长 沙 学 院

课程设计说明书

|  |  |
| --- | --- |
| 题目 | 报到系统的模拟设计 |
| 系(部) | 数学与计算机系 |
| 专业(班级) | 软件工程专业（五班） |
| 姓名 | 谢晗阳 |
| 学号 | B20150304523 |
| 指导教师 | 刘钢钦 |
| 起止日期 | 2015-12-28 至 2015-12-29 |

项目实训任务书

**实训题目：**报到系统的模拟设计

1. **已知技术参数和设计要求：**

报到系统业务描述如下：

某学校为摆脱招生困难的困境，决定降分录取。于是向符合降分后录取条件的学生发送了通知书。因为考虑到并非所有收到通知书的学生都会来报到。于是请你帮他们开发一款报到系统，方便统计已经报到的学生信息。这些信息包括学生的姓名、语数外三门课的成绩以及这三门课的总分和已经交费的情况。学校提出的基本功能如下：登录系统、报到功能、缴费功能、查询功能、排序功能等。本系统各个业务有如下注意点：

1. 登陆系统

用户登陆之前，必须在登陆界面输入其账号，通过其正确输入的密码登陆系统。

1. 报到功能

报到功能主要是管理人员通过查询学生的材料将学生的信息（姓名、成绩等）填入系统中。

1. 缴费功能

管理人员通过收取学生的费用后将学生的学费情况填入系统中。这里我们假设学生学费统一收取5000元。那么学生可能在0~5000范围内缴费。

1. 补交功能

学生可以在系统中全部或者部分补交所欠学费。是否欠费或者欠费多少，程序应该可以判断。程序应该能通过计算列出该生欠缴的学费额度。

1. 查询功能

管理人员应该可以通过学生姓名查询学生的情况。包括姓名、成绩和学费欠缴情况。也可以通过设定条件如：总分低于220分的，语文低于80分等来查询。

1. 排序功能（可选）

程序应该能将学生基于某一门成绩进行冒泡排序。

**二、各阶段具体要求：**

1、需求分析阶段

熟悉系统业务，从业务中抽取出系统的需求，形成完善的需求说明书。

2、系统设计阶段

根据需求，进行程序设计，包括定义系统的界面、定义系统数据的存储方式等，形成完善的设计说明书。

3、编码实现阶段

（1）完成代码编写

（2）要求代码编写规范

4、系统测试阶段

（1）完成功能调试

（2）要求完成必要的测试工作

5、交付实施阶段

（1）提交可正常执行的系统

（2）提交系统需求说明书、设计说明书、程序代码

（3）撰写课程设计报告书

（4）要求规范地书写文档

**三、设计工作量：**

（1）软件设计：完成问题陈述中所提到的所有需求功能。

（2）论文：要求撰写不少于3000个文字的文档，详细说明各阶段具体要求。

**四、工作计划：**

理论课结束后两周进行课程设计，软件开发如下，一周完成。其中，两教学课时用于题目分析与介绍。其他教学可是用于程序设计。

1) 选定题目

2) 需求分析

3) 系统设计

4) 编码实现

5) 系统测试

6) 交付实施

**五、注意事项**

**1）提交文档**

长沙学院课程设计任务书 （每学生1份）

长沙学院课程设计论文 （每学生1份）

长沙学院课程设计鉴定表 （每学生1份）

指导教师签名：　　 　　　　日期：

教研室主任签名：　　　　　　　 日期：

系主任签名：　　　　　　　　　 日期：

长沙学院课程设计鉴定表

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | | 谢晗阳 | | 学号 | B2015030453 | 专业 | 软件工程 | | 班级 | | 五班 |
| 设计题目 | | | 报到系统的模拟设计 | | | | | 指导教师 | | 刘钢钦 | |
| 指导教师意见：  评定成绩： 教师签名： 日期： | | | | | | | | | | | |
| 答辩小组意见：  评定成绩：　　　　　答辩小组长签名：　　　　　日期： | | | | | | | | | | | |
| 教研室意见：  最终评定等级：　　　　　教研室主任签名：　　　　　日期： | | | | | | | | | | | |
| 说明 | 课程设计成绩分“优秀”、“良好”、“中等”、“及格”、“不及格”五等。 | | | | | | | | | | |

