

重庆师范大学

《数据结构课程设计》

课程名称： 数据结构

题 目： 航空订票系统

学 院： 计算机与信息科学学院

专业年级： 19 级计科 3+2

小组组长： 易立

小组成员： 王晓露、潘友琴、冯海娟、邓星云

指导教师： 张万里 职称： 讲师

2022年 1月 7日

目录

一 需求分析	- 2 -
1. 用户需求分析	- 2 -
2. 管理员需求分析	- 2 -
3. 程序总体设计图	- 3 -
二 问题描述	- 3 -
三 基本要求	- 4 -
四 概要设计	- 5 -
1. 算法思路	- 5 -
2. 工作分配	- 5 -
3. 程序模块	- 5 -
五 主程序	- 6 -
1. 结构体设计	- 6 -
2. 函数设计	- 7 -
3. 功能模块函数设计	- 7 -
4. 程序流程图设计	- 8 -
六 调试分析	- 12 -
1. 程序调试	- 12 -
1.1 用户方面	- 13 -
1.2 管理员方面	- 16 -
2. 系统改进设想	- 20 -
七 总结	- 21 -

航空订票系统

一 需求分析

1. 用户需求分析

(1) 用户使用此程序所要达成的工作主要为：查问所有航线信息、查看已订票客户信息、查问航线、办理订票业务、办理退票业务。经过此系统能够方便的进行上述工作。

(2) 每条航线所涉及的信息有：终点站名、航班号、飞行时间、票价、乘员定额、余票量。

(3) 已订票的客户名单信息包含姓名、订票量，舱位等级（1，2或3）。

(4) 等待替补的客户名单信息包含姓名、所需票量。

(5) 查问航线功能能够依据游客提出的终点站名输出航班号、飞行时间、票价、乘员定额、余票量。

(6) 订票业务功能依据客户提出的要求（航班号、订票数额）查问该航班票额状况，若另有余票，则为客户办理订票手续，输出座位号；若已满员或余票额少于定票额，则需从头咨询客户要求。若需要，可登记排队候补购票。

