**程序设计项目论文——《高校宿舍管理系统》**

**目录：**

**第一章：前言**

**第二章：系统功能描述**

**第三章：系统设计过程**

**第四章：系统测试**

**第五章：总结**

**英文摘要：**

**This system is in the Windows operating system as well as the VS2010 compilation environment. Through the DOS console window as interface to operate. This system operation is simple. Not too much complicated operation and program statements. And the operation environment is quite simple. The program readability is higher.**

**中文摘要：**

本系统是在windows操作系统以及VS2010的编译环境下进行的，通过dos控制台窗口作为界面进行操作的，本系统操作简单，没有过多复杂的操作以及程序语句，且运行环境简单，适合大部分机型，同时本程序可读性较高。

**中文关键词：**

管理、统计、方便。

**英文关键词**

**Management、statistical、convenient.**

1. **前言**

由于此门课程属于必修课程的范畴，故为了满足老师对此门课程的要求，我选择编写一个《高校宿舍管理系统》，同时也因为高校宿舍的管理比较贴近我的生活，因此选择《高校宿舍管理系统》作为我的项目题目，并通过我的生活体验来编写这个项目，既能够让我们更加了解我们的生活。

通过这个系统，我们能够录入学生信息，并且能通过搜索功能提取学生信息，并且还具有学生入宿、退宿登记，同时还能够批量删除学生记录，还拥有宿舍基本信息管理功能等。

1. **系统功能描述**

1. 系统初始界面：要求管理员必须输入正确的用户名和密码才能进入系统。

2. 提供对宿舍基本信息的管理。

3. 提供学生住宿情况的基本登记。

4. 提供学生每学期学生的离校处理。

5. 提供早出晚归登记及结束访问的详细登记。

6. 提供查询功能，以方便用户对学生和宿舍信息的查询

7、登录管理：允许管理员登录和退出系统。

8. 学生基本信息管理：

提供添加、修改、删除、查询学生基本信息的功能。有些基本信息在入宿后就不允许修改或删除，如性别入宿后不能修改、删除只能在入宿，且入宿后就不能删除。

9. 住宿管理：

宿舍基本信息管理——添加、修改、删除、查询所有宿舍基本信息。

入宿登记——登记、修改学生入宿信息。

10、出入管理：学生早出登记、学生晚归登记。

11. 信息查询 ：通过关键字或信息进行搜索，从而得到该学生信息。

**三、系统设计过程**

1、登录管理：通过布尔代数的方式进行判断管理员所输入的密码是正确只有当返回值为true的时候才能打开下面所编写的程序文件，才能得到访问所有宿舍信息的权限。在开始时先设置密码，并将其值用布尔代数输出返回值true，使其在下次输入时能够满足条件，得以进入系统。如果不符合所设置的密码，则输出false，并进行报错，提醒管理员重新输入密码以获得进入系统的权限。

2、学生基本信息管理：通过编写一个student类文件的形式，将学生各项信息当做私有成员进行输入数据，并采用类的继承的方法进行其他类的编写，这样就能有效地减少其他不必要的操作，减少错误的发生，同时还能防止学生信息被他人修改而造成不必要的损失。通过构造成员函数的形式进行对学生基本信息的录入、读取及其保存；之后运用new和delete的语句对学生信息创建一个存储空间，并且使其可以在某些时刻进行删除学生信息操作。

3、住宿管理（此功能未完善）：其大致的操作流程与学生信息差不多，也是通过建立一个dormitory类文件，并通过类的继承的方式进行内容的扩充，运用new和delete的语句进行对宿舍信息的保存和删除。

4、出入管理：通过对学生出入信息的数据的录入以及赋予其临时的存储空间，以便于删除数据以及查看数据，此处需要管理员自主输入信息，从而才能进行保存信息。

5、信息查询：通过对学生信息的关键字，比如姓名、学号、宿舍、和床号等方式进行检索通过文件流的方式进行学生信息数据的提取和输出，同样也要利用类的继承的方法，从而得以调用student类文件的信息，而不至于报错。