目 录

第1章 需求分析……………………………………………………………………1

1.1 设计需求……………………………………………………………………1

1.2 任务需求……………………………………………………………………1

1.3 技术需求……………………………………………………………………2

第二章 报道系统的设计……………………………………………………………3

2.1 设计分析……………………………………………………………………3

2.2 设计主题……………………………………………………………………3

2.3 设计方案……………………………………………………………………3

2.3.1 用户选择界面………………………………………………………3

2.3.2 学生登陆界面………………………………………………………4

2.3.3 学生导航界面………………………………………………………4

2.3.4 学生补交界面………………………………………………………5

2.3.5 学生改密界面………………………………………………………6

2.3.6 管理员登陆界面……………………………………………………7

2.3.7 导航界面……………………………………………………………7

2.3.8 报道服务界面………………………………………………………8

2.3.9 缴费界面……………………………………………………………9

2.3.10 查询界面……………………………………………………………10

2.3.11 排序界面……………………………………………………………12

2.3.12 修改账号密码界面…………………………………………………13

2.3.13 文件读取和录入……………………………………………………13

2.4 设计流程图…………………………………………………………………14

第三章 结论…………………………………………………………………………15

附 录…………………………………………………………………………………16

# 第1章 需求分析

## 1.1 设计需求

报道系统能为用户提供登陆、报到、缴费、补交、排序、改密的功能。为了模拟真实的报道环境，本系统必须实现登陆、报到、缴费、补交、排序、改密的功能。管理员通过登陆系统自带账号密码，在系统中实现登陆、报到、缴费、补交、排序、改密的操作。学生通过登陆学号和相应密码，在系统中实现补交、改密的操作。

## 1.2 任务需求

为了使用户使用起来更方便，看起来更明了，同时也满足用户在报道系统中功能到位，本系统应包含这么几个业务：

1. 登陆系统

用户登陆之前，必须在登陆界面输入其账号，通过其正确输入的密码登陆系统。

1. 报到功能

报到功能主要是管理人员通过查询学生的材料将学生的信息（姓名、成绩等）填入系统中。

1. 缴费功能

管理人员通过收取学生的费用后将学生的学费情况填入系统中。这里我们假设学生学费统一收取5000元。那么学生可能在0~5000范围内缴费。

1. 补交功能

学生可以在系统中全部或者部分补交所欠学费。是否欠费或者欠费多少，程序应该可以判断。程序应该能通过计算列出该生欠缴的学费额度。

1. 查询功能

管理人员应该可以通过学生姓名查询学生的情况。包括姓名、成绩和学费欠缴情况。也可以通过设定条件如：总分低于220分的，语文低于80分等来查询。

1. 排序功能

程序应该能将学生基于某一门成绩进行冒泡排序。

（7）改密功能

程序应该能更改管理人员账号密码和学生的密码。

1

## 1.3 技术需求

使用结构体储存学生数据，同时同步把数据保存至文件中，使用系统自动调用之前的数据，自动更新数据。switch函数的使用使得用户在选择功能时变得更加的方便以及简单。以及函数的调用，使得 main()函数代码量大大减少，程序更加清晰。使用清屏函数，使运行时屏幕保持整洁，输入密码时使用\*代替输入数据，使运行的安全性高，数据最后都储存在文件中，开始通过文件读取数据，方便多次使用。使用for(;;)来继续循环等等。

具体操作技术需求：

（1）对于每个功能模块，应划分成操作层和数据层

（2）对于每个操作层，应尽量考虑用户操作的简单，便捷

（3）对于每个数据层，必须使用结构体进行管理

（4）在用结构体进行管理时，必须用到数组的检索和排序

（5）对于数据层的操作完的数据必须保存到文件中

2

# 第二章 报道系统的设计

## 2.1 设计分析

报道系统能为用户提供登陆、报到、缴费、补交、排序、改密的功能。为了模拟真实的报道环境，本系统必须实现登陆、报到、缴费、补交、排序、改密的功能。管理员通过登陆系统自带账号密码，在系统中实现登陆、报到、缴费、补交、排序、改密的操作。学生通过登陆学号和相应密码，在系统中实现补交、改密的操作。

## 2.2 设计主题

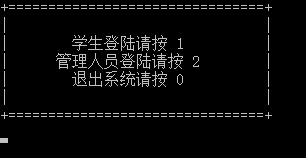
本报道系统的设计主题是结构体。从main函数出发，涉及各个方面。

## 2.3 设计方案

首先通过欢迎界面，进入用户选择界面。选择后进入相应功能界面：学生的补交和改密，管理者的登陆、报到、缴费、补交、排序、改密界面。最后在进过main函数中多次循环直至用户选择退出系统。

