中i+=2输出数据，再i-1输出横线。，再p往后移。

N-S图：

4)查询记录：

算法设计和思路：

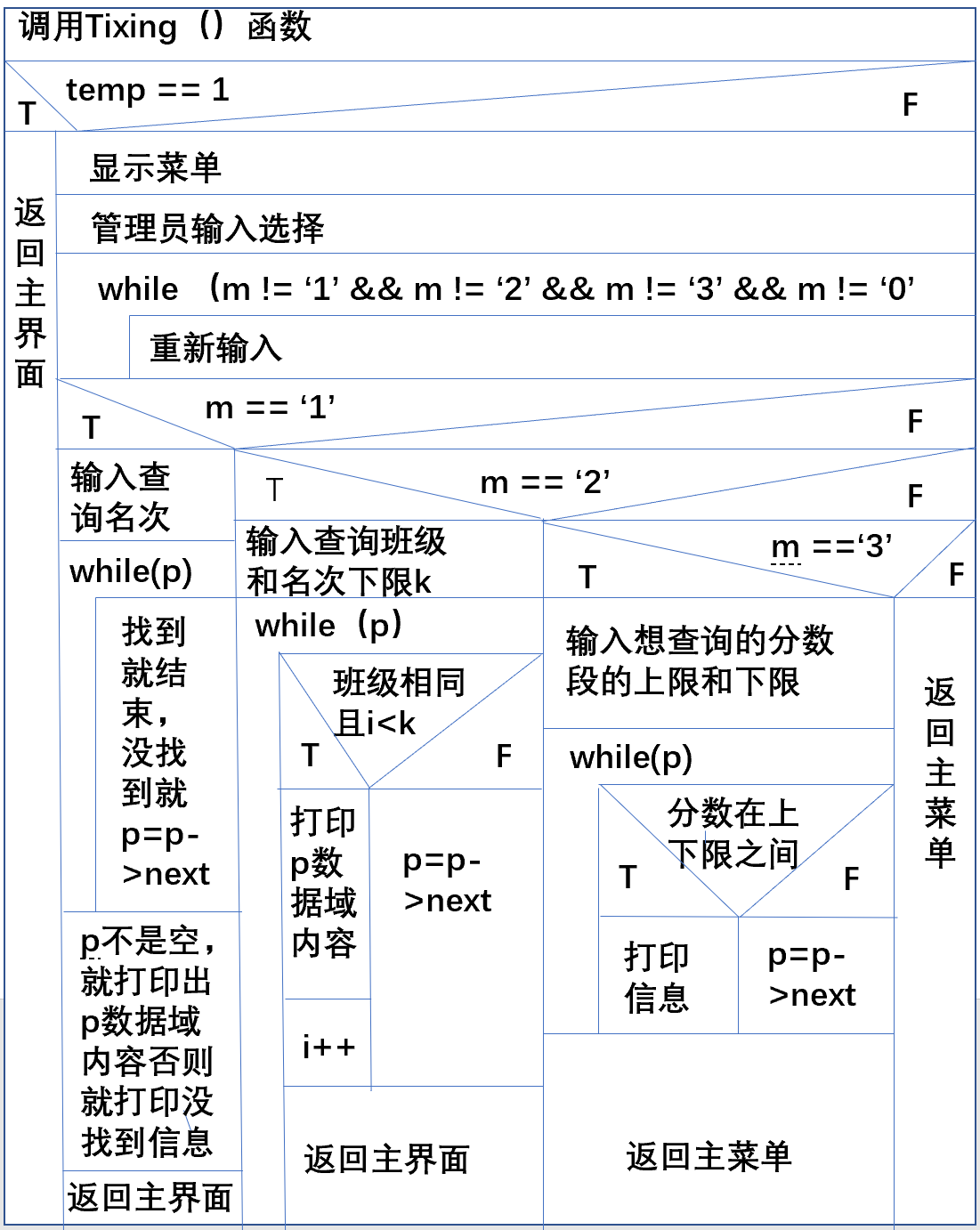
1.当管理员选择查询功能时，有3个选项，1是按照名次查询，2是按照班级查询，3是按照分数段查询，0是返回主菜单，输入其他会提示报错，用while（m != ’1’ && m != ’2’&&m != ‘0’）来实现报错提醒，定义一个Node类型的指针变量，如果头结点的后一位没有储存数据，那么打印提示信息，并返回主界面，如果有数据则显示查询界面。

2.管理员输入选择，输入1，if判断，进入名次查询，用while（p）语句中的if语句寻找单链表中相同名次的结点，如果找到就p不为空并跳出循环，没找到，p为空并结束循环，如果p不为空就执行接下来的if语句，打印想要查询的学生信息否则就打印提示信息：没找到。

3.输入2，进入else if 同样是如上执行，只不过再while中的if语句中设置一个限定条件想要查询前几名的限制条件。管理员首先输入查询哪个班级，然后再输入想要查询前几名。按任意键返回。

4.输入3，按分数段查询，定义两个float类型的变量，用于保存管理员想查询分数段的上下限，然后再依次查找，符合条件的可以输出。

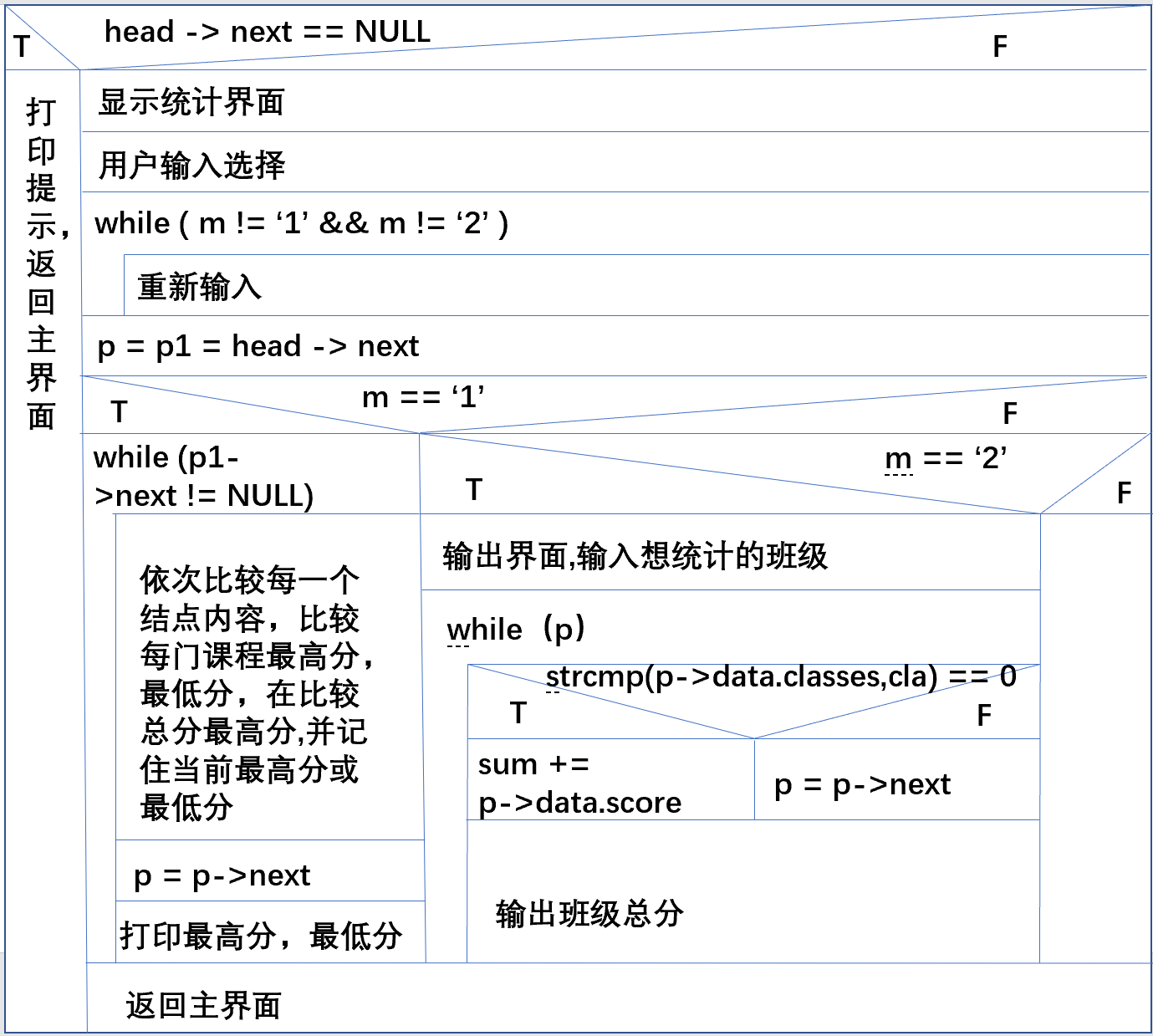
N-S图：



5)统计记录：

算法设计和思路：统计模块开发出两种功能，一是总计，总计每门课程的最高分，最高分，以及年级最高分。传入单链表的地址到统计函数，定义两个指针变量，临时记录，先判断头结点后一位有无数据，没有数据就打印提示信息，否则输入选择，选择1，则找出每门课程的最高分最低分，先都初始化为头结点后一位数据域的值，再用if找出最大最小。选择二，则输入管理员想要了解的班级的总分，输入班级，再进入while（p）判断是否找到该班级学生，若找到则sum+=p->score直到退出循环，再输入num的值，实现班级总分。

N-S图：



6)删除记录:

算法设计和思路:传入单链表的首地址，定义两个Node类型的指针变量p和r , r是用来记住要删除结点的前驱结点（当要删除一个结点的时候，只需前驱结点指向后驱结点），p指向头结点的后一位，如果首元结点没有数据就打印“没有学生记录，请添加一些信息吧”，返回主界面。否则，打印删除界面。输入自己的选择，如果选择1，则输入想要删除学生的学号，进入while（p）循环判断p此时结点数据域的信息是否就是删除的学号，如果是的，那么break，跳出循环；如果不是，则p继续向后移。循环结束后，用if（p）来确定是否找到，如果找到则执行if语句，if语句中，首先跳出界面“确认管理员是否删除”若输入1，则确认删除，输入2则取消删除；r指向头结点，当r->next!=p也就是r的指针域没有指向目标结点的时候，继续后移，找到后就跳出循环，执行r->next=p->next,直接跳过目标结点，free（p）释放内存，再把saveflag置为1，提醒存盘，，如果没找到，就打印提示信息。按照姓名删除也遵循上述思想，这里不再赘述。