四、系统测试

在进行了多次对程序的编译和测试，发现了很多之前没有考虑到的问题，比如创建多文件的时候，编译时会出现类未定义的错误，面对这种情况迟迟无法得到解决，因此我选择用一个项目来运行这个系统，能够免去一些不必要的麻烦，同时在面对宿舍类的算法，**比如对宿舍床位的批量分配的问题，始终无法得到一个可以解决的算法，所以我决定省去这个功能**，**改成人工输入学生宿舍信息**，其它的功能也基本得到了实现，**由于没有设计好宿舍类，故只有一个初始化功能，将数据全部取消。**

五、总结

通过这次的系统项目的编写，我能够更好的掌握关于类定义的知识，同时也初步掌握了有关文件流的使用，了解了通过什么操作和程序语句可以达到存储数据、读取数据和删除数据的作用，也为接下来的数据结构课程做了一个充分的准备。除此之外，我也通过这次程序项目课程的学习中也发现了自身在学习上的弱点，以致于迟迟无法完成项目的全部功能，因此在接下来的学习中，我会继续加强对编程的学习，让自己可以从容的解决今后可能会遇到的编程难题，让自己可以掌握更多知识。

附程序代码：

#include <iostream>

#include <fstream>

#include <string>

#include <iomanip>

#include <conio.h>

#include <stdio.h>

#include <windows.h>

using namespace std;

class Student{

private:

string S\_user,S\_password; //用户名，密码

string S\_name; //学生姓名

string S\_sex; //性别

string S\_major; //专业

int S\_class; //班级

string S\_id; //学号

string S\_quarters;

int S\_bednum; //寝室号

string S\_date; //日期

string S\_time; //时间

char S\_sign; //早出晚归标志

bool Open\_flag; //文件是否载入标志

bool Alter\_flag; //修改标志

bool Exit\_flag; //退出标志

int Recond,U\_count; //记录学生信息数，用户数

int S\_position; //查找学生的位置

public:

Student();

virtual ~Student(); //析构函数

void Start(Student &L); //启动界面

void Register(Student &L); //注册

void Login(Student &L); //登陆

void Menu(Student &L); //菜单

void Load(Student &L); //验证是否载入信息

void ReadFromFile(Student &L); //从文件中读出数据

void Search(Student &L); //搜索数据成员

void Search\_On\_Name(Student &L); //根据姓名搜索

void Search\_On\_Id(Student &L); //根据学号搜索

void ModifyMember(Student &L); //修改成员数据

void DeleteMember(Student &L); //删除成员

void PrintAll(Student &L); //打印所有数据

void AddMember(Student &L); //添加成员

void SaveToFile(Student &L); //保存文件

void ReturnMenu(Student &L); //返回菜单

void S\_exit(Student &L); //退出系统

void Print(Student &L); //输出早出晚归信息

void SaveFile(Student &L); //保存早出晚归信息

void ReadFile(Student &L); //读取早出晚归系统

void BackandOut(Student &L); //输入早出晚归信息

};

//构造函数

Student::Student()

{

Open\_flag=false;

Alter\_flag=false;

Exit\_flag=true;

Recond=U\_count=0;

S\_position=-1;

}

Student stu[50]; //定义类成员-学生信息数

Student admin[50]; //定义类成员-管理员数量

Student::~Student(){}

//初始化界面函数

void Student::Start(Student &L)

{

system("cls");

cout<<"\n\n\n\n\t\t\t\ \*欢迎进入华南师范大学宿舍管理系统\*\n\n"<<endl;

cout<<"\t\t1.用户注册";

cout<<"\t\t2.用户登陆";

cout<<"\t\t0.退出系统 "<<endl;

cout<<"\n\n\t\t\t请选择你的操作： ";

int Key;

point:

cin>>Key;

switch(Key)

{

case 0:exit(0);break;

case 1:L.Register(L);break;

case 2:L.Login(L);break;

default:

{

cout<<"\n输入错误，请重新选择： ";

L.Start(L);

goto point;break;

}

}

}

//注册函数

void Student::Register(Student &L)

