中南大学2019级

高级程序设计实践（C++）

课

程

设

计

报

告

报告人：

报告时间：2020/6/24

目录

1. 本次课题的基本目的和要求

1.1课设基本要求

1.2课题要求

1. 软件整体规划及设计

2.1设计分析

2.2类的设计

2.3程序结构

1. 程序详细设计

3.1用户功能设计

3.2 管理员功能设计

1. 调试运行结果并显示
2. 软件特色与不足

5.1软件特色

5.2软件不足

1. 设计体会
2. 致谢
3. 参考文献

一、本次课题的基本目的和要求

**1.1课设基本要求**

1、数据管理：要求对数据进行保存

2、要求在程序中体现出函数重载、运算符重

载、类的继承与派生等语法，并尽量避免使用全局变量。

3、用户管理功能：所设计的系统要求能用于多个用户，每个用户管理自己的数据和登录密码。

4、实用要求：界面美观、可操作性强、稳定性好。

**1.2课题要求**

见附页1

二、软件整体规划与设计

2.1设计分析

本次课题为某市地铁售票管理及客流量预测系统，其使用对象为用户、和管理员，主要涉及到用户购票系列功能、地铁线路、地铁车次、管理人员操作这四大功能。为使程序直观清晰，保证程序功能的齐全，我分别设计了：users（用户类），lines（线路类），trains（车次类），managers（管理员类），分别设置其数据成员并添加对应功能的函数以完成各项功能。

其中，users应实现的功能有：用户登录、新用户注册、地铁购票、退票、查询已购车票、查询线路途经站点、查询线路人流量、查询本账户余额等信息、余额充值、提现、办理会员及销毁账户。

managers应实现的功能有：管理人员登录、添加车次、删除车次、显示正在运行的车次、查看地铁统计数据等。

lines与trains的功能会通过用户与管理员的操作实现。

2.2类的设计

见附页2

2.3程序结构 见附页3

三、程序详细设计

3.1源代码（在DevC环境下调试通过）

见附页4

3.2用户功能设计

1、登录功能：用户凭借账号及密码登录

2、注册用户：建立新的用户档案

3、购票：

**附页1 课题具体要求**

课设题目：某市地铁售票管理与客流量预测系统

1. 基本要求：
2. 编写某市地铁售票管理系统，实现对地铁售票的管理，能够根据售票记录预测不同时间段的乘车人数，提示错峰出行。系统须支持多条地铁线路、多趟车次、多用户购票，记录地铁线路信息、车次信息，以及用户的购票信息。
3. 用户有不同的类型，比如：普通用户、月卡用户、年票用户，每种用户购票的折扣不同。用户的信息根据需要自己确定。定义地铁的线路，每条线路包含：代码、名称、经过的站点、承载的车次、运载人数等信息，其中每个运行时段乘客量的预测通过成员函数实现。车票信息包括以下信息：车次、始发地、目的地、乘车区间、车票价格、发车时间等。
4. 基本管理功能：
5. 购票管理：针对车票信息进行管理。包括：

购票：添加车票信息。

显示：根据购票时间显示某一时间段的购票信息。

退票：删除相应车票信息。

1. 用户管理：主要是针对月卡用户和年卡用户，包括：

买卡：用户添加，增加月卡用户和年卡用户；

续费：修改其中的余额；

退卡：删除用户；

显示：显示当前持卡用户信息；

查询：根据卡号查询相应的余额等信息

1. 地铁线路管理：

添加：为某条线路增加车次；

删除：为某条线路删除车次；

显示：显示某条线路处于运行状态的车次。

1. 统计功能：根据需要设计合理的统计功能，比如：每天乘客的数量、发车次数、运行收入等。
2. 退出功能：要求点击退出，可以退出系统。

**附页2 类的设计**

**class users //用户类**

{

public:

void log\_in(); //登录

void registering(); //注册用户

void home(); //首页

void account(); //账户管理

void in\_charge(); //充值

void out\_charge(); //提现

void up\_grade(); //办理会员

void destroy(); //销户

void search(lines &l); //地铁查询

void search\_query(); //查询账户信息

void search\_other(); //查询其他账户信息

void buy\_1(); //购票 线路选择

void buy\_2(string x); //购票 车次选择

void buy\_3(string x); //购票 确认支付

void buy\_4(string x,float y);//购票 支付成功

void buy\_4(string x,int y);//函数重载

void buy\_5(); //购票 支付失败

void buy\_6(string x); //购票 显示所有车次

void refund\_1(); //退票 选择车票

void refund\_2(string x,float y);//退票 确认退票

void refund\_3(); //退票 退票成功

void search\_tickt(); //查看已购车票

protected:

static string login\_card;//(静态数据成员)登录卡号

long card; //卡号

string key; //密码

string name; //持卡人

string id\_card; //身份证号

string type; //会员类型

float balance; //余额

string tickt\_name[20]; //车票 车次

int tickt\_hour[20]; //车票 发车时间

int tickt\_minute[20]; //车票 发车时间

float tickt\_type[20]; //车票 车票类型0.5/1

int tickt\_number[20]; //车票 车票数量

};

**class lines //线路类**

{

friend void users::search(lines &l);

public:

void statistics(); //管理人员查看统计数据

void line\_information\_1();//显示线路站点信息

void line\_information\_2();//显示当前人流量

protected:

string line\_name[4]; //线路编号

int passengers[4]; //乘客数量

int run\_number[4]; //发车次数

double income[4]; //收入

};

**class managers //管理员类**

{

public:

void log\_in(); //登录

void home(trains &t); //管理人员首页

protected:

long card;

string key;

};

**class trains:public lines //车次类**

{

friend void managers::home(trains &t);

public:

void add(); //添加车次

void deleting(); //删除车次

void search(); //显示正在运行的车次

protected:

string train\_name[20]; //车次

int set\_hour[20]; //发车时间(小时)

int set\_minute[20]; //发车时间(分钟)

int passengers[20]; //乘客量

double in\_come[20]; //收入

};