2. 管理员需求分析

(1) 增加航线：根据需求添加相应的航线。

(2) 查看航班订单：用户提交订票信息后，可以查看用户的航班订单。

(3) 三次密码锁定：输入三次密码错误，将锁定密码。

(4) 查看所有航班：可以查看所有的航班信息。

(5) 查看指定航线的候补订单：有些顾客没有抢到票，就排队候补，管理员就查看顾客候补情况，如果有人退票，可以根据候补为顾客进行订票手续。

(6) 删除航线：特殊原因影响，可以删除一些不需要的航线，避免顾客买错票。

(7) 退票业务功能依据客户供给的状况（航班号、姓名）为客户办理退票手续，而后查

问该航班能否有人排队候补，第一咨询排在第一的客户，若所退票额能知足他的要求，则为他办理订票手续，不然挨次咨询其余排队候补的客户。

3. 程序总体设计图

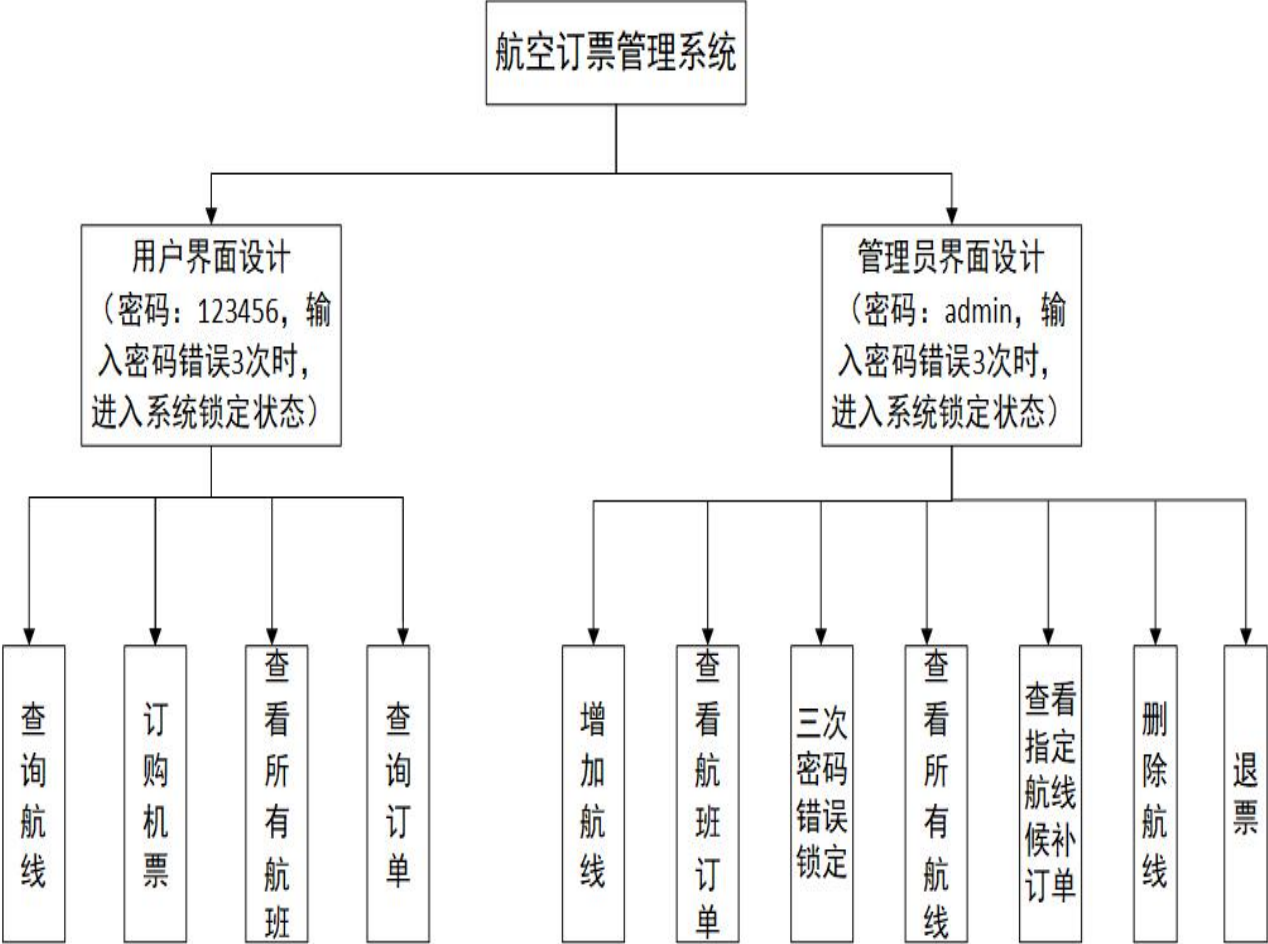


图 1 程序总体设计图

二 问题描述

因为传统订票系统比较繁琐，不方便，所以采用了网上订票的方式，使得乘客和航空公司更加方便、快捷、安全可靠，所以设计了航空订票系统，通过此系统可以实现如下功能：

增加：可以增加航班的所有消息。

查询：可以查询某个航线的情况（如，输入航班号，查询起降飞时间，起飞抵达城市，航班票价，确定航班是否满仓）；可以输入起飞抵达城市，查询飞机航班情况；

订票：可以订票，如果该航班已经无票，可以提供相关可选择航班；退票：可退票，退

票后修改相关数据文件；客户资料有姓名，证件号，订票数量及航班情况，订单要有编号。

退票：根据顾客的情况（如行程改变）为客户办理退票手续，然后查询该航班是否有人排队候补，首先询问排在第一的客户，若所退票额能满足他的要求，则为他办理订票手续，否则依次询问其它排队候补的客户