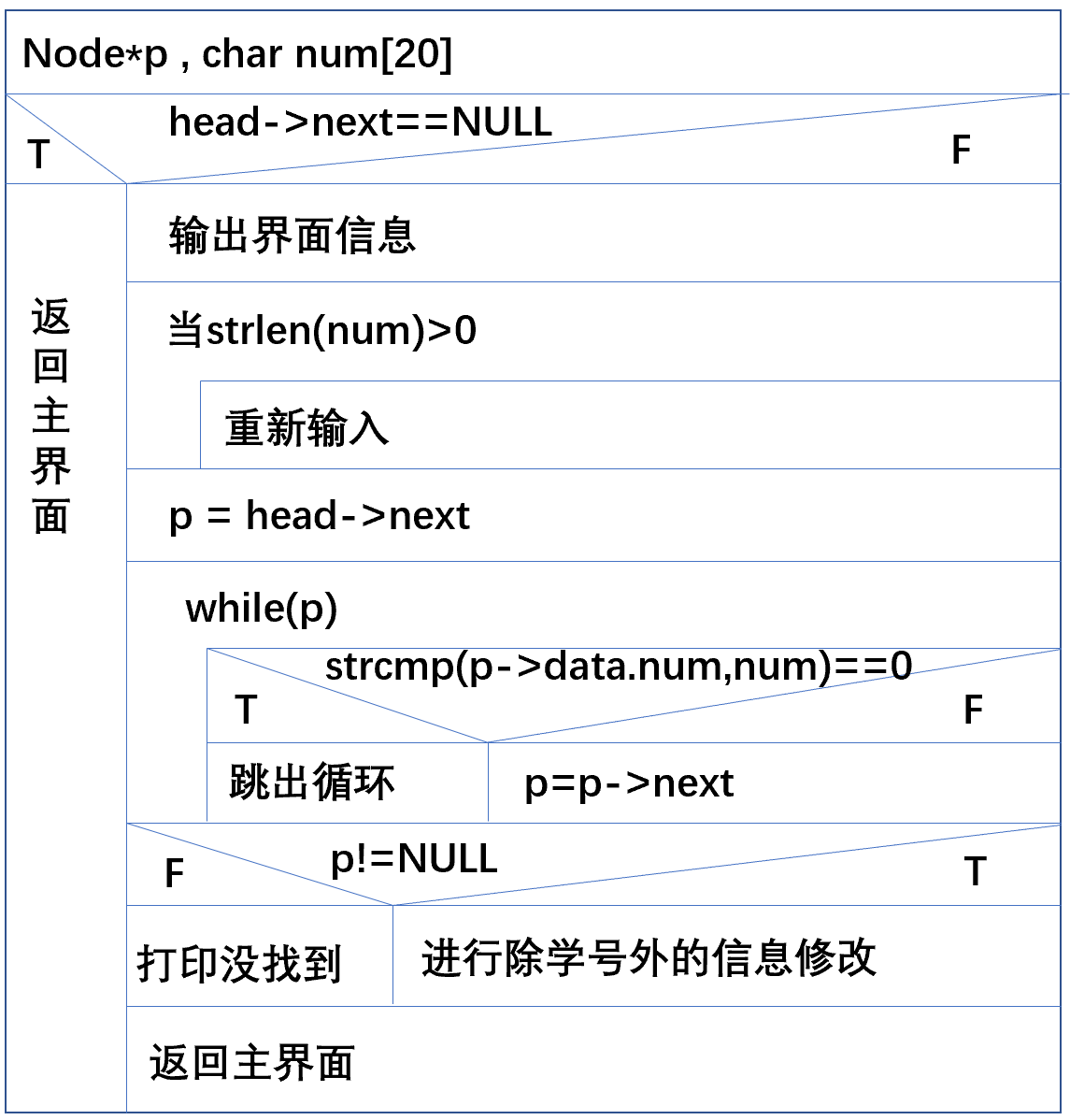
流程图:



7)修改信息：

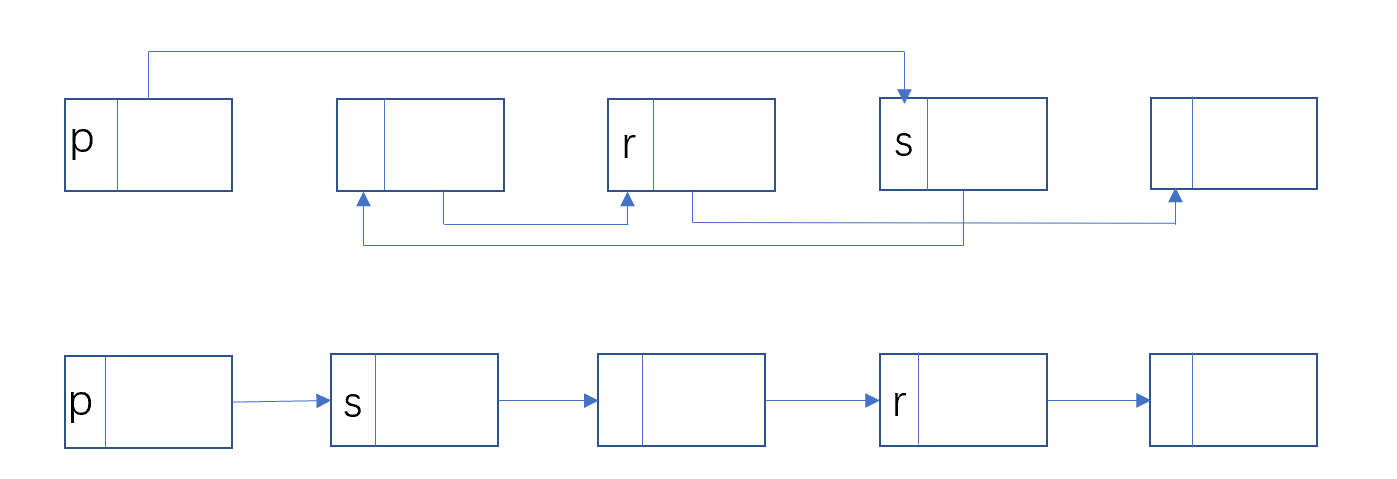
算法设计和思路：这个模块比较容易实现，系统按照输入的学号查找到相关记录，然后提醒用户只能修改学号之外的值，编号不能修改。传入单链表的首地址到函数，定义一个Node类型的指针p，如果头结点后一位没有存放数据则会打印提示信息，否则进入修改界面。输入要修改学生信息的学号，这里如果学号违规输入，则会打印报错提示。P指向头结点的后一位，进入while（p）循环，如果找到了相同学号就break跳出循环，否则p继续后移。结束循环后，用if（p）判断是否找到信息，如果找到就执行if语句，进行信息修改，否则就打印提示学号不存在。

N-S图：

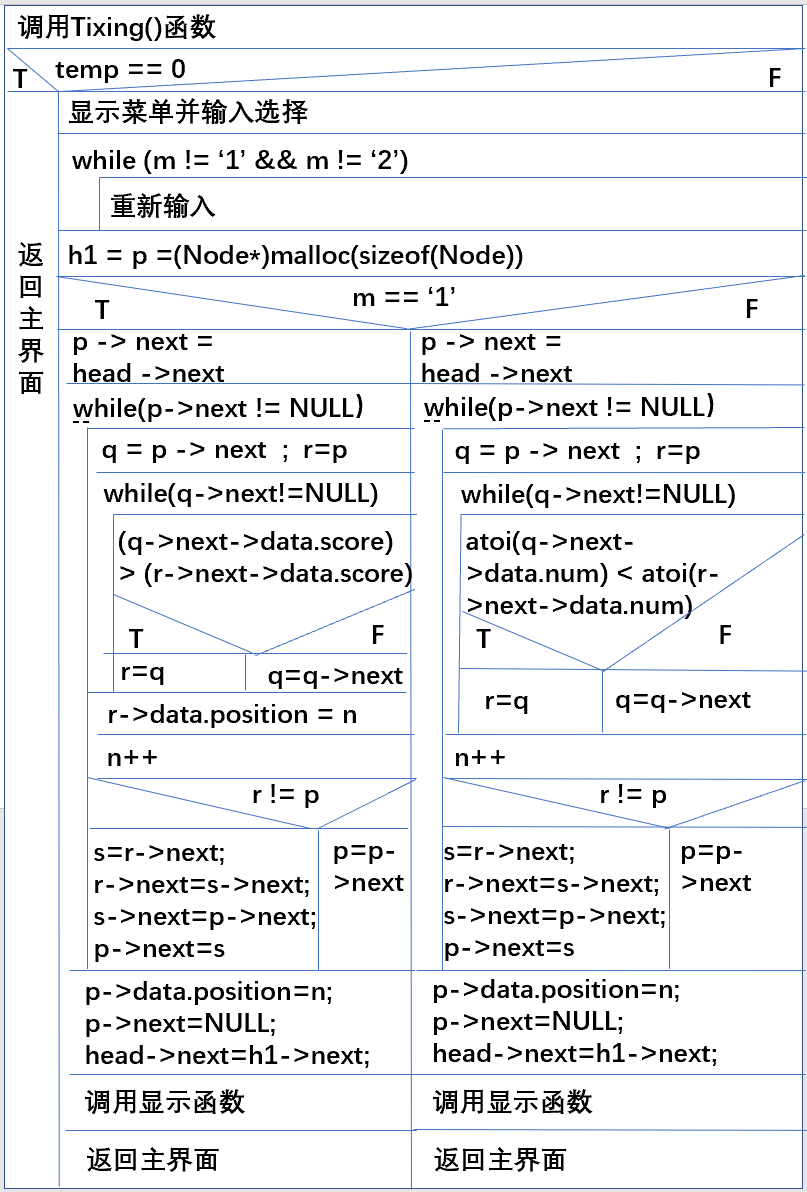


8)排序：

算法设计和思路：排序用的算法是选择排序法，先定义4个Node类型的指针变量p，q，r，h1，r用于记住目标最小或最大结点的前一位结点，h1和p重新开辟一个新的空间，且h1=p，使p的指针域指向头结点的后一位。P成为了新的头结点，当p->next不为空的时候，执行while语句内的操作，内循环的目的是找到当前最大或最小的目标结点， 用r指向目标结点的前一位，当目标结点的前一位不是上一次内循环找到的最小或最大结点时，就执行if语句，r的指针域直接指向目标结点的后驱结点，目标结点脱离链表，然后插到上一次找到的最小或最大结点的后一位，具体实现是，s->next=p->next ; p -> next = s ;总的来说，就是先找到目标结点，然后在记住目标结点的前驱结点，前驱结点的作用就是把目标结点脱离链表，具体实现就是前驱结点的指针域指向后驱结点，再把目标结点插入到当前内循环首结点的后一位，插入后，p后移，每一次都是当前内循环的首结点。对学号的排序也是用了选择排序法。排序完后再调用显示函数，显示排序后的学生信息。