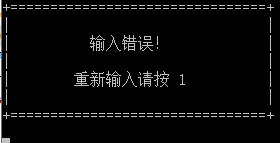
### 2.3.1 用户界面

由用户选择是学生还是管理人员，输入相应编号进行下一步。还可以选择退出此系统。输入错误会提示错误。本界面通过if…else…进行选择。



用户选择界面

3

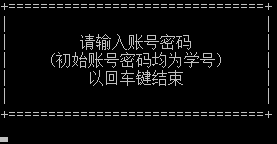


输入错误报错界面

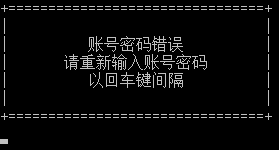
### 2.3.2 学生登陆界面

账号密码默认都是学号，后面可进行修改。输入错误会提示请重新输入。

通过getch()函数的妙用使密码输入用\*号显示。调用自定义validate\_stu(count,zh,mm)；函数判断账号密码是否正确。



账号密码输入界面

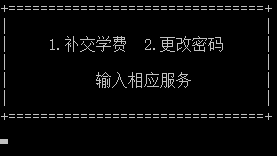


输入错误时提示界面

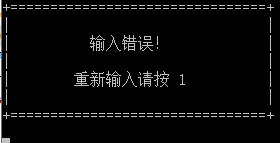
### 2.3.3 学生导航界面

选择是补交学费还是修改密码，输入错误会报错。本界面通过if…else…进行选择。

4



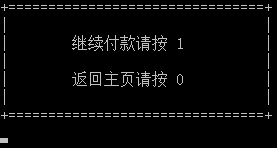
学生登陆导航



学生导航输入错误

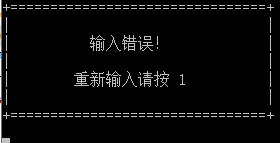
### 2.3.4 学生补交界面

进入补交界面或者付款成功后都会进入交费选择界面，选择继续付款或者返回用户选择界面，输入错误同样会报错。选择继续付款会提示付款金额，输入金额后会提示付款成功，还会提示当前欠费。本界面通过if…else…进行选择。调用自定义payment\_stu(count,zh);函数进行补交。

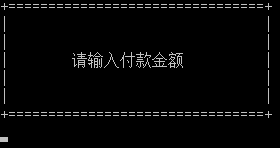


交费选择界面

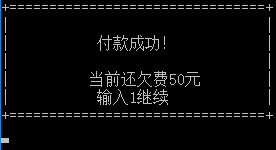
5



学生补交界面输入错误



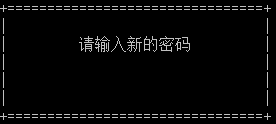
付款金额输入



付款成功界面

### 2.3.5 学生改密界面

进入改密界面会提示用户输入新的密码。输入成功后会提示。返回会进入用户选择界面。通过gaim\_stu(int count,char zh[])函数进行改密。密码储蓄在stu[].mm[]中。



输入新密码界面

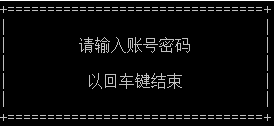
6



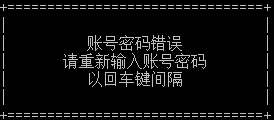
修改成功界面

### 2.3.6管理员登陆界面

进入管理员登陆界面会提示输入账号密码，系统自带账号密码均为1234，之后可进行修改。账号密码错误会报错并提示重新输入。通过getch()函数的妙用使密码输入用\*号显示。调用自定义validat(zh[],mm[])；函数判断账号密码是否正确。密码储存在文件mima.in中。打开文件读取然后再关闭文件。



输入账号密码界面

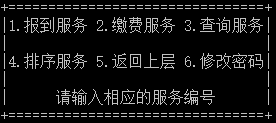


输入错误界面

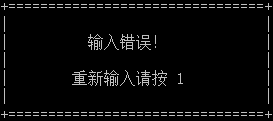
### 2.3.7导航界面

在导航界面你可以选择需要的服务输入相应的编号，输入错误会报错，输入5会返回上层的用户选择界面。本界面通过switch进行选择。

7



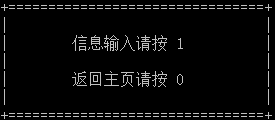
导航界面



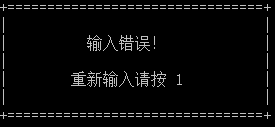
输入错误报错界面

### 2.3.8报道服务界面

进入界面后或者输入成功后可选择信息输入或返回导航界面。输入错误会提示重新输入。调用自定义entering(int count);函数，在输入界面会提示依次输入姓名、学号、数学成绩、英语成绩、语文成绩和总成绩。数据录入stu[]结构体数组中。最后录入文件old.data中。

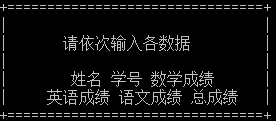


报道服务选择



输入错误提示

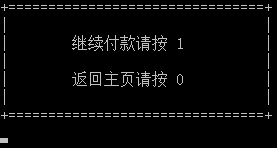
8



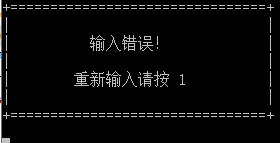
数据录入界面

### 2.3.9缴费界面

进入交费界面或者付款成功后都会进入交费选择界面，输入0会返回导航界面，输入错误同样会报错。选择继续付款会提示输入付款人，付款人输入错误会提示查不到此人，输入正确姓名和金额后会提示付款成功，还会提示当前欠费。本界面通过if…else…进行选择。调用自定义payment(count,zh);函数进行缴费。