三 基本要求

每条航班所涉及的信息有：航班号，航班机型，日期，起飞时间，价格，乘员定额，余票量，订定票的客户名单(包括姓名，订票量，舱位等级（1/2/3等座）以及等候替补的客户名单（包括姓名、所需数量）。采用链表存储结构。

系统能实现的操作和功能如下：

（1）航班信息管理。

（2）查询航线

按以下几种方式查询：按航班号查询；按终点站查询；每种查询方式中，查询后输出如下信息：航班号，航班机型，起飞时间，价格，可乘坐旅客数量，余票量。

（3）承办订票业务

根据客户提出的要求(航班号，订票数额)查询该航班票额情况，若有余票，则为客户办理订票手续，输出座位号；若已满员或余票少于订票额,则需重新询问客户要求。若需要，可登记排队候补。

（4）承办退票业务

根据客户提出的情况(日期，航班号)，为客户办理退票手续，然后查询该航班是否有人排队候补，首先询问排在第一的客户，若所退票额能满足他的要求，则为他办理订票手续，否则依次询问其它排队候补的客户。

四 概要设计

1. 算法思路

使用了DevC++进行整个程序的实现，使用了带头结点双向链表，带头结点单链表，链表和结点的创建，和遍历、队列的灵活使用。

2. 工作分配

(1) 框架设计：由邓星云进行构思，同时参考其他人的意见。

(2) 系统设计：由冯海娟进行框架的编写。

(3) 程序设计：由易立进行各项功能的编辑。

(4) 程序调试：由王晓露进行程序的调试与调整。

(5) 文档制作，由潘友琴完成文档的编辑工作。

3. 程序模块

航空订票系统主要包括以下功能：

登录功能：在登录的主界面，我们可以采用普通用户登录和管理员身份进行登录，并且都需要输入密码进行验证登录。

删除功能：可以在管理员的界面选择删除功能，需要输入正确的航班号才能对其进行有效的删除。在管理员的功能中还有退票是需要通过输入相对应的信息进行退票。

增加功能：可以在管理员身份登录的情况下选择航线的增加功能，并且需要输入对应所需要完善的增加航班的信息，在输入完成后还要输入随即得到的验证码才能够添加成功。

订购功能：可以在普通用户的界面选择订购机票功能，可以通过目的地或者航班号查询过后再进行订购，并且会根据你所购买的舱位等级还有购买的票数进行计算最终需要支付的金额。

退票功能：在管理员的功能中还有退票是需要通过输入相对应的信息进行退票。

查询功能：在普通用户的功能中可以查询所有的航班信息，也可以通过用户想要到达的目的地进行查看航班信息，并且可以查看自己订票的信息。

在管理员的功能中可以查看所有客户的订单，还有查看航班候补客户的订单，当然也可以查看所有航班信息。

五 主程序

1. 结构体设计

表 1 结构体设计表

结构体定义	变量	说明
typedef struct clientNodehaha{ }	char name[MAXSIZE];//姓名 char ID_num[MAXSIZE];//证件号 char flight_num[10];//航班号 char time[MAXSIZE];//起飞时间 int order_num;//订单号 int amout;//订票数量 int rank;//舱位等级	用带头结点双向链表，创建订票客户名单
typedef struct waitNode{ }	char name[MAXSIZE];//姓名 char ID_num[MAXSIZE];//证件号 int amout;//订票数量	用带头结点单链表，创建等候候补客户名单
typedef struct Flight{ }	char destination[MAXSIZE];//终点站 char flightId[MAXSIZE];//航班名，唯一 char planeId[MAXSIZE];//飞机号 char time[MAXSIZE];//起飞时间 float money;//票价 int num;//乘员余额 int tickets;//余票量 clientListL CLientc;//已订票客户名单 waitPtr wqueue;//等候替补的客户名单	用带头结点的单链表，创建航班总信息链表

