四、类的设计

界面类：

class Interface//界面类

{

public:

};

static void IntroduceFace();//欢迎界面

static void Show();//操作界面

static void ManagerFace();//管理员功能界面

static void ManagerLoginFace();//管理员登录后界面

static void CustomerFace();//顾客功能界面

售票机类：

class TicketSeller//售票机类

{

public:

TicketSeller();//构造函数

void setMovieName(string m);//设置影片名

void setYear(int y);//设置年份

void setMonth(int m);//设置月份

void setDay(int d);//设置日期

void setPlayTime(int p);//设置场次

void setPrice(int p);//设置票价

void setFlag(int i,int j,int f);//设置标致

string getMovieName();//返回影片名

int getYear();//返回年份

int getMonth();//返回月份

int getDay();//返回日期

int getPlayTime();//返回场次

int getPrice(int i,int j);//返回票价

int getFlag(int i,int j);//返回标致

bool operator==(const int &pt)const//重定义==运算符函数

{

return (playTime==pt);

}

protected:

};

string movieName;//影片名称

int year;//影片上映年份

int month;//影片上映月份

int day;//影片上映日期

int playTime;//影片上映场次

int price[10][10];//price价格数组，存放某行某列座位的票价

int flag[10][10];//flag标致数组，某行某列座位为空则为0

售票机管理操作类：

class Manager:public TicketSeller//售票机管理操作类

{

private:

fstream ioFile;//文件流对象

vector<TicketSeller> M;//向量容器

M vector<TicketSeller>::iterator posM;//迭代器

posM static string ManagerName;//用户名

static string Password;//密码

public:

Manager();//构造函数

void Read();//文件数据读入向量容器

void Save();//向量容器数据写入文件

void SetInformation();//录入影片信息

void DelInformation();//删除影片信息

};

void Display();//显示影片信息

void Inquire();//查询售票信息

string GetManagerName();//获取登录名

string GetPassword();//获取密码

void Run();//管理系统运行按钮

售票机顾客操作类：

class Customer:public TicketSeller//售票机顾客操作类

{

private:

fstream ioFile;//文件流对象

vector<TicketSeller> C;//向量容器

C vector<TicketSeller>::iterator posC;//迭代器

posC public:

};

Customer();//构造函数

void Book();//定票

void Refund();//退票

五、程序代码与说明

界面类：

SYSTEMTIME t;//SYSTEMTIME为windows头文件里的类，用来获取系统时间

void Interface::IntroduceFace()

{

cout<<"\n\n\n\t\t !\n"

}

void Interface::Show()

{

cout<<"\n\n\t\t\t"<<"欢迎使用MBC电影世界自助售票机"<<endl<<endl <<"\t\t\t"<<"------------------------"<<endl

<<"\t\t\t"<<"| 1、管理员系统 |"<<endl <<"\t\t\t"<<"|

2、订退票系统 |"<<endl <<"\t\t ! / \\ ! \n" <<"\t\t / | \\ <MBC 电影世界> / | \\ \n" <<"\t\t -------||\_\_\_||\_\_\_\_\_\_\_\_\_\_\_\_\_\_||\_\_\_||------- \n" <<"\t\t [=][=] ||\_\_\_||\_\_\_\_\_\_||\_\_\_\_\_\_||\_\_\_|| [=][=] \n" <<"\t\t |\_||\_| ||\_\_\_||\_\_\_\_\_\_||\_\_\_\_\_\_||\_\_\_|| |=||=| \n" <<"\t\t |\_||\_| ||\_\_\_||\_\_\_\_\_\_||\_\_\_\_\_\_||\_\_\_|| |=||=| \n" <<"\t\t |\_||\_| ||\_\_\_||\_\_\_\_\_\_||\_\_\_\_\_\_||\_\_\_|| |=||=| \n" <<"\t\t |\_||\_| ||\_\_\_||\_\_\_\_\_\_||\_\_\_\_\_\_||\_\_\_|| |=||=| \n" <<"\t\t |\_||\_|\_||\_\_\_||\_\_\_\_\_\_||\_\_\_\_\_\_||\_\_\_||\_|=||=| \n\n\n\n";

cout<<"\t\t\t正在进入自助售票机，请稍后..."; \_sleep(2\*1000);

system("cls"); <<"\t\t\t"<<"------------------------"<<endl;

do{ char choice;

cout<<"\n\t\t\t"<<"请选择：";

cin>>choice; switch(choice)