**附页3 程序结构**

附页4 源代码

#include<iostream>

#include<fstream>

#include<math.h>

#include<stdlib.h>

#include<stdio.h>

#include<Windows.h>

#include<time.h>

#include<sstream>

using namespace std;

class lines;class trains; //超前使用说明

class users //声明用户类

{

public:

void log\_in(); //登录

void registering(); //注册用户

void home(); //首页

void account(); //账户管理

void in\_charge(); //充值

void out\_charge(); //提现

void up\_grade(); //办理会员

void destroy(); //销户

void search(lines &l); //地铁查询

void search\_query(); //查询账户信息

void search\_other(); //查询其他账户信息

void buy\_1(); //购票 线路选择

void buy\_2(string x); //购票 车次选择

void buy\_3(string x); //购票 确认支付

void buy\_4(string x,float y);//购票 支付成功

void buy\_4(string x,int y);//函数重载

void buy\_5(); //购票 支付失败

void buy\_6(string x); //购票 显示所有车次

void refund\_1(); //退票 选择车票

void refund\_2(string x,float y);//退票 确认退票

void refund\_3(); //退票 退票成功

void search\_tickt(); //查看已购车票

protected:

static string login\_card;//(静态数据成员)登录卡号

long card; //卡号

string key; //密码

string name; //持卡人

string id\_card; //身份证号

string type; //会员类型

float balance; //余额

string tickt\_name[20]; //车票 车次

int tickt\_hour[20]; //车票 发车时间

int tickt\_minute[20]; //车票 发车时间

float tickt\_type[20]; //车票 车票类型0.5/1

int tickt\_number[20]; //车票 车票数量

};

string users::login\_card="0"; //静态数据成员类外初始化

class lines //声明线路类

{

friend void users::search(lines &l);

public:

void statistics(); //管理人员查看统计数据

void line\_information\_1();//显示线路站点信息

void line\_information\_2();//显示当前人流量

protected:

string line\_name[4]; //线路编号

int passengers[4]; //乘客数量

int run\_number[4]; //发车次数

double income[4]; //收入

};

class managers //声明管理人员类

{

public:

void log\_in(); //登录

void home(trains &t); //管理人员首页

protected:

long card;

string key;

};

class trains:public lines //声明车次类

{

friend void managers::home(trains &t);

public:

void add(); //添加车次

void deleting(); //删除车次

void search(); //显示正在运行的车次

protected:

string train\_name[20]; //车次

int set\_hour[20]; //发车时间(小时)

int set\_minute[20]; //发车时间(分钟)

int passengers[20]; //乘客量

double in\_come[20]; //收入

};

void users::log\_in() //用户登录

{

system("cls"); //清屏

string m;

//设置中间变量

loop\_start:

cout<<endl

<<" ☆ 长沙地铁通 ☆"<<endl<<endl

<<" 用户请登录："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 卡号："; cin>>login\_card; login\_card=login\_card+".txt"; //输入卡号并转换为文件名

cout<<" 密码："; cin>>m; //输入登录密码

cout<<endl;

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

ifstream user(login\_card.c\_str(),ios::in); //读文件,ifstream只打开现有文件

if(!user) //用户不存在的情况

{

cout<<" 用户不存在！"<<endl<<endl

<<" 按回车键以返回";

getchar(); getchar();

system("cls"); //清屏

goto loop\_start; //回到初始界面

}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

if(m!=key) //密码错误的情况

{

cout<<" 密码错误！"<<endl<<endl

<<" 按回车键以返回";

getchar(); getchar();

system("cls"); //清屏

goto loop\_start; //回到初始界面

}

cout<<" 登录成功！";

getchar();getchar();

system("cls"); //清屏

home();

}

string f(long x) //数据类型转化

{

stringstream ss;

ss<<x;

return ss.str();

}

void users::registering() //新用户注册

{

system("cls"); //清屏

balance=0;type="Normal"; //初始化部分数据

cout<<endl

<<" 新用户注册"<<endl<<endl<<endl

<<" 您好！欢迎使用长沙地铁通！请输入："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 持卡人: ";

cin>>name; //输入持卡人姓名

cout<<" 身份证号：" ;

cin>>id\_card; //输入身份证号

cout<<" 请输入密码：" ;

cin>>key;cout<<endl; //输入密码

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

srand((unsigned)time(NULL)); //初始化随机数种子

for(int i=0;;i++) //随机生成卡号

{

card=10\*rand();

if(card>=100000&&card<=999999) break;

}

cout<<endl

<<" 注册成功！"<<endl<<endl

<<" 您的卡号为："<<card<<endl

<<" ";

login\_card=f(card)+".txt"; //利用辅助函数将card转化为string型

fstream user(login\_card.c\_str(),ios::out); //创建文件(若文件已存在则替换)

if(!user){cout<<"open erro!"<<endl;exit(1);}

user<<card<<endl<<key<<endl<<name<<endl<<id\_card<<endl<<type<<endl<<balance<<endl; //写入文件

for(int i=0;i<100;i++) user<<"0"<<endl;

user<<"END";

user.close();

getchar();getchar();

home();

}

void users::home() //用户首页

{

system("cls"); //清屏

char m; //设置中间变量

fstream user(login\_card.c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

cout<<endl

<<" 首页"<<endl<<endl<<endl

<<" "<<name<<" 您好！请选择服务项目：" <<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" A.地铁购票"<<endl

<<" B.退票办理"<<endl

<<" C.地铁查询"<<endl

<<" D.账户管理"<<endl

<<" E.退出系统"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" ";cin>>m;

lines l;

switch(m)

{

case'A':buy\_1();break;

case'B':refund\_1();break;

case'C':search(l);break;

case'D':account();break;

case'E':exit(1);break;

}

}

void users::buy\_1() //地铁购票>>线路选择

{

system("cls"); //清屏

string m; //设置中间变量

cout<<endl;

cout<<" 地铁购票>>线路选择"<<endl<<endl<<endl

<<" 线路信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 线路 始发地 目的地 半程票价 全程票价 运行时间" <<endl<<endl

<<" line1 尚双塘 开福区政府 3 6 06:30-23:00"<<endl

<<" line2 光达 梅溪湖西 3 6 06:30-23:00"<<endl

<<" line4 罐子岭 杜家坪 3 6 06:30-23:00"<<endl

<<" maglev 高铁站 机场站 10 20 07:00-22:30"<<endl<<endl

<<" ☆路站点详细信息请使用 地铁查询 "<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 请选择您的线路："; cin>>m; getchar();

system("cls"); //清屏

buy\_2(m); //进行下一步

}

void users::buy\_2(string x) //地铁购票>>线路选择>>车次选择

{

string n; int now,time[20]; //设置中间变量

string train\_name[20]; int set\_hour[20]; int set\_minute[20]; int passengers[20]; double in\_come[20]; //设置中间变量

fstream train((x+".txt").c\_str()); //根据m打开对应line的文件

if(!train){cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>passengers[i]>>in\_come[i];

train.seekg(0, ios::beg); //读文件

SYSTEMTIME sysTime;

ZeroMemory(&sysTime, sizeof(sysTime));

GetLocalTime(&sysTime); //获取系统时间

now=sysTime.wHour\*60+sysTime.wMinute;

for(int i=0;i<20;i++)