交费选择界面

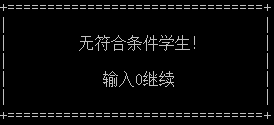


学生补交界面输入错误

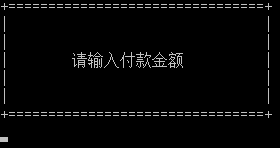


输入付款人界面

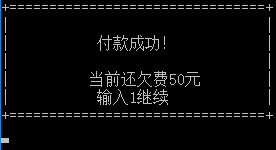
9



查不到付款人界面



付款金额输入



付款成功界面

### 2.3.10查询界面

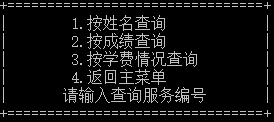
界面会提示你输入各个代号进入不同服务。输入4返回导航界面。调用自定义query(int count);函数进行查询。

按姓名查询：输入学生姓名就可以查询到学生的基本信息，输入错误会显示无此学生。

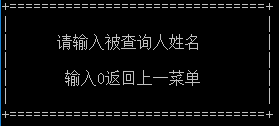
按成绩查询：输入查询科目和成绩范围会显示出相应人员。并提示共查到人数。

按学费情况查询：输入相应的欠费范围查询出相应学生的信息。

10



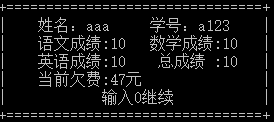
查询界面



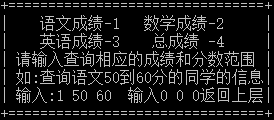
输入姓名界面



姓名错误界面

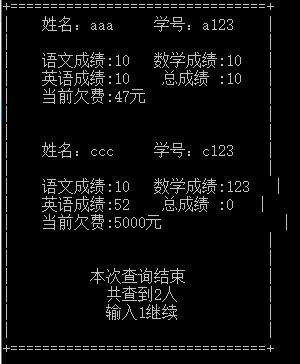


学生信息界面

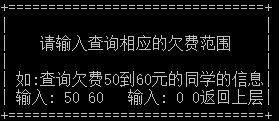


按成绩查询界面

11



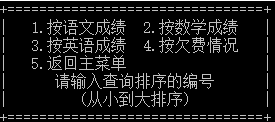
查询结果界面



欠费查询界面

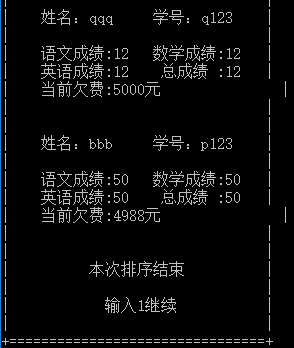
### 2.3.11排序界面

输入相应编号进入相应界面。输入5返回导航界面。查询项目默认按照从小到大排序，进入各项排序界面即可得到按相应排序的学生的各种信息。调用自定义sort(int count)函数进行排序排序中调用自定义ex(int i,int j);函数进行交换。



排序界面

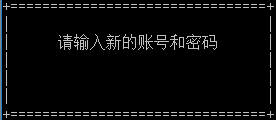
12



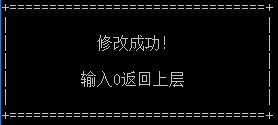
排序结果

### 2.3.12修改账号密码界面

进入界面直接进行账号密码修改，修改成功后返回到导航界面。更改密码后将密码储存在mima.in中。调用自定义gaim();函数进行改密。



输入新帐号密码界面



修改成功界面

### 2.3.13文件的读取和录入

调用自定义old\_data(int count);函数通过fscanf读取文件old.data中的数据。

调用自定义add(int count);函数通过fprintf把数据录入文件old.data中。

13

## 2.4 设计流程图

为了更加的展示本报道系统的操作流程，我们将以流程图的形式向用户更加直观的展示本报道的操作流程。

欢迎界面

用户选择

学生登陆

管理者登陆

按姓名查询

按成绩查询

按学费查询

按姓名排序

按成绩排序

按学费排序

改密

排序

报道

缴费

改密

补交

查询

# 

14

# 第三章 结论

这个学期开始接触c语言感觉有点难度，但发现学懂后c语言其实很有趣，慢慢的通过自己不断实际操作，渐渐改正逻辑与语法上的错误，也渐渐地到编程原来是一件有趣的事情，它锻炼了你的思考能力，逻辑的思考能力，改掉粗心的毛病，刚学的时候各种错误什么忘记打分号忘记打括号，忘记打取地址，但是进过一段时间的训练发现错误慢慢的少了，慢慢的做一个程序有了很强的成就感。慢慢的我的c语言就好了起来

if…else()在我的程序上运用的很多，但我同时也发现switch也能起到同样地作用。开始只想到用结构体储存数据后来发现数据用了一次后第二次要重新打进去感觉好麻烦，后来采用文件进行编写，更加简便，通过一步步的改正，思路越来越清晰。但是我发现我的数据总量是限制的在一个范围内，不能动态调用内存，这是我要改的最大的地方，因为在链表方面学的不是很好，所以在接下来的学习中，努力学习链表争取下次把程序改好后可以进行空间的动态调用。

