****

高级程序设计语言课程设计

设计题目:

C/C++实现学生信息管理

专 业：\_\_\_\_计算机类\_\_\_\_\_\_\_\_\_\_\_

班 级：\_\_\_\_\_\_\_03班\_\_\_\_\_\_\_\_\_\_\_\_

设 计 者：\_\_202100800538高捷\_\_\_

\_\_202100800609 孙义捷\_

\_\_202100800570林正阳\_

2022年 6月 8 日

本系统实现了学生成绩全部查询，搜索查询，录入，删除；学生和教师账号注册，删除等功能，各成员分工如下：

高捷：工作量34%

软件总体架构的实现 界面颜色设计 加载页面设计

管理员账户权限确定 教师账户权限确定 学生账户权限确定

功能类别设计 录入学生成绩功能主体实现

阅览学生成绩功能主体实现 查询学生成绩功能主体实现

更新学生成绩功能主体实现 删除学生功能主体实现

计算功能主体实现 排序功能主体实现 删除用户功能修补

软件运行测试 主体函数与各函数链接 用户注册功能实现

孙义捷：工作量33%

文档主体部分编写 流程图设计和制作 流程视频录制 软件运行测试 录入学生成绩功能修补 阅览学生成绩功能修补

查询学生成绩功能修补 更新学生成绩功能修补

删除学生功能修补 计算功能主体修补 排序功能主体修补

删除用户功能修补 用户类设计 类内函数设计 类内函数链接

界面结构设计 页面美观化设计

林正阳：工作量33%

删除用户功能主体实现 学生结构设计和链接

录入学生成绩功能修补 阅览学生成绩功能修补

查询学生成绩功能修补 更新学生成绩功能修补

删除学生功能修补 计算功能主体修补 排序功能主体修补

删除用户功能修补 链表插入方式设计 软件运行测试

流程视频录制和剪辑 软件运行测试 用户保存方式设计

学生成绩保存方式设计 用户注册方式设计

目录

目录

**[一、 系统介绍 5](#_Toc21073)**

[1. 系统功能介绍 5](#_Toc25450)

[2. 系统使用用户: 系统拥有方（管理员）、教师、学生 5](#_Toc14514)

[3.主要使用场景: 对学生成绩进行管理、查询 5](#_Toc15700)

**[二、 概要设计 5](#_Toc1903)**

**[三、 详细设计 5](#_Toc12494)**

[主函数 5](#_Toc11922)

[初始化函数 5](#_Toc21712)

[面板函数 6](#_Toc7139)

[管理函数 6](#_Toc4081)

**[四、 功能模块 7](#_Toc13285)**

[信息存储实现 7](#_Toc29666)

[用户记录的删除 8](#_Toc27601)

[学生信息的删除 8](#_Toc27134)

**[五、 系统实现 9](#_Toc21629)**

[登录与注册界面 9](#_Toc24863)

[加载界面 10](#_Toc9542)

[主界面 10](#_Toc28507)

**[六、 测试中的问题和解决方式 18](#_Toc3956)**

**[总结： 18](#_Toc11855)**

# 系统介绍

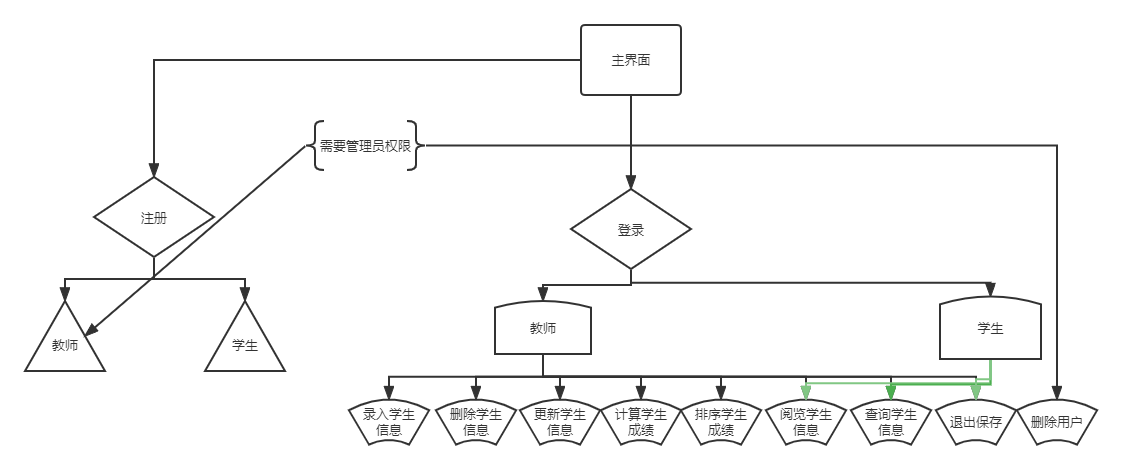
1. 系统功能介绍
2. 进行登录使用的账户注册和账户登录
3. 根据登录账户类型（教师或者学生）决定可运行的功能。
4. 学生信息录入、删除，更新，计算，排序（需要教师或者管理员权限）；学生信息查阅、查询（分为姓名查询，学号查询，性别查询）（这两项功能无需特别权限）；删除登录使用的账户（需要管理员权限）。
5. 系统使用用户: 系统拥有方（管理员）、教师、学生