{

system("cls");

string \_user,\_password;

ifstream infile;

infile.open("administrator.txt",ios\_base::in);

if(!infile){

cerr<<"打开失败！"<<endl;

exit(1);

}

for(int i=0;!infile.eof();++i)

{infile>>admin[i].S\_user>>admin[i].S\_password;U\_count = i;}

cout<<"\t\t\t \*欢迎使用华南师范大学宿舍管理系统，请先进行注册!\*"<<endl;

cout<<"\n\n\n\t\t请输入用户名（长度不超过10位）： ";

cin>>\_user;

cout<<"\n\n\t\t请输入密码（密码长度为6位）： ";

cin>>\_password;

for(int j=0;j<U\_count;++j)

if(admin[j].S\_user==\_user) {

cout<<"\t\t\*用户已存在！请重新注册！\*"<<endl;

getch();

L.Register(L);

}

cout<<"\t\t注册成功！\n";

infile.close();

U\_count++;

admin[U\_count-1].S\_user=\_user;

admin[U\_count-1].S\_password=\_password;

ofstream outfile;

outfile.open("administrator.txt",ios::out);

for(int k=0;k<U\_count;k++)

outfile<<admin[k].S\_user<<" "<<admin[k].S\_password<<endl;

outfile.close();

L.Start(L);

system("cls");

}

//登录验证函数

void Student::Login(Student &L)

{

ifstream infile("administrator.txt",ios::in);

if(!infile) {

cout<<"文件无法打开!"<<endl;

exit(1);

}

for(int i=0;!infile.eof();++i)

{infile>>admin[i].S\_user>>admin[i].S\_password;U\_count = i;}

char \_user[10],\_password[7];

bool u=false,p=false;

int c,t=1;

do {

system("cls");

cout<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\t \*欢迎使用华南师范大学宿舍管理系统，请先进行登陆!\*"<<endl;

cout<<endl<<endl<<endl<<endl<<endl<<endl;

cout<<"\t\t\t请输入用户名:";

fflush(stdin);

gets(\_user);

cout<<endl<<"\t\t\t请输入密码:";

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

c=getch();

if(c==13) break;

if(i&&(c=='\b')){

i-=2;

cout<<'\b';

cout<<' ';

cout<<'\b';

}

else if(c=='\b')

i--;

else {

\_password[i]=c;

cout<<"\*";

}

}

\_password[6]='\0';

getch();

for(int j=0;j<=U\_count;++j){

if(admin[j].S\_user==\_user) {

u=true;

if(admin[j].S\_password==\_password) p=true;

}

}

if(!u){

cout<<"\n\t\t用户不存在！请重新登陆！"<<endl;

getch();

t++;

continue;

}

else if(!p){

cout<<"\n\t\t密码错误，请重新登陆！"<<endl;

getch();

t++;

continue;

}

else L.Menu(L);

}while(t<=3);

if(t>3){

cout<<"\n\n\t\t连续3次登陆失败，系统自动关闭"<<endl;

if(getch()=='\r') cout<<"\b";

exit(1);

}

infile.close();

system("cls");

}

//主菜单

void Student::Menu(Student &L)

{

system("cls");

cout<<"\n\n\n\n\t\t------------华南师范大学宿舍管理系统------------\n\n"<<endl;

cout<<"\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\t\t1、初始化系统 \t\t2、添加学生信息"<<endl;

cout<<"\t\t3、查询学生信息\t\t4、修改学生信息"<<endl;

cout<<"\t\t5、删除学生信息\t\t6、保存学生信息"<<endl;

cout<<"\t\t7、输出学生信息\t\t8、早出晚归登记"<<endl;

cout<<"\t\t9、早出晚归信息\t\t0、退出系统"<<endl;

cout<<"\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\n\n\t请选择功能： ";

int i;

point:

cin>>i;

switch(i)

{

case 1:L.ReadFromFile(L);break;

case 2:L.AddMember(L);break;

case 3:L.Search(L);break;

case 4:L.ModifyMember(L);break;

case 5:L.DeleteMember(L);break;

case 6:L.SaveToFile(L);break;

case 7:L.PrintAll(L);break;

case 8:L.BackandOut(L);break;

case 9:L.Print(L);break;

case 0:exit(1);break;

default:

{

cout<<"命令输入错误，请重新选择：";

goto point;break;

}

}

}

//从文件中读取数据成员函数

void Student::ReadFromFile(Student &L)

{

system("cls");

ifstream file;

file.open("Student.txt",ios\_base::in);

if(!file){

cerr<<"can't open the file!"<<endl;

exit(1);

}

cout<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

for(int i=0;!file.eof();++i)

{file>>stu[i].S\_name>>stu[i].S\_sex>>stu[i].S\_major>>stu[i].S\_class>>stu[i].S\_id>>stu[i].S\_quarters>>stu[i].S\_bednum;Recond = --i;}

Open\_flag=true;

cout<<"\t\t载入数据完毕！请选择操作："<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

file.close();

L.ReturnMenu (L);

}

void Student::ReadFile(Student &L)

{

system("cls");

ifstream file;

file.open("BackandOut.txt",ios\_base::in);

if(!file){

cerr<<"can't open the file!"<<endl;

exit(1);

}

cout<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

for(int i=0;!file.eof();++i)

{file>>stu[i].S\_name>>stu[i].S\_sex>>stu[i].S\_id>>stu[i].S\_sign>>stu[i].S\_date>>stu[i].S\_time;Recond = --i;}

Open\_flag=true;

cout<<"\t\t载入数据完毕！请选择操作："<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

file.close();

L.ReturnMenu (L);

}

//载入信息函数

void Student::Load(Student &L)

{

char load\_temp;

if(!Open\_flag){

cout<<"尚未载入信息，现在是否载入信息（Y/N）?"<<endl;

cin>>load\_temp;

if(load\_temp=='Y'||load\_temp=='y'){

L.ReadFromFile(L);

}

else

exit(1);

}

}

//打印成员函数

void Student::PrintAll(Student &L)

{

system("cls");

L.Load(L);

cout<<setw(6)<<"姓 名"<<setw(8)<<"性 别"<<setw(10)<<"专 业"<<setw(8)<<"班 级"

<<setw(11)<<"学 号"<<setw(9)<<"寝室号"<<setw(7)<<"床号"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

for(int i=0;i<=Recond;i++){

cout<<setw(6)<<stu[i].S\_name<<setw(8)<<stu[i].S\_sex<<setw(10)<<stu[i].S\_major

<<setw(6)<<stu[i].S\_class<<setw(14)<<stu[i].S\_id<<setw(6)<<stu[i].S\_quarters<<setw(6)<<stu[i].S\_bednum<<endl;

}

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\n输出完毕！请选择操作："<<endl;

L.ReturnMenu (L);

}

void Student::Print(Student &L)

{

system("cls");

L.Load(L);

cout<<" "<<"姓 名"<<" "<<"性 别"<<" "<<"学 号"

<<" "<<"早出/晚归"<<" "<<"日期"<<" "<<"时间"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

for(int i=0;i<=Recond;i++){

cout<<" "<<stu[i].S\_name<<" "<<stu[i].S\_sex<<" "

<<stu[i].S\_id<<" "<<stu[i].S\_sign<<" "<<stu[i].S\_date<<" "<<stu[i].S\_time<<endl;

}

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\n输出完毕！请选择操作："<<endl;

L.ReturnMenu (L);

}

//添加成员函数

void Student::AddMember(Student &L)

{

system("cls");

L.Load(L);

Recond++;

cout<<"请输入姓名："<<endl;

cin>>stu[Recond].S\_name;

cout<<"请输入性别："<<endl;

cin>>stu[Recond].S\_sex;

cout<<"请输入专业："<<endl;

cin>>stu[Recond].S\_major;

cout<<"请输入班级"<<endl;

cin>>stu[Recond].S\_class;

cout<<"请输入学号："<<endl;

cin>>stu[Recond].S\_id;

cout<<"请输入寝室号："<<endl;

cin>>stu[Recond].S\_quarters;

cout<<"请输入床号："<<endl;

cin>>stu[Recond].S\_bednum;

cout<<"\n添加信息完毕！请选择操作："<<endl;

char temp;

cout<<"是否保存(Y/N)?"<<endl;

cin>>temp;

if(temp=='y'||temp=='Y'){

L.SaveToFile(L);