2. 函数设计

表 2 函数设计表

函数	说明
Status initFlight(flightList &F)	初始化航班列表
Status init_passengerList(clientListL & CLient)	初始化乘客链表
Status init_waitQueue(waitPtr &Q)	初始化候补乘客链表
flightlist creat_Flight(char *des,char *fid,char *pid,char *time, float money,int num,int tickets)	创建航班结点
clientListL creat_client(char *name,char *ID_num,char *flight_num,char *time,int order_num,int amout, int rank)	创建乘客结点
Status insert_flight(Flight *&h,Flight *p)	在航班总信息链表中插入新航班信息
Status insert_clientList(clienListL &totalList,clientListL &cp)	在乘客链表中插入新结点
Status en_waitQueue(waitPtr &wqueue,char *n,int amount,char *ID_num)	创建候补乘客结点，并且将其插入链表尾部
int length_queue(waitPtr Q)	使用队列求候补乘客人数
int delete_flight(flightList &f,char *flightId)	删除航班链表结点
Status delete_clientList(clientListL &cl,clientListL &dp)	删除乘客链表结点

3. 功能模块函数设计

表 3 功能模块函数设计表

功能模块函数	说明
void adminhome()	管理员主页面
void adminView()	管理员登录界面
void mixmima()	三次密码错误后界面
void home()	登录主页面
void displayClientInfo(flightList air)	输入航班号查询信息

void displaywaitClientInfo(flightList air)	查看候补人员信息
void show(flightList &air,waitNode *&Q)	退票时处理，候补队列
int refund(flightList &air)	退票
void userhome()	普通用户主页面
void mimal(int &tags1)、 void mima(int &tags)	密码设计
void userView()	普通用户登录界面
int main()	主函数
void find(flightList air)	查询航线
int booking(clientList *&airCLient,int amount,int sum,int order_num,char *name,char *ID_num,char *flightId,char *time)	订票
void recommend(Flight *air,clientList *Client,char *flightId,char *destination,int amount)	推荐航班路线
int wiat(waitPtr &wqueue,int amount)	加入候补队列
int bookTickets(flightList &air)	引导用户进行订票
void displayAllInfo(Flight *F)	显示所有航班信息
void findddingdan(flightList air)	查询订单
void addFlight(flightList &air)	增加航线
void deleteflight(flightList &air)	删除航线