3.主要使用场景: 对学生成绩进行管理、查询

# 概要设计

数据结构:

本程序设计中数据结构主要为两个主体:用户和学生。用户通过c++的类进行数据的录入，然后利用iFile功能修改存储用的user.txt文档备份、读取相关信息进行登录和权限确定；学生则以结构链表形式保存，链表类是一个模板类，可以讲结构体转化为链表，只要定义了学生结构体，套用在模板类中，就可以对其进行链表操作了。在登录时，遍历学生链表，判断能否修改，在搜索时，查找链表，在进行修改链表信息的功能时将修改同步储存在磁盘中。



# 详细设计

主函数

设置i变量用while循环表示主界面中登录和注册过程，登录完成后i变换进入下一步，设置head学生信息struct的头，然后进入菜单函数menu().

初始化函数

主要以user的READ函数为主，用于读取已有的user数据进行比对。

面板函数

主要由start()函数和menu()对登录、注册界面和功能界面的主体进行管理，start下连user类中的login和register函数进行登录和注册，menu函数则连接了search\_record()、delete\_record()、change\_record()等实现系统功能。

两个函数都应用了SetColorAndBackground()函数来调整界面的背景颜色，使界面更加美观。

管理函数

根据登录类别是教师还是学生或者管理员决定接下来能够使用的功能，假若权限不够会弹出提示，并重新加载界面。

都可使用的功能:

查询学生信息:

查询可以通过学号、姓名、性别三种元素查询，主要通过在链表中遍历查找。查询后可选择返回或者继续查询。

列出学生信息:

通过for循环对链表逐个表达，显示学生姓名、学号、性别、年龄、手机号、语文成绩、数学成绩、英语成绩，并较为美观的隔开方便观看。

计算学生成绩:

根据需求可进行平均分计算和总分计算。计算成绩为全体学生都进行计算，拥有翻页显示功能，信息较多可逐页查看。

教师权限使用的功能为：

录入学生成绩，更新学生成绩，删除学生信息，排序学生成绩。实现原理分别如下:

录入学生成绩:

首先判断是否有权限，无权限则返回，有权限后创建新学生信息结构体，依次输入名字，学号，性别，年龄，手机号，语文、数学、英语成绩。确定保存后通过尾插法插入链表。

更新学生成绩:

判断权限后输入学生学号，遍历法搜索链表中是否有对应学号的结构，然后录入新成绩保存，显示新成绩后选择返回或继续更新。

删除学生信息：

根据学号进行遍历查询，找到对应学生信息后确认删除，并重新连接链表确保不断链。选择继续进行删除或者返回主界面。

排序学生成绩:

根据输入选择不同的排序方式，分别为按学号排序、按语文、按数学、按英语、按年龄。排序会改变链表中的学生信息顺序，使用时需注意学生信息顺序的改变。可以直接退回主界面不进行排序。

管理员权限使用功能:

删除用户:

当登录管理员账户后，所有功能都可使用。删除用户时通过对user.txt读取查找是否有对应账户，确定有后将账户的手机号、密码、权限等级覆盖为空，实现账户的删除功能。

1. 功能模块

信息存储实现

学生信息

通过struct student建立链表临时存储学生信息，main函数中首先建立了head头指引，录入时进行尾插录入链表，根据输入决定保存与否。若保存，转至save\_record函数，通过FILE函数打开goblinimformation.txt并用fprintf()将链表输入至goblinimformation.txt，fclose关闭FILE指向，在下次进入程序时读取goblinimformation.txt获得已有学生信息存入链表。

账户信息

利用类User建立账户信息，void User::save()用于保存注册的用户，首先建立ofstream ofile然后ofile.open("user.txt", ios::out)打开存储用户信息的user文档，将新注册的账户的手机号、密码和权限等级录入后ofile.close()关闭，结束函数。User.txt保存后可在登录和注册教师账户时用于读取来对比账号和密码的错误与否，在进行功能的使用时也会确认账户权限等级，判断功能的使用是否可行。

