程序设计综合实验报告

题目**： 学生选课系统**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 学号 | 姓名 | 班级 | 分工 | 贡献率 |
| 20201202054 | 徐嘉福 | 2020计科-2班 | 主函数设计及函数适配 | 25% |
| 20201202080 | 吴金硕 | 2020计科-2班 | 搜索模块 | 25% |
| 20201202078 | 冷浩甫 | 2020计科-2班 | 课表模块 | 25% |
| 20201202024 | 赵苗苗 | 2020计科-2班 | 课表模块 | 25% |

网络空间安全与计算机学院

2020年12月30日

#### 一、实验题目

学生选课系统

#### 二、实验内容

设计一个选课系统供学生选课，管理员修改相关信息

学生选课系统的基本功能包括：

1、密码登录

2、信息保存

3、选课退课

#### 三、实验目的和要求

**1．目的**

训练文件操作能力，训练页面设计，熟练使用多分支选择，学会进行流程控制。

1. **要求：**

**学生能够正常选课，退课。**

四、程序的框架结构

1．主要的数据类型定义：

//课程信息结构体

typedef struct course

{

int num; //课程编号

int score; //课程学分

int totalPeriod; //总学时

int classPeriod; //每星期学时

int week; //开课周

} COURSE;

//学生选课信息结构体

typedef struct student

{

int stuNo; //学号偶数为男，奇数为女。

int num[6]; //已选课程编号0表示没有选，非0表示选择

int score; //还需要选的学分

} STUDENT;

//课程名称

char name[6][26]={ "operations research","database","java","visual computing","computing system","algorithm design analysis"};

/\*

COURSE cour[6]= { {20201201,4,3,243647,1},

{20201202,2,2,3441,2},

{20201203,1,2,3753,4},

{20201204,3,3,122552,2},

{20201205,2,3,273167,3},

{20201206,3,2,6172,6}

};

STUDENT stu [4]=

{

{2020201,{0,1,0,0,0,0},3},{2020202,{0,1,0,0,0,0},3},{2020203,{0,0,0,0,0,0},5},{2020104,{0,0,0,0,0,0},5}

};//原始数据\*/

1. **本程序包括以下8个功能模块：**
2. 选课模块

选课是登入学生端后可以进行的操作，想法是通过学生带的数组进行模拟选课。

如果是1，就是选了课；否则就是没有，另外就是基本查找操作。（附:考虑了网络问题

，学生多人选课选不上的情况，所以做了一个随机函数进行模拟）

1. 输入模块

采取了强制的想法，将用户不符合要求的输入进行屏蔽，读取但不显示，利用了getch函数

只读不显示的特性，对于不符合的数据直接不放入我的字符串数组里边，同时还利用了标记变量去限制长度，防止缓冲区的溢出问题。这个模块我一共做了三个函数，分别去检测数字输入，字符串输入，真假判断。数字输入我将数字以字符型的模式读入，最后再将其转化为

整形。

1. 主程序模块

加法模块

输出模块

输入模块

通过goto语句实现流程调控，system进行界面优化。然后主体通过switch语句进行调控，防止一些不可预见情况。

1. 浏览信息模块

主要就是排版的问题，其他问题不大。

1. 查找模块

其实就通过顺序查找，然后按各种方式进行了排序而已。

1. 日志模块

日志模块在学生端没有什么感觉，在管理员端可以查看日志，日志中有学生的选课记录，包括具体到秒的时间，学生是谁，对什么课程进行了什么操作。（附：我对日志的书写和查看是用的绝对路径，在自己电脑上运行可能会出问题）

1. 密码模块

密码板块也没什么，就通过比较是否相同，如此而已。

1. 课表模块

这个算是我这么多板块唯一需要脑子的了，当然得好好说下。首先先说明以下来源吧！

最早是李焱向我提出这个要求的，当时是让组内其他成员去实现。最后他们给我了一个想法，即用坐标去模拟课程位置，并给了我一个简单的函数去配套理解。但是那个函数只能够去实现一节课在课表中的位置，而且还不能够实现我的多个学生，配套多个已经选择没有选择的课，最大的问题还是我并没有存放课程位置的数据。对上述问题，我做了如下改动：

1. 将我的课程信息的总学时变成了课程位置的数据，当时是两两配套，分别是那课第一节在哪里，先是X坐标，然后是Y坐标，就这样我将所有的信息用一个数据表达了（其实这样做还是因为懒，但没想到让问题这么复杂）。
2. 这样很自然的带来了第二个问题。如何去做XY坐标的分离。我的想法是用定义了一个结构体，结构体里面用两个数组去分别放XY坐标，两个数组的下标分别是第几个。然后就带来了个新问题，如何去分离那些数呢？我想到的是将数据拆位，用一个循环做出来，奇数次放x数组，偶数次放y数组（这个赋值操作不好写）。
3. 于是接下来的问题就是如何去打印这个表。我采取的是将每一块看成一个小方块，对没有使用到的部分赋值为空（也就是指定长度的空格）， 有的便赋值为课程名。但效果不如人意。

整体特别挤，而且课程名之间重叠。我开始是一点一点去调整位置。但是会出现课程名过长影响布局，课程和星期难以配套的麻烦。最后采取的方案是让它们右对齐，并在每两行间用了多个换行最终达到了现在的效果。

#### 五、源代码

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <conio.h>

#include <time.h>

#include <math.h>

#include <windows.h>

#include <ctype.h>

void check()//欢乐颂

{

int ff=62;//音符个数别忘了写

unsigned a[]=

{

659,659,698,784,

784,698,659,587,

523,523,587,659,

659,587,587,

659,659,698,784,

784,698,659,587,

523,523,587,659,

587,523,523,

587,659,523,587,659,698,659,523,

587,659,698,659,587,

523,587,392,659,

659,659,698,784,

784,698,659,587,

523,523,587,659,

587,523,523,

//音符

};

unsigned b[]=

{

400,400,400,400,

400,400,400,400,

400,400,400,400,

600,200,800,

400,400,400,400,

400,400,400,400,

400,400,400,400,

600,200,800,

600,400,400,

400,200,200,400,400,

400,200,200,400,400,

400,400,400,400,

400,400,400,400,

400,400,400,400,

400,400,400,400,

600,400,800,

//音长

};

for(int i=0; i<ff/2; i++)

{

// printf("第%d个音符\n",i+1);//显示出来好找错

Beep(a[i],b[i]);