{

time[i]=set\_hour[i]\*60+set\_minute[i];

if(time[i]==0) time[i]=9999; //保证0数据不被输出

}

cout<<endl

<<" 地铁购票>>线路选择>>车次选择"<<endl<<endl<<endl

<<" 最近车次："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 车次 发车时间 现有乘客数"<<endl<<endl;

for(int i=0;i<20;i++) //输出后两小时内的车次

if((time[i]-now)>=0&&(time[i]-now)<=120) cout<<" "<<train\_name[i]<<" "<<set\_hour[i]<<"："<<set\_minute[i]<<" "<<passengers[i]<<endl;

cout<<endl;

cout<<" A显示所有有效车次"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 请选择车次：";cin>>n;cout<<endl<<endl;

system("cls"); //清屏

if(n!="A"&&n!="B") buy\_3(n); //进入下一步

if(n=="A") buy\_6(x); //显示所有车次

}

void users::buy\_6(string x) //地铁购票>>线路选择>>车次选择>>显示所有车次

{

string n; int now,time[20]; //设置中间变量

string train\_name[20]; int set\_hour[20]; int set\_minute[20]; int passengers[20]; double in\_come[20]; //设置中间变量

fstream train((x+".txt").c\_str()); //根据x打开对应line的文件

if(!train){cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>passengers[i]>>in\_come[i];

train.seekg(0, ios::beg); //读文件

SYSTEMTIME sysTime;

ZeroMemory(&sysTime, sizeof(sysTime));

GetLocalTime(&sysTime); //获取系统时间

now=sysTime.wHour\*60+sysTime.wMinute;

for(int i=0;i<20;i++)

{

time[i]=set\_hour[i]\*60+set\_minute[i];

if(time[i]==0) time[i]=9999; //保证0数据不被输出

}

cout<<endl

<<" 地铁购票>>线路选择>>车次选择"<<endl<<endl<<endl

<<" 最近车次："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 车次 发车时间 现有乘客数"<<endl<<endl;

for(int i=0;i<20;i++) //输出所有有效车次

if((time[i]-now)>=0&&(time[i]-now)<=1440) cout<<" "<<train\_name[i]<<" "<<set\_hour[i]<<"："<<set\_minute[i]<<" "<<passengers[i]<<endl;

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

<<" 请选择车次：";cin>>n;cout<<endl<<endl;

system("cls"); //清屏

buy\_3(n); //进入下一步

}

void users::buy\_3(string x) //地铁购票>>线路选择>>车次选择>>确认支付

{

string u=x; float price; string m; //设置中间变量

string e1; int e2,e3,e5; float e4,e6; //设置中间变量

string train\_name[20]; int set\_hour[20]; int set\_minute[20]; int passengers[20]; double in\_come[20]; //设置中间变量

const char \*p=x.c\_str();

switch(\*p)

{

case'A':u="line1";price=6;break;

case'B':u="line2";price=6;break;

case'C':u="line4";price=6;break;

case'D':u="maglev";price=20;break;

}

fstream train((u+".txt").c\_str()); //打开对应车次文件

if(!train) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>passengers[i]>>in\_come[i];

train.seekg(0, ios::beg);

fstream user(login\_card.c\_str(),ios::in|ios::out); //打开对应用户文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

if(type=="Year") e6=0.7; //根据会员类型确定折扣

if(type=="Month") e6=0.8;

if(type=="Normal") e6=1;

int ii;

for(ii=0;ii<20;ii++) if(train\_name[ii]==x) break; //用ii定位对应车次

cout<<endl

<<" 地铁购票>>路线选择>>车次选择>>确认支付"<<endl<<endl<<endl

<<" 请核对您的购票信息：" <<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 车次 发车时间 半程票价 全程票价"<<endl

<<" "<<x<<" "<<set\_hour[ii]<<"："<<set\_minute[ii]<<" "<<price\*0.5<<" "<<price<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 请输入购票张数：";cin>>e5;cout<<endl;

cout<<" 请选择车票类型： A.半程 B.全程 "; cin>>m; if(m=="A") e4=0.5; if(m=="B") e4=1;

cout<<endl

<<" 确认支付"<<price\*e4\*e5\*e6<<"元？ A.确认 B.取消 "; cin>>m; cout<<endl<<endl;

if(m=="A"&&(balance>=price\*e4\*e5\*e6)) //支付成功后，扣除余额写入车票信息

{

passengers[ii]+=e5; //向车次信息中写入

in\_come[ii]+=price\*e4\*e5\*e6;

balance-=price\*e4\*e5\*e6;

for(int i=0;i<20;i++)

if(tickt\_name[i]=="0"||(tickt\_name[i]==x&&tickt\_type[i]==e4)) //向用户信息中写入

{

tickt\_name[i]=x;

tickt\_hour[i]=set\_hour[ii];

tickt\_minute[i]=set\_minute[ii];

tickt\_type[i]=e4;

tickt\_number[i]+=e5;

for(int i=0;i<20;i++) train<<train\_name[i]<<endl<<set\_hour[i]<<endl<<set\_minute[i]<<endl<<passengers[i]<<endl<<in\_come[i]<<endl;

train<<"END";

train.close(); //写文件

user<<card<<endl<<key<<endl<<name<<endl<<id\_card<<endl<<type<<endl<<balance<<endl;

for(int i=0;i<20;i++) user<<tickt\_name[i]<<endl<<tickt\_hour[i]<<endl<<tickt\_minute[i]<<endl<<tickt\_type[i]<<endl<<tickt\_number[i]<<endl;

user<<"END";

user.close(); //写文件

break;

}

}

system("cls"); //清屏

if(m=="A"&&(balance>=price\*e4\*e5\*e6)) buy\_4(x,e4); //进入支付成功页面

if(m=="A"&&(balance<price\*e4\*e5\*e6)) buy\_5(); //进入购票失败页面

}

void users::buy\_4(string x,float y)//地铁购票>>线路选择>>车次选择>>确认支付>>支付成功

{

fstream user(login\_card.c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

int ii; //设置中间变量

for(ii=0;ii<20;ii++) if(tickt\_name[ii]==x&&tickt\_type[ii]==y) break;

cout<<endl

<<" 地铁购票>>路线选择>>车次选择>>确认支付>>购票成功"<<endl<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 购票成功!"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 车票信息："<<endl<<endl

<<" 车次 发车时间 车票类型 车票数量"<<endl

<<" "<<x<<" "<<tickt\_hour[ii]<<"："<<tickt\_minute[ii];

if(tickt\_type[ii]==0.5) cout<<" "<<"半程票";

if(tickt\_type[ii]==1) cout<<" "<<"全程票";

cout<<" "<<tickt\_number[ii]<<endl<<endl<<endl

<<" E.返回首页"<<endl<<endl

<<" ";

user.close();

getchar();getchar();

system("cls"); //清屏

home();

}

void users::buy\_4(string x,int y)//(函数重载)

{

fstream user(login\_card.c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

int ii; //设置中间变量

for(ii=0;ii<20;ii++) if(tickt\_name[ii]==x&&tickt\_type[ii]==y) break;

cout<<endl

<<" 地铁购票>>路线选择>>车次选择>>确认支付>>购票成功"<<endl<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 购票成功!"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 车票信息："<<endl<<endl