N-S图：

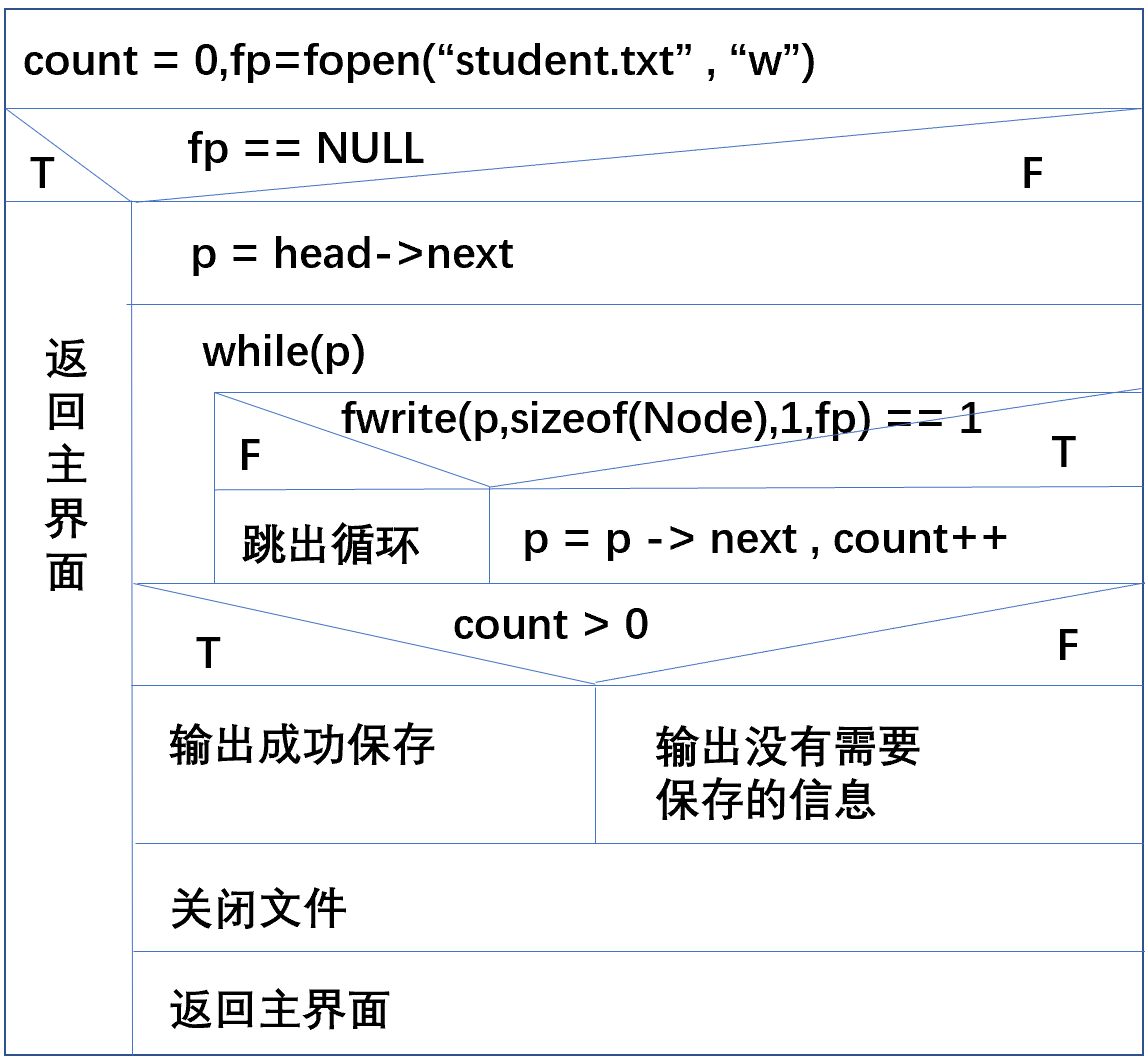


9)保存记录：

算法设计和思路：定义文件指针fp，定义Node类型的指针变量p，如果打开文件失败就退出程序，p指向头结点的后一位，while（p）再执行if（fwrite（p,sizeof( Node ),1,fp）== 1）

如果满足则p继续后移，数目加1，否则就break跳出循环，如果有记录就输出保存记录条数，否则就输出没有更新记录。关闭文件。

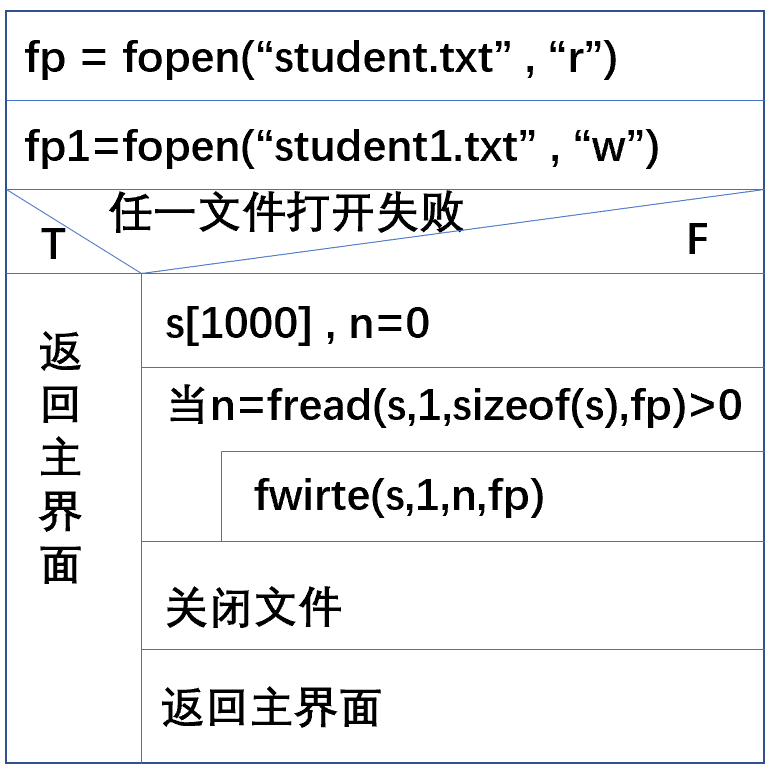
N-S图：



10)复制文件:

算法设计和思路：定义两个文件指针fp指向源文件，fp1指向目标文件，如果文件打开失败则提示出错，并结束程序。再定义一个字符数组，循环读取源文件内容到s中，再写到目标文件中。再关闭两个文件。

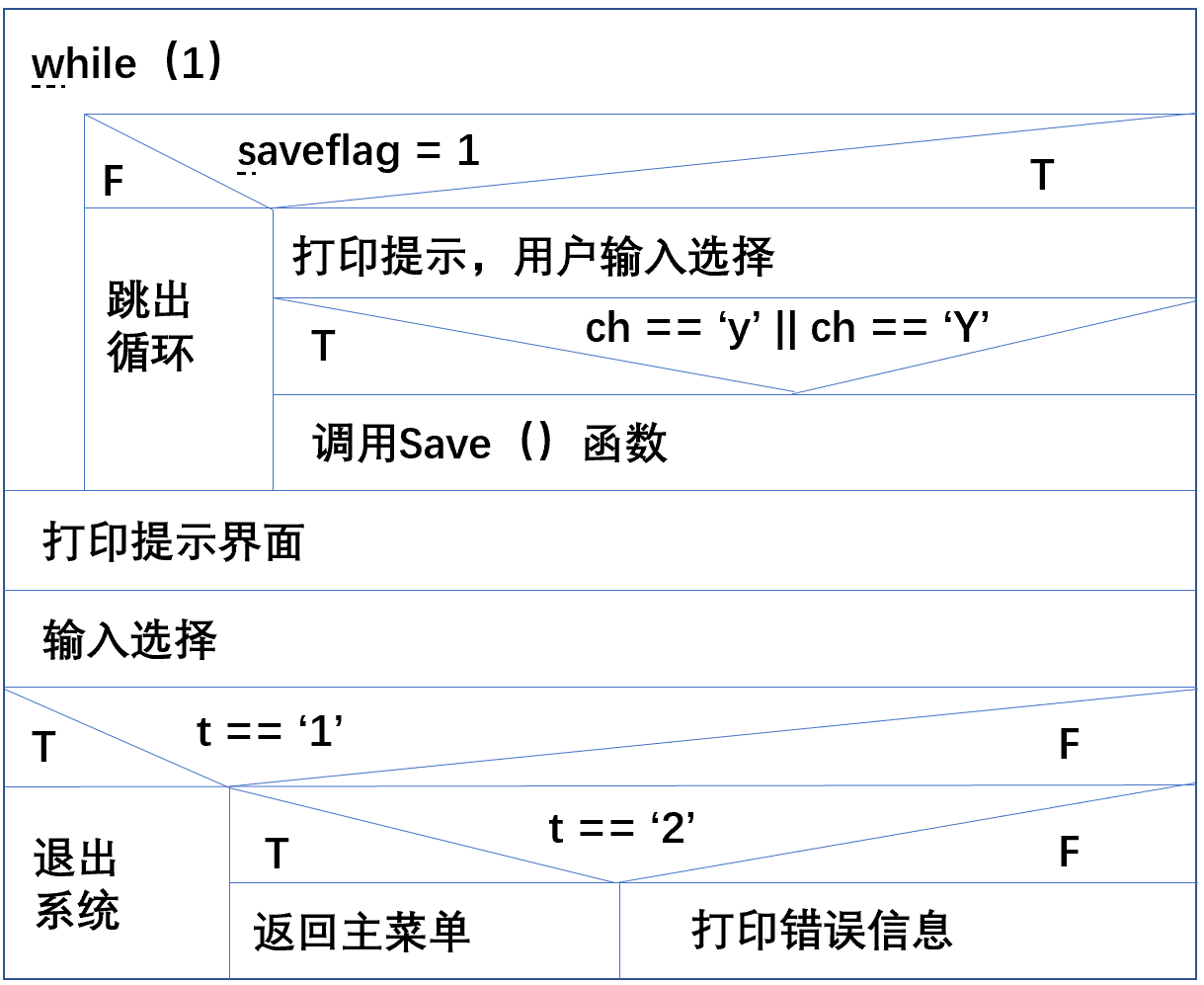
N-S图：



11)退出系统：

算法设计和思路：传入链表的首地址，进入while（1）的循环，如果saveflag==1，就打印是否存盘的信息，输入y或者Y则调用保存记录函数，否则就跳出循环进入退出界面。输入1就执行case1退出程序，输入2就返回主菜单，输入其他视为错误处理，并打印错误提醒，用while（1）实现可重复输入。