}

}

void PengYou()//朋友

{

int ff=131;

unsigned a[]=

{

0,587,659,523,587,659,587,659,784,

880,784,659,659,587,659,523,587,659,392,587,659,

587,440,523,

587,659,523,587,659,523,587,659,

784,784,880,523,440,523,

587,587,523,440,440,523,

587,659,587,523,587,587,659,587,523,

523,587,587,659,523,587,659,

784,784,784,880,523,440,523,

587,587,523,440,440,392,440,

523,0,

659,659,784,

784,784,784,880,784,880,988,

1046,880,880,784,659,659,587,523,

523,880,784,659,587,523,

523,440,587,659,659,784,

784,784,784,880,784,880,988,

1046,880,880,784,659,659,587,523,

523,880,784,659,587,523,

440,440,523,523,

};

unsigned b[]=

{

900,450,450,900,450,450,900,450,450,

900,450,450,900,450,450,900,450,450,900,450,450,

900,450,4950,

450,450,900,450,450,900,450,450,900,450,450,900,450,450,

900,450,450,900,450,450,

450,450,450,225,225,900,450,225,225,

900,450,225,225,900,450,450,

900,450,225,225,900,450,450,

900,450,225,225,900,450,450,

1800,900,

450,225,225,

450,450,450,225,1125,450,450,

450,450,450,225,1125,450,225,1125,

450,225,1125,450,225,1125,

225,225,1350,450,225,225,

450,450,450,225,1125,450,450,

450,450,450,225,1125,450,225,1125,

450,225,1125,450,225,1125,

450,225,675,1350,

};

for(int i=0; i<ff/2; i++)

{

//printf("第%d个音符\n",i);

Beep(a[i],b[i]);

}

}

int work (int x,int i)

{

int y;

y = x/(int)(pow(10,i-1)+0.5)%10;

return y;

}

typedef struct xy

{

int x[3];

int y[3];

} XY;

//课程信息结构体

typedef struct course

{

int num; //课程编号

int score; //课程学分

int totalPeriod; //每星期学时

int classPeriod; //每星期学时其实是在那一届课，不怎么想改代码了

int week; //开课周

int people; //还可以选择的人数

} COURSE;

//学生选课信息结构体

typedef struct student

{

int stuNo; //学号偶数为男，奇数为女。

int num[6]; //已选课程编号0表示没有选，非0表示选择

int score; //还需要选的学分

} STUDENT;

//课程名称，作为随程序生存的数据，由于许多地方要用，所有将其设置为全局变量

char name[6][26]= { "operations research","database","java","visual computing","computing system","algorithm design analysis"};

char name2[6][26]= { "运筹学","数据库","Java","可视化","计算系统","算法设计"};

void readfile (COURSE cour[],STUDENT stu[])//读取文件信息储存在结构体数组中

{

FILE \*fp;

int n,m;

int i,j;

if((fp=fopen("data.txt","r"))==NULL)

{

printf("打开失败");

exit(0);

}

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

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

for(i=0; i<6; i++)

{

fscanf(fp," %d",&cour[i].num);

fscanf(fp," %d",&cour[i].score);

fscanf(fp," %d",&cour[i].classPeriod);

fscanf(fp," %d",&cour[i].totalPeriod);

fscanf(fp," %d",&cour[i].week);

fscanf(fp," %d",&cour[i].people);

}

for(i=0; i<4; i++)

{

fscanf(fp," %d",&stu[i].stuNo);

for(j=0; j<6; j++)

{

fscanf(fp," %d",&stu[i].num[j]);

}

fscanf(fp," %d",&stu[i].score);

}

fclose(fp);

}

void writefile (COURSE cour[],STUDENT stu[])//将结构体信息写入文件

{

int i,j;

int n=4,m=6;

FILE \*fp;

if((fp=fopen("data.txt","w"))==NULL)

{

printf("打开失败");

exit(0);

}

fprintf(fp,"%d %d",n,m);

for(i=0; i<6; i++)

{

fprintf(fp," %d %d %d %d %d %d",cour[i].num,cour[i].score,cour[i].classPeriod,cour[i].totalPeriod,cour[i].week,cour[i].people);

}

for(i=0; i<4; i++)

{

fprintf(fp," %d",stu[i].stuNo);

for(j=0; j<6; j++)

{

fprintf(fp," %d",stu[i].num[j]);

}

fprintf(fp," %d",stu[i].score);

}

fclose(fp);

}

void writepassword (char password[][26],int n)//将密码写入文件

{

int i;

FILE \*fp;

if((fp=fopen("password.txt","w"))==NULL)

{

printf("打开失败");

exit(0);

}

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

fprintf(fp,"%s ",password[i]);

fclose(fp);

}

void readpassword (char password[][26],int n)//从文件中读取密码

{

int i;

FILE \*fp;

if((fp=fopen("password.txt","r"))==NULL)

{

printf("打开失败");

exit(0);

}

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

{

fscanf(fp,"%s",password[i]);

}

fclose(fp);

}

void writejn(void)//创建一个关于日志的文本文件

{

FILE \*fp;

if((fp=fopen("1.txt","w"))==NULL)

{

printf("打开失败");

exit(0);

}

fprintf(fp,"年月日时分 课程 学生 操作\n");

fclose(fp);

}

void addjournal(char \*t,int i,int num,int n)//向日志中加入新信息,参数分别是时间，课程编号,学生编号，学生的操作

{

FILE \*fp;

if((fp=fopen("1.txt","a"))==NULL)

{

printf("打开失败");

exit(0);

}

fprintf(fp,"%s",t);

fprintf(fp," %s ",name[i]);

fprintf(fp,"%d",num);

if(n) fprintf(fp," 选课\n");

else fprintf(fp," 退课\n");

fclose(fp);

}

int correct(void)//用来实现真假判断

{

int i=1,flag=0;

char num;

while(i++<6) //用来输入

{

num=getch();//屏蔽非数字用户输入和模拟用户输入内容，只有单个

if(num==8)//满足输入条件，显示给用户,8是为了让用户能删除

{

putchar(8);//显示给用户

putchar(32);

putchar(8);

}

else if (num=='1')

{

flag=1;

putchar(num);

}

else if (num=='0')

{

flag=0;

putchar(num);

}

else

{

if(num==13)//回车作为结束标准

{

goto END;

}

else//吃掉非法数据

{

putchar(32);//空格

putchar(8);//退格

putchar(32);//空格

putchar(8);//再次退格

}

}

}

END:

return flag;

}

char\* hide (int lazy)//用来实现信息隐蔽

{

char num[26];

int i=0;

char \*pstr=num;

int n;