}

else{

cout<<"若暂不保存，";

L.ReturnMenu (L);

}

}

void Student::BackandOut(Student &L)

{

system("cls");

L.Load(L);

Recond++;

cout<<"请输入姓名："<<endl;

cin>>stu[Recond].S\_name;

cout<<"请输入性别："<<endl;

cin>>stu[Recond].S\_sex;

cout<<"请输入学号："<<endl;

cin>>stu[Recond].S\_id;

cout<<"晚归输入B，早出输入O："<<endl;

cin>>stu[Recond].S\_sign;

cout<<"请输入日期："<<endl;

cin>>stu[Recond].S\_date;

cout<<"请输入时间："<<endl;

cin>>stu[Recond].S\_time;

cout<<"\n添加信息完毕！请选择操作："<<endl;

char temp;

cout<<"是否保存(Y/N)?"<<endl;

cin>>temp;

if(temp=='y'||temp=='Y'){

L.SaveFile(L);

}

else{

cout<<"若暂不保存，";

L.ReturnMenu (L);

}

}

//搜索函数

void Student::Search(Student &L)

{

system("cls");

L.Load(L);

cout<<"搜索条件："<<endl;

cout<<"\t1、姓名 \t 2、学号 \t 0、退出搜索"<<endl;

int i;

cin>>i;

switch(i)

{

case 1:L.Search\_On\_Name(L);break;

case 2:L.Search\_On\_Id(L);break;

case 0:exit(1);break;

default:

{

cout<<"查找失败！";

L.ReturnMenu (L);

}

}

}

//根据姓名搜索

void Student::Search\_On\_Name(Student &L)

{

cout<<"寻找成员： ";

string \_name;

cin>>\_name;

for(int i=0;i<=Recond;i++){

if(\_name==stu[i].S\_name)

S\_position=i;

}

if(S\_position==-1){

cout<<"查找失败！";

L.ReturnMenu (L);

}

system("cls");

cout<<setw(6)<<"姓 名"<<setw(8)<<"性 别"<<setw(8)<<"专 业"<<setw(8)<<"班 级"

<<setw(9)<<"学 号"<<setw(10)<<"寝室号"<<setw(8)<<"床号"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<setw(5)<<stu[S\_position].S\_name<<setw(8)<<stu[S\_position].S\_sex<<setw(10)<<stu[S\_position].S\_major

<<setw(6)<<stu[S\_position].S\_class<<setw(14)<<stu[S\_position].S\_id<<setw(6)<<stu[S\_position].S\_quarters<<setw(6)<<stu[S\_position].S\_bednum<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\n搜索成功！";

L.ReturnMenu (L);

}

//根据学号搜索

void Student::Search\_On\_Id(Student &L)

{

cout<<"寻找成员： ";

string \_id;

cin>>\_id;

for(int i=0;i<=Recond;i++){

if(\_id==stu[i].S\_id)

S\_position=i;

}

if(S\_position==-1){

cout<<"查找失败！是否返回菜单(Y/N)？";

char temp;

cin>>temp;

if(temp=='y'||temp=='Y'){

L.Menu(L);

}

else

exit(1);

}

system("cls");

cout<<setw(6)<<"姓 名"<<setw(8)<<"性 别"<<setw(8)<<"专 业"<<setw(8)<<"班 级"

<<setw(9)<<"学 号"<<setw(10)<<"寝室号"<<setw(10)<<"床号"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<setw(6)<<stu[S\_position].S\_name<<setw(8)<<stu[S\_position].S\_sex<<setw(10)<<stu[S\_position].S\_major

<<setw(6)<<stu[S\_position].S\_class<<setw(14)<<stu[S\_position].S\_id<<setw(6)<<stu[S\_position].S\_quarters<<setw(6)<<stu[S\_position].S\_bednum<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\n搜索成功！";