用户记录的删除

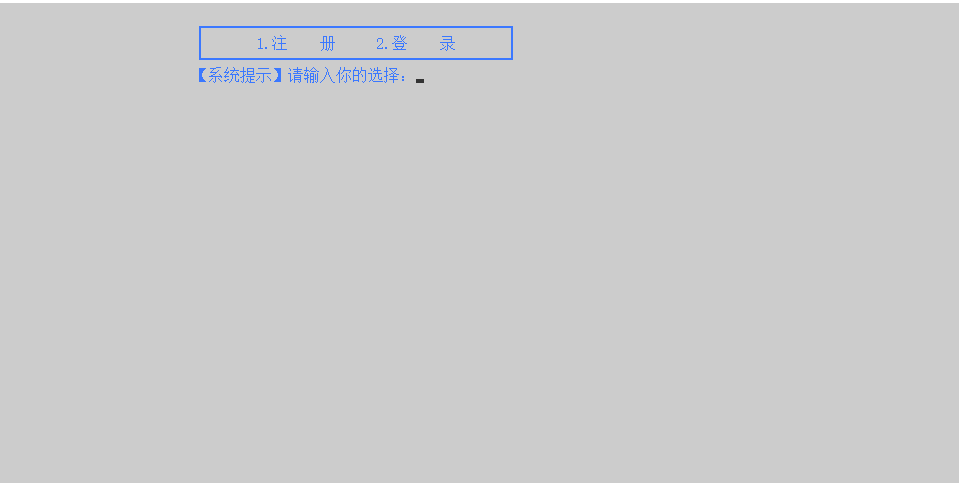
用户记录删除主要依靠void User::READ()函数对user.txt进行读取来对比删除的成功与否。将user.txt 输入到函数void deleteByName()中，对输入的手机号进行搜索，然后利用中间文件temp.txt 将用户的手机号、密码和权限等级覆盖为空，最后对user.txt重新保存备份。

学生信息的删除

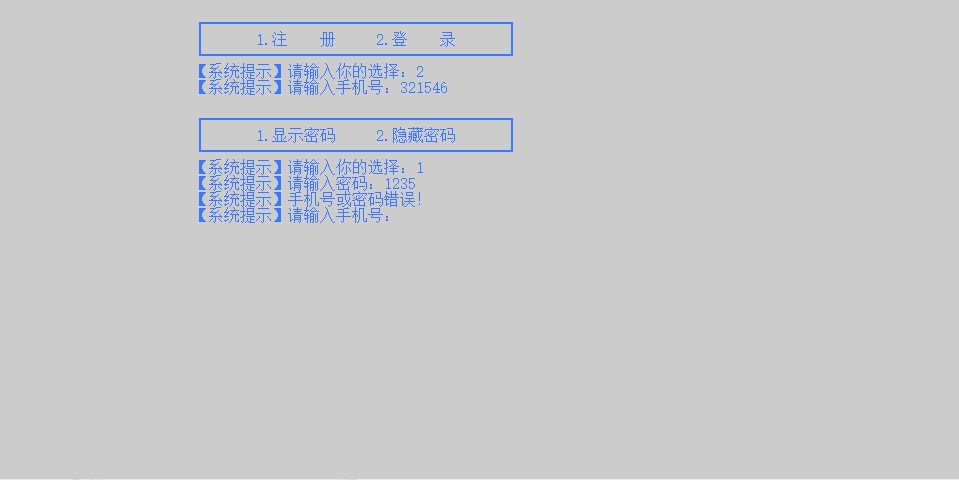
学生信息的删除通过链表实现。查找到对应学生信息在链表中的位置后清除该结构体， 并链接前后结构，最后通过save\_record()函数将改变后的链表即时存入goblinimformation.txt备份，确保不会出错。