START:

for(i=0; i<25; i++) //用来输入

{

num[i]=getch();//屏蔽非数字用户输入和模拟用户输入内容，只有单个

if((num[i]>='0'&&num[i]<='9')||num[i]==8||(num[i]>='a'&&num[i]<='z')||(num[i]>='A'&&num[i]<='Z'))//满足输入条件，显示给用户,8是为了让用户能删除

{

if(num[i]==8)

{

i-=2;

putchar(8);

putchar(32);

putchar(8);

}

else

{

putchar('\*');//显示给用户

}

}

else

{

if(num[i]==13)//回车作为结束标准

{

goto END;

}

else//吃掉非法数据

{

putchar(32);//空格

putchar(8);//退格

putchar(32);//空格

putchar(8);//再次退格

i--;

}

}

}

END:

num[i]='\0';

if(!lazy)

{

printf("\n请检查有无错误,如果有请按1，没有按0\n");

Sleep(1000);

if(correct())

goto START;

}

return pstr;

}

int correctint(int lazy)//用来与用户交互，通过数字

{

char num[9];

int i=0;

char \*pstr=num;

int n,flag;

START:

for(i=0; i<8; i++) //用来输入

{

num[i]=getch();//屏蔽非数字用户输入和模拟用户输入内容，只有单个

if(num[i]>='0'&&num[i]<='9'||num[i]==8||num[i]==' ')//满足输入条件，显示给用户,8是为了让用户能删除

{

putchar(num[i]);//显示给用户

if(num[i]==8)

{

i-=2;

putchar(32);

putchar(8);

}

}

else

{

if(num[i]==13)//回车作为结束标准

{

goto END;

}

else//吃掉非法数据

{

putchar(32);//空格

putchar(8);//退格

putchar(32);//空格

putchar(8);//再次退格

i--;

}

}

}

num[8]='\0';

END:

n=atoi(pstr);

if(!lazy)

{

printf("\n请检查有无错误,如果有请按1，没有按0\n");

Sleep(1000);

flag=correct();

if(flag)

goto START;

}

return n;

}

char\* correctchar (int lazy)//用来与用户交互，用过数字和字符

{

char num[26];

int i=0;

char \*pstr=num;

int n;

START:

for(i=0; i<25; i++) //用来输入

{

num[i]=getch();//屏蔽非数字用户输入和模拟用户输入内容，只有单个

if((num[i]>='0'&&num[i]<='9')||num[i]==8||(num[i]>='a'&&num[i]<='z')||(num[i]>='A'&&num[i]<='Z')||num[i]==' ')//满足输入条件，显示给用户,8是为了让用户能删除

{

putchar(num[i]);//显示给用户

if(num[i]==8)

{

i-=2;

putchar(32);

putchar(8);

}

}

else

{

if(num[i]==13)//回车作为结束标准

{

goto END;

}

else//吃掉非法数据

{

putchar(32);//空格

putchar(8);//退格

putchar(32);//空格

putchar(8);//再次退格

i--;

}

}

}

END:

num[i]='\0';

if(!lazy)

{

printf("\n请检查有无错误,如果有请按1，没有按0\n");

Sleep(1000);

if(correct())

goto START;

}

return pstr;

}

int format (const COURSE cour[],const STUDENT stu[],XY pos[])//将所有人和所有课程的信息打印在屏幕上

{

system("title 学生和课程信息");

int flag,k;

printf("\n\t按0回到上一级\n");

printf("\n\t继续操作按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

int i,j;

printf(" 课程名称\t");//15个空格

printf("课程编码\t");

printf("课程的学分\t");

printf(" 时间\t");//3个空格

printf(" 每周学时\t");//8个空格

printf(" 开课周\t");//4个空格

printf(" 剩余人数\t");//1个空格

printf("\n");

for(i=0; i<6; i++)

{

printf("%-25s\t",name[i]);

printf("%-8d\t",cour[i].num);

printf("%-8d\t",cour[i].score);

printf("%-8d\t",cour[i].classPeriod);

printf("%-8d\t",cour[i].totalPeriod);

printf("%-10d\t", cour[i].week);

printf("%-8d\t",cour[i].people);

printf("\n");

}

printf("\n\n");

for(i=0; i<4; i++)

{

printf("学生学号\t");

printf("%-8d\t", stu[i].stuNo);

printf("已选课程\t");

for(j=0; j<6; j++)

if(stu[i].num[j])

printf("%-25s",name[j]);

printf("还需要选的学分\t");

printf("%-3d\t",stu[i].score);

printf("\n");

}

printf("\n\t时间两两一组，第一个为星期，第二个为多少节课\n");

system("pause");

printf("\n\t如果你无法理解请按1继续操作按0\n");

flag=correct();

printf("\n\n");

if(flag)

{

for(k=0; k<6; k++)

{

printf("\t%-25s ",name[k]);

for(j=0; j<cour[k].totalPeriod; j++)

{

printf("\t星期%-5d", pos[k].y[j]);

printf("第%-2d节 \t",pos[k].x[j]);

}

printf("\n");

}

}

printf("\n\t按0回到上一级\n");

printf("\n\t退出系统按1\n");

flag=correct();

if(flag)

return 0;

else

return 1;

}

int journal (int lazy,COURSE cour[],STUDENT stu[],int i,int n,int d)//参数分别是是否省略，课程，学生数组，课程位置,学生名，操作

{

system("title 操作保存");

system("cls");

time\_t rawtime;

struct tm \* timeinfo;

time(&rawtime);

timeinfo=localtime(&rawtime);

char \*s=asctime(timeinfo);

addjournal(s,i,stu[n].stuNo,d);

return 0;

}

int change(char password[][26],int n,int lazy)//更改密码

{

system("title 修改密码");

char str[26];

char newstr[26];

char last[26];

int i,k,flag;

printf("------------------------------------更改密码(密码不可带空格)\n");

printf("\n\t返回上一级按0\n");

printf("\n\t继续操作按1\n");

flag=correct();

if(!flag)

return 0;

START:

system("cls");

printf("\n\t如果你想让你的密码不可见请按1否则按0\n");

k=correct();

if(!k)

{

printf("\n\t请输入原密码\n");

strcpy(last,correctchar(lazy));

printf("\n\t请输入你的新密码\n");

strcpy(str,correctchar(lazy));

printf("\n\t请再次输入你的新密码\n");

strcpy(newstr,correctchar(lazy));

}

else

{

printf("\n\t请输入原密码\n");

strcpy(last,hide(lazy));

printf("\n\t请输入你的新密码\n");

strcpy(str,hide(lazy));

printf("\n\t请再次输入你的新密码\n");

strcpy(newstr,hide(lazy));