15

# 附 录（运行环境：Dev-C++ 5.11）

源代码：

#include<conio.h>

#include<stdio.h>

#include<string.h>

#include<windows.h>

int validate(char zh[],char mm[]);//登陆系统(调用储存账号文件进行匹配)

int validate\_stu(int count,char zh[],char mm[]);

int entering(int count);//报到功能(信息录入:姓名,语数外成绩及总分,学费)

int payment(int count);//缴费功能(共5000,显示欠费和已交费)

int payment\_stu(int count,char sum\_number[]);//缴费功能(共5000,显示欠费和已交费)

// int payment2(char );//补交功能(共5000,显示欠费和已交费)

int query(int count);//查询功能(姓名、成绩和学费欠缴情况)

int sort(int count);//排序功能(成绩和学费欠缴分别排序)

int ex(int i,int j);//交换

int old\_data(int count);//从文件中读取原有信息

int add(int count);//把本次数据放到文件中保存

void gaim();//改密

int gaim\_stu(int count,char zh[]);

struct student

{

char name[20];

16

int math;

int english;

int chinese;

int all;

int money;

char number[20];//交费情况

char mm[20];

}stu[100000];

int main()

{

int a=0/\*退出系统标志\*/,x/\*代号\*/,t=0;

int count=0/\*人数\*/;

char zh[20],mm[20];

int i,abc;

int man;

printf("+================================+\n");

printf("| |\n");

printf("| 欢迎使用报道系统 |\n");

printf("| |\n");

printf("| 请输入1登陆 |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%d",&abc);

system("cls");

count=old\_data(count);

for(;;)

{

printf("+================================+\n");

printf("| |\n");

printf("| 学生登陆请按 1 |\n");

printf("| 管理人员登陆请按 2 |\n");

printf("| 退出系统请按 0 |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%d",&man);

system("cls");

if(man!=1&&man!=2&&man!=0)

{

printf("+================================+\n");

printf("| |\n");

printf("| 输入错误！ |\n");

printf("| |\n");

printf("| 重新输入请按 1 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

else if(man==2)

{

printf("+================================+\n");

printf("| |\n");

printf("| 请输入账号密码 |\n");

printf("| |\n");

printf("| 以回车键结束 |\n");

printf("| |\n");

printf("+================================+\n");

for(t=0;t==0;)

{

scanf("%s",zh);

for(i=0;;i++)

{

mm[i]=getch();

putchar('\*');

if(mm[i]==13) break;

}

mm[i]='\0';

t=validate(zh,mm);

if(t==0)

{

system("cls");

printf("+================================+\n");

printf("| |\n");

printf("| 账号密码错误 |\n");

printf("| 请重新输入账号密码 |\n");

printf("| 以回车键间隔 |\n");

printf("| |\n");

printf("+================================+\n");

}

}

system("cls");

for(a=0;a==0;)

{

printf("+================================+\n");

printf("|1.报到服务 2.缴费服务 3.查询服务|\n");

printf("| |\n");

printf("|4.排序服务 5.返回上层 6.修改密码|\n");

printf("| |\n");

printf("| 请输入相应的服务编号 |\n");

printf("+================================+\n");

scanf ("%d",&x);

system("cls");

if(x!=1&&x!=2&&x!=3&&x!=4&&x!=5&&x!=6)

{

printf("+================================+\n");

printf("| |\n");

printf("| 输入错误！ |\n");

printf("| |\n");

printf("| 重新输入请按 1 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

else switch(x)

{

case 1://报道

count=entering(count);break;

case 2://交费

count=payment(count);break;

case 3://查询

count=query(count);break;

case 4://排序

count=sort(count);break;

case 5:

a=1;break;

case 6://改密

gaim();break;

}

}

}

else if(man==1)

{

printf("+================================+\n");

printf("| |\n");

printf("| 请输入账号密码 |\n");

printf("| (初始账号密码均为学号) |\n");

printf("| 以回车键结束 |\n");

printf("| |\n");

printf("+================================+\n");

for(t=0;t==0;)

{

scanf("%s",zh);

for(i=0;;i++)

{

mm[i]=getch();

putchar('\*');

if(mm[i]==13) break;

}

mm[i]='\0';

t=validate\_stu(count,zh,mm);

if(t==0)

{

system("cls");

printf("+================================+\n");

printf("| |\n");

printf("| 账号密码错误 |\n");

printf("| 请重新输入账号密码 |\n");

printf("| 以回车键间隔 |\n");

printf("| |\n");

printf("+================================+\n");

}

}

system("cls");

printf("+================================+\n");

printf("| |\n");

printf("| 登陆成功 |\n");

printf("| |\n");

printf("| 输入1继续 |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%d",&abc);

system("cls");

printf("+================================+\n");

printf("| |\n");

printf("| 1.补交学费 2.更改密码 |\n");

printf("| |\n");

printf("| 输入相应服务 |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%d",&abc);

system("cls");

if(abc==1) count=payment\_stu(count,zh);

else count=gaim\_stu(count,zh);