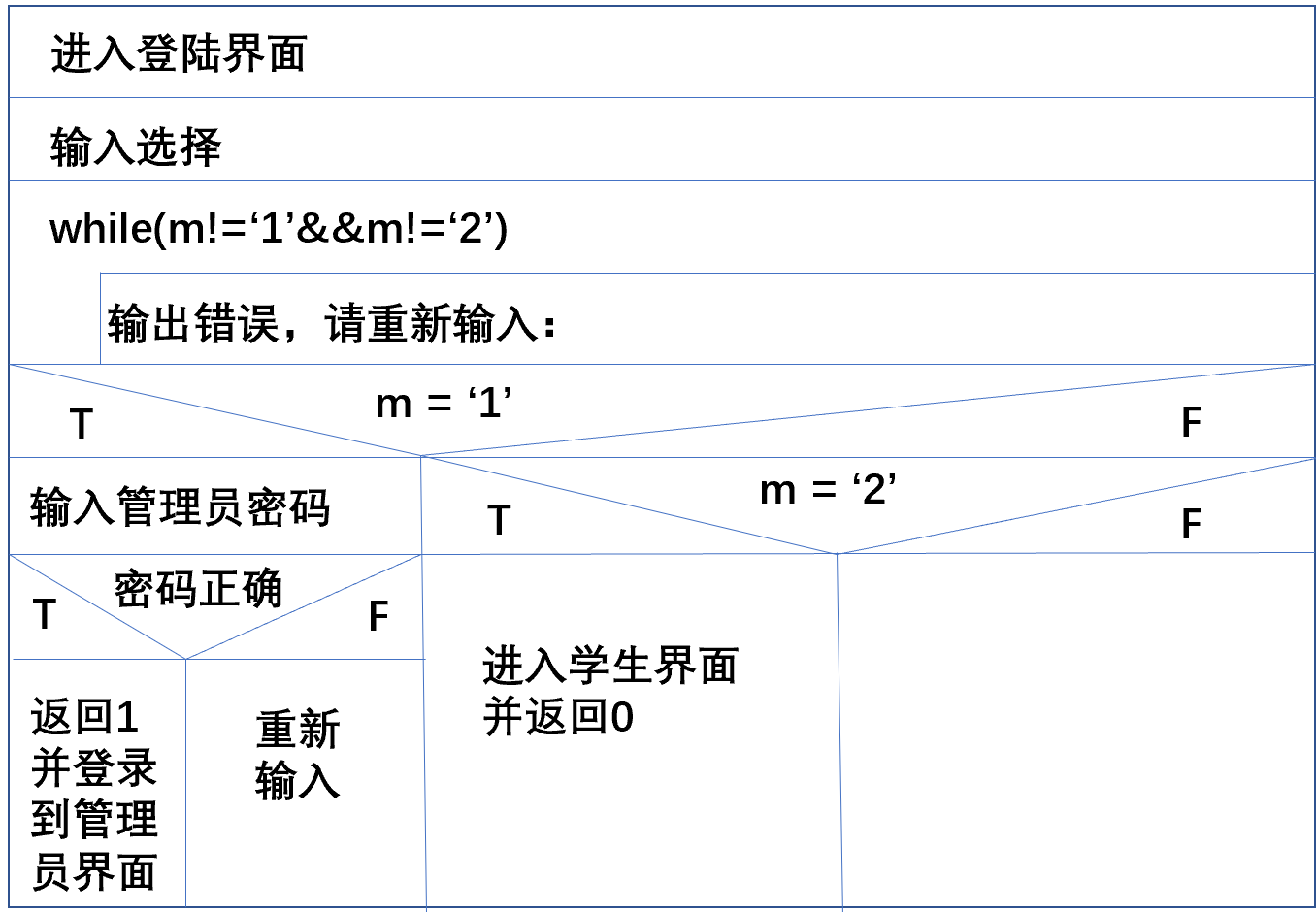
N-S图:



12)登录函数：

算法设计和思路：登录函数的返回值是int类型，先显示登录界面，确认身份，用户输入选择，输入错误则会有报错提醒，当输入1，进入管理员登录界面，用do while 循环实现密码输入，如果输入正确则进入学生奖学金管理系统，输入错误则重新输入。如果选择2，先调用函数creat，然后传到menu2函数，进入学生界面，学生可以进行对应操作。

N-S图:



对文件处理方面的知识点

【size\_t fwrite(const void\* buffer, size\_t size, size\_t count, FILE\* stream);

返回值：返回实际写入的[数据块](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%9D%97" \t "_blank)数目

（1）buffer：是一个[指针](https://baike.baidu.com/item/%E6%8C%87%E9%92%88" \t "_blank)，对fwrite来说，是要获取数据的地址；

（2）size：要写入内容的单字节数；

（3）count:要进行写入size字节的[数据项](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E9%A1%B9" \t "_blank)的个数；

（4）stream:目标[文件指针](https://baike.baidu.com/item/%E6%96%87%E4%BB%B6%E6%8C%87%E9%92%88" \t "_blank)；

（5）返回实际写入的数据项个数count。】

【fread函数： fread(const void \*ptr, size\_t size, size\_t nmemb, FILE \*stream) size-这是要被写入的每个元素的大小，以字节为单位。nmemb这是元素的个数，每个元素的大小为 size 字节。stream-这是指向 FILE 对象的指针，该 FILE 对象指定了一个输出流】

【feof】是C语言标准库函数，其原型在stdio.h中，其功能是检测流上的文件结束符，如果文件结束，则返回非0值，否则返回0（即，文件结束：返回非0值；文件未结束：返回0值）

API函数方面的了解

1.void color（）函数：解释：【GetStdHandle（）】返回标准的输入、输出或错误的设备的句柄，也就是获得输入、输出/错误的屏幕缓冲区的句柄。此处【STD\_OUTPUT\_HANDLE】为标准输出句柄。【SetConsoleTextAttribute()】函数的参数：（控制台屏幕缓冲区的句柄且句柄必须具有GENERIC\_READ访问权限，字符属性）（虽然我也不是很懂，但是介于对界面美观的强迫症，上网学习了一下简单用法）

【API函数的学习】：（参考了microsoft的API函数中SetConsoleTextAttribute函数设置控制台字体的颜色）；设置前景色和背景色使用的函数是：SetConsoleTextAttribute函数 靠一个字节的低四来控制前景色，高四位来控制背景色。

BOOL  SetConsoleTextAttribute(HANDLE hConsoleOut句柄 ， WORD wAttributes文本属性 ）；

参数的说明：HANDLE handle\_out = GetStdHandle(STD\_OUTPUT\_HANDLE);    //获得标准输出设备句柄

文本属性，其实就是颜色属性，有背景色和前景色（就是字符的颜色）两类，每一类只提供三原色（红，绿，蓝）和加强色（灰色，可与其他颜色搭配使用，使颜色变亮）；

若要恢复原来字体的颜色，可以在设置颜色的语句后面加上下面的语句：

 SetConsoleTextAttribute(GetStdHandle(STD\_OUTPUT\_HANDLE),FOREGROUND\_RED | FOREGROUND\_GREEN | FOREGROUND\_BLUE);

2.void HideCursor（）函数: CONSOLE\_CURSOR\_INFO包含有关控制台光标的信息;/\*BOOL WINAPI SetConsoleCursorInfo(\_In\_HANDLE hConsoleOutput, \_In\_const CONSOLE\_CURSOR\_INFO \*lpConsoleCursorInfo);指向CONSOLE\_CURSOR\_INFO结构的指针，该结构为控制台屏幕缓冲区的光标提供新的规范\*/

3.void toxy(int x,int y): COORD pos = { x , y }; 定义表示一个字符在控制台屏幕上的坐标的对象; SetConsoleCursorPosition是API中定位光标位置的函数。

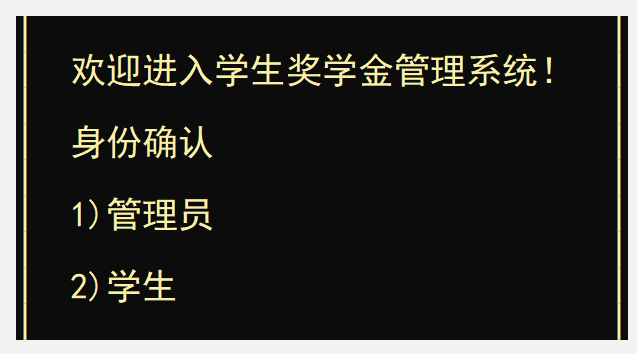
4.void printheader1（）和void printheader2（）

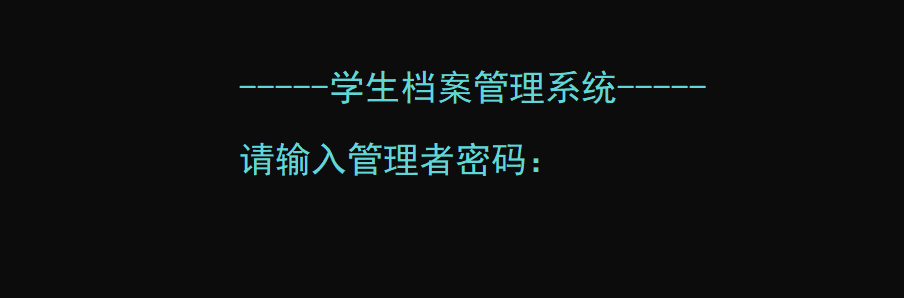
void Tixing函数用于提醒是否文件为空，因为这三个函数经常使用，所以独立出来，实现代码简洁。