}

if(strcmp(last,password[n])==0)

{

if(strcmp(str,newstr)==0)

{

strcpy(password[n],newstr);

system("pause");

}

else

{

printf("\n\t两次密码不相同,如果您想再次输入，请按1，退出请按0\n");

flag=correct();

if(flag)

goto START;

}

}

else

{

printf("\n\t原密码输入错误，如果您想再次输入，请按1，退出请按0\n");

flag=correct();

if(flag)

goto START;

}

system("cls");

printf("\n\t改变成功！\n");

return 1;

}

int passwords (char password[][26],int m,int lazy)//输入密码检查

{

int number;

int count=0;

int n[5]= {2020,2020201,2020202,2020203,2020104};

char ps[26];

int i,k;

printf("\t请在下方输入您的账号与密码\n");

NUMBER:

printf("\n账号:\n");

number=correctint(lazy);

printf("\n如果你想让你的密码不可见请按1否则按0(默认不可见)\n");

k=correct();

printf("\n密码:\n");

if(k)

{

strcpy(ps,hide(lazy));

}

else

{

strcpy(ps,correctchar(lazy));

}

for(i=0; i<5; i++)

{

if(number==n[i])

{

if(strcmp(password[i],ps)==0)

return i;

}

}

printf("\n账号或密码输入错误,按0退出系统，按1重新输入\n");

int a;

a=correct();

if(a)

{

if(count++<5)

goto NUMBER;

else

{

printf("\n您输入错误次数过多，请1分钟稍后再试\n");

Sleep(60\*1000);

}

}

else

return -1;

}

int choose (int lazy,COURSE cour[],STUDENT stu[],int n)//选课或者退课

{

system("title 选课或退课");

int flag;

int number;//课程编号

char \*s;//课程名称

int i,k,j;

printf("\t按0回到上一级\n");

printf("\t继续操作按1\n");

flag=correct();

if(!flag)

return 1;

START:

system("cls");

printf("\n\t请输入你要选/退的课的\n");

printf("\n\t名称（按0）\n");

printf("\n\t编号（按1）\n");

flag=correct();

printf("\n\t如果你忘记了名称或编号按1查看，没有忘记按0继续操作\t\n");

if(correct())

{

printf(" 课程名称\t");//15个空格

printf("课程编码\t");

printf("课程的学分\t");

printf(" 时间\t");//3个空格

printf(" 每周学时\t");//8个空格

printf(" 开课周\t");//4个空格

printf(" 剩余人数\t");//1个空格

printf("\n");

for(i=0; i<6; i++)

{

printf("%-25s\t",name[i]);

printf("%-8d\t",cour[i].num);

printf("%-8d\t",cour[i].score);

printf("%-8d\t",cour[i].classPeriod);

printf("%-8d\t",cour[i].totalPeriod);

printf("%-10d\t", cour[i].week);

printf("%-8d\t",cour[i].people);

printf("\n");

}

printf("\n\n");

}

if(flag)

{

printf("\n\t请输入课程编号\n");

number=correctint(lazy);

printf("\n\n");

for(i=0; i<6; i++)

{

if(number==cour[i].num)

{

FIRST:

system("pause");//用来暂停，防止太快看不清

system("cls");//防止信息过多

printf("\n\n");//打印那门课的信息

printf(" 课程名称\t");//15个空格

printf("课程编码\t");

printf("课程的学分\t");

printf(" 时间\t");//3个空格

printf(" 每周学时\t");//8个空格

printf(" 开课周\t");//4个空格

printf(" 剩余人数\n");//1个空格

printf("%25s\t",name[i]);

printf("%8d\t",cour[i].num);

printf("%8d\t",cour[i].score);

printf("%8d\t",cour[i].classPeriod);

printf("%8d\t",cour[i].totalPeriod);

printf("%8d\t", cour[i].week);

printf("%3d\t",cour[i].people);

printf("\n");

//以下为选课与退课操作

printf("\t按1选课\n");

printf("\t按2退课\n");

printf("\t按0退出\n");

flag=correctint(lazy);

if(flag==1)

{

if(stu[n].score<-5)//防止一个人选太多课

{

printf("\n\t对不起，你选的太多了\t\n");

goto END;

}

if(!stu[n].num[i]&&cour[i].people>0)//学生没选这节课，这节课还能选

{

srand((int)time(NULL));

if(rand()%2+0)//选课成功跳转到日志界面保存

{

cour[i].people--;//课程人数减1

stu[n].score=stu[n].score-cour[i].score;//学生的学分减去课程的学分

stu[n].num[i]++;//表示学生选的什么课

journal(lazy,cour,stu,i,n,1);//保存到课程日志

printf("\n\t你是一个幸运儿，选上了这节课！\n\t");

check();

system("pause");

goto END;

}

else//滑稽

{

printf("\n\t由于一些不可知的原因，你没有选上课，让我心疼你3秒钟\n");

PengYou();

static const unsigned int table[] =

{

0x00000000, 0x00000000, 0x000FF000, 0x00300C00,

0x00C00300, 0x01000080, 0x02000040, 0x04000020,

0x08700E10, 0x098C3190, 0x127E4FC8, 0x118C3188,

0x20000004, 0x20000004, 0x20000004, 0x20000004,

0x24000024, 0x24000024, 0x24000024, 0x22000044,

0x12000048, 0x11000088, 0x09000090, 0x08800110,

0x04600620, 0x02181840, 0x0107E080, 0x00C00300,

0x00300C00, 0x000FF000, 0x00000000, 0x00000000,

};

for(int y = 0; y < 32; ++y)

{

for(int x = 0; x < 64; ++x)

{

putchar((table[y] & (1 << (x >> 1))) ? 'X' : ' ');

}

putchar('\n');

system("pause");

system("cls");

goto START;

}

}

}

else//输入错误判断

{

printf("\n\t不能重复选同一节课！或者这节课已经不能选了\n");

printf("\n\t将在3秒后回到选课界面\n");

Sleep(3\*1000);

system("cls");

goto START;

}

}

else if (flag==2)//退课

{

printf("%d\n",stu[n].num[i]);

if(stu[n].num[i]<=0)//防止没选就退课

{

printf("\n你还没选呢!\n");

printf("\n将在3秒后回到选课界面\n");

Sleep(3\*1000);

system("cls");

goto START;

}

else//选了可以退

{

srand((int)time(NULL));

if(rand()%2+0)//退课成功

{

cour[i].people++;

stu[n].score=stu[n].score+cour[i].score;

stu[n].num[i]--;

journal(lazy,cour,stu,i,n,0);

printf("\n退课成功\n");

check();

goto END;