<<" 车次 发车时间 车票类型 车票数量"<<endl

<<" "<<x<<" "<<tickt\_hour[ii]<<"："<<tickt\_minute[ii];

if(tickt\_type[ii]==0.5) cout<<" "<<"半程票";

if(tickt\_type[ii]==1) cout<<" "<<"全程票";

cout<<" "<<tickt\_number[ii]<<endl<<endl<<endl

<<" E.返回首页"<<endl<<endl

<<" ";

user.close();

getchar();getchar();

system("cls"); //清屏

home();

}

void users::buy\_5() //地铁购票>>线路选择>>车次选择>>确认支付>>购票失败

{

char m;

cout<<endl

<<" 地铁购票>>路线选择>>车次选择>>确认支付>>购票失败"<<endl<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 购票失败！"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 您的余额已不足，请及时充值"<<endl<<endl

<<" F.前往充值"<<endl

<<" E.返回首页"<<endl

<<" ";cin>>m;

switch(m)

{

case'F':in\_charge();break;

case'E':home();break;

}

}

void users::refund\_1() //退票办理>>选择车票

{

system("cls"); //清屏

int time[20],now; string u,m;//设置中间变量

fstream user(login\_card.c\_str(),ios::in|ios::out); //打开对应用户文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

SYSTEMTIME sysTime;

ZeroMemory(&sysTime, sizeof(sysTime));

GetLocalTime(&sysTime); //获取系统时间

now=sysTime.wHour\*60+sysTime.wMinute;

for(int i=0;i<20;i++)

{

time[i]=tickt\_hour[i]\*60+tickt\_minute[i];

if(time[i]==0) time[i]=9999; //保证0数据不被输出

}

cout<<endl

<<" 退票办理>>选择车票"<<endl<<endl<<endl

<<" 请选择您要退订的车票："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 编号 车次 发车时间 车票类型 车票数量"<<endl<<endl;

int i; float n;

for(int i=0;i<20;i++)

if(time[i]>=now&&time[i]<=1440)

{

cout<<" "<<i<<" "<<tickt\_name[i]<<" "<<tickt\_hour[i]<<"："<<tickt\_minute[i];

if(tickt\_type[i]==0.5) cout<<" "<<"半程票";

if(tickt\_type[i]==1) cout<<" "<<"全程票";

cout<<" "<<tickt\_number[i]<<endl;

}

cout<<endl;

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

<<" 请输入车票编号：";cin>>i;cout<<endl<<endl;

m=tickt\_name[i]; n=tickt\_type[i];

user.close();

system("cls"); //清屏

refund\_2(m,n); //进入下一步

}

void users::refund\_2(string x,float y) //退票办理>>选择车票>>确认退票

{

string u=x,n; float m,price;

string train\_name[20]; int set\_hour[20]; int set\_minute[20]; int passengers[20]; double in\_come[20]; //设置中间变量

fstream user(login\_card.c\_str(),ios::in|ios::out); //打开对应用户文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

const char \*q=x.c\_str();

switch(\*q)

{

case'A':u="line1";price=6;break;

case'B':u="line2";price=6;break;

case'C':u="line4";price=6;break;

case'D':u="maglev";price=20;break;

}

fstream train((u+".txt").c\_str()); //打开对应车次文件

if(!train) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>passengers[i]>>in\_come[i];

train.seekg(0, ios::beg);

if(type=="Year") m=0.7; //根据会员类型确定折扣

if(type=="Month") m=0.8;

if(type=="Normal") m=1;

int ii;

for(ii=0;ii<20;ii++) if(tickt\_name[ii]==x&&tickt\_type[ii]==y) break;

cout<<endl

<<" 退票办理>>选择车票>>确认退票"<<endl<<endl<<endl

<<" 请核对您的退票信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 线路 车次 发车时间 车票类型 退票数量 应退金额"<<endl<<endl

<<" "<<u<<" "<<x<<" "<<tickt\_hour[ii]<<"："<<tickt\_minute[ii];

if(tickt\_type[ii]==0.5) cout<<" "<<"半程票";

if(tickt\_type[ii]==1) cout<<" "<<"全程票";

cout<<" "<<tickt\_number[ii]<<" "<<price\*tickt\_type[ii]\*tickt\_number[ii]\*m<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 确认退票？ A.确认 B.取消 ";cin>>n;cout<<endl<<endl;

if(n=="A")

{

for(int i=0;i<20;i++)

if(train\_name[i]==x)

{

passengers[i]-=tickt\_number[ii]; //向车次信息中写入

in\_come[i]-=price\*tickt\_type[ii]\*tickt\_number[ii]\*m;

break;

}

balance+=price\*tickt\_type[ii]\*tickt\_number[ii]\*m;

tickt\_name[ii]="0";

tickt\_hour[ii]=0;

tickt\_minute[ii]=0;

tickt\_type[ii]=0;

tickt\_number[ii]=0;

for(int i=0;i<20;i++) train<<train\_name[i]<<endl<<set\_hour[i]<<endl<<set\_minute[i]<<endl<<passengers[i]<<endl<<in\_come[i]<<endl;

train<<"END";

train.close(); //写文件

int time[20]; string e1; int e2,e3,e5,e6; float e4; //设置中间变量

for(int i=0;i<20;i++)

{

time[i]=tickt\_hour[i]\*60+tickt\_minute[i];

if(time[i]==0) time[i]=1440;

}

for(int i=0;i<19;++i)

for(int j=0;j<19-i;++j)

if(time[j]>time[j+1]) //写入前排序，按时间顺序排序

{

e1=tickt\_name[j];tickt\_name[j]=tickt\_name[j+1];tickt\_name[j+1]=e1;

e2=tickt\_hour[j];tickt\_hour[j]=tickt\_hour[j+1];tickt\_hour[j+1]=e2;

e3=tickt\_minute[j];tickt\_minute[j]=tickt\_minute[j+1];tickt\_minute[j+1]=e3;

e4=tickt\_type[j];tickt\_type[j]=tickt\_type[j+1];tickt\_type[j+1]=e4;

e5=tickt\_number[j];tickt\_number[j]=tickt\_number[j+1];tickt\_number[j+1]=e5;

e6=time[j];time[j]=time[j+1];time[j+1]=e6;

}

user<<card<<endl<<key<<endl<<name<<endl<<id\_card<<endl<<type<<endl<<balance<<endl;

for(int i=0;i<20;i++) user<<tickt\_name[i]<<endl<<tickt\_hour[i]<<endl<<tickt\_minute[i]<<endl<<tickt\_type[i]<<endl<<tickt\_number[i]<<endl;

user<<"END";

user.close(); //写文件

system("cls"); //清屏

refund\_3(); //进入下一步

}

}

void users::refund\_3() //退票办理>>选择车票>>退票成功

{

cout<<endl

<<" 退票办理>>选择车票>>确认退票>>退票成功"<<endl<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 退票成功！" <<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 应退金额已返回您的账户"<<endl<<endl