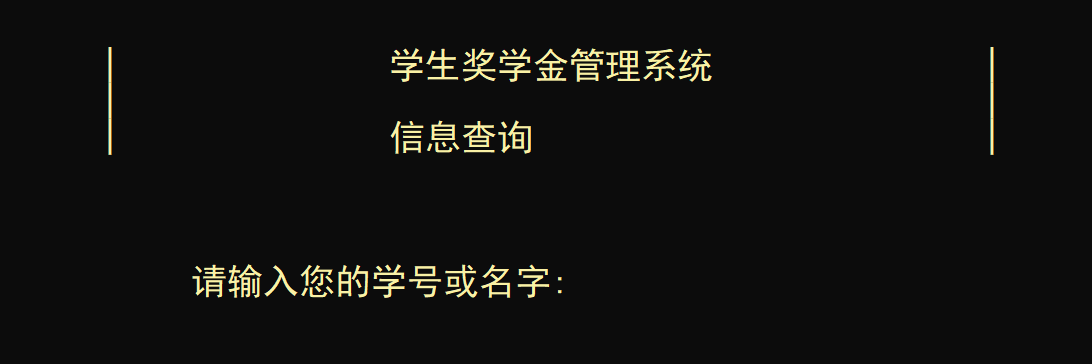
* 1. **实验结果**

**6.1各模块输入输出结果：**

**（**1）当用户进入系统时，其欢迎界面如下：用户可以选择身份，如果是管理员则输入1，转到管理员密码输入界面；是学生则输入2，转到学生个人信息查询界面；输入其他，则显示出错提醒。实现如下：

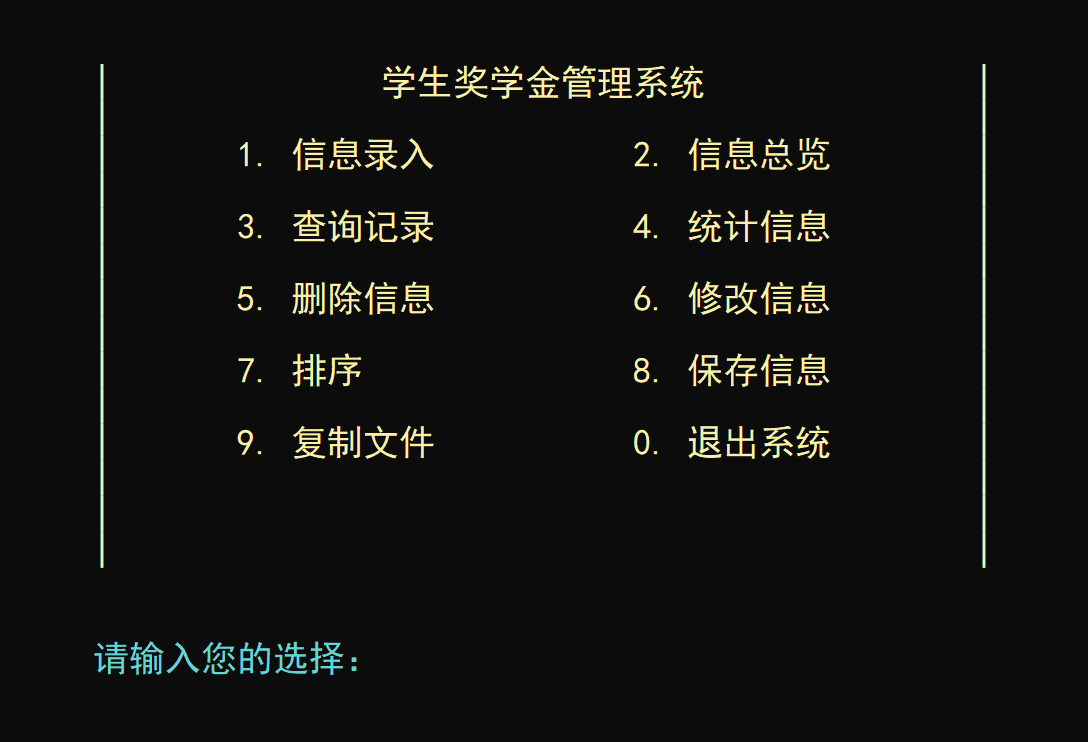
**欢迎界面：**

**输入1进入：**

**输入2进入：**

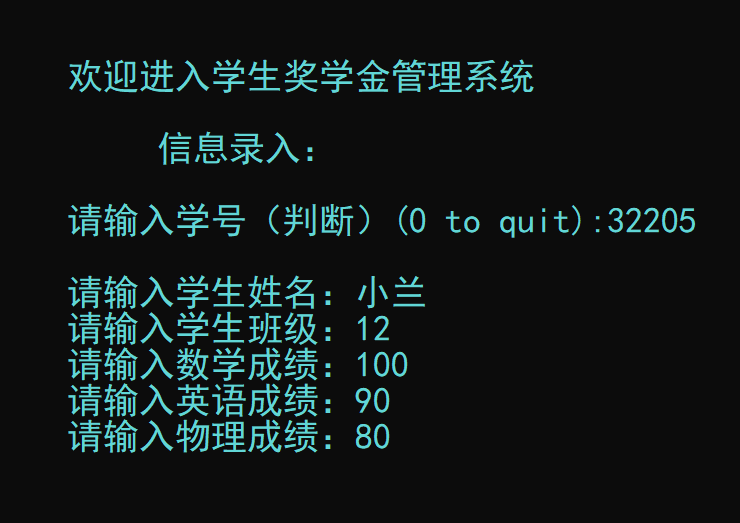
（2）管理员主界面

当密码输入正确时，提示“正在登陆“，管理员可以进入主界面，主界面如下。此时系统已经把student.txt文件打开，如果文件没有空的话，则会执行creat（）函数，将数据从文件中逐条读出，并写入单链表中，返回单链表的首地址给主函数，管理员可以选择0~9之间的数值，调用相应的功能进行操作，输入其他，则会提示出错处理。



（3）信息录入

当管理员输入1后并按Enter键后，即可进入数据输入界面。其输入过程如下所示。先输入12条信息，当输入学号为0时，会结束输入过程，回到主菜单界面。



（4）信息总览

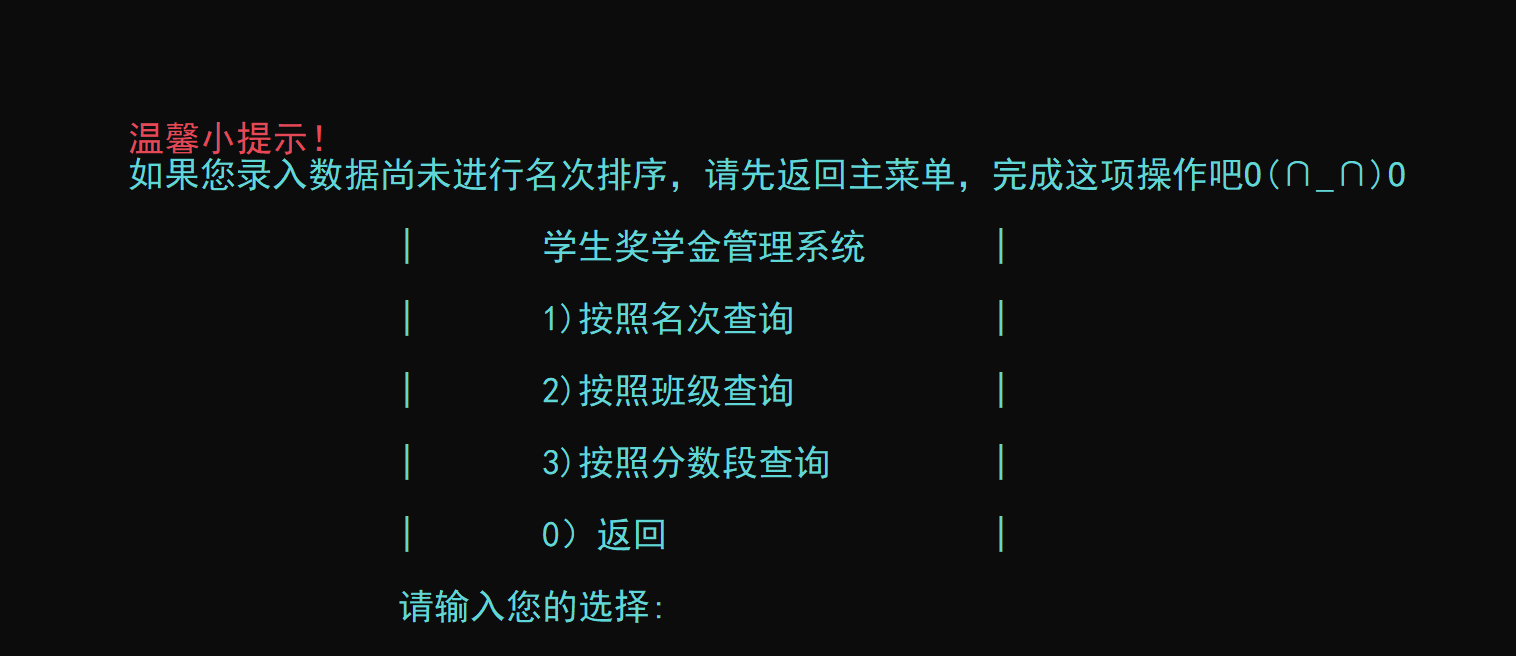
当管理员执行了输入记录或者是从数据文件中读取了文件之后，即可输入2，并且按下Enter键后，可以查看当前链表的内容，如下所示：