# 系统实现

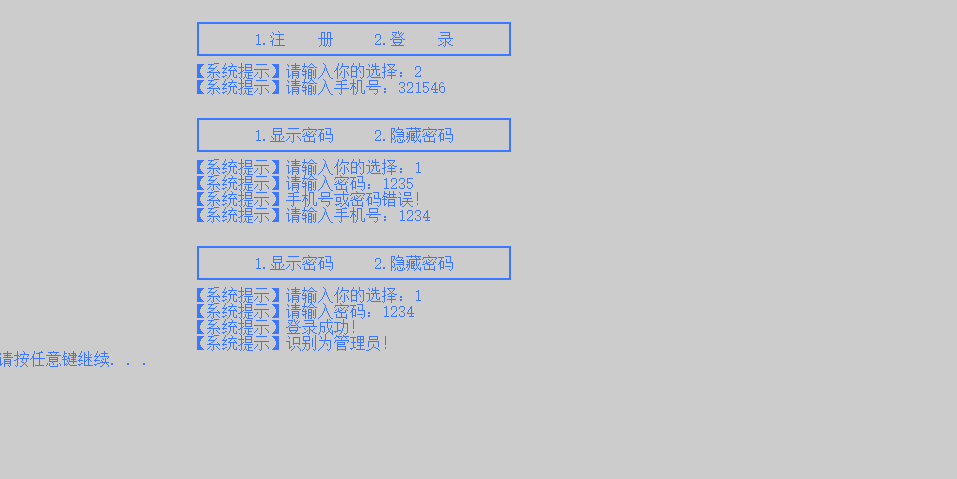
登录与注册界面



登录出错



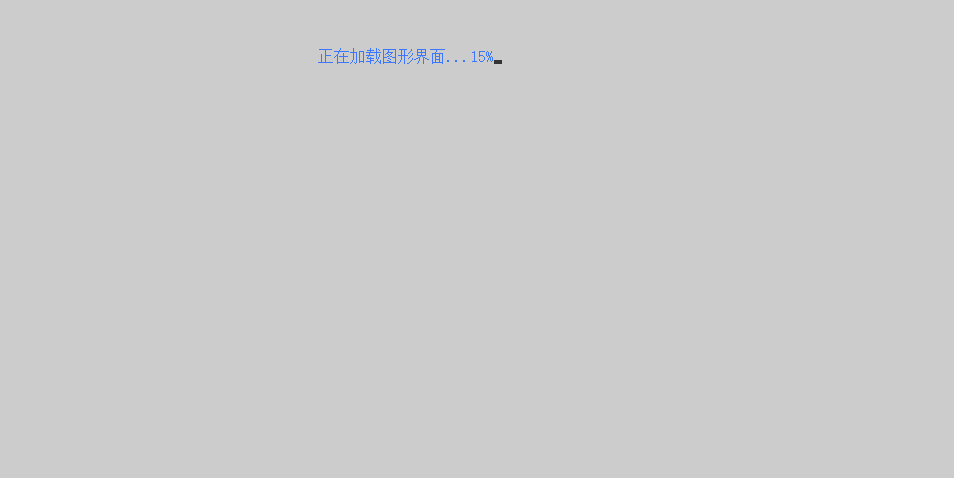
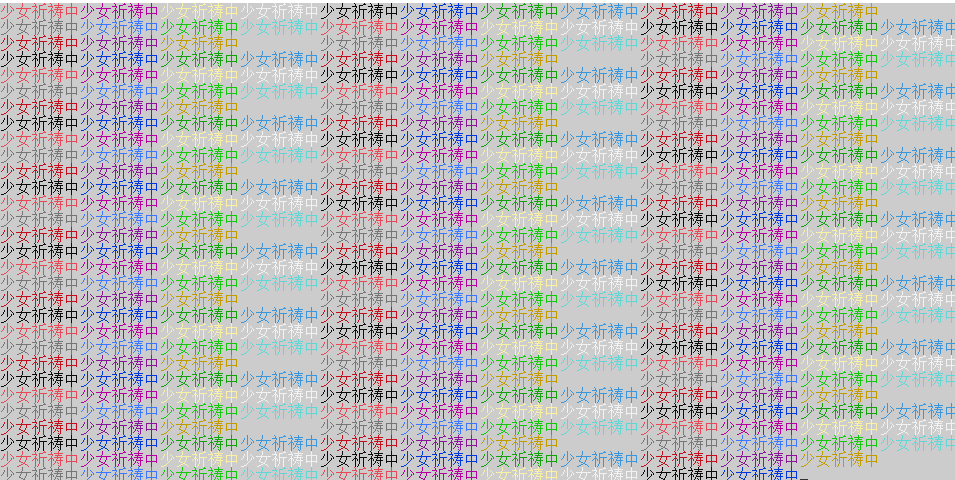
登录成功