<<" E.返回首页"<<endl

<<" ";

getchar();getchar();

system("cls"); //清屏

home();

}

void users::search(lines &l) //地铁查询

{

system("cls"); //清屏

char m;

cout<<endl

<<" 地铁查询>>选择服务"<<endl<<endl<<endl

<<" 请选择查询内容："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" A.查询已购车票"<<endl

<<" B.查询线路途经站点"<<endl

<<" C.查询线路人流量"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" ";cin>>m;

switch(m)

{

case'A':search\_tickt();break;

case'B':l.line\_information\_1();home();break;

case'C':l.line\_information\_2();home();break;

}

}

void users::search\_tickt() //地铁查询>>查询已购车票

{

system("cls"); //清屏

string u[20]; float m,price[20]; const char \*t[20]; int now,time[20];//设置中间变量

fstream user(login\_card.c\_str(),ios::in|ios::out); //打开对应用户文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

SYSTEMTIME sysTime;

ZeroMemory(&sysTime, sizeof(sysTime));

GetLocalTime(&sysTime); //获取系统时间

now=sysTime.wHour\*60+sysTime.wMinute;

for(int i=0;i<20;i++)

{

time[i]=tickt\_hour[i]\*60+tickt\_minute[i];

if(time[i]==0) time[i]=9999; //保证0数据不被输出

t[i]=tickt\_name[i].c\_str();

switch(\*t[i])

{

case'A':u[i]="line1";price[i]=6;break;

case'B':u[i]="line2";price[i]=6;break;

case'C':u[i]="line4";price[i]=6;break;

case'D':u[i]="maglev";price[i]=20;break;

}

}

if(type=="Year") m=0.7; //根据会员类型确定折扣

if(type=="Month") m=0.8;

if(type=="Normal") m=1;

cout<<endl

<<" 地铁查询>>选择服务>>已购车票"<<endl<<endl<<endl

<<" 我的车票信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 线路 车次 发车时间 车票种类 车票数量 支付金额 是否有效"<<endl<<endl;

for(int i=0;i<20;i++)

if(tickt\_name[i]!="0")

{

cout<<" "<<u[i]<<" "<<tickt\_name[i]<<" "<<tickt\_hour[i]<<"："<<tickt\_minute[i];

if(tickt\_type[i]==0.5) cout<<" "<<"半程票";

if(tickt\_type[i]==1) cout<<" "<<"全程票";

cout<<" "<<tickt\_number[i]<<" "<<price[i]\*tickt\_type[i]\*tickt\_number[i]\*m;

if(now>time[i]) cout<<" "<<"已过期"<<endl;

if(now<=time[i]) cout<<" "<<" 有效"<<endl;

}

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

<<" E.返回首页"<<endl

<<" ";

user.close();

getchar();getchar();

home();

}

void lines::line\_information\_1()//地铁查询>>查询线路站点

{

system("cls"); //清屏

cout<<endl;

cout<<" 地铁查询>>选择服务>>线路站点"<<endl<<endl<<endl

<<" 各线路途经站点："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 线路 站点1 站点2 站点3 站点4 站点5"<<endl<<endl

<<" line1 开福区政府 培元桥 五一广场 黄土岭 尚双塘"<<endl

<<" line2 梅西湖西 橘子洲 五一广场 火车南站 光达"<<endl

<<" line4 罐子岭 溁湾镇 湖南大学 阜埠河 杜家坪"<<endl

<<" maglev 磁浮高铁站 磁浮槊梨站 磁浮机场站"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" ☆更多资讯请访问https://baike.baidu.com/item/长沙轨道交通"<<endl<<endl

<<" E.返回首页"<<endl

<<" ";

getchar();getchar();

}

void lines::line\_information\_2()//地铁查询>>查询当前人流量

{

system("cls"); //清屏

int now,n=0; //设置中间变量

string train\_name[20]; int set\_hour[20],set\_minute[20],train\_passengers[20]; double in\_come[20];

SYSTEMTIME sysTime; //获取系统时间

ZeroMemory(&sysTime, sizeof(sysTime));

GetLocalTime(&sysTime);

now=sysTime.wHour\*60+sysTime.wMinute;

fstream train1("line1.txt");

if(!train1) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train1>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>train\_passengers[i]>>in\_come[i];

train1.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++)

if(train\_name[i]!="0"&&((now-set\_hour[i]\*60-set\_minute[i])>=0&&(now-set\_hour[i]\*60-set\_minute[i])<=120))

{

n+=train\_passengers[i];

}

train1.close();

fstream train2("line2.txt");

if(!train1) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train2>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>train\_passengers[i]>>in\_come[i];

train2.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++)

if(train\_name[i]!="0"&&((now-set\_hour[i]\*60-set\_minute[i])>=0&&(now-set\_hour[i]\*60-set\_minute[i])<=120))

{

n+=train\_passengers[i];

}

train2.close();

fstream train3("line4.txt");

if(!train3) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train3>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>train\_passengers[i]>>in\_come[i];

train3.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++)

if(train\_name[i]!="0"&&((now-set\_hour[i]\*60-set\_minute[i])>=0&&(now-set\_hour[i]\*60-set\_minute[i])<=120))

{

n+=train\_passengers[i];

}

train3.close();

fstream train4("maglev.txt");

if(!train4) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train4>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>train\_passengers[i]>>in\_come[i];

train4.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++)

if(train\_name[i]!="0"&&((now-set\_hour[i]\*60-set\_minute[i])>=0&&(now-set\_hour[i]\*60-set\_minute[i])<=120))

{

n+=train\_passengers[i];

}

train4.close();

cout<<endl;

cout<<" 地铁查询>>选择服务>>查询线路人流量"<<endl<<endl<<endl

<<" 现在时间 "<<sysTime.wHour<<":"<<sysTime.wMinute<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 昨日此时段人流量："<<n<<endl

<<" 今日预计人流量："<<(n+141)<<endl

<<" 今日预计拥堵情况：";

if(n<=6600) cout<<"良好"<<endl;

else cout<<"拥堵"<<endl;

cout<<" 出行提示：";

if(n<=6600) cout<<"道路通畅，可以出行"<<endl;

else cout<<"地铁拥堵，请错峰出行"<<endl;

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

<<" E.返回首页"<<endl

<<" ";

getchar();getchar();

}

void users::account() //账户管理

{

system("cls"); //清屏

char m;

cout<<endl

<<" 账户管理>>选择服务"<<endl<<endl<<endl

<<" 请选择服务项目："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" A.账户信息"<<endl

<<" B.余额充值"<<endl

<<" C.余额提现"<<endl

<<" D.办理会员"<<endl

<<" F.销毁账户"<<endl

<<" E.返回首页"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" ";cin>>m;

switch(m)

{

case'A':search\_query(); break;

case'B':in\_charge(); break;

case'C':out\_charge(); break;

case'D':up\_grade(); break;

case'F':destroy(); break;

case'E':home(); break;