}

if(man==0) {count=add(count); return 0;}

}

}

int gaim\_stu(int count,char zh[])

{

int i,abc;

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

if(strcmp(zh,stu[i].number)==0) break;

printf("+================================+\n");

printf("| |\n");

printf("| 请输入新的密码 |\n");

printf("| |\n");

printf("| |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%s",stu[i].mm);

system("cls");

printf("+================================+\n");

printf("| |\n");

printf("| 修改成功 |\n");

printf("| |\n");

printf("| 输入0继续 |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%d",&abc);

system("cls");

return count;

}

void gaim()//改密

{

char zh[20],mm[20];

int abc;

printf("+================================+\n");

printf("| |\n");

printf("| 请输入新的账号和密码 |\n");

printf("| |\n");

printf("| |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%s%s",zh,mm);

system("cls");

FILE \*fp;

fp=fopen("mima.in","w");

fprintf(fp,"%s\n%s",zh,mm);

fclose(fp);

printf("+================================+\n");

printf("| |\n");

printf("| 修改成功！ |\n");

printf("| |\n");

printf("| 输入0返回上层 |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%d",&abc);

system("cls");

}

int add(int count)

{

FILE \*fp;

fp=fopen("old.data","w");

fprintf(fp,"%d\n",count);

for(int i=0;i<count;i++)

fprintf(fp,"%s %s %d %d %d %d %d %s\n",stu[i].name,stu[i].number,stu[i].math,stu[i].english,stu[i].chinese,stu[i].all,stu[i].money,stu[i].mm);

fclose(fp);

return count;

}

int old\_data(int count)

{

FILE \*fp;

int i;

fp=fopen("old.data","r");

fscanf(fp,"%d",&count);

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

fscanf(fp,"%s%s%d%d%d%d%d%s",stu[i].name,stu[i].number,&stu[i].math,&stu[i].english,&stu[i].chinese,&stu[i].all,&stu[i].money,stu[i].mm);

fclose(fp);

return count;

}

int sort(int count)//排序功能(成绩和学费欠缴分别排序)

{

int sum\_x,i,j,abc;

for(;;)

{

printf("+================================+\n");

printf("| 1.按语文成绩 2.按数学成绩 |\n");

printf("| 3.按英语成绩 4.按欠费情况 |\n");

printf("| 5.返回主菜单 |\n");

printf("| 请输入查询排序的编号 |\n");

printf("| (从小到大排序) |\n");

printf("+================================+\n");

scanf ("%d",&sum\_x);

system("cls");

if(sum\_x!=1&&sum\_x!=2&&sum\_x!=3&&sum\_x!=4&&sum\_x!=5)

{

printf("+================================+\n");

printf("| |\n");

printf("| 输入错误！ |\n");

printf("| |\n");

printf("| 重新输入请按 1 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

else switch(sum\_x)

{

case 1://语文成绩

for(i=0;i<(count-1);i++)

for(j=i+1;j<count;j++)

if(stu[i].chinese>stu[j].chinese)

ex(i,j);

printf("+================================+\n");

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

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

}

printf("| 本次排序结束 |\n");

printf("| |\n");

printf("| 输入1继续 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

break;

case 2://数学成绩

for(i=0;i<(count-1);i++)

for(j=i+1;j<count;j++)

if(stu[i].math>stu[j].math)

ex(i,j);

printf("+================================+\n");

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

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

}

printf("| 本次排序结束 |\n");

printf("| |\n");

printf("| 输入1继续 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

break;

case 3://英语成绩

for(i=0;i<(count-1);i++)

for(j=i+1;j<count;j++)

if(stu[i].english>stu[j].english)

ex(i,j);

printf("+================================+\n");

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

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

}

printf("| 本次排序结束 |\n");

printf("| |\n");

printf("| 输入1继续 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

break;

case 4://欠费情况

for(i=0;i<(count-1);i++)

for(j=i+1;j<count;j++)

if(stu[i].money<stu[j].money)

ex(i,j);

printf("+================================+\n");

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

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

}

printf("| 本次排序结束 |\n");

printf("| |\n");

printf("| 输入1继续 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

break;

case 5://主菜单

return count;

}

}

}

int ex(int i,int j)//交换

{

int t;

char sum;

for(t=0;t<strlen(stu[j].name);t++)

{

sum=stu[i].name[t];

stu[i].name[t]=stu[j].name[t];

stu[j].name[t]=sum;

}

t=stu[i].math;

stu[i].math=stu[j].math;

stu[j].math=t;

t=stu[i].chinese;

stu[i].chinese=stu[j].chinese;

stu[j].chinese=t;

t=stu[i].english;

stu[i].english=stu[j].english;

stu[j].english=t;

t=stu[i].all;

stu[i].all=stu[j].all;

stu[j].all=t;

t=stu[i].money;

stu[i].money=stu[j].money;

stu[j].money=t;