4. 程序流程图设计

(1) 查询航线流程图

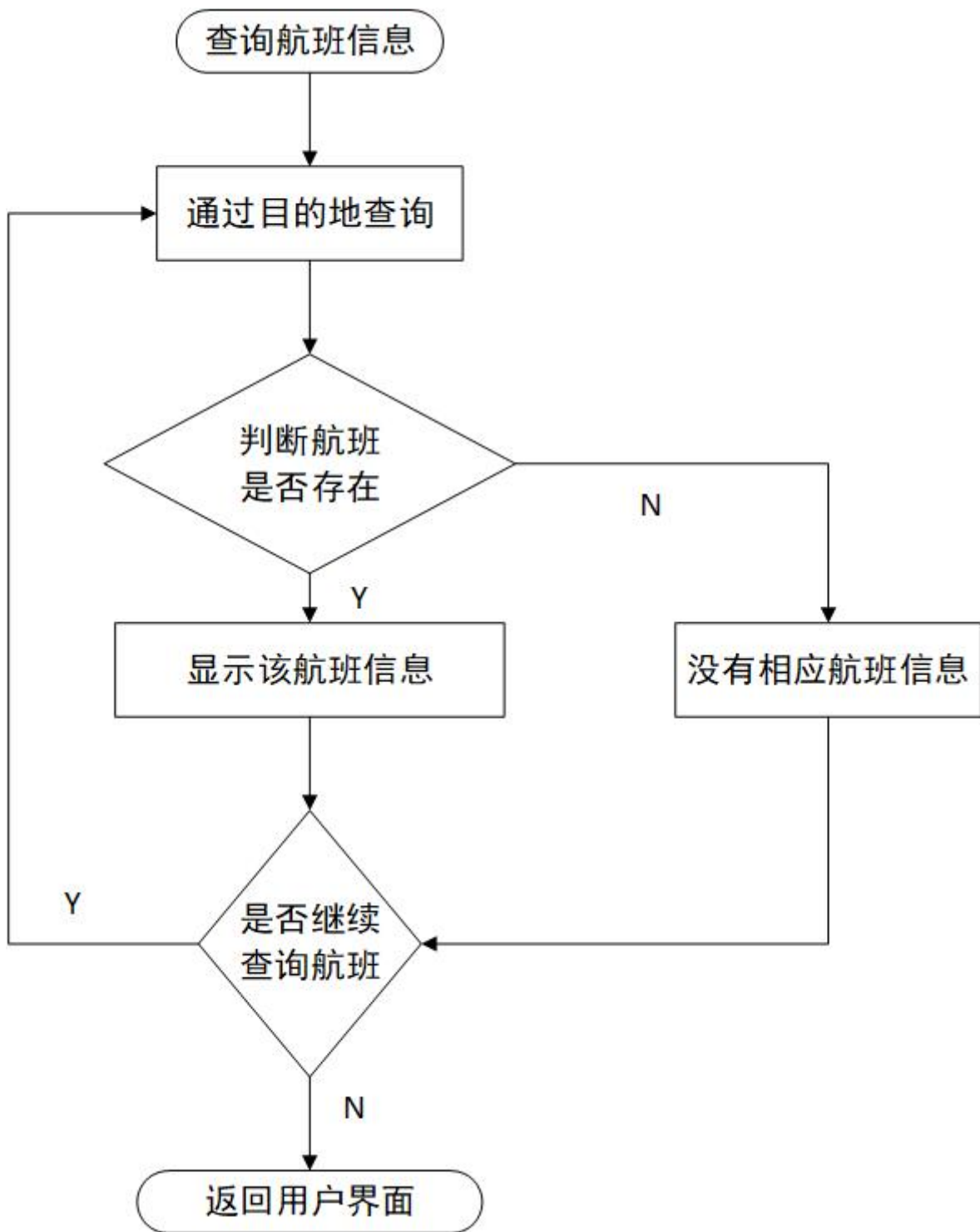


图 1 查询航线流程图

(2) 增加航线流程图

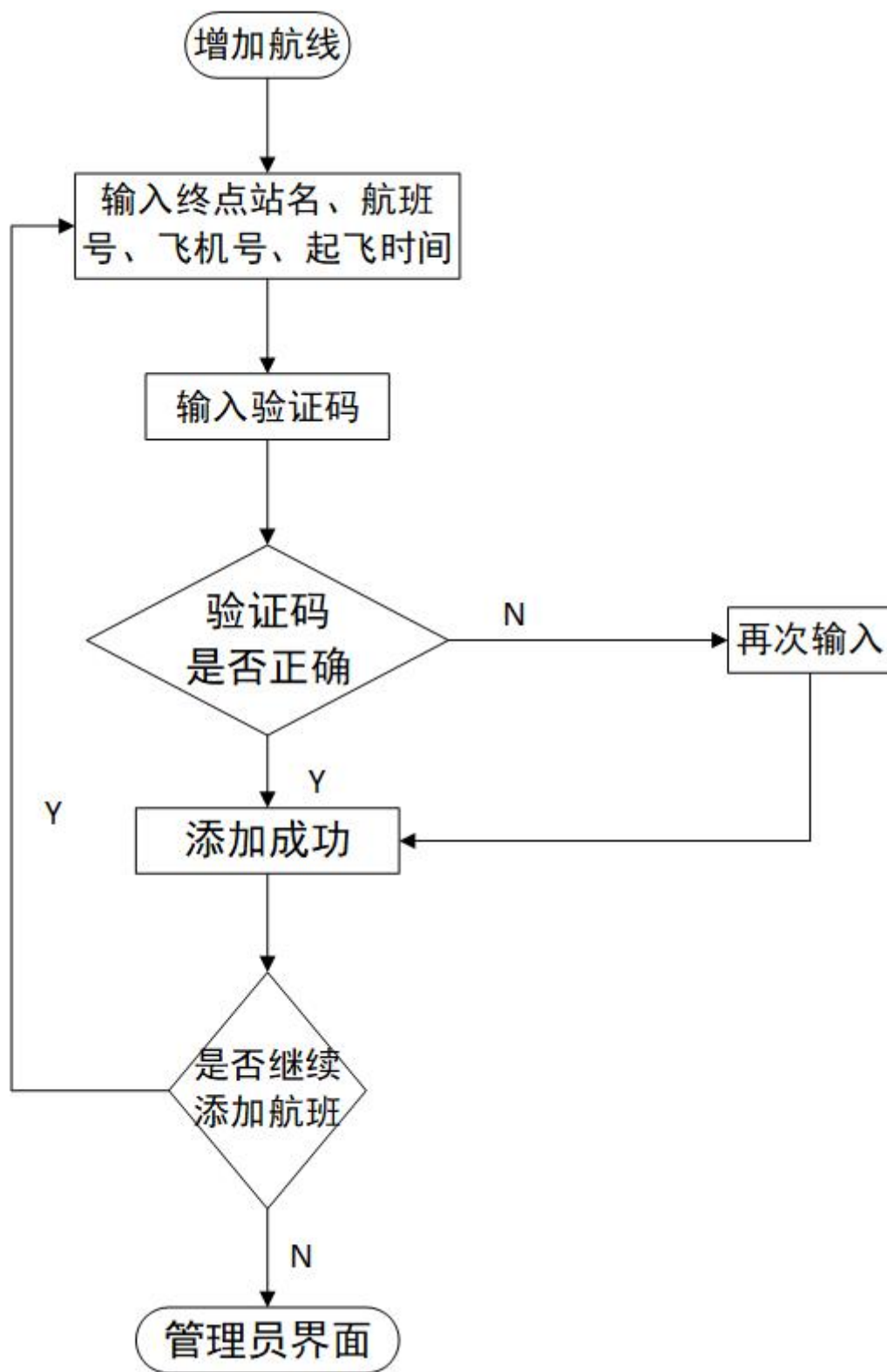


图 2 增加航线流程图

(3) 订票功能流程图

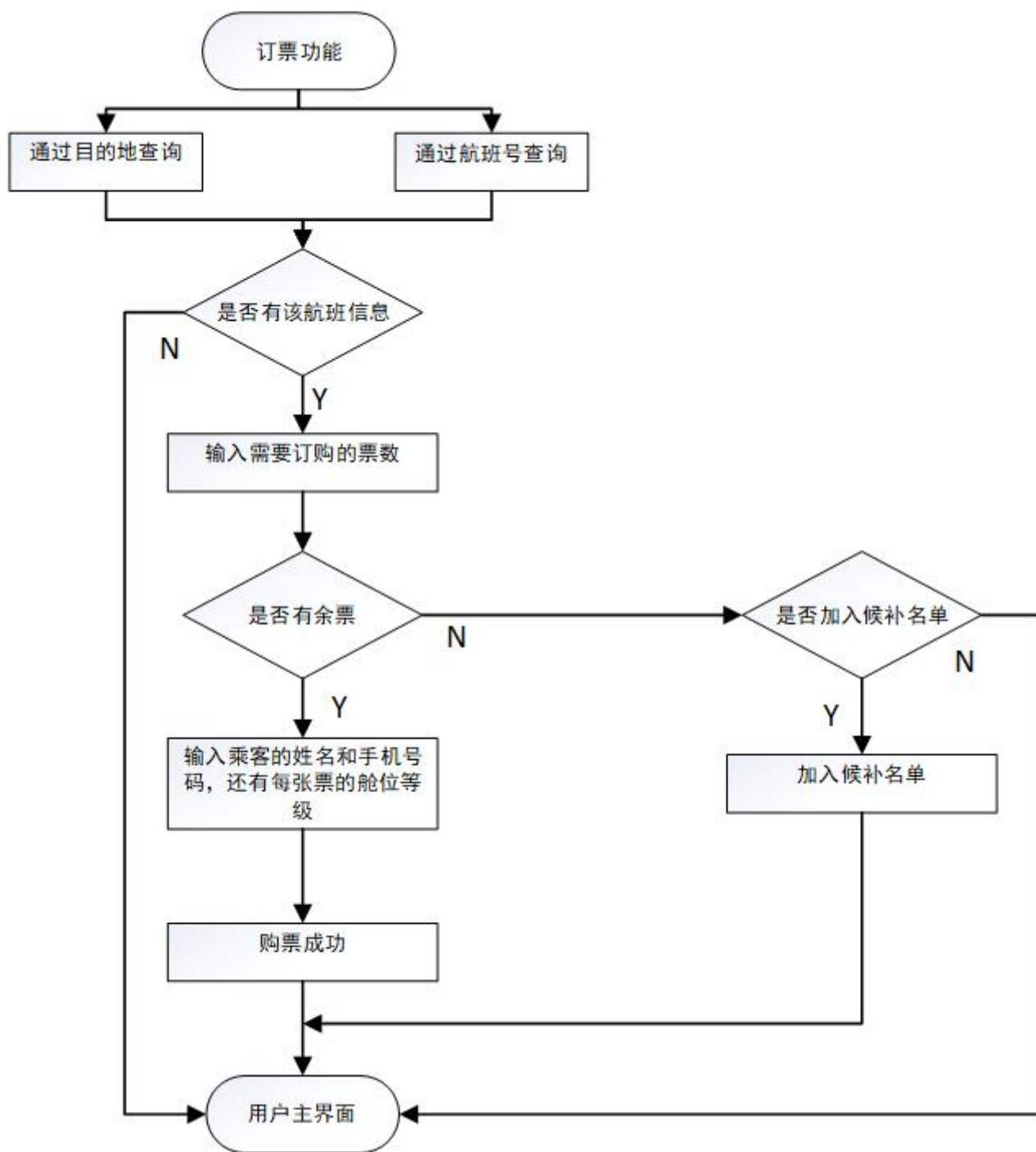


图 3 订票功能流程图

(4) 退票功能流程图