}

}

void users::search\_query() //账户管理>>查询账户信息

{

system("cls"); //清屏

string m; //设置中间变量

fstream user(login\_card.c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

cout<<endl

<<" 账户管理>>选择服务>>查询" <<endl<<endl<<endl

<<" 我的账户信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 持卡人："<<name<<endl

<<" 卡号："<<card<<endl

<<" 会员类型："<<type<<endl

<<" 余额："<<balance<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" F.查询其他卡号信息"<<endl

<<" E.返回账户管理 "<<endl

<<" ";cin>>m;

user.close();

if(m=="F") search\_other();

if(m=="E") account();

}

void users::search\_other() //账户管理>>查询其他账户信息

{

system("cls"); //清屏

long m; string n;

cout<<endl

<<" 账户管理>>选择服务>>查询>>查询其他卡号信息"<<endl<<endl<<endl

<<" 请输入待查："<<endl<<endl

<<" 卡号：";cin>>m;

cout<<" 密码：";cin>>n;cout<<endl<<endl;

fstream user((f(m)+".txt").c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

if(key==n)

{

cout<<" 账户信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 持卡人："<<name<<endl

<<" 卡号："<<card<<endl

<<" 会员类型："<<type<<endl

<<" 余额："<<balance<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" E.返回账户管理"<<endl

<<" ";

getchar();getchar();

account();

}

if(key!=n) {cout<<" 密码错误!";getchar();getchar();account();}

}

void users::in\_charge() //账户管理>>余额充值

{

fstream user(login\_card.c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

system("cls"); //清屏

float m=0;string n="0"; //设置中间变量

cout<<endl

<<" 账户管理>>选择服务>>充值"<<endl<<endl<<endl

<<" 请确认您的账户信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 持卡人："<<name<<endl

<<" 卡号："<<card<<endl

<<" 会员类型："<<type<<endl

<<" 余额："<<balance<<endl

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

cout<<" 请输入充值金额：";

cin>>m;cout<<endl;

cout<<" 请输入银行卡/微信账户：";

cin>>n;cout<<endl;

cout<<" 请输入支付密码：";

cin>>n;cout<<endl;

cout<<" 确认充值？ A确认 B取消 ";

cin>>n;

if(n=="A") balance+=m;

user<<card<<endl<<key<<endl<<name<<endl<<id\_card<<endl<<type<<endl<<balance<<endl; //写入文件

for(int i=0;i<20;i++) user<<tickt\_name[i]<<endl<<tickt\_hour[i]<<endl<<tickt\_minute[i]<<endl<<tickt\_type[i]<<endl<<tickt\_number[i]<<endl;

user<<"END";

user.close();

system("cls"); //清屏

cout<<endl

<<" 账户管理>>选择服务>>充值>>充值成功"<<endl<<endl<<endl

<<" 卡号："<<card<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 充值成功！"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 当前余额："<<balance<<endl

<<" E.返回账户管理"<<endl

<<" ";

getchar();getchar();

account();

}

void users::out\_charge() //账户管理>>余额提现

{

fstream user(login\_card.c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

system("cls"); //清屏

float m; string n="0"; //设置中间变量

cout<<endl

<<" 账户管理>>选择服务>>提现"<<endl<<endl<<endl

<<" 请确认您的账户信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 持卡人："<<name<<endl

<<" 卡号："<<card<<endl

<<" 会员类型："<<type<<endl

<<" 余额："<<balance<<endl

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

cout<<" 请输入提现金额：";

cin>>m;cout<<endl;

cout<<" 请输入银行卡/微信账户：";

cin>>n;cout<<endl;

cout<<" 确认提现？ A确认 B取消 ";

cin>>n;

if(n=="A") balance-=m;

user<<card<<endl<<key<<endl<<name<<endl<<id\_card<<endl<<type<<endl<<balance<<endl; //写入文件

for(int i=0;i<20;i++) user<<tickt\_name[i]<<endl<<tickt\_hour[i]<<endl<<tickt\_minute[i]<<endl<<tickt\_type[i]<<endl<<tickt\_number[i]<<endl;

user<<"END";

user.close();

system("cls"); //清屏

cout<<endl

<<" 账户管理>>选择服务>>提现>>提现成功"<<endl<<endl<<endl

<<" 卡号："<<card<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 提现成功！"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 当前余额："<<balance<<endl

<<" E.返回账户管理"<<endl

<<" ";

getchar();getchar();

account();

}

void users::up\_grade() //账户管理>>办理会员

{

system("cls"); //清屏

fstream user(login\_card.c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

string m,n; //设置中间变量

cout<<endl;

cout<<" 账户管理>>选择服务>>办理会员" <<endl<<endl<<endl

<<" 请选择会员类型："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" A.Month"<<endl

<<" 月卡会员:当月乘车享受0.8折优"<<endl

<<" 惠。无需会员费当月乘车额需达"<<endl

<<" 到20元不足20将按照20元收费"<<endl<<endl

<<" B.Year"<<endl

<<" 年卡会员:本年乘车享受0.7折优"<<endl

<<" 惠。无需会员费本年乘车额需达"<<endl

<<" 到240元不足240将按照240元收费"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 请选择会员类型： A.Month B.Year ";

cin>>m;

cout<<" 确认办理？ Y.确认 N.取消 ";

cin>>n;

if(m=="A"&&n=="Y") {type="Month";cout<<endl<<" 办理成功！";}

if(m=="B"&&n=="Y") {type="Year";cout<<endl<<" 办理成功！";}

cout<<endl

<<" E.返回账户管理"<<endl

<<" ";

user<<card<<endl<<key<<endl<<name<<endl<<id\_card<<endl<<type<<endl<<balance; //写

for(int i=0;i<20;i++) user<<tickt\_name[i]<<endl<<tickt\_hour[i]<<endl<<tickt\_minute[i]<<endl<<tickt\_type[i]<<endl<<tickt\_number[i]<<endl;

user<<"END";

user.close();

getchar();getchar();

account();

}

void users::destroy() //账户管理>>销毁账户

{

system("cls"); //清屏

fstream user(login\_card.c\_str(),ios::in|ios::out); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user>>card>>key>>name>>id\_card>>type>>balance;

for(int i=0;i<20;i++) user>>tickt\_name[i]>>tickt\_hour[i]>>tickt\_minute[i]>>tickt\_type[i]>>tickt\_number[i];

user.seekg(0, ios::beg);

string m;

cout<<endl;

cout<<" 账户管理>>选择服务>>销毁账户"<<endl<<endl<<endl

<<" 请确认账户信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 持卡人："<<name<<endl

<<" 卡号："<<card<<endl

<<" 会员类型："<<type<<endl

<<" 余额："<<balance<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 若您想销毁您的账户，请先阅读并同意销户协议："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 长沙地铁通用户销户协议"<<endl<<endl