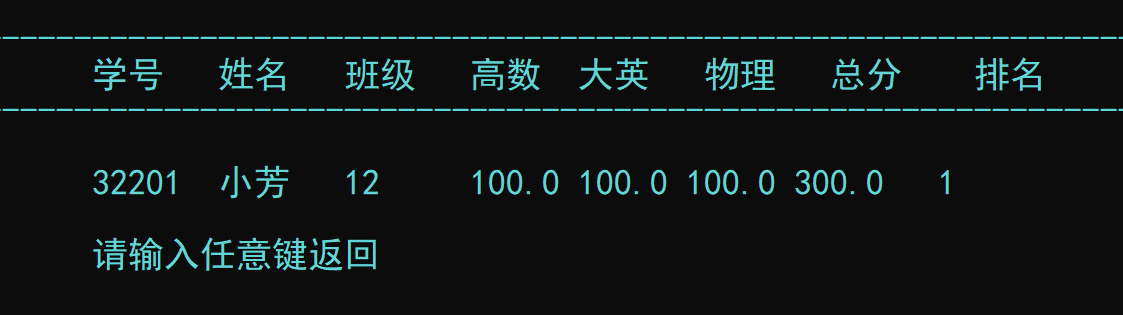


（5）查询信息

当管理员输入3时进入查询界面，有三个选择，输入1时按照名次查询，本次查询第一名；输入2，查询班级前三名；输入3，按照成绩段查询。实现如下：

查询界面：

****

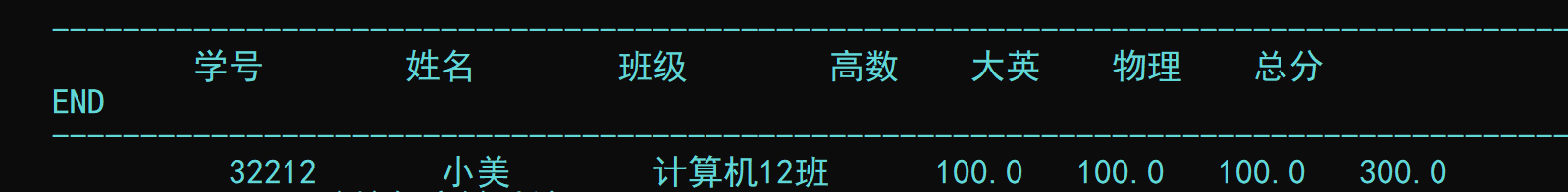
输入1：

输入2：

输入3：



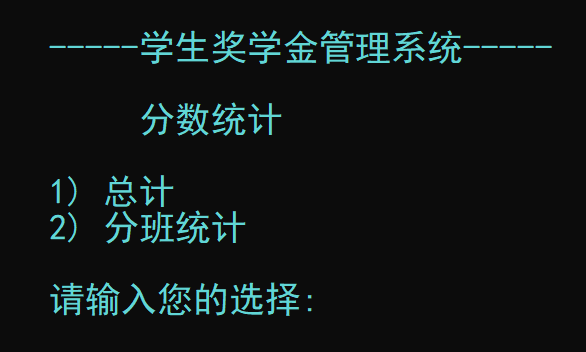
输出：

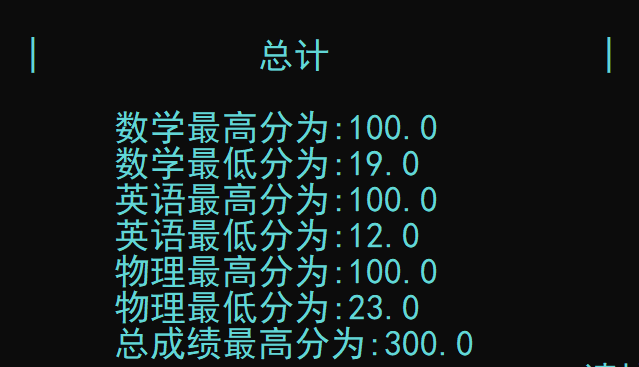


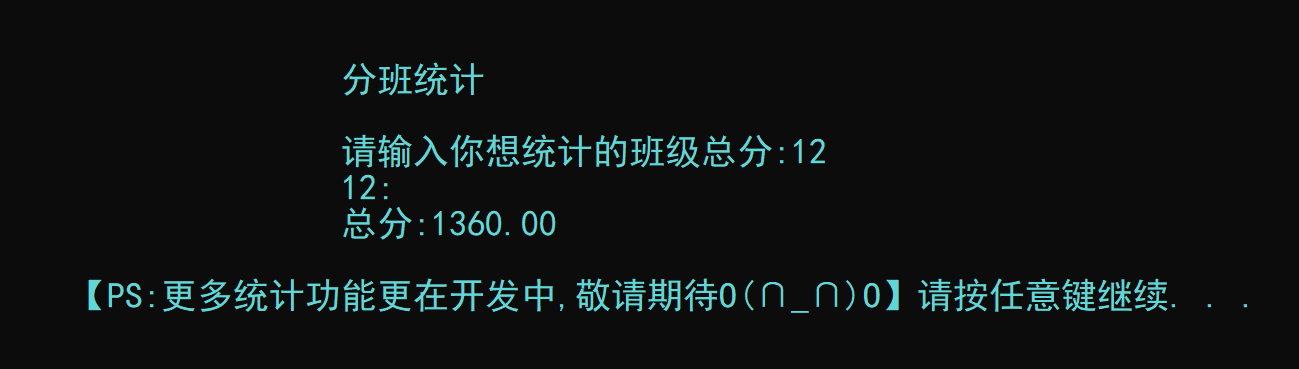
输入0：返回主界面

（6）统计信息

管理员输入4时进入统计界面，有两个选择，输入1是总计，输入2是分班统计，实现如下：

统计界面：

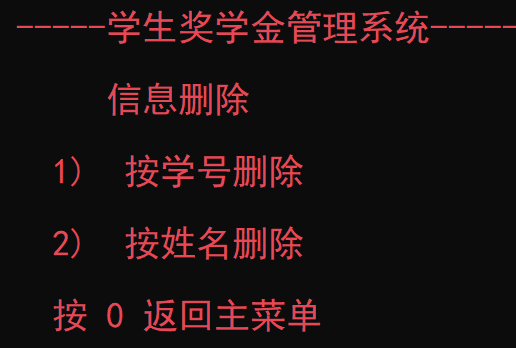
输入1：

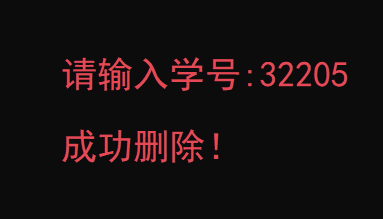
输入2：

输入0：返回主界面

（7）删除信息

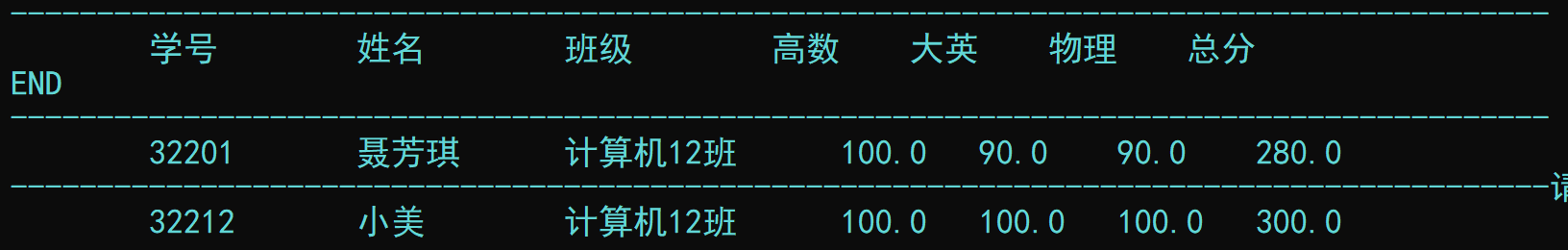
当管理员进入2时并按入Enter键后，即可进入删除界面，我要删除小兰的记录，删除后再enter退回主界面时，按键2进入信息总览，可见信息成功删除，如图下：（本次记录均为一次操作）

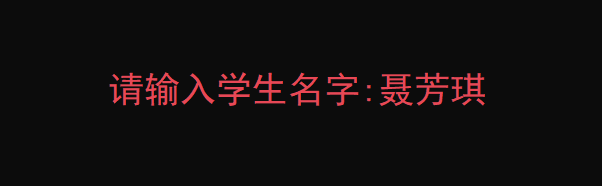
删除界面：

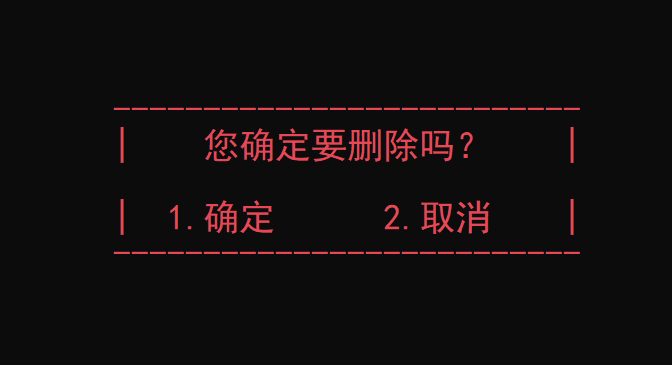
输入1： 

删除成功：

输入2：删除前



输入名字：

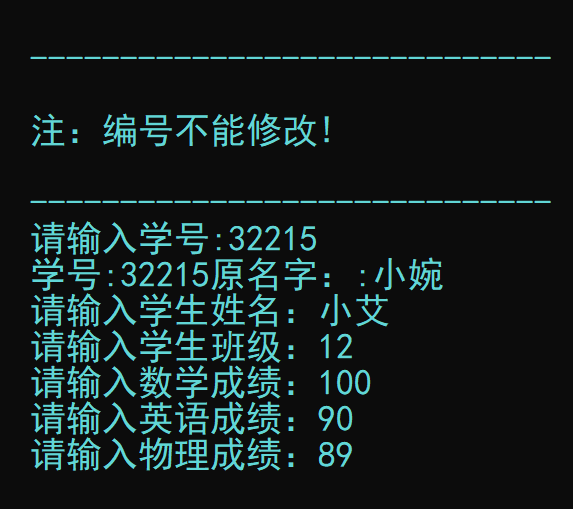
删除成功：

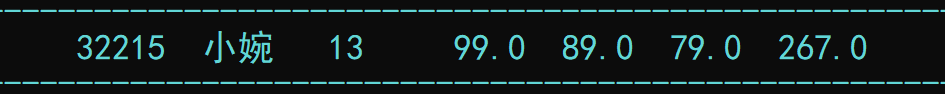


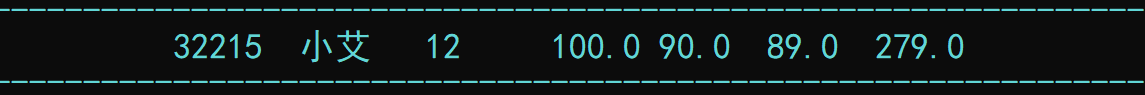
输入0：返回主界面

（8）修改信息

当管理员输入6进入修改信息界面，如下，我选择修改小婉信息：修改前后信息如下：

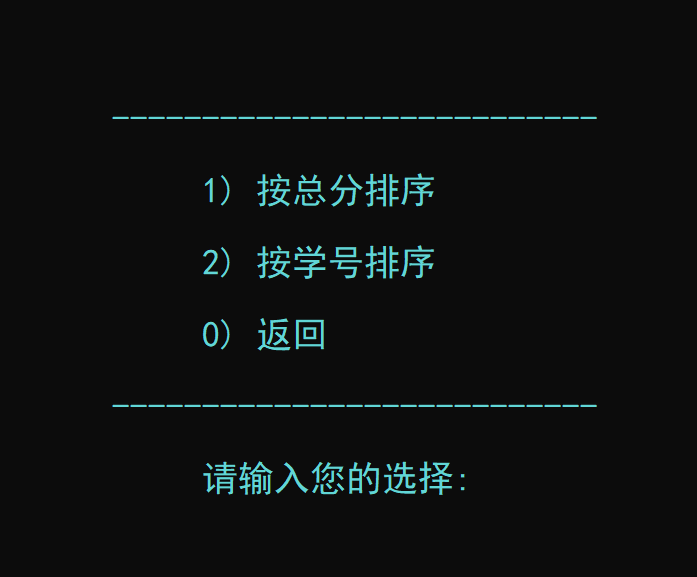
修改界面：

修改前：

修改后：

（9）排序

当用户输入7并按入Enter键时，即可进入记录排序界面，首先会显示排序前的信息，然后再按入Enter键，出现选择界面如下，输入1，按照总分进行排序；输入2，按照学号进行排序，如图：

排序界面：

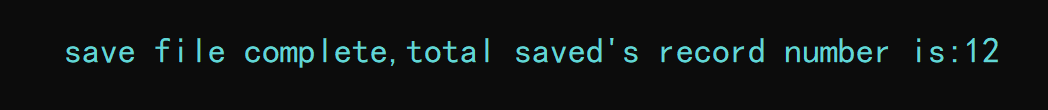
输入1：

输入2：

输入0：返回主界面

（10）保存信息

界面显示如下：



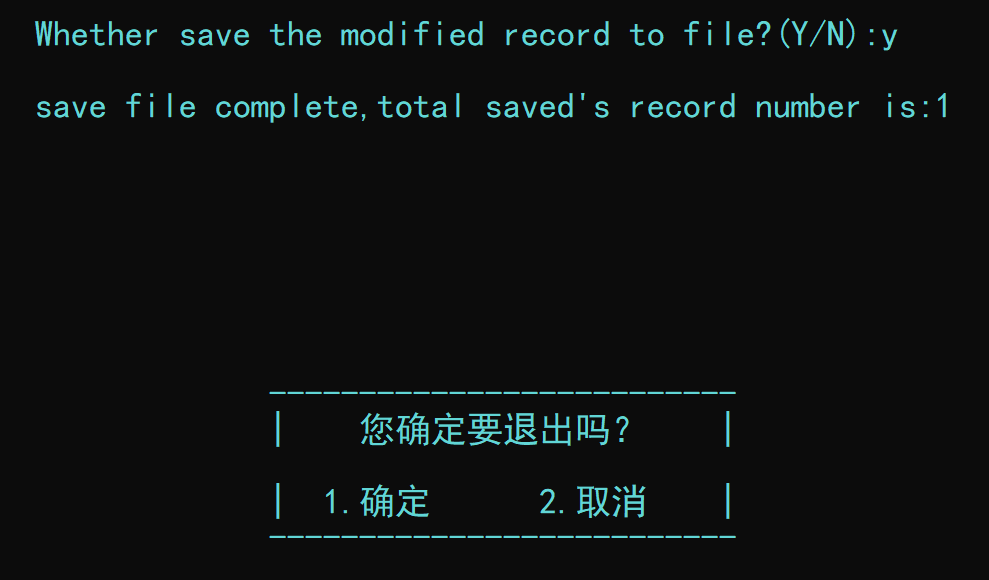
（11）复制文件

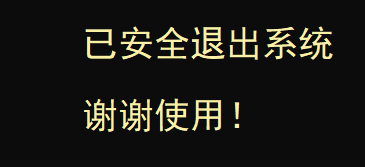
文件复制功能不在界面操作，系统内部执行次步骤，管理员可在建立的student.txt文件中查看是否复制，如图下：系统成功复制。