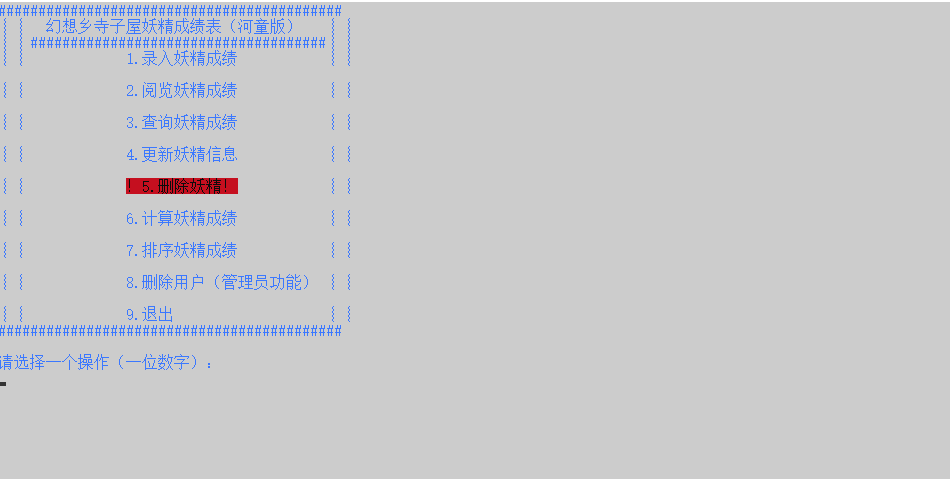
注册格式出错

注册教师时管理员账户的登录

加载界面

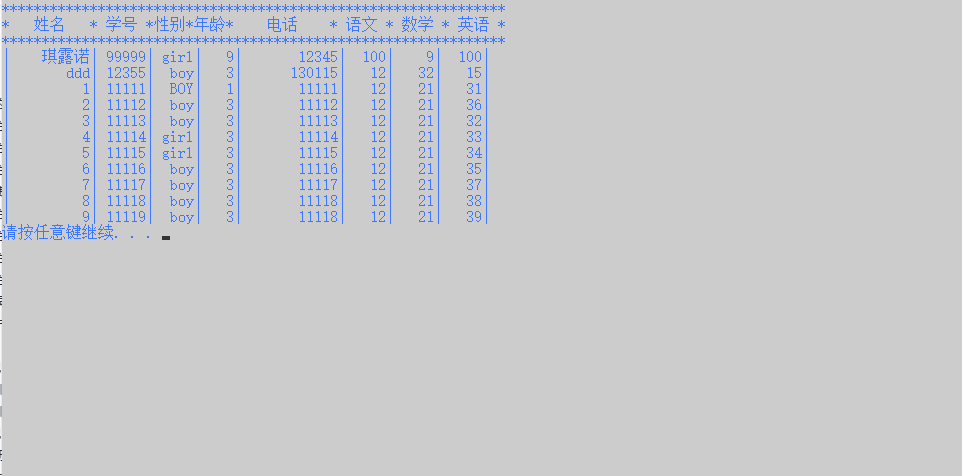
主界面



录入学生成绩



阅览学生成绩



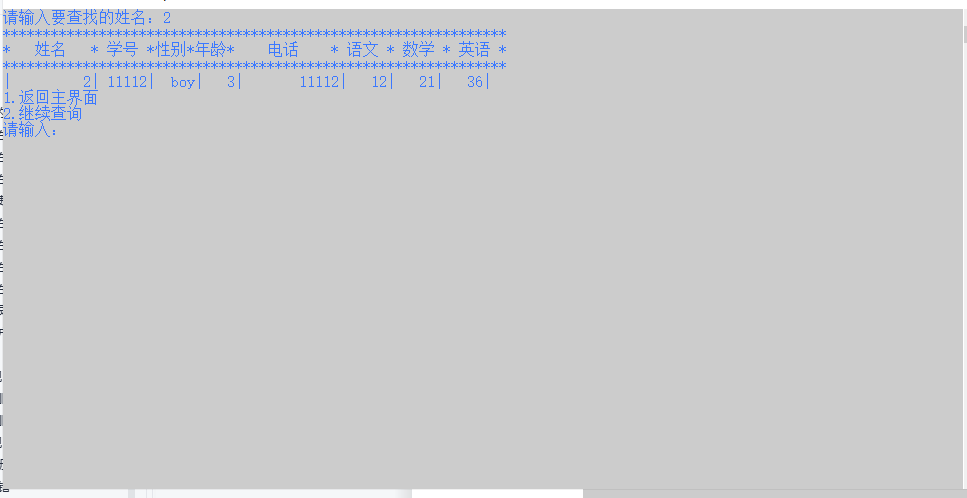
查询学生成绩

查询方式



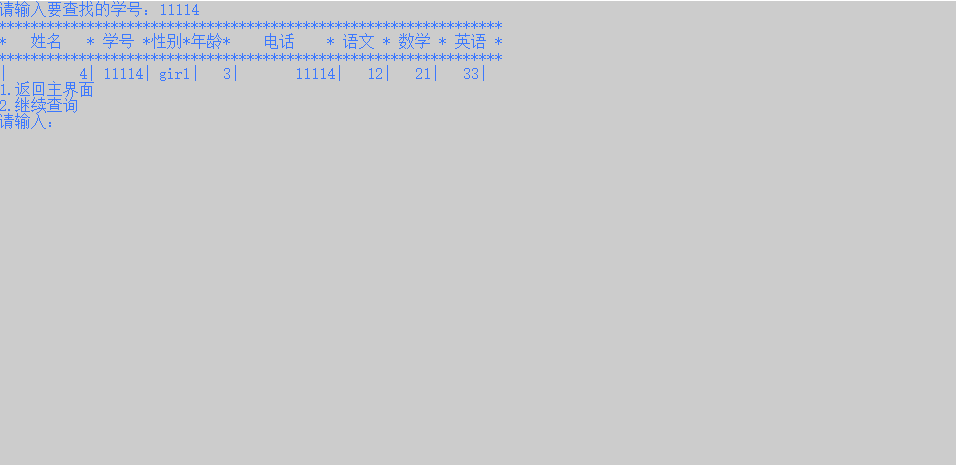
姓名定向





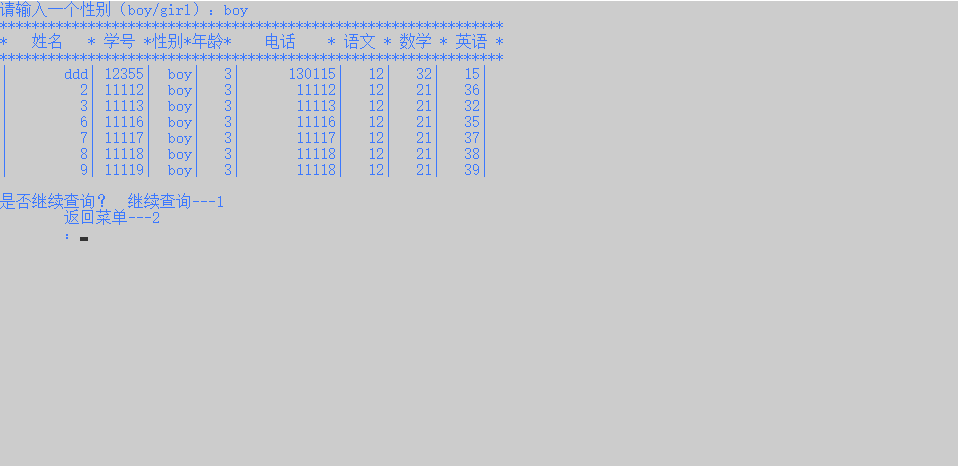
学号定向



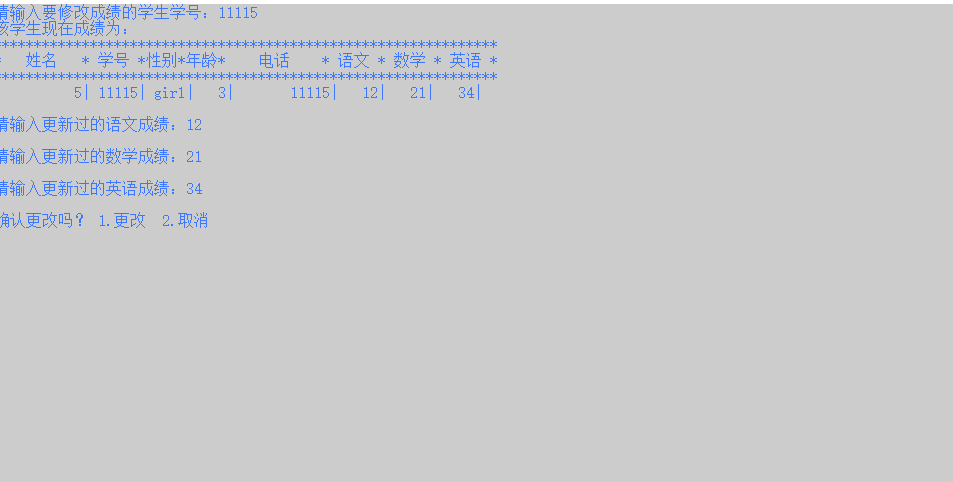


性别批量





更新学生信息



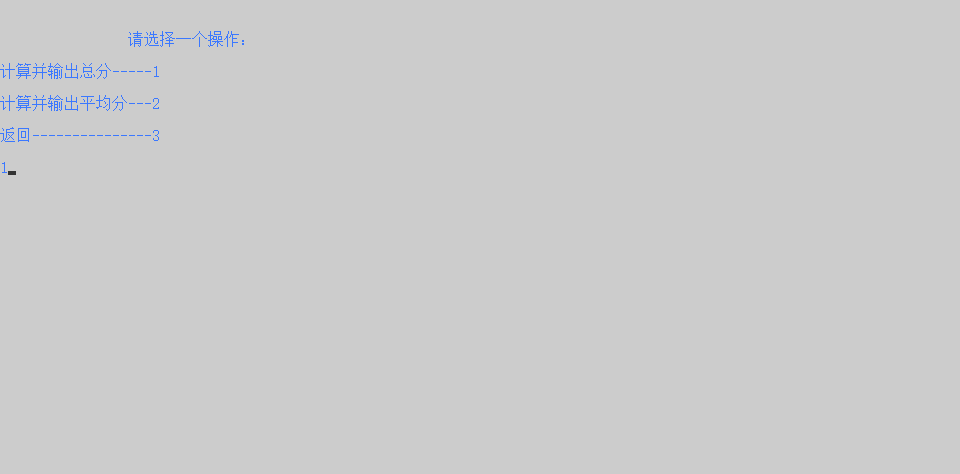


删除学生

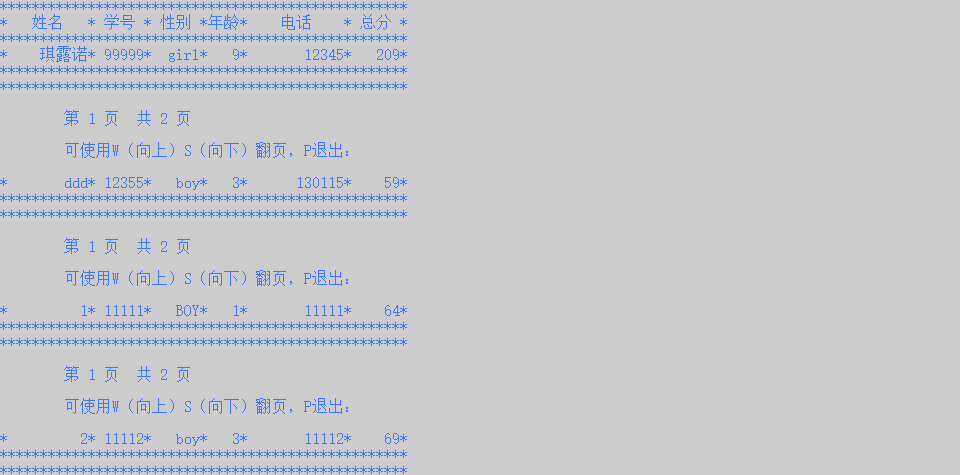