}

int query(int count)//查询功能(姓名、成绩和学费欠缴情况)

{

int sum\_x,i,abc,sum\_y,sum\_man;

char sum\_name[20];

int sum\_a,sum\_b,sum\_c;

for(;;)

{

printf("+================================+\n");

printf("| 1.按姓名查询 |\n");

printf("| 2.按成绩查询 |\n");

printf("| 3.按学费情况查询 |\n");

printf("| 4.返回主菜单 |\n");

printf("| 请输入查询服务编号 |\n");

printf("+================================+\n");

scanf ("%d",&sum\_x);

system("cls");

if(sum\_x!=3&&sum\_x!=1&&sum\_x!=2&&sum\_x!=4)

{

printf("+================================+\n");

printf("| |\n");

printf("| 输入错误！ |\n");

printf("| |\n");

printf("| 重新输入请按 1 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

else switch(sum\_x)

{

case 1://按姓名查询

for(sum\_y=0;;)

{

sum\_y=0;

printf("+================================+\n");

printf("| |\n");

printf("| 请输入被查询人姓名 |\n");

printf("| |\n");

printf("| 输入0返回上一菜单 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%s",sum\_name);

system("cls");

if(sum\_name[0]=='0') break;

else

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

if(strcmp(stu[i].name,sum\_name)==0)

{sum\_y=1;break;}

if(sum\_y==1)

{

printf("+================================+\n");

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| 输入0继续 |\n");

printf("+================================+\n");

}

else

{

printf("+================================+\n");

printf("| |\n");

printf("| 无符合条件学生！ |\n");

printf("| |\n");

printf("| 输入0继续 |\n");

printf("| |\n");

printf("+================================+\n");

}

scanf ("%d",&abc);

system("cls");

}

break;

case 2://按成绩查询

for(;;)

{

sum\_man=0;

printf("+================================+\n");

printf("| 语文成绩-1 数学成绩-2 |\n");

printf("| 英语成绩-3 总成绩 -4 |\n");

printf("| 请输入查询相应的成绩和分数范围 |\n");

printf("| 如:查询语文50到60分的同学的信息|\n");

printf("| 输入:1 50 60 输入0 0 0返回上层|\n");

printf("+================================+\n");

scanf ("%d%d%d",&sum\_a,&sum\_b,&sum\_c);

system("cls");

if(sum\_a==0) break;

else

{

printf("+================================+\n");

if(sum\_a==1)//语文

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

if(stu[i].chinese>=sum\_b&&stu[i].chinese<=sum\_c)

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

sum\_man++;

}

if(sum\_a==2)

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

if(stu[i].math>=sum\_b&&stu[i].math<=sum\_c)

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

sum\_man++;

}

if(sum\_a==3)

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

if(stu[i].english>=sum\_b&&stu[i].english<=sum\_c)

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

sum\_man++;

}

if(sum\_a==4)

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

if(stu[i].all>=sum\_b&&stu[i].all<=sum\_c)

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

sum\_man++;

}

printf("| 本次查询结束 |\n");

printf("| 共查到%d人 |\n",sum\_man);

printf("| 输入1继续 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

}

break;

case 3://按学费情况查询

for(;;)

{

sum\_man=0;

printf("+================================+\n");

printf("| |\n");

printf("| 请输入查询相应的欠费范围 |\n");

printf("| |\n");

printf("| 如:查询欠费50到60元的同学的信息|\n");

printf("| 输入: 50 60 输入: 0 0返回上层|\n");

printf("+================================+\n");

scanf ("%d%d",&sum\_b,&sum\_c);

system("cls");

if(sum\_b==0&&sum\_c==0) break;

else

{

printf("+================================+\n");

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

if((5000-stu[i].money)>=sum\_b&&(5000-stu[i].money)<=sum\_c)

{

printf("| 姓名：%s 学号：%s |\n",stu[i].name,stu[i].number);

printf("| |\n");

printf("| 语文成绩:%d 数学成绩:%d |\n",stu[i].chinese,stu[i].math);

printf("| 英语成绩:%d 总成绩 :%d |\n",stu[i].english,stu[i].all);

printf("| 当前欠费:%d元 |\n",5000-stu[i].money);

printf("| |\n");

printf("| |\n");

sum\_man++;

}

printf("| 本次查询结束 |\n");

printf("| 共查到%d人 |\n",sum\_man);

printf("| 输入1继续 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

}

break;

case 4://返回主菜单

return count;

}

}

}

int payment(int count)//缴费功能(共5000,显示欠费和已交费)

{

int sum\_1,sum\_money,i,abc,sum\_man;

char sum\_name[20];

for(;;)

{

printf("+================================+\n");

printf("| |\n");

printf("| 继续付款请按 1 |\n");

printf("| |\n");

printf("| 返回主页请按 0 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&sum\_1);

system("cls");

if(sum\_1!=0&&sum\_1!=1)