L.ReturnMenu (L);

if(!Exit\_flag) L.Menu(L);

}

//修改成员函数

void Student::ModifyMember(Student &L)

{

system("cls");

L.Load(L);

int \_point;

Alter\_flag=false;

cout<<"修改学生信息:\n"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

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

cout<<"\t"<<i<<"、"<<stu[i].S\_name<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

cin>>\_point;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

for(int i=0;i<=Recond;i++){

if(\_point==i){

cout<<setw(6)<<stu[\_point].S\_name<<setw(8)<<stu[\_point].S\_sex<<setw(10)<<stu[\_point].S\_major

<<setw(6)<<stu[\_point].S\_class<<setw(14)<<stu[\_point].S\_id<<setw(6)<<stu[\_point].S\_quarters<<setw(6)<<stu[\_point].S\_bednum<<endl;

Alter\_flag=true;

}

}

if(!Alter\_flag){

cerr<<"修改失败！";

L.ReturnMenu (L);

}

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"修改信息"<<endl;

cout<<"1、姓名\t2、性别\t3、专业\n4、班级\t5、学号\t6、寝室号7、床号"<<endl;

int t;

char second;

char b;

point:

cin>>t;

switch(t)

{

case 1:

{

cout<<"请输入姓名：";

cin>>stu[\_point].S\_name;

cout<<"修改成功，是否还要修改（Y/N）?"<<endl;

cin>>second;

if(second=='y'||second=='Y')

goto point;

else {

cout<<"是否保存修改（Y/N）?"<<endl;

cin>>b;

if(b=='y'||b=='Y')

L.SaveToFile(L);

else exit(1);

}

break;

}

case 2:

{

cout<<"请输入性别：";

cin>>stu[\_point].S\_sex;

cout<<"修改成功，是否还要修改（Y/N）?"<<endl;

cin>>second;

if(second=='y'||second=='Y')

goto point;

else {

cout<<"是否保存修改（Y/N）?"<<endl;

cin>>b;

if(b=='y'||b=='Y')

L.SaveToFile(L);

else exit(1);

}

break;

}

case 3:

{

cout<<"请输入专业：";

cin>>stu[\_point].S\_major;

cout<<"修改成功，是否还要修改（Y/N）?"<<endl;

cin>>second;

if(second=='y'||second=='Y')

goto point;

else {

cout<<"是否保存修改（Y/N）?"<<endl;

cin>>b;

if(b=='y'||b=='Y')

L.SaveToFile(L);

else exit(1);

}

break;

}

case 4:

{

cout<<"请输入班级：";

cin>>stu[\_point].S\_class;

cout<<"修改成功，是否还要修改（Y/N）?"<<endl;

cin>>second;

if(second=='y'||second=='Y')

goto point;

else {

cout<<"是否保存修改（Y/N）?"<<endl;

cin>>b;

if(b=='y'||b=='Y')

L.SaveToFile(L);

else exit(1);

}

break;

}

case 5:

{

cout<<"请输入学号：";

cin>>stu[\_point].S\_id;

cout<<"修改成功，是否还要修改（Y/N）?"<<endl;

cin>>second;

if(second=='y'||second=='Y')

goto point;

else {

cout<<"是否保存修改（Y/N）?"<<endl;

cin>>b;

if(b=='y'||b=='Y')

L.SaveToFile(L);

else exit(1);

}

break;

}

case 6:

{

cout<<"请输入寝室号：";

cin>>stu[\_point].S\_quarters;

cout<<"修改成功，是否还要修改（Y/N）?"<<endl;

cin>>second;

if(second=='y'||second=='Y')

goto point;

else {

cout<<"是否保存修改（Y/N）?"<<endl;

cin>>b;

if(b=='y'||b=='Y')

L.SaveToFile(L);

else L.Menu(L);

}

break;

}

case 7:

{

cout<<"请输入床号：";

cin>>stu[\_point].S\_bednum;

cout<<"修改成功，是否还要修改（Y/N）?"<<endl;

cin>>second;

if(second=='y'||second=='Y')

goto point;

else {

cout<<"是否保存修改（Y/N）?"<<endl;

cin>>b;

if(b=='y'||b=='Y')

L.SaveToFile(L);

else exit(1);

}

break;

}

default:

{

cerr<<"修改失败！";

L.ReturnMenu (L);

if(!Exit\_flag) L.Menu(L);

}

}

}

//删除成员函数

void Student::DeleteMember(Student &L)

{

system("cls");

L.Load(L);

cout<<"删除成员"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

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

cout<<"\t"<<i<<"、"<<stu[i].S\_name<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

int \_point;

bool Del\_flag;

cin>>\_point;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

for(int i=0;i<=Recond;i++){

if(\_point==i){

cout<<setw(6)<<stu[\_point].S\_name<<setw(8)<<stu[\_point].S\_sex<<setw(10)<<stu[\_point].S\_major

<<setw(6)<<stu[\_point].S\_class<<setw(14)<<stu[\_point].S\_id<<setw(6)<<stu[\_point].S\_quarters<<setw(6)<<stu[\_point].S\_bednum<<endl;

Del\_flag=true;

}

}

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

if(Del\_flag){

cout<<"确定要删除？（Y/N）"<<endl;

char \_temp;

cin>>\_temp;

if(\_temp=='y'||\_temp=='Y'){

for(int j=\_point;j<=Recond;j++){

stu[j].S\_name=stu[j+1].S\_name;

stu[j].S\_sex=stu[j+1].S\_sex;

stu[j].S\_major=stu[j+1].S\_major;

stu[j].S\_class=stu[j+1].S\_class;

stu[j].S\_id=stu[j+1].S\_id;

stu[j].S\_quarters=stu[j+1].S\_quarters;

stu[j].S\_bednum=stu[j+1].S\_bednum;

}

Recond--;

cout<<"删除成功！是否保存修改(Y/N)？"<<endl;

char temp;

cin>>temp;

if(temp=='y'||temp=='Y'){

L.SaveToFile(L);

}

else {

cout<<"若暂不保存,";

L.ReturnMenu (L);

if(!Exit\_flag) L.Menu(L);

}

}

else L.Menu(L);

}

else L.ReturnMenu(L);

if(!Exit\_flag) L.Menu(L);

}

//保存文件函数

void Student::SaveToFile(Student &L)

{

system("cls");

L.Load(L);

ofstream ofile;

ofile.open("Student.txt",ios\_base::out);

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

ofile<<stu[i].S\_name<<' '<<stu[i].S\_sex<<' '<<stu[i].S\_major<<' '

<<stu[i].S\_class<<' '<<stu[i].S\_id<<' '<<stu[i].S\_quarters<<' '<<stu[i].S\_bednum<<endl;

ofile.close();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\t\t信息保存完毕！"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

L.ReturnMenu (L);

if(!Exit\_flag) L.Menu(L);

}

void Student::SaveFile(Student &L)

{

system("cls");

L.Load(L);

ofstream ofile;

ofile.open("BackandOut.txt",ios\_base::out);

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

ofile<<stu[i].S\_name<<' '<<stu[i].S\_sex<<' '

<<stu[i].S\_id<<' '<<stu[i].S\_date<<' '<<stu[i].S\_time<<endl;

ofile.close();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\t\t信息保存完毕！"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

L.ReturnMenu (L);

if(!Exit\_flag) L.Menu(L);

}

//返回菜单函数

void Student::ReturnMenu (Student &L)

{

cout<<"是否返回菜单(Y/N)?"<<endl;

char temp;

cin>>temp;

if(temp=='y'||temp=='Y')

L.Menu(L);

else

L.S\_exit(L);

}

//退出系统函数

void Student::S\_exit(Student &L)

{

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"确定退出系统(Y/N)?"<<endl;

char temp;

cin>>temp;

if((temp=='y'||temp=='Y'))

exit(0);

else

Exit\_flag=false;

}

int main()

{

Student L;

L.Start(L);

return 0;

}