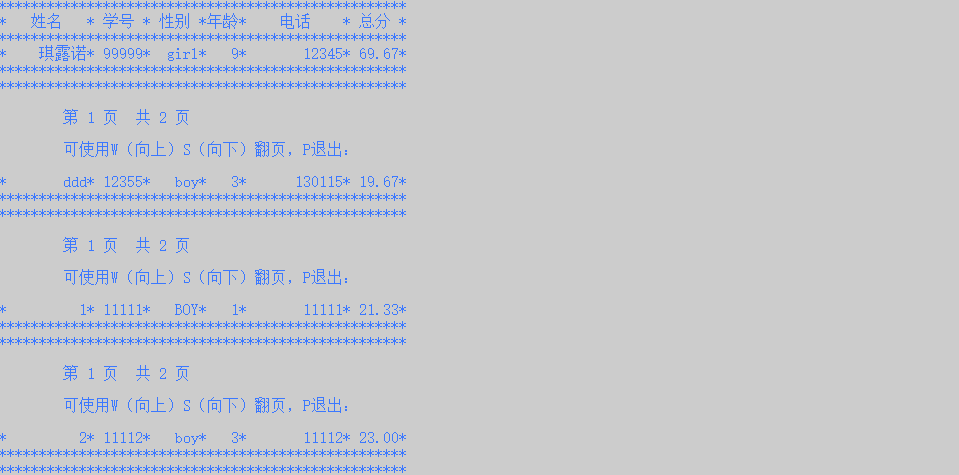
计算学生成绩



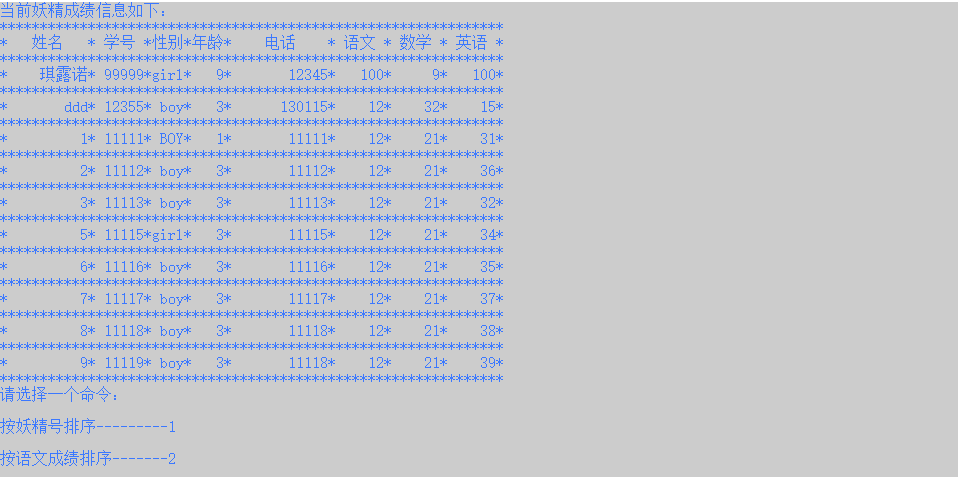
计算总分



计算平均分

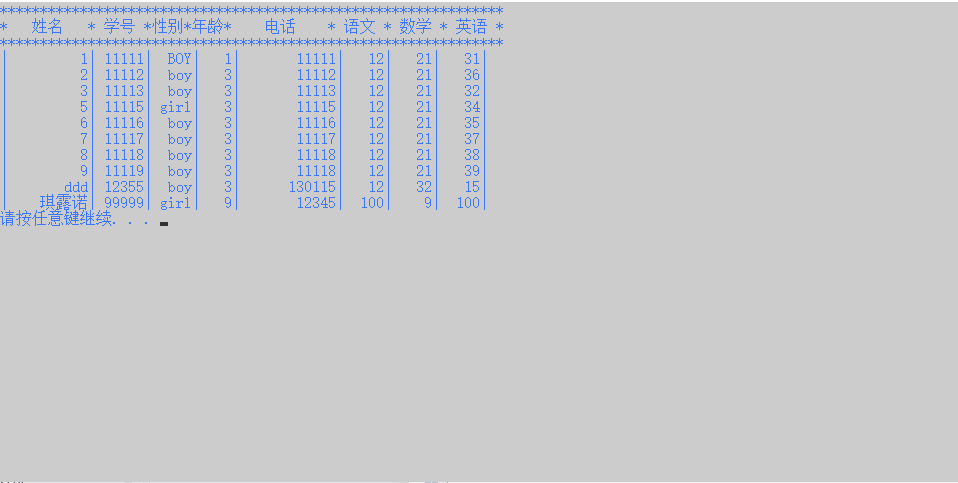


排序学生信息

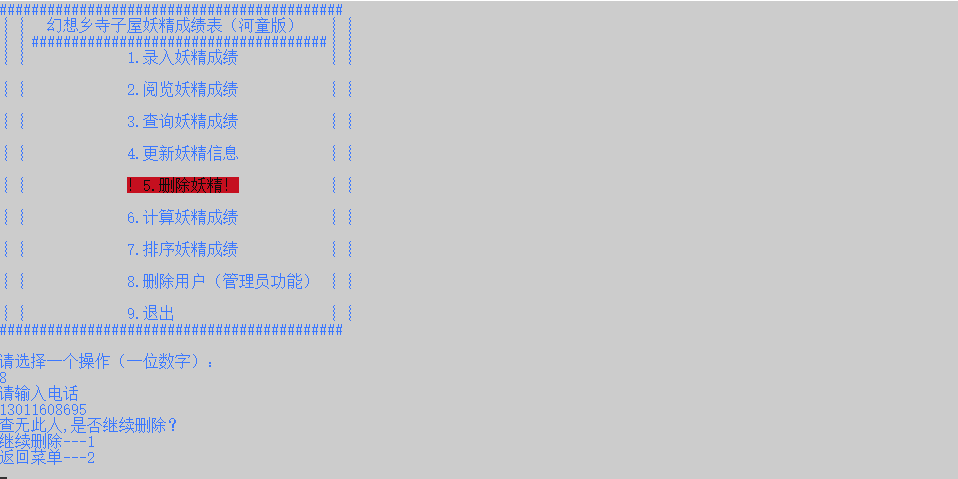




按学号排序



删除用户



1. 测试中的问题和解决方式

**问题：**在设计删除用户模块时，遇到了会把与目标字符串相同的字符串删除的问题。

**解决方式：**因为在文件中每个用户的信息分为三行存储，且删除的逻辑是创建一个中间文件，把出了目标字符串的字符串从原文件写入中间文件，再把删除成功后的剩余字符串从中间文件重新覆盖写入原文件。