{

printf("+================================+\n");

printf("| |\n");

printf("| 输入错误！ |\n");

printf("| |\n");

printf("| 重新输入请按 1 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

else if(sum\_1==0) return count;

else if(sum\_1==1)

{

sum\_man=0;

printf("+================================+\n");

printf("| |\n");

printf("| |\n");

printf("| 请输入付款人 |\n");

printf("| |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%s",sum\_name);

system("cls");

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

if(strcmp(sum\_name,stu[i].name)==0)

{

printf("+================================+\n");

printf("| |\n");

printf("| |\n");

printf("| 请输入付款金额 |\n");

printf("| |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%d",&sum\_money);//////////////////如果没有欠费怎么办！！

system("cls");

stu[i].money=stu[i].money+sum\_money;

sum\_man=1;

break;

}

if(sum\_man==1)

{

printf("+================================+\n");

printf("| |\n");

printf("| 付款成功！ |\n");

printf("| |\n");

printf("| 当前还欠费%d元 |\n",5000-stu[i].money);

printf("| 输入1继续 |\n");

printf("+================================+\n");

}

else

{

printf("+================================+\n");

printf("| |\n");

printf("| 无符合条件学生！ |\n");

printf("| |\n");

printf("| 输入0继续 |\n");

printf("| |\n");

printf("+================================+\n");

}

scanf("%d",&abc);///////////////////////////////如果没有欠费怎么办！！

system("cls");

}

}

}

int payment\_stu(int count,char sum\_number[])//学生缴费功能(共5000,显示欠费和已交费)

{

int sum\_1,sum\_money,i,abc;

for(;;)

{

printf("+================================+\n");

printf("| |\n");

printf("| 继续付款请按 1 |\n");

printf("| |\n");

printf("| 返回主页请按 0 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&sum\_1);

system("cls");

if(sum\_1!=0&&sum\_1!=1)

{

printf("+================================+\n");

printf("| |\n");

printf("| 输入错误！ |\n");

printf("| |\n");

printf("| 重新输入请按 1 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

else if(sum\_1==0) return count;

else if(sum\_1==1)

{

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

if(strcmp(sum\_number,stu[i].number)==0)

{

printf("+================================+\n");

printf("| |\n");

printf("| |\n");

printf("| 请输入付款金额 |\n");

printf("| |\n");

printf("| |\n");

printf("+================================+\n");

scanf("%d",&sum\_money);//////////////////如果没有欠费怎么办！！

system("cls");

stu[i].money=stu[i].money+sum\_money;

break;

}

printf("+================================+\n");

printf("| |\n");

printf("| 付款成功！ |\n");

printf("| |\n");

printf("| 当前还欠费%d元 |\n",5000-stu[i].money);

printf("| 输入1继续 |\n");

printf("+================================+\n");

scanf("%d",&abc);///////////////////////////////如果没有欠费怎么办！！

system("cls");

}

}

}

int entering(int count)//报到功能(信息录入:姓名,语数外成绩及总分,学费)

{

int sum\_1=1,abc;

for(;;)

{

printf("+================================+\n");

printf("| |\n");

printf("| 信息输入请按 1 |\n");

printf("| |\n");

printf("| 返回主页请按 0 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&sum\_1);

system("cls");

if(sum\_1!=0&&sum\_1!=1)

{

printf("+================================+\n");

printf("| |\n");

printf("| 输入错误！ |\n");

printf("| |\n");

printf("| 重新输入请按 1 |\n");

printf("| |\n");

printf("+================================+\n");

scanf ("%d",&abc);

system("cls");

}

else if(sum\_1==0) return count;

else if(sum\_1==1)

{

printf("+================================+\n");

printf("| |\n");

printf("| 请依次输入各数据 |\n");

printf("| |\n");

printf("| 姓名 学号 数学成绩 |\n");

printf("| 英语成绩 语文成绩 总成绩 |\n");

printf("+================================+\n");

scanf("%s%s%d%d%d%d",stu[count].name,stu[count].number,&stu[count].math,&stu[count].english,&stu[count].chinese,&stu[count].all);

stu[count].money=0;

for(abc=0;abc<20;abc++)

stu[count].mm[abc]=stu[count].number[abc];

count++;

}

system("cls");

}

}

int validate(char zh[],char mm[])//登陆系统(调用储存账号文件进行匹配)

{

FILE \*fp;

fp=fopen("mima.in","r");

int t=0;

char zh0[20]/\*={'1','2','3'}\*/,mm0[20]/\*={'1','2','3'}\*/;/////////////////文件读取密码账号

fscanf(fp,"%s%s",zh0,mm0);

if(strcmp(zh,zh0)==0&&strcmp(mm,mm0)==0)

t=1;

fclose(fp);

return t;

}

int validate\_stu(int count,char zh[],char mm[])//学生登陆系统(调用储存账号文件进行匹配)

{

int i,t=0;

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

{

if(strcmp(zh,stu[i].number)==0&&strcmp(mm,stu[i].mm)==0)

t=1;

}

return t;

}