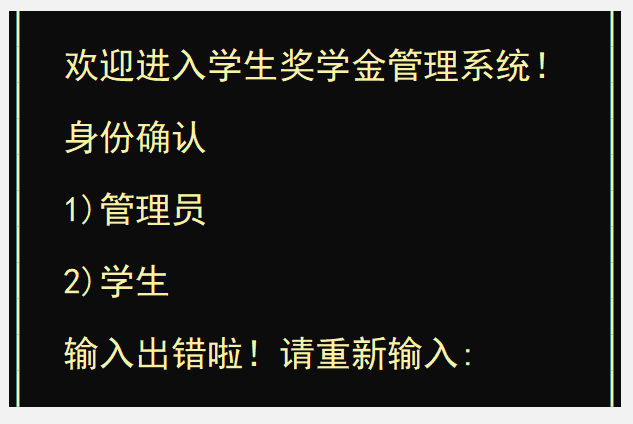
（12）退出系统

当管理员输入0时，退出系统可能有两种界面一种是直接退出，一种是当管理员忘记保存修改信息时，会跳出提示界面。实现如下：



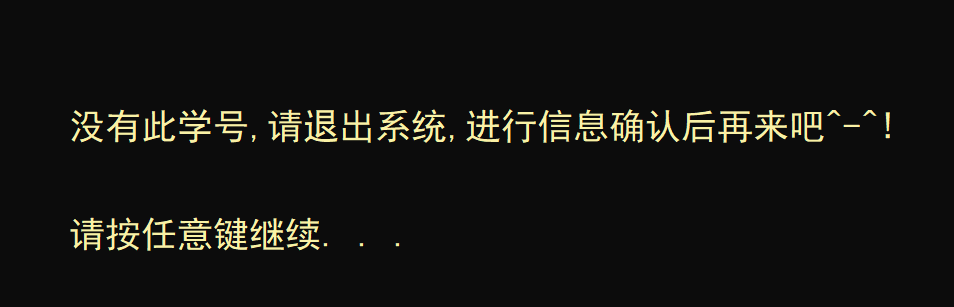


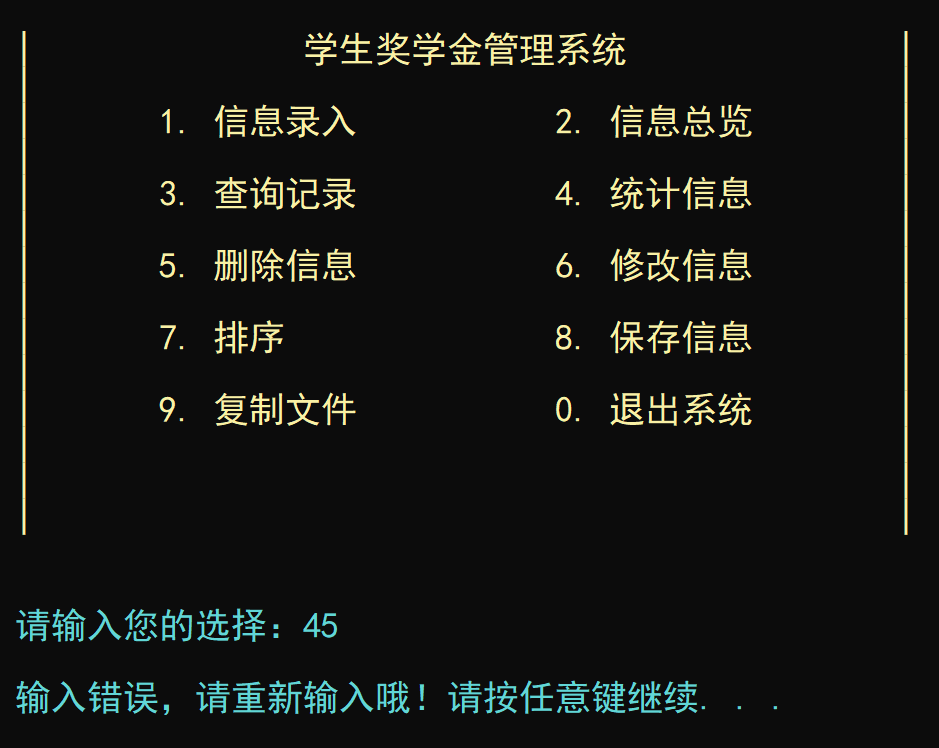
**6.2出错处理情况**

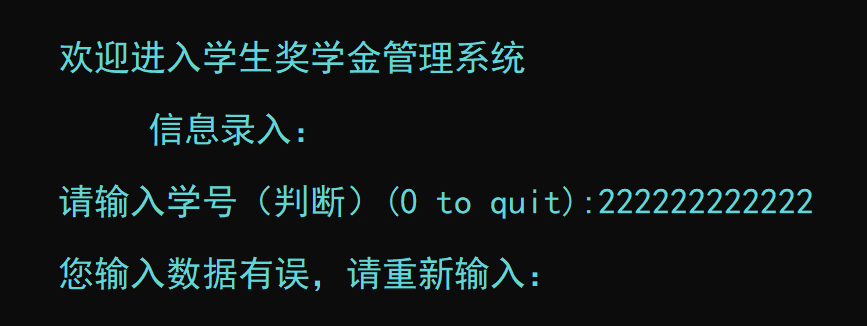
欢迎界面若输入其他：

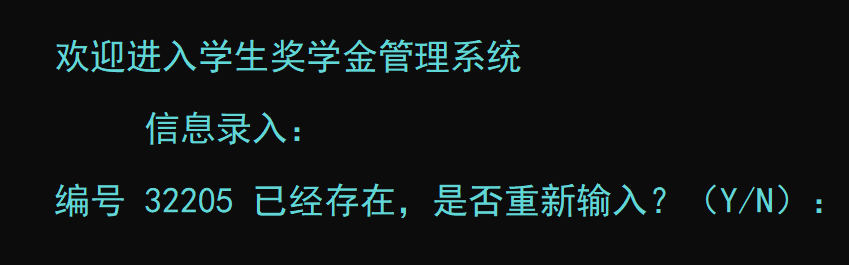
如果管理员密码错误：提示输入错误请重新输入

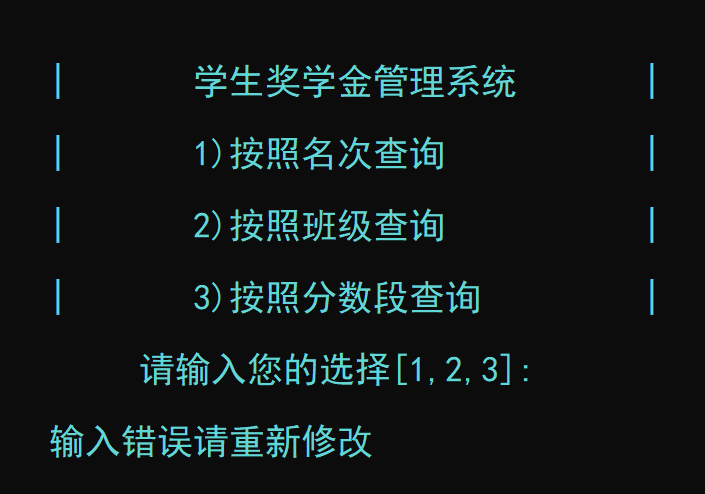
如果学生查询信息不存在会提示：

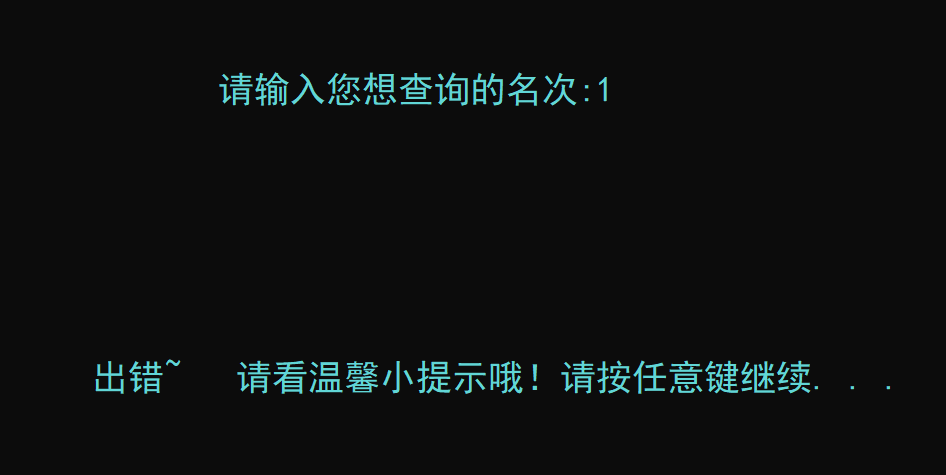
****

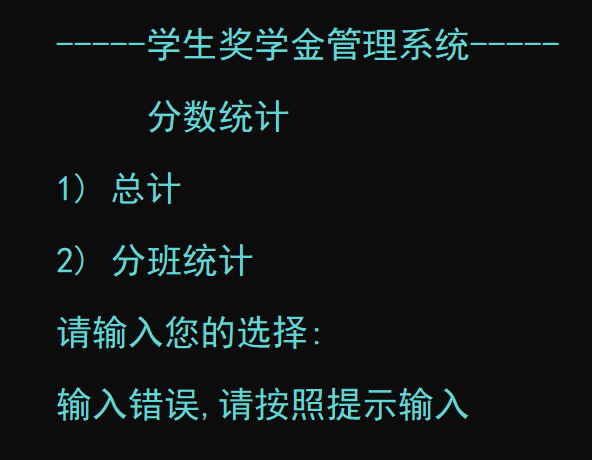
主菜单出错提示：****

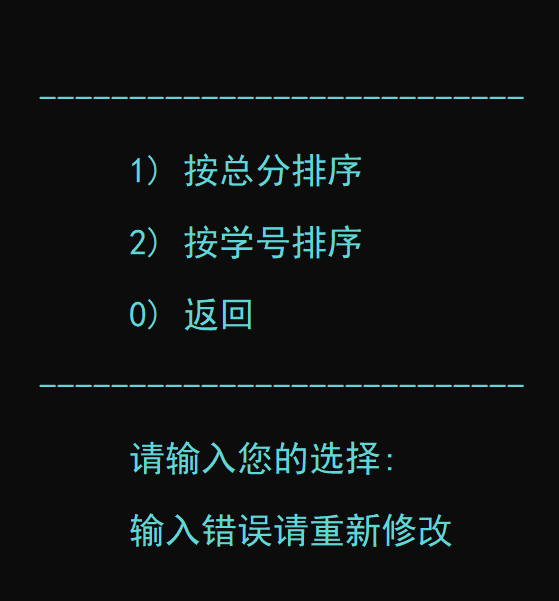
信息录入出错提示：****

提示编号重复：

名次查询出错提示：****

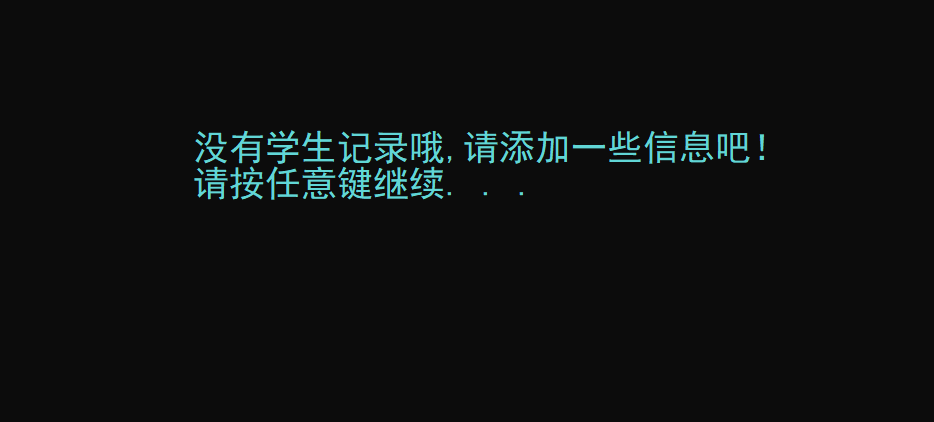
如果是在录入信息后没有查看温馨小提示内容，（没有进行总分排序）直接进行了名次查询，会出现如下界面: ****