所以就定义了一个变量j，当读取到目标字符串时，使其加一，当目标字符串全部读取完毕时，变量j就让程序不进入删除的循环，从而解决了会把与目标字符串相同的字符串删除的问题。

**问题：**在程序输入输出时，在处理一些特殊输入时会出问题，因为程序在修改前有些部分采用了for循环cin读到一个特殊的数再停止的方法，这就会导致输入字符串中的空格、换行等字符对下一次输入造成影响。

**解决方式：**使用getline等忽略空格、换行符的方式进行读取字符串。

**问题：**在读取文件内容时，某些情况下会出现文件中存在的换行符影响输入的问题。

**解决方式：**每读取一行都使用一次读取掉换行符的函数。

总结：

**沟通协作**

小组成员之间的沟通协作十分重要，在程序编写的准备阶段，小组成员之间要做好充分的沟通工作，分配好各自负责的部分，定义好每一个函数接口，以便于进行衔接。

**命名规范**

小组成员之间要建立一个统一的命名规范，不能随意给函数、变量等取名字，以免造成小组成员之间沟通的困难，在互相看对方写的代码的时候，很有可能出现因为随便取名字而不知道函数模块、变量作用的现象，所以在编写程序之前一定要建立一个统一的命名规范，以便增加程序的可读性。

**注释**

在编写代码的时候一定一定要写注释！不要因为一时之便而不加注释，不加注释的话，不仅对修改自己写的代码的人来说是一个灾难，而且等到隔了一段时间来修改自己之前写的代码时，就会发现不加注释，自己都看不懂，所以不要做编写代码不加注释这种既恶心自己，又恶心别人的行为。

**合理使用技巧来缩减代码长度**

在编写函数时，可以合理使用函数重载、模版类、函数模版等技巧来重复利用已经编写好的代码。一方面缩减了代码体积，又一方面提高了程序可读性。

**分模块调试**

当程序出现问题时，可能只是某个函数的某个部分出现问题，在主程序中调试可能比较麻烦，所以就可以把可能有问题的函数模块单独拿出来进行调试，以提高调试效率。另外，一个函数模块一定要经过测试，没有问题了之后再与主体部分进行衔接，以减少出现主程序出现bug的问题。