}

else//模拟网络问题

{

printf("\n\t由于一些不可知的原因，你退不了课了，让我心疼你3秒钟\n");

PengYou();

static const unsigned int table[] =

{

0x00000000, 0x00000000, 0x000FF000, 0x00300C00,

0x00C00300, 0x01000080, 0x02000040, 0x04000020,

0x08700E10, 0x098C3190, 0x127E4FC8, 0x118C3188,

0x20000004, 0x20000004, 0x20000004, 0x20000004,

0x24000024, 0x24000024, 0x24000024, 0x22000044,

0x12000048, 0x11000088, 0x09000090, 0x08800110,

0x04600620, 0x02181840, 0x0107E080, 0x00C00300,

0x00300C00, 0x000FF000, 0x00000000, 0x00000000,

};

for(int y = 0; y < 32; ++y)

{

for(int x = 0; x < 64; ++x)

{

putchar((table[y] & (1 << (x >> 1))) ? 'X' : ' ');

}

putchar('\n');

Sleep(1000\*3);

system("cls");

goto START;

}

}

}

}

else if(flag==0)

{

goto END;

}

else

{

printf("\n\t输入错误，3秒后可重新输入\n");

Sleep(3\*1000);

system("cls");

goto START;

}

}

}

printf("\n\t未找到！\n");

printf("\n\t将在3秒后跳回选课界面\n");

Sleep(3\*1000);

system("cls");

goto START;

}

else

{

printf("\n\t请输入课程名称(全部小写，单词分隔用一个空格)\n");

char \*pt=correctchar(lazy);

for(i=0; i<6; i++)

{

if(strcmp(pt,name[i])==0)

{

goto FIRST;

}

}

printf("\n\t未找到！\n");

printf("\n\t将在3秒后跳回选课界面\n");

Sleep(3\*1000);

system("cls");

goto START;

}

END:

system("pause");

printf("\n\t按1回到上一级\n");

printf("\n\t继续操作按0\n");

flag=correct();

if(flag)

return 1;

else

return 0;

return 0;

}

int search (int lazy,COURSE cour[])//找课，通过名称，编号，学分

{

system("title 查找课程");

int flag;

printf("\n\t按0回到上一级\n");

printf("\n\t继续操作按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

int a;

char s[26];

int i,k,label=1;

system("cls");

START:

system("cls");

printf("\n\t请选择你想查询的方式：\n1.课程名称查询\n2.课程编号查询\n3.学分查询\n4.周学时查询\n5.退出系统\n6.返回上一级\n请选择序号:\t\n");

a=correctint(lazy);

switch(a)

{

case 1:

printf("\n输入课程名称:(单词间以一个空格作为分割,请输入小写)\n");

strcpy(s,correctchar(lazy));

for(i=0; i<6; i++)

{

if(strcmp(s,name[i])==0)

{

printf("\n\t----------------------------------------------------------------\n");

printf("课程编号：%d|---课程学分：%d|---每星期学时：%-6d|---开课周：%d|\n",cour[k].num,cour[k].score,cour[k].classPeriod,cour[k].week);

label=0;

}

}

if(label)

{

printf("\n未找到，将在2秒后跳转回选择页面\n");

goto START;

}

printf("\n\t---------------------------------------------------------------------------------\n");

printf("\n是否继续查询1继续0退出\n");

flag=correct();

if(flag) goto START;

else goto END;

break;

case 2:

system("cls");

printf("\n\t输入课程编号:\n");

a=correctint(lazy);

printf("\n\t---------------------------------------------------------------------------------\n");

for(i=0; i<6; i++)

{

if(a==cour[i].num)

{

printf("课程名称：");

printf("%-25s|",name[i]);

printf("---课程学分：%-8d|---每星期学时：%-6d|---开课周：%-2d|\n",cour[i].score,cour[i].classPeriod,cour[i].week);

label=0;

}

if(label)

{

printf("\n未找到，将在2秒后跳转回选择页面\n");

Sleep(1000\*2);

goto START;

}

system("pause");

printf("\n\t---------------------------------------------------------------------------------\n");

printf("\n是否继续查询1继续0退出\n");

flag=correct();

if(flag) goto START;

else goto END;

}

break;

case 3:

system("cls");

printf("\n\t输入你想查的学分:\n");

a=correctint(lazy);

printf("\n\t----------------------------------------------------------------------------------------\n");

for(i=0; i<6; i++)

{

if(a==cour[i].score)

{

printf("课程名称：");

printf("%-25s|",name[i]);

printf("---课程编号：%-8d|---每星期学时：%-6d|---开课周：%-2d|\n",cour[i].num,cour[i].totalPeriod,cour[i].week);

label=0;

}

}

if(label)

{

printf("\n未找到，将在2秒后跳转回选择页面\n");

Sleep(1000\*2);

goto START;

}

system("pause");

printf("\n\t---------------------------------------------------------------------------------\n");

printf("\n是否继续查询1继续0退出\n");

flag=correct();

if(flag) goto START;

else goto END;

break;

case 4:

system("cls");

printf("\n\t课程包括每周2学时课程和每周3学时课程\n");

a=correctint(lazy);

if(a!=2||a!=3)

{

printf("\n输入错误，请重输\n");

goto START;

}

printf("\n\t--------------------------------------------------------------------------------------\n");

for(i=0; i<6; i++)

{

if(a==cour[i].classPeriod)

{

printf("课程名称：");

printf("%-25s|",name[i]);

printf("---课程编号：%-8d|---课程学分：%-6d|---开课周：%-2d|\n",cour[i].num,cour[i].score,cour[i].week);

label=0;

}

}

if(label)

{

printf("\n\t未找到，将在2秒后跳转回选择页面\n");

goto START;

}

system("pause");

printf("\n\t---------------------------------------------------------------------------------\n");

printf("\n\t是否继续查询1继续0退出\n");

flag=correct();

if(flag) goto START;

else goto END;

break;

case 5:

exit(0);

break;

case 6:

goto END;

break;

default:

printf("\n\t输入错误,将在3秒后跳回选择页面\n");

Sleep(1000\*3);

goto START;

}

END:

printf("\n\t按0回到上一级\n");

printf("\n\t退出系统按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

return 0;