统计模块出错处理：****

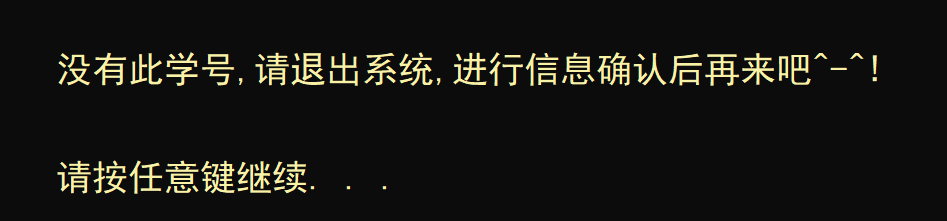
排序模块出错处理：

空文档提示：

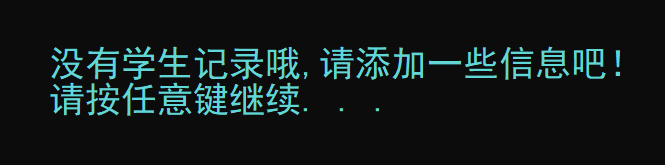
如果一开始是空文档，除了点开信息录入复制文件以及退出系统，点开其他都会提示“没有学生记录，返回主菜单信息录入

”

如果在没有数据打开系统，打开学生界面输入学号会提示：



选择除信息录入的功能之外会提示：



* 1. **体会总结**

**（课程设计完成了哪些功能，有什么扩展功能有什么地方需要改进，通过课程设计有何收获，调试程序的体会等）**

1. 存在的问题及解决方法

从一开始的代码设计到最终完成共耗时两周，这两周里感觉对c语言有了更深的理解，完全不像之前做的anyview，之前做的实验报告。是一种全新的体验。

从刚开始学习如何操作文件，如何操作链表，我首先写课设的代码使用的是直接头指针就存放数据，但是因为头指针就存放数据，如果我要查找删除，以及排序的时候使用选择排序法，就得先取下一个结点，再插入一个结点的时候，我都需要特殊考虑头指针是目标结点的问题，但是如果我用头结点的链表那么所存放的数据都可以视作普通情况。就感觉降低了某个问题的实现难度，而且减少了代码。

比如我排序用的是选择排序法，每一次内循环找到目标点，然后再判断是否是本次内循环的首结点，如果不是就交换，头结点的使用，当我的交换对象是第一个数据，只需要在头结点的后面插入这个结点。

然后还想到有一种就是只交换数据域的内容，感觉这个也很好实现，就是用头指针也很方便，不移动结点，只进行数据的交换。不过还没有操作过。

还有一个好处是：在遍历输出链表数据时，带头结点是while(head->next!=NULL)，而不带头结点是while(head!=NULL)，虽然头指针可以在初始时设定，但是对于特殊情况如只有一个节点会出现问题，就还要考虑这个情况。

只不过在使用头结点的时候传递的参数一定是二级指针，即指向头结点的地址。也就是保存单链表的首地址的指针变量。因为不带头结点的声明Node \*head 的时候，C编译器将其自动初始化为NULL，于是根本不需要调用Link head; 也即不带头结点的初始化是个伪操作。而带头结点的初始化在堆开辟了一段内存，需要修改head指针变量指向的地址(即head的值)，所以要修改head的值，必须传保存head变量的地址(即二维指针)。如果直接定义Node head相当于传head变量的值，那我要对单链表的内容进行修改的时候形参是不能传回实参的，也就是根本不会对实参的值产生作用。这个bug找了好久，主要是忘记了二级指针的这个操作~~~

还有一个就是每一次大致完成了一个模块，debug的时候停不下来，总是想，要是我输入这个系统会给我什么提醒，报错提醒如何设置，由于操作的不可控性，我主要加强了一些主流出错，比如如果数据是空的该如何反馈，如果用户误点如何返回主界面，比如用户输入了除规定选项外的其他输入，该如何提醒？

刚开始想到的使用if语句，分别对应情况，后来发现，代码太多，用if很难看清程序，比较麻烦，就想到用第二个版本，使用while语句，先用户输入选项，当选项不是其中的任何一个就执行while语句，也就是报错提醒和重新输入，感觉增加了代码的可读性。

第三个阶段就是设置全局变量，如果数据为空，那么每一个模块都应该提醒“抱歉，数据为空”我直接另设一个提醒函数，提醒函数传入的是单链表的首地址，在提醒函数里判断，如果头结点的后一位为空就打印提醒，并把全局变量temp置为1，然后再在函数首先初始化temp为0，再执行提醒函数，如果temp还是0说明数据不为空，可执行后续操作。

第四个阶段就是不用全局变量，直接把提醒函数设为int类型，如果数据空就返回1，否则返回0.没有用全局变量。

感觉这一步一步调试，一步一步debug的过程真的很上瘾，以及当我可以不断地使用更好的办法去解决一个事情的时候，感觉很自豪。

当然也有很痛苦的阶段解释对文件的处理这一块，感觉没有真正懂得如何二进制转换时，如何在文本文件里显示正确的格式，这一次由于时间的问题，我还没有得到很好的答案，在之后的时间，我一定要搞清楚这个问题。

还有就是在录入信息模块花费了很久的时间，首先要确定一个录入结束的判断，其次还要保证没有重复录入已有的编号，因为学生编号相当于他的代号，是独一无二的，要保证学号不能重复，我就再设了一个指针变量，用于遍历链表查询是否有重复，重复就会提醒管理员。

还有就是，花费了很多时间在界面的处理上，如何有效打印报错提醒。

写完程序的第一次整体观感，有很多重复代码，于是分别用宏定义定义一些常量，或者是格式化输入，减少了很多重复代码。

然后也上网学习了很多函数的用法，比如如何把字符数组学号转换为整形然后比较，就用了atoi函数，如何显示界面时更加模拟平常使用的程序，就采用了Sleep（）函数，还有就是设置颜色，对应坐标输出，使界面更加有设计感，还有就是不回显输入（这个部分还是有点没设计好比如输入密码部分应该也要用不回显输入，但有些地方有没必要使用getch）等等。

当然在整个系统运行的时候，也遇到了很多小细节问题，比如为什么一闪而过，为什么没有清屏等等，逐个击破，逐个解决。

1. 完成情况

2.1：完成的功能：

完成了信息录入的功能，可以从空链表开始也可以从已有链表开始，再追加信息；信息察看功能；查询记录，可以按照班级，名次，分数段查询；统计信息功能；删除信息，可以按照学号删除也可以按照姓名删除；修改信息功能；排序功能，按照学号排序或者按照名次排序；保存信息，复制文件，最后退出系统；也完成了管理员和学生的权限分开。

2.1：扩展的地方：

扩展的功能是：1.可以修改学生信息

2.使用了API函数，使界面美观，赏心悦目

3.使用睡眠函数，显示安全退出系统，用户友好型

4.详细的报错提醒，对每一个分界面都有“误点 友好返回”。

2.3：需要改进的地方：

功能大致完成，文件处理有欠缺。其实还挺想加入一些其他功能，但是由于时间问题，之后再进行改善。毕竟这是第一个自己写出来的程序！！！

功能改进：

1.修改功能还有待改进，当我要修改一个学生的信息，我可能只是想修改某一个局部，而不是所有信息，那除了修改的信息之外，是否其他信息可以不用重复输入呢？或者说，如果我可以直接把文件内容显示在界面上，然后在界面上用光标移动到我想修改的位置，然后直接修改，这样既方便了信息的总览，也减少了很多重复工作。

2.学生可执行的功能还有待开发

3.查询功能还有待增加，比如可以实现模糊查询（可用于管理员不确定情况下对某类学生信息的查询）

4.登录界面，如果管理员一直没有输入正确的密码，是否设定一个限制输入次数，超过次数，系统就会自动退出，关于密码，还有很大的改进，首先密码的输入是否显示，一般情况下是不会显示输入的密码的，其次密码应该有修改功能。本系统由于时间问题，没有设定这个功能，但本系统将在后续更新。

5.排序功能部分，如果遇到几个学生正好是相同的分数时，排序就不能以简单的名次排序，而是把相同分数的人是相同分数，之后的按照正常排序。其次排序应该也要设置每个科目的排序，这样子让每个老师可以大致了解自己学生的学习情况，但由于是一个奖学金管理系统，最主要知道的还是总体的排名

6.统计功能做的还不够好，没有真正实现方便管理员对学生成绩的分析，以及奖学金的分配，也就是最后要实现奖学金的评定，还需要管理员手动抉择，实际上一个良好的奖学金管理系统应该有自动生成一份奖学金评定报告的功能。

7.删除数据的部分也做得不是很友好，主要是没有恢复删除的功能，虽然有提醒界面，但是也不能排除忽略提醒而执行了删除，删除后这个数据就不能找回了。

8.文件处理部分使用了二进制打开文本文件，没有实现解决文本文件乱码的问题，在此次课程设计之后，一定要把这个问题解决！

9.还有一个图形化界面模块，如何实现系统界面显示和菜单选项控制，我感觉一个程序应该要有这两部分。

10.还有一个小细节，登录后系统为什么会闪一下，这个问题还没有得到解决。

代码改进：

1.下一次希望写程序的时候 有些可以使用if语句的地方是否可以改写为switch语句，上网搜索了一下，如果一个分语句要执行很多条语句，用switch效率会更高，而且会使代码简洁很多。

2.主函数有点复杂，应该把菜单选项放在菜单函数里，就可以只在主函数里调用菜单函数。

3.多文件编程自定义头文件，在这一次课设之后，一定要好好学会！

1. 收获和体会

首先在对于代码的设计上：如果不是此次课设有完整的题目，有完整的课设要求，我可能会对于做一个程序，所要花费的庞大精力，而望而却步，就是给你一个任务，你会不停的想他会有什么功能，他应该要执行什么操作，但是却不知道从哪一步开始。

这一次课程设计我体会很深的一点就是，如何设计一个程序，如何设计自己的代码，如何把自己所想用代码实现。首先我应该要了解清楚某个功能模块（或者整个功能）具体要干什么事情，才能够知道具体要如何做设计。而不是找一个设计方案，能够实现主要功能就行了，其他功能只需在其基础上修修补补。设计一定是从整体到局部设计出来的。从局部构造整体，不可能得到一个好的的设计。我最开始写过一段300行代码，但最后把这三百行弃掉了，最主要的原因是感觉自己的思路很乱，对我要做的学生奖学金管理系统没有一个整体的概念，以及对链表和文件的处理这两块不太熟悉。

在对于算法的设计上，有一种体会就是，自己心中所想难以用代码实现，说明自己的算法还有待加强，以及如何更加深刻理解自己语言工具，并利用自己的工具与成熟的算法，解决实际的问题，这是我最应该思考的事情了吧。

**八、参考文献**

参考了网上csdn的算法和csdn里相关知识和代码。

参考了microsoft的文件，关于对文件的处理以及简单界面函数的处理。参考了网上搜索的一些算法，如选择排序法，还有一些库函数的使用。

参考了图书馆七楼的一本书，具体是啥名字，忘记了，好像是《C语言课程设计》。