}

{

case '1': ManagerFace();

break;

case '2': CustomerFace();

break;

default: cout<<"\n\t\t\t操作错误！"<<endl;

break;

} }while(1);

void Interface::ManagerFace() {

system("cls"); Manager manager;//管理员对象

cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*管理员系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl; cout<<"\t\t\t"<<"------------------------"<<endl <<"\t\t\t"<<"|

1、登录 |"<<endl <<"\t\t\t"<<"|

2、返回 |"<<endl <<"\t\t\t"<<"------------------------"<<endl;

do{ char choice; cout<<"\n\t\t\t"<<"请选择：";

cin>>choice;

switch(choice) { case '1':

manager.Run();

break;

case '2': system("cls");

Show();

} break;

default: cout<<"\n\t\t\t操作错误!"<<endl;

break;

} }while(1);

void Interface::ManagerLoginFace() {

}

void Interface::CustomerFace() {

system("cls"); Customer customer;//顾客对象 cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*订退票系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl; system("cls"); cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*管理员系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl; cout<<"\t\t\t"<<"--------------------------"<<endl <<"\t\t\t"<<"| 1、设置信息 |"<<endl <<"\t\t\t"<<"| 2、删除信息 |"<<endl <<"\t\t\t"<<"| 3、显示信息

|"<<endl <<"\t\t\t"<<"| 4、查询信息 |"<<endl <<"\t\t\t"<<"| 5、注销账户 |"<<endl <<"\t\t\t"<<"| 6、关闭售票机 |"<<endl <<"\t\t\t"<<"--------------------------"<<endl <<"\t\t\t选择操作："; cout<<"\t\t\t"<<" -------------------------"<<endl <<"\t\t\t"<<"| 1、订票 |"<<endl <<"\t\t\t"<<"| 2、退票 |"<<endl <<"\t\t\t"<<"| 3、返回 |"<<endl <<"\t\t\t"<<" ------------------------"<<endl; do{ char choice;

cout<<"\n\t\t\t"<<"请选择："; cin>>choice; switch(choice) { case '1': system("cls"); cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*订退票系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

GetLocalTime(&t);//获取时间的函数 cout<<"\t\t\t北京时间："<<t.wYear<<"年"<<t.wMonth<<"月"<<t.wDay<<"日"<<" "<<t.wHour<<":"<<t.wMinute

<<":"<<t.wSecond<<endl<<endl

<<"\t\t\t"<<"--------------选择场次：------------"<<endl

<<"\t\t\t"<<"| 场次1：00:00-02:00 |"<<endl

<<"\t\t\t"<<"| 场次2：08:00-10:00 |"<<endl <<"\t\t\t"<<"| 场次3：10:00-12:00 |"<<endl

<<"\t\t\t"<<"| 场次4：12:00-14:00 |"<<endl

<<"\t\t\t"<<"| 场次5：14:00-16:00 |"<<endl

<<"\t\t\t"<<"| 场次6：16:00-18:00 |"<<endl

<<"\t\t\t"<<"| 场次7：18:00-20:00 |"<<endl

<<"\t\t\t"<<"| 场次8：20:00-22:00 |"<<endl

<<"\t\t\t"<<"| 场次9：22:00-24:

00 |"<<endl <<"\t\t\t"<<"------------------------------------"<<endl<<endl<<endl; customer.Book(); break; case '2': system("cls"); cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*订退票系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

customer.Refund(); break; case '3': system("cls"); Show(); break;

}

default: cout<<"\n\t\t\t操作错误!"<<endl; break; } }while(1);

售票机类：

TicketSeller::TicketSeller()//初始化flag标致数组 {}

void TicketSeller::setMovieName

(string m) {}

void TicketSeller::setYear(int y)

{

}

void TicketSeller::setMonth(int m)

{

}

void TicketSeller::setDay(int d)

{

}

void TicketSeller::setPlayTime(int p)

{

} playTime=p; day=d; month=m; year=y; movieName=m; memset(flag,0,sizeof(flag));//将flag标致数组全部置0

void TicketSeller::setPrice(int p) {

}

void TicketSeller::setFlag(int i,int j,int f) {

}

string TicketSeller::getMovieName() {

}

int TicketSeller::getYear(){}

int TicketSeller::getMonth(){}

int TicketSeller::getDay(){

return month; return year; returnmovieName; flag[i][j]=f; int i,j;

/\*普通座票价设置\*/for(i=0;i<10;i++) for(j=0;j<10;j++) price[i][j]=p;

/前排座(2-5行4-7列)票价设置\*/ for(i=1;i<5;i++)

for(j=3;j<7;j++)

price[i][j]=p+10;