}

int scan(int lazy,int n,STUDENT stu[])//学生浏览自己的信息

{

system("title 相关信息");

int flag,j;

printf("\n\t按0回到上一级\n");

printf("\n\t继续操作按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

printf("\t你的信息如下：\n");

printf("学生学号\t");

printf("%d\t",stu[n].stuNo);

printf("\n已选课程\t");

for(j=0; j<6; j++)

if(stu[n].num[j])

printf("\n\t%-25s\t\n",name[j]);

printf("\n还需要选的学分\t");

printf("%d\t",stu[n].score);

printf("\n");

system("pause");

printf("\n\t按1回到上一级\n");

printf("\n\t退出系统按0\n");

flag=correct();

if(flag)

return 1;

else

return 0;

}

void trans (COURSE cour[],XY pos[])

{

int k,i;

for(k=0; k<6; k++)

{

for(i=0; i<cour[k].totalPeriod\*2; i++)

{

if((i+1)%2)

{

pos[k].y[cour[k].totalPeriod-(i+1)/2-1]=work(cour[k].classPeriod,i+1);

}

else

{

pos[k].x[cour[k].totalPeriod-(i+2)/2]=work(cour[k].classPeriod,i+1);

}

}

}

}

int timetable(int lazy,int n,COURSE cour[],STUDENT stu[])//课表打印

{

system("title 课表");

int flag;

printf("\n\t按0回到上一级\n");

printf("\n\t继续操作按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

XY pos[6];

trans(cour,pos);

char cl[100][26];

int i,j,k;

char pr[8][9][26];

char s0[26]=" ";

for(i=0; i<9; i++)

{

for(j=0; j<8; j++)

{

strcpy(pr[i][j],s0);

}

}

for(i=0; i<6; i++)

{

if(stu[n].num[i])

{

for(k=0; k<cour[i].totalPeriod; k++)

{

strcpy(pr[pos[i].x[k]][pos[i].y[k]],name2[i]);

}

}

}

strcpy(pr[0][0]," ");

strcpy(pr[1][0]," 星期一");

strcpy(pr[2][0]," 星期二");

strcpy(pr[3][0]," 星期三");

strcpy(pr[4][0]," 星期四");

strcpy(pr[5][0]," 星期五");

strcpy(pr[6][0]," 星期六");

strcpy(pr[7][0]," 星期天");

strcpy(pr[0][1],"第一节");

strcpy(pr[0][2],"第二节");

strcpy(pr[0][3],"第三节");

strcpy(pr[0][4],"第四节");

strcpy(pr[0][5],"第五节");

strcpy(pr[0][6],"第六节");

strcpy(pr[0][7],"第七节");

strcpy(pr[0][8],"第八节");

for(i=0; i<8; i++)

printf("%s",pr[i][0]);

printf("\n");

printf("\n");

for(j=1; j<8; j++)

{

for(i=0; i<8; i++)

{

printf("%s ",pr[i][j]);

}

printf("\n");

printf("\n");

}

system("pause");

printf("\n\t按0回到上一级\n");

printf("\n\t退出系统按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

return 0;

}

int lookjournal (int lazy,int n)//查看日志

{

system("title 日志");

int flag;

printf("\n\t按0回到上一级\n");

printf("\n\t继续操作按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

system("type D:\\CBproject\\学生选课系统之菜单\\1.txt");

system("pause");

printf("\n\t按0回到上一级\n");

printf("\n\t退出系统按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

return 0;

}

int changestudent(int lazy,int n,STUDENT stu[],COURSE cour[])//改变学生相关信息

{

system("title 学生信息修改");

int flag;

printf("\n\t按0回到上一级\n");

printf("\n\t继续操作按1\n");

flag=correct();

if(!flag)

return 1;

START:

system("cls");

printf("\n\t请选择要修改谁的信息(请输入编号)退出请按0\n");

flag=correctint(lazy);

if(flag==0)

goto EXIT;

int i,k,m=0;

for(i=0; i<4; i++)

{

if(flag==stu[i].stuNo)

{

CHANGE:

system("cls");

printf("------------------------------------------------------您想修改什么信息？\n");

printf("------------------------------------------------------按0退出页面\n");

printf("------------------------------------------------------学生选课按1\n");

printf("------------------------------------------------------学生退课按2\n");

printf("------------------------------------------------------学生学分修改按3\n");

flag=correctint(lazy);

switch(flag)

{

case 1:

m++;

case 2:

printf("\n请选择是课程的名称（0）或编号（1）\n");

flag=correct();

if(flag)

{

printf("\n请输入编号\n");

flag=correctint(lazy);

for(k=0; k<6; k++)

{

if(flag==cour[k].num)

{

printf("\n请确定1确定0退出\n");

flag=correct();

if(flag)

{

if(!m)

{

cour[i].people++;

stu[n].score=stu[n].score+cour[i].score;

stu[n].num[i]--;

printf("\n\t修改成功\n");

Sleep(1000\*2);

system("cls");

goto CHANGE;

}

else

{

cour[i].people--;//课程人数减1

stu[n].score=stu[n].score-cour[i].score;//学生的学分减去课程的学分

stu[n].num[i]++;//表示学生选的什么课

printf("\n\t修改成功\n");

Sleep(1000\*2);

system("cls");

goto CHANGE;

}

}

else

{

goto CHANGE;

}

}

}

printf("\n未找到！将跳转回原来界面！\n");

goto CHANGE;

}

else

{

printf("\n请输入名称\n");

char \*s;

strcpy(s,correctchar(lazy));

for(k=0; k<6; k++)

{

if(strcmp(s,name[k]))

{

printf("\n请确定按1继续操作按0不修改\n");

flag=correct();

if(flag)

{

cour[i].people++;

stu[n].score=stu[n].score+cour[i].score;

stu[n].num[i]--;

printf("修改成功！");

system("pause");

goto EXIT;

}

else

{

goto CHANGE;

}

}

}

printf("\n未找到！将跳转回原来界面！\n");

system("cls");

goto CHANGE;

}

break;

case 3:

CHANGE3:

system("cls");

printf("\n请输入你想要的值\n");

stu[i].score=correctint(lazy);

printf("\n如果还想继续修改该学生的信息按0\n");

printf("\n如果想修改别的学生的信息按1\n");

printf("\n如果想退出系统按2\n");

flag=correctint(lazy);

switch(flag)

{

case 1:

goto START;

break;

case 2:

goto EXIT;

break;

case 0:

goto CHANGE;

break;

default:

{

printf("\n输入错误，请重输!\n");

goto CHANGE3;

}

}

break;

case 0:

goto EXIT;

break;

default:

{

printf("\n输入错误，请重输！\n");

goto CHANGE;

}

}

}

}

printf("\n没有找到！将在3秒后跳回操作页面\n");

Sleep(1000\*3);

goto START;