<<" 尊敬的 "<<name<<" :"<<endl

<<" 感谢您长期以来对长沙地铁通的支"<<endl

<<" 持！若您因为某些原因需要销毁您的账"<<endl

<<" 户请先将余额全部提现。账户销毁后，"<<endl

<<" 系统将删除您的一切数据并且无法找回"<<endl

<<" 未退回的余额将会被清空。"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 是否继续？ A.继续,我已阅读并同意该协议"<<endl

<<" B.返回"<<endl

<<" ";

cin>>m;

if(m=="A")

{

system("cls"); //清屏

cout<<endl;

cout<<" 账户管理>>选择服务>>销毁账户>>确认销户"<<endl<<endl<<endl

<<" 请再次确认账户信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 持卡人："<<name<<endl

<<" 卡号："<<card<<endl

<<" 会员类型："<<type<<endl

<<" 余额："<<balance<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 警告！是否销户？ A.销户 B.返回"<<endl<<endl

<<" ";

cin>>m;

if(m=="A")

{

system("cls");

cout<<endl;

cout<<"账户管理>>选择服务>>销毁账户>>确认销户>>销户成功"<<endl<<endl<<endl

<<" 卡号："<<card<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" 销户成功！"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" E.退出程序"<<endl

<<" ";

user.close();

ofstream user(login\_card.c\_str(),ios::trunc); //读文件

if(!user){cout<<"open erro!"<<endl;exit(1);}

user.close();

getchar();getchar();

exit(1);

}

}

}

void managers::log\_in() //管理人员登录

{

system("cls"); //清屏

long m; string n;

loop\_start:

cout<<endl

<<" 管理人员登录" <<endl<<endl<<endl

<<" 管理人员请登录：" <<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 账号：";cin>>m;

cout<<" 密码：";cin>>n;cout<<endl;

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

ifstream manager(("Manager"+f(m)+".txt").c\_str(),ios::in); //读文件,ifstream只打开现有文件

if(!manager) //用户不存在的情况

{

cout<<" 账号不存在！"<<endl<<endl

<<" 按回车键以返回";

getchar(); getchar();

system("cls"); //清屏

goto loop\_start; //回到初始界面

}

manager>>card>>key;

manager.seekg(0, ios::beg);

if(n!=key) //密码错误的情况

{

cout<<" 密码错误！"<<endl<<endl

<<" 按回车键以返回";

getchar(); getchar();

system("cls"); //清屏

goto loop\_start; //回到初始界面

}

cout<<" 登录成功！";

getchar();getchar();

system("cls"); //清屏

trains t;

home(t);

}

void managers::home(trains &t) //管理人员首页

{

system("cls"); //清屏

char m;

cout<<endl

<<" 管理人员首页"<<endl<<endl<<endl

<<" 请选择需执行的操作："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" A.添加车次"<<endl

<<" B.删除车次"<<endl

<<" C.显示当前运行的车次"<<endl

<<" D.查看统计数据"<<endl

<<" E.退出系统"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" ";cin>>m;

switch(m)

{

case'A':t.add(); break;

case'B':t.deleting(); break;

case'C':t.search(); break;

case'D':t.statistics(); break;

case'E':exit(1); break;

}

}

void trains::add() //添加车次

{

system("cls"); //清屏

string xline,m1; int m2,m3; //设置中间变量

cout<<endl

<<" 添加车次"<<endl<<endl<<endl

<<" 请输入新车次信息："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" 线路：";cin>>xline;

cout<<" 车次：";cin>>m1;

cout<<" 发车时间(hour)：";cin>>m2;

cout<<" 发车时间(minute)：";cin>>m3;cout<<endl;

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

cout<<" 添加成功！"<<endl

<<" E.返回首页"<<endl

<<" ";

fstream train((xline+".txt").c\_str()); //根据xline打开对应线路的文件

if(!train){cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>passengers[i]>>in\_come[i];

train.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++) //将最前方的0车次覆盖

if(train\_name[i]=="0") { train\_name[i]=m1;set\_hour[i]=m2;set\_minute[i]=m3;break; }

int time[20]; string e1; int e2,e3,e4,e6; double e5; //设置中间变量

for(int i=0;i<20;i++)

{

time[i]=set\_hour[i]\*60+set\_minute[i];

if(time[i]==0) time[i]=1440;

}

for(int i=0;i<19;++i)

for(int j=0;j<19-i;++j)

if(time[j]>time[j+1]) //写入前排序，按时间顺序排序

{

e1=train\_name[j];train\_name[j]=train\_name[j+1];train\_name[j+1]=e1;

e2=set\_hour[j];set\_hour[j]=set\_hour[j+1];set\_hour[j+1]=e2;

e3=set\_minute[j];set\_minute[j]=set\_minute[j+1];set\_minute[j+1]=e3;

e4=passengers[j];passengers[j]=passengers[j+1];passengers[j+1]=e4;

e5=in\_come[j];in\_come[j]=in\_come[j+1];in\_come[j+1]=e5;

e6=time[j];time[j]=time[j+1];time[j+1]=e6;

}

for(int i=0;i<20;i++) train<<train\_name[i]<<endl<<set\_hour[i]<<endl<<set\_minute[i]<<endl<<passengers[i]<<endl<<in\_come[i]<<endl;

train<<"END";

train.close(); //写文件

getchar();getchar();

managers m;trains t;

m.home(t);

}

void trains::deleting() //删除车次

{

system("cls"); //清屏

string m,u; int n=0; //设置中间变量

cout<<endl

<<" 删除车次"<<endl<<endl<<endl

<<" 请先查询该车次："; cin>>m; cout<<endl<<endl;

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

<<" 查询结果："<<endl<<endl;

const char \*p=m.c\_str();

switch(\*p)

{

case'A':u="line1";break;

case'B':u="line2";break;

case'C':u="line4";break;

case'D':u="maglev";break;

default:cout<<" 无此车次！"<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" E.返回首页"<<endl

<<" ";

getchar();getchar();

managers mmm;trains tt;

mmm.home(tt);

break;

}

fstream train((u+".txt").c\_str());

if(!train) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>passengers[i]>>in\_come[i];

train.seekg(0, ios::beg); //读文件

int ii=0; //设置中间变量

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

if(m==train\_name[ii]) {n=1;/\*表示有此车次\*/ break;}

if(n==1)

{

cout<<" 线路 车次 发车时间 乘客量 收入"<<endl<<endl

<<" "<<u<<" "<<train\_name[ii]<<" "<<set\_hour[ii]<<"："<<set\_minute[ii]<<" "<<passengers[ii]<<" "<<in\_come[ii]<<endl;

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

<<" 是否删除该车次？ A.删除 B.返回 ";cin>>u;cout<<endl;

if(u=="A")