/\*情侣座(9-10行3-8列)票价设置\*/ for(i=8;i<10;i++)

for(j=2;j<8;j++)

price[i][j]=p+20;

}

return day;

int TicketSeller::getPlayTime(){}

int TicketSeller::getPrice(int i,int j){}

int TicketSeller::getFlag(int i,int j){}

return flag[i][j]; return price[i][j];

return playTime;

售票机管理操作类：

#include"Manager.h"

#include"Interface.h"

SYSTEMTIME mt;//SYSTEMTIME为windows头文件里的类，用来获取系统时间

string Manager::ManagerName="Chung";

string Manager::Password="19940711";

Manager::Manager(){}

void Manager::SetInformation(){

system("cls");

ioFile.open("movieInfo.txt",ios::out|ios::app);//写入和添加方式打开文件 cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*管理员系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl;

cout<<"\n\t\t\t\t设置影片信息\n"<<endl; while(1) {

TicketSeller Movie;

int date[3],pt,p;

} }

string mn; char ch;

cout<<"\n请输入日期(格式如2013-12-11)：";

cin>>date[0]>>ch>>date[1]>>ch>>date[2];

Movie.setYear(date[0]);

Movie.setMonth(date[1]);

Movie.setDay(date[2]);

cout<<"\n请输入放映场次(1到9)：";

cin>>pt;

Movie.setPlayTime(pt);

cout<<"\n请输入电影名称：";

cin>>mn;

Movie.setMovieName(mn); cout<<"\n请输入电影票价：";

cin>>p;

Movie.setPrice(p);

M.push\_back(Movie);//存入容器

ioFile.write((char \*)&Movie,sizeof(TicketSeller));//写入文件

cout<<"\n是否继续添加（y/n）：";

cin>>ch; if(ch=='y') continue;

if(ch=='n') { } cout<<"\n\t\t\t操作成功，正在跳转...";

break;

ioFile.close();

void Manager::DelInformation() {system("cls");

string mn;

} int find=0; cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*管理员系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl; cout<<"\n\t\t\t\t删除影片信息\n"<<endl; cout<<"\n请输入需要删除的影片名："; cin>>mn; for(posM=M.begin();posM!=M.end();posM++)//用迭代器遍历容器 { } if(find)//若找到 else cout<<"\n\t\t\t未找到结果，删除失败！\n"; cout<<"\n\t\t\t删除成功！\n"; /\*更新容器数据\*/ if((\*posM).getMovieName()==mn) { } posM=M.erase(posM); find=1;//找到p场次，find赋值为1 break; \_sleep(1\*1000);

void Manager::Display()

{

cout<<(\*posM).getYear()<<"-"<<(\*posM).getMonth()<<"-"<<(\*posM).getDay()<<"\t"<<(\*posM).getPlayTime()<<"\t"<<(\*posM).getMovieName()<<"\

t\t"; system("cls"); cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*管理员系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl; cout<<"\n\t\t\t\t显示影片信息\n"<<endl; cout<<"\n\t\t\t所有电影信息如下："<<endl; cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl; cout<<"日-----期\t场-次\t影---片---名\t普通座票价(元)前排票价(元)情侣座票价(元)"<<endl<<endl; for(posM=M.begin();posM!=M.end();posM++)//用迭代器遍历容器 {

} } cout<<(\*posM).getPrice(0,0)<<"\t\t"<<(\*posM).getPrice(3,3)<<"\t\t"<<(\*posM).getPrice(9,7); cout<<endl; cout<<"\n\t\t\t5s后自动返回管理员系统..."; \_sleep(5\*1000); system("cls");

void Manager::Inquire() {

system("cls"); cout<<"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*管理员系统\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"<<endl; cout<<"\n\t\t\t\t查询售票信息\n"<<endl; int count,added=0; for(int pt=1;pt<10;pt++) { count=0; for(posM=M.begin();posM!=M.end();posM++)//用迭代器遍历容器 { } added=added+count;//当天售票总张数added if((\*posM).getPlayTime()==pt) { } for(int i=0;i<10;i++) { } cout<<"\n\t\t\t场次"<<pt<<"--------已售"<<count<<"张票"<<endl; for(int j=0;j<10;j++) { } if((\*posM).getFlag(i,j)!=0) { } count++;

} GetLocalTime(&mt);//获取时间的函数 cout<<"\n\t\t\t"<<mt.wYear<<"/"<<mt.wMonth<<"/"<<mt.wDay<<" 的售票总数为:"<<added<<"张"<<endl;