EXIT:

printf("按0回到上一级");

printf("\n\t退出系统按1\n");

flag=correct();

if(!flag)

return 1;

system("cls");

return 0;

}

int main(void)//由密码板块，文件板块，修改板块，打印板块组成

{

system("title 教务系统");

//创建密码数组，学生数组，课程数组

char password[5][26];

COURSE cour[6];

STUDENT stu [4];

XY pos1[6];

//读取文件信息，赋值给上述数组

//writefile(cour,stu);

readfile(cour,stu);

readpassword(password,5);

trans(cour,pos1);

//format(cour,stu,pos);

// writejn();

int c;

//以下为主菜单

printf("\t 欢迎来到教务平台\t\n");

Sleep(1000);

PASS:

system("title 教务系统登录");

system("cls");

printf("\t请注意中途退出不会保存\n\n\t在第一次输入前您可以选择相信自己，无须确认,\n\n\t或者每次输入都检查一遍。\n\n\t如果不想检查请按1，否则0\n\n\t以回车作为结束标准(默认0，且所有0，1输入以最后一个为准,最多可输入5次)\n");

int lazy=correct();

printf("\n");

int flag=passwords(password,5,lazy);//返回值为在密码数组中的位置，我将管理员放在第0个

if(flag>0)//学生板块

{ flag--;//这个是为和学生数组匹配

START:

system("cls");

system("title 教务系统学生端");

printf("-------------------按1检索课程相关信息\n");

printf("-------------------按2查看自己信息\n");

printf("-------------------按3查看当前课表\n");

printf("-------------------按4修改密码\n");

printf("-------------------按5选课或退课\n");

printf("-------------------按6退出系统\n");

printf("-------------------按7切换账号\n");

c=correctint(lazy);

switch(c)

{

case 1:

system("cls");

if(search(lazy,cour))

goto START;

else

goto END;

break;

case 2:

system("cls");

if(scan(lazy,flag,stu))

goto START;

else

goto END;

break;

case 3:

system("cls");

if(timetable(lazy,flag,cour,stu))

goto START;

else

goto END;

break;

case 4:

system("cls");

if(change(password,flag+1,lazy))

goto START;

else

goto END;

break;

case 5:

system("cls");

if(choose(lazy,cour,stu,flag))

goto START;

else

goto END;

break;

case 6:

system("cls");

goto END;

break;

case 7:

system("cls");

goto PASS;

break;

default:

{

printf("\n\t输入错误，将在5秒后跳转回系统首页\n");

Sleep(5\*1000);

system("cls");

goto START;

}

}

}

else if(flag==0)//管理员板块

{

START1:

system("cls");

system("title 教务系统管理员");

printf("-------------------按1查看课程和学生的信息\n");

printf("-------------------按2对学生的信息进行修改\n");

printf("-------------------按3查看课程日志\n");

printf("-------------------按4修改密码\n");

printf("-------------------按5退出系统\n");

c=correctint(lazy);

switch(c)

{

case 1:

system("cls");

if(format(cour,stu,pos1))

goto START1;

else

goto END;

break;

case 2:

system("cls");

if(changestudent(lazy,flag,stu,cour))

goto START1;

else

goto END;

break;

case 3:

system("cls");

if(lookjournal(lazy,flag))

goto START1;

else

goto END;

break;

case 4:

system("cls");

if(change(password,0,lazy))

goto START1;

else

goto END;

break;

case 5:

system("cls");

goto END;

break;

default:

{

printf("\n\t输入错误，将在5秒后跳转回系统首页\n");

Sleep(5\*1000);

system("cls");

goto START1;

}

}

}

else//欢迎再次光临

{

END:

system("cls");

system("title 欢迎再次光临");

printf("\n\t欢迎再次光临\n");

}

//将更改后的信息写入文件

writepassword (password,5);

writefile(cour,stu);

return 0;