{

cout<<" 删除成功！"<<endl

<<" E.返回首页"<<endl

<<" ";

train\_name[ii]="0";set\_hour[ii]=0;set\_minute[ii]=0;passengers[ii]=0;in\_come[ii]=0;

}

}

if(n==0) cout<<" 无此车次！"<<endl<<endl<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" E.返回首页"<<endl

<<" ";

int time[20]; string e1; int e2,e3,e4,e6; double e5; //设置中间变量

for(int i=0;i<20;i++)

{

time[i]=set\_hour[i]\*60+set\_minute[i];

if(time[i]==0) time[i]=1440;

}

for(int i=0;i<19;++i)

for(int j=0;j<19-i;++j)

if(time[j]>time[j+1]) //写入前排序，按时间顺序排序

{

e1=train\_name[j];train\_name[j]=train\_name[j+1];train\_name[j+1]=e1;

e2=set\_hour[j];set\_hour[j]=set\_hour[j+1];set\_hour[j+1]=e2;

e3=set\_minute[j];set\_minute[j]=set\_minute[j+1];set\_minute[j+1]=e3;

e4=passengers[j];passengers[j]=passengers[j+1];passengers[j+1]=e4;

e5=in\_come[j];in\_come[j]=in\_come[j+1];in\_come[j+1]=e5;

e6=time[j];time[j]=time[j+1];time[j+1]=e6;

}

for(int i=0;i<19;i++) train<<train\_name[i]<<endl<<set\_hour[i]<<endl<<set\_minute[i]<<endl<<passengers[i]<<endl<<in\_come[i]<<endl;

train<<"END";

train.close(); //写文件

getchar();getchar();

managers mm;trains t;

mm.home(t);

}

void trains::search() //显示正在运行的车次

{

system("cls"); //清屏

string m; int now,time[20]; //设置中间变量

SYSTEMTIME sysTime;

ZeroMemory(&sysTime, sizeof(sysTime));

GetLocalTime(&sysTime); //获取系统时间

cout<<endl

<<" 正在运行的车次"<<endl<<endl<<endl

<<" 现在时间 "<<sysTime.wHour<<":"<<sysTime.wMinute<<endl<<endl

<<" 请输入要查询的线路："; cin>>m;cout<<endl;

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

<<" 查询结果："<<endl<<endl

<<" 线路 车次 发车时间 运载人数"<<endl<<endl;

fstream train((m+".txt").c\_str());

if(!train) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>passengers[i]>>in\_come[i];

train.seekg(0, ios::beg); //读文件

now=sysTime.wHour\*60+sysTime.wMinute;

for(int i=0;i<20;i++)

{

time[i]=set\_hour[i]\*60+set\_minute[i];

if(time[i]==0) time[i]=9999; //保证0数据不被输出

}

for(int i=0;i<20;i++) //将前一小时内发车的车次调出

if((now-time[i])<=60&&(now-time[i])>=0) cout<<" "<<m<<" "<<train\_name[i]<<" "<<set\_hour[i]<<"："<<set\_minute[i]<<" "<<passengers[i]<<endl;

cout<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl

<<" E.返回首页"<<endl

<<" ";

train.close();

getchar();getchar();

managers a;trains r;

a.home(r);

}

void lines::statistics() //查看统计数据

{

system("cls"); //清屏

string train\_name[20]; int set\_hour[20],set\_minute[20],train\_passengers[20]; double in\_come[20];

fstream line("lines.txt",ios::in|ios::out); //读文件

if(!line){cout<<"open erro!"<<endl;exit(1);}

line>>line\_name[0]>>passengers[0]>>run\_number[0]>>income[0]

>>line\_name[1]>>passengers[1]>>run\_number[1]>>income[1]

>>line\_name[2]>>passengers[2]>>run\_number[2]>>income[2]

>>line\_name[3]>>passengers[3]>>run\_number[3]>>income[3];

line.seekg(0, ios::beg);

fstream train1("line1.txt");

if(!train1) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train1>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>train\_passengers[i]>>in\_come[i];

train1.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++)

if(train\_name[i]!="0")

{

run\_number[0]+=1;

passengers[0]+=train\_passengers[i];

income[0]+=in\_come[i];

}

train1.close();

fstream train2("line2.txt");

if(!train2) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train2>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>train\_passengers[i]>>in\_come[i];

train2.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++)

if(train\_name[i]!="0")

{

run\_number[1]+=1;

passengers[1]+=train\_passengers[i];

income[1]+=in\_come[i];

}

train2.close();

fstream train3("line4.txt");

if(!train3) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train3>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>train\_passengers[i]>>in\_come[i];

train3.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++)

if(train\_name[i]!="0")

{

run\_number[2]+=1;

passengers[2]+=train\_passengers[i];

income[2]+=in\_come[i];

}

train3.close();

fstream train4("maglev.txt");

if(!train4) {cout<<"open erro!"<<endl;exit(1);}

for(int i=0;i<20;i++) train4>>train\_name[i]>>set\_hour[i]>>set\_minute[i]>>train\_passengers[i]>>in\_come[i];

train4.seekg(0, ios::beg); //读文件

for(int i=0;i<20;i++)

if(train\_name[i]!="0")

{

run\_number[3]+=1;

passengers[3]+=train\_passengers[i];

income[3]+=in\_come[i];

}

train4.close();

float aver\_1[4],aver\_2[4]; //设置中间变量

cout<<endl;

cout<<" 查看统计数据" <<endl<<endl<<endl;

cout<<" 查询结果："<<endl<<endl;

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

<<" 线路 发车次数 乘客量 运行收入 单次发车收入 单次发车乘客量" <<endl;

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

{

aver\_1[i]=(float)income[i]/(float)run\_number[i];

aver\_2[i]=(float)passengers[i]/(float)run\_number[i];

cout<<" "<<line\_name[i]<<" "<<run\_number[i]<<" "<<passengers[i]<<" "<<income[i]<<" "<<aver\_1[i]<<" "<<aver\_2[i]<<endl;

}

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

<<" E.返回首页"<<endl

<<" ";

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

{

line<<line\_name[i]<<endl<<passengers[i]<<endl<<run\_number[i]<<endl<<income[i]<<endl;

}

line.close();

getchar();getchar();

managers a;trains r;

a.home(r);

}

char log\_in() //初始登录界面

{

char m;

cout<<endl

<<" ☆ 长沙地铁通 ☆"<<endl<<endl

<<" 请选择登录方式："<<endl<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" A.用户登录"<<endl

<<" B.新用户注册"<<endl

<<" C.管理人员登录"<<endl

<<" E.退出系统"<<endl

<<" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl<<endl

<<" ";cin>>m;

return m;

}

int main() //主函数

{

trains train;

managers manager;

lines line;

users user;

char m;

m=log\_in();

switch(m)

{

case'A':user.log\_in();break;

case'B':user.registering();break;

case'C':manager.log\_in();break;

case'E':exit(1);break;

}

return 0;

}