}

string Manager::GetManagerName()

{

}

string Manager::GetPassword()

{

}

void Manager::Run()

{

char choose1,choose2; string mn,p; ioFile.open("movieInfo.txt",ios::in);//读方式打开文件 M.clear();//清空容器 TicketSeller is; ioFile.read((char \*)&is,sizeof(TicketSeller));//读入数据 while(!ioFile.eof())//当文件不为空 { } ioFile.close();//关闭文件 do{ cout<<"\n\t\t\t请输入账号："; cin>>mn; cout<<"\n\t\t\t请输入密码："; M.push\_back(is);//存入容器 ioFile.read((char \*)&is,sizeof(TicketSeller));//读入数据 return Password; return ManagerName; cout<<"\n\t\t\t5s后自动返回管理员系统..."; \_sleep(5\*1000); system("cls");

cin>>p;

if((mn!=GetManagerName())||(p!=GetPassword()))//若账号或密码与默认不同 { } else

break;

cout<<"\n\t\t\t账号或密码错误！\n"<<endl; cout<<"\n\t\t\t继续请按G，退出请按B：";

cin>>choose1; switch(choose1) { case 'B':

system("cls"); Interface::Show(); break;

case 'G':

; break;

default: }

cout<<"\n\t\t\t操作错误!"<<endl; break;

}while(1);

if((mn==GetManagerName())&&(p==GetPassword())) {

cout<<"\n\t\t\t正在登录管理员系统，请稍后..."; \_sleep(2\*1000); do{

Interface::ManagerLoginFace();//登陆后界面 cin>>choose2;

switch(choose2) {

case '1':

SetInformation();

break;

case '2':

DelInformation();

ioFile.open("movieInfo.txt",ios::out|ios::trunc);//写入和清空方式打开文件

for(posM=M.begin();posM!=M.end();posM++)//用迭代器遍历容器 {

/\*更新文件数据\*/

TicketSeller s;

s.setYear((\*posM).getYear()); s.setMonth((\*posM).getMonth()); s.setDay((\*posM).getDay());

s.setPlayTime((\*posM).getPlayTime()); s.setMovieName((\*posM).getMovieName()); s.setPrice((\*posM).getPrice(0,0));

}

ioFile.close();//关闭文件

for(int i=0;i<10;i++) { }

ioFile.write((char \*)&s,size of

(TicketSeller));

for(int j=0;j<10;j++) { }

if((\*posM).getFlag(i,j))

s.setFlag(i,j,1);

break;

case '3':

Display();

break;

case '4':

Inquire();

break; case '5': system("cls");

}

Interface::Show();

break;

case '6':

exit(1);

default:

cout<<"\n\t\t\t操作错误！"<<endl;

break;

} }

}while(1);

售票机顾客操作类：

Customer::Customer(){} void Customer::Book() {

int p,i,j,find=0; string chioce;

C.clear();//使用容器前先清空

ioFile.open("movieInfo.txt",ios::in);//读方式打开文件 TicketSeller s;

ioFile.read((char \*)&s,sizeof(TicketSeller));//读入数据 while(!ioFile.eof())//当文件不为空 { }

ioFile.close();//关闭文件

cout<<"日-----期\t场-次\t影---片---名\t普通座票价(元)前排票价(元)情侣座票价(元)"<<endl<<endl; for(posC=C.begin();posC!=C.end();posC++)//用迭代器遍历容器里的数据 {

C.push\_back(s);//存入容器

ioFile.read((char \*)&s,sizeof(TicketSeller));//读入数据

cout<<(\*posC).getYear()<<"-"<<(\*posC).getMonth()<<"-"<<(\*posC).getDay()<<"\t"<<(\*posC).getPlay

Time()<<"\t"<<(\*posC).getMovieName()<<"\t\t";

} do{

cout<<"\n\t\t\t请选择场次："; cin>>p;

for(posC=C.begin();posC!=C.end();posC++) {

/\*在屏幕上打印p场次的座位情况\*/ if(p==(\*posC).getPlayTime()) {

find=1;//找到p场次，find赋值为1cout<<"\n\t\t\t场次 "<<p<<" 剩余座位情况："<<endl;

cout<<(\*posC).getPrice(0,0)<<"\t\t"<<(\*posC).getPrice(3,3)<<"\t\t"<<(\*posC).getPrice(7,7); cout<<endl;

cout<<"\n\*----------------------------------大屏------------------------------------\*"<<endl; for(i=0;i<10;i++) {

for(j=0;j<10;j++) {

if((\*posC).getFlag(i,j))//若标致为1

cout<<"【"<<"有人"<<"】";

else

cout<<"【"<<"----"<<"】";