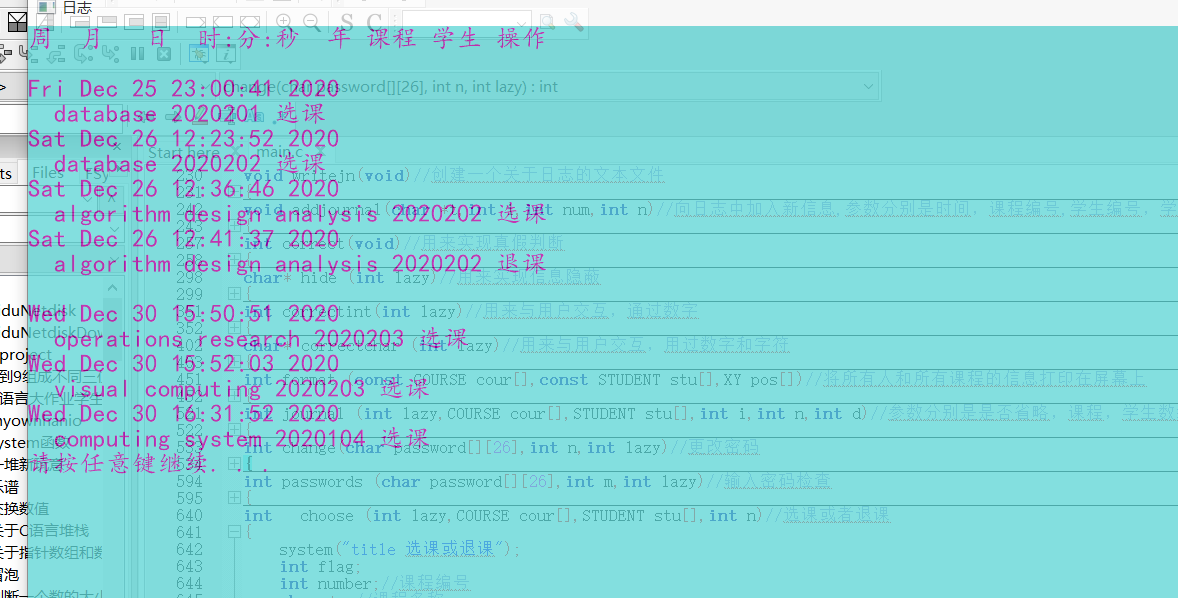
}

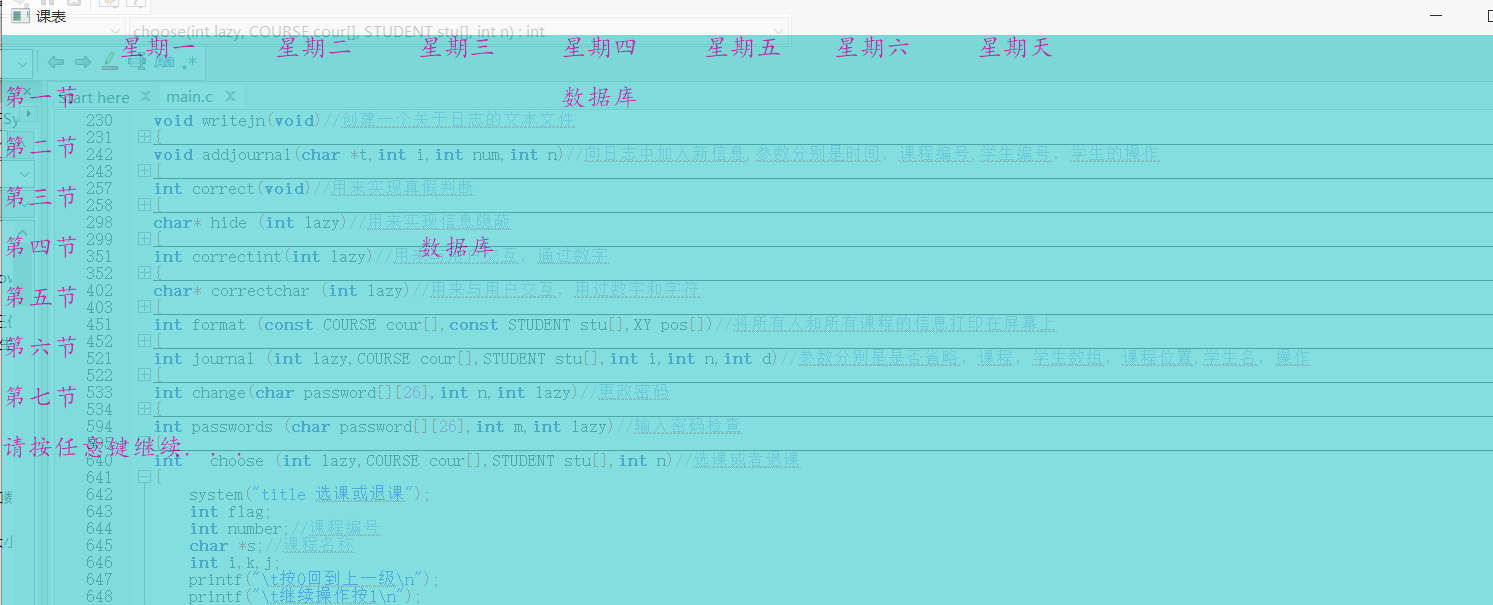
#### 六、实验结果

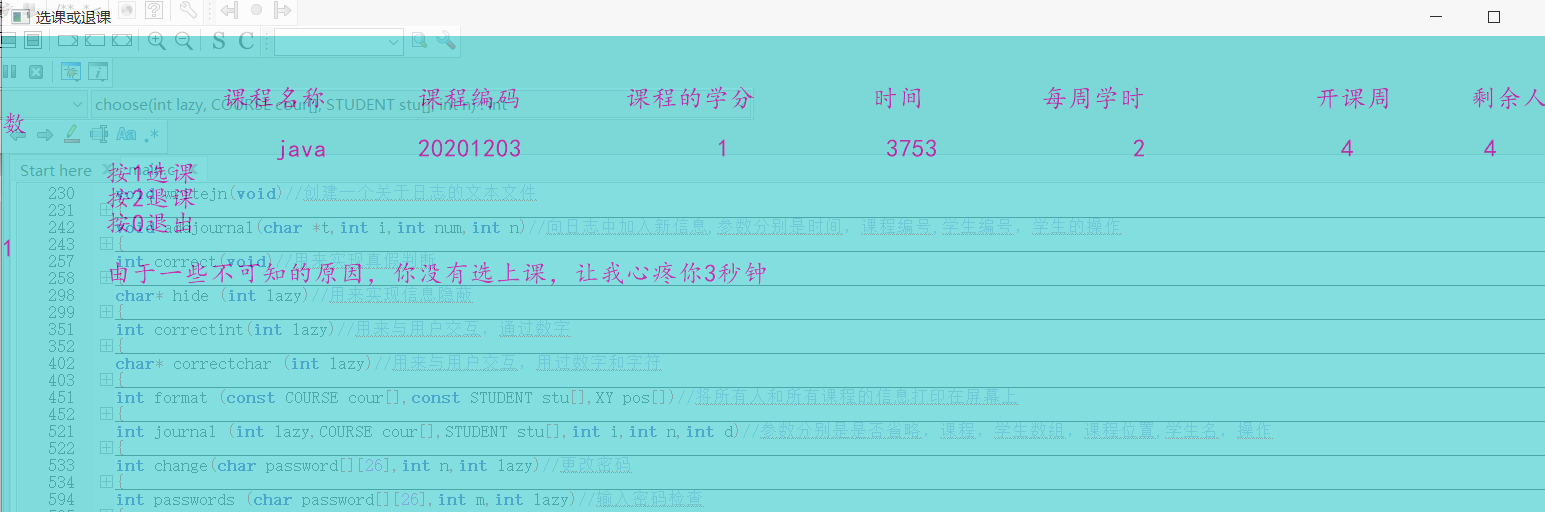
**1．使用说明：**

需要提前知道账号密码，其他按提示来就行了。另外就是别点太快，我的输入函数将enter作为结束标准，这样返回值就可能不是用户想要的了。

1. 运行截图：







#### 七、遇到的问题及解决方法

**1、输入函数无法使用退格键，输入的字符如何转化为整形。解决方法是将退格键的实现变为先退格再空格再退格再空格，达到将用户数据吃掉的效果。输入字符转换为整形本来打算用atoi函数去做，但是不知道为什么就是行不通，所以自己用了一个小函数实现了这个功能。**

**2、文件操作遇到的最大的问题就是带空格的字符串用scanf读取不方便，操作不方便，解决方法就是规避掉这个问题，当然最开始想的是用fgets 和fgetc但是怕写不完， 后来发现写的完了，又不想去修改了。**

**3、在密码板块遇到的一个问题是字符串复制的问题，就是丢了一点，最后怎么解决的忘了。**

**4、在其他的板块也有许多问题，但是没记住，太多了。不过最后全部解决了。**

**5、整个程序最大的问题就是简单，甚至连指针都没用，更别说链表，栈，队列了。但是**

**这也就是说可以使用其他语言去再次编写，体会不同语言的优缺点，为未来的学习做好准备。**