}

cout<<endl;

}

cout<<"\n\t\t\t选择座位行数：";cin>>i;

i=i-1;

cout<<"\n\t\t\t选择座位列数："; cin>>j;

j=j-1;

if(!(\*posC).getFlag(i,j))

{ }

cout<<"\n\t\t\t该座票价为"<<(\*posC).getPrice(i,j)<<"元"<<endl; (\*posC).setFlag(i,j,1);

else }

find=0;//重新给find赋值0 do{

cout<<"\n\t\t\t继续请按G，退出请按B："; cin>>chioce; if(chioce=="B") {

ioFile.open("movieInfo.txt",ios::out|ios::trunc);//写入和清空方式打开文件 /\*更新文件数据\*/

for(posC=C.begin();posC!=C.end

();posC++) {

TicketSeller s;

s.setYear((\*posC).getYear()); s.setMonth((\*posC).getMonth()); s.setDay((\*posC).getDay());

s.setPlayTime((\*posC).getPlayTime()); s.setMovieName((\*posC).getMovieName()); s.setPrice((\*posC).getPrice(0,0)); for(i=0;i<10;i++) {

for(j=0;j<10;j++) {

if((\*posC).getFlag(i,j))

cout<<"\n\t\t\t该座位已被预定,请重新选择！"<<endl; }

if((posC==C.end()-1)&&(!find))//若遍历容器没有找到该场次信息 { }

cout<<"\n\t\t\t无此场次！"<<endl;

}}

}

}

}

s.setFlag(i,j,1);

ioFile.write((char \*)&s,sizeof(TicketSeller));//写入文件

ioFile.close();//更新数据后关闭文件 system("cls"); Interface::Show();

else if(chioce=="G") { } else

cout<<"\n\t\t\t操作错误！"<<endl; break;

}while(1);

}while(1);

void Customer::Refund() {

int p,i,j,find=0; string chioce;

C.clear();//使用容器前先清空

ioFile.open("movieInfo.txt",ios::in);//读方式打开文件 TicketSeller s;

ioFile.read((char \*)&s,sizeof(TicketSeller));//读入数据 while(!ioFile.eof())//当文件不为空 { }

ioFile.close();//关闭文件 do{

cout<<"\n\t\t\t输入退票场次："; C.push\_back(s);//存入容器

ioFile.read((char \*)&s,sizeof(TicketSeller));//读入数据

cin>>p;

cout<<"\n\t\t\t退票座位行数：";

cin>>i;

i=i-1;

cout<<"\n\t\t\t退票座位列数：";

cin>>j;

j=j-1;

for(posC=C.begin();posC!=C.end

();posC++) { }

if(!find)//若没找到 { }

find=0;//重新给find赋值0 do{

cout<<"\n\t\t\t继续请按G，退出请按B："; cin>>chioce; if(chioce=="B") {

ioFile.open("movieInfo.txt",ios::out|ios::trunc);//写入和清空方式打开文件 /\*更新文件数据\*/

for(posC=C.begin();posC!=C.end();posC++) {

TicketSeller s;

s.setYear((\*posC).getYear()); s.setMonth((\*posC).getMonth()); s.setDay((\*posC).getDay());

cout<<"\n\t\t\t退票信息错误！"<<endl; if((\*posC).getFlag(i,j))//若标致为1 { }

(\*posC).setFlag(i,j,0);//标致改为0 find=1;//找到p场次，find赋值为1cout<<"\n\t\t\t退款（原价80%）"<<0.8\*(\*posC).getPrice(i,j)<<"元"<<endl;//退票

}

}

}

s.setPlayTime((\*posC).getPlayTime());

s.setMovieName((\*posC).getMovieName());

s.setPrice((\*posC).getPrice(0,0)); for(i=0;i<10;i++) { }

ioFile.write((char \*)&s,sizeof(TicketSeller));//写入文件

for(j=0;j<10;j++) { }

if(!(\*posC).getFlag(i,j))

s.setFlag(i,j,0);

ioFile.close();//关闭文件 system("cls"); Interface::Show();

else if(chioce=="G") else

cout<<"\n\t\t\t操作错误！"<<endl; break;

}while(1);

}while(1);

主函数：

void main() { }

TicketSeller ticket\_sller; Interface::IntroduceFace(); Interface::Show();