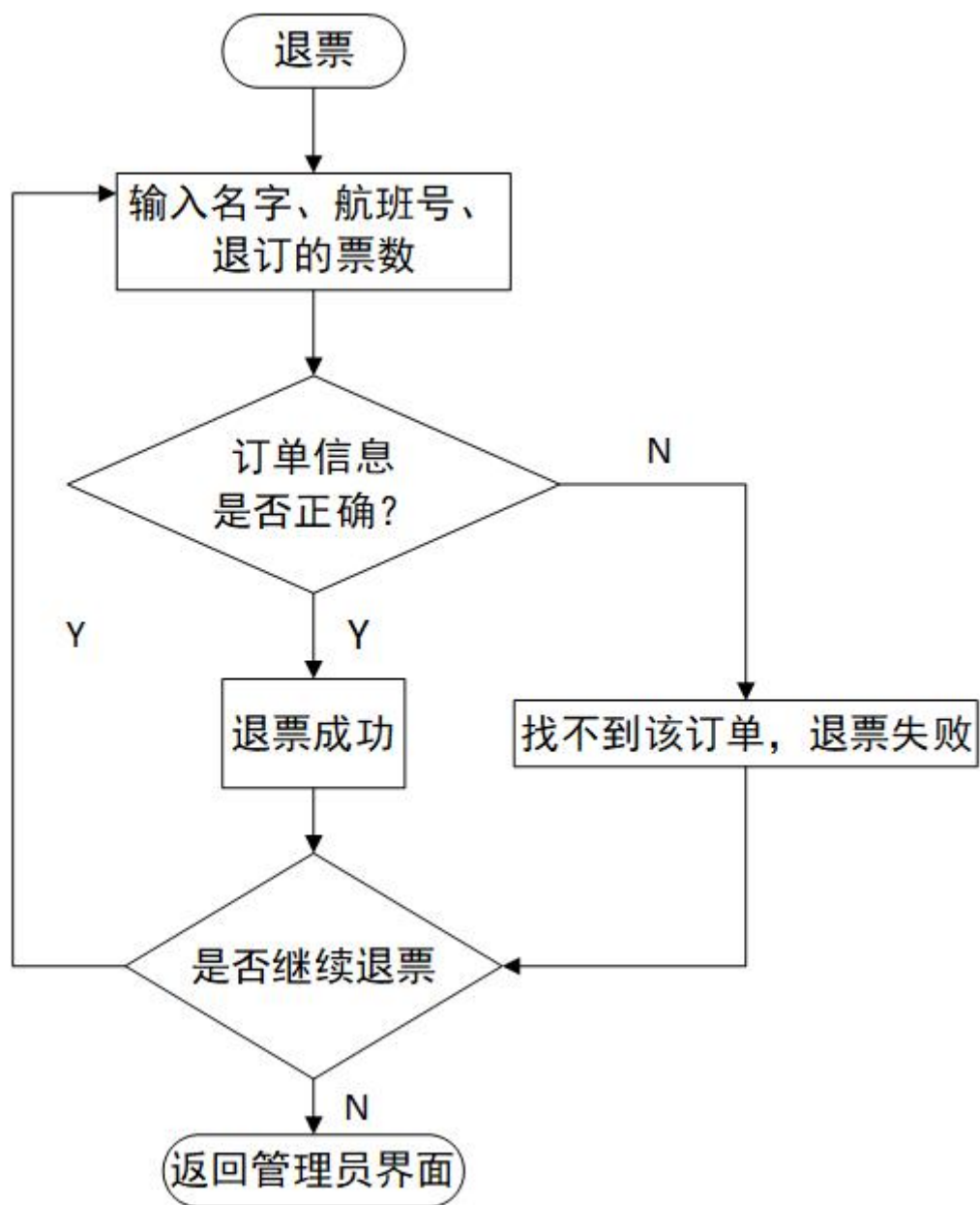


图 4 退票流程图

六 调试分析

1. 程序调试

系统登陆主页面，定义主页面函数 `void home()`，在这里我们可以选择用户或者管理员的身份进行登录，同时也定义了密码函数 `void mima(int &tags)`，三次密码过后退出系统并重新登录，如图 5 所示。



图 5 系统主界面

1.1 用户方面

(1) 首先通过普通用户的身份进行登录，并且输入密码（123456）进入到普通用户的界面，如图 6 所示，当密码输错 3 次时，会调用 mixmima()函数，会提示退出系统重登。



图 6 用户主界面

(2) 用户主界面也使用了 switch case 语句进行页面的跳转，当选择数字 1 时，会调用 void find(flightList air)函数，并且从头结点后一结点开始遍历航班，实现查询航线的功能，如图 7 所示。



图 7 输入目的地查询航线

(3) 当选择数字 2 时, 会调用 `int bookTickets(flightList &air)` 函数, 引用了订票函数, 推荐路线函数, 加入候补队列函数, 从头结点后一结点开始查询, 可以输入你的目的地(按 1)或者航班号(按 2)来查询, 如图 8, 图 9 所示。如果所需票数大于剩余票数会调用 `int wiat(waitPtr &wqueue, int amount)` 函数, 是否加入候补队列。如果根据航班号订票, 航班满员或者余票不足将会调用 `void recomand()` 函数, 是否需要推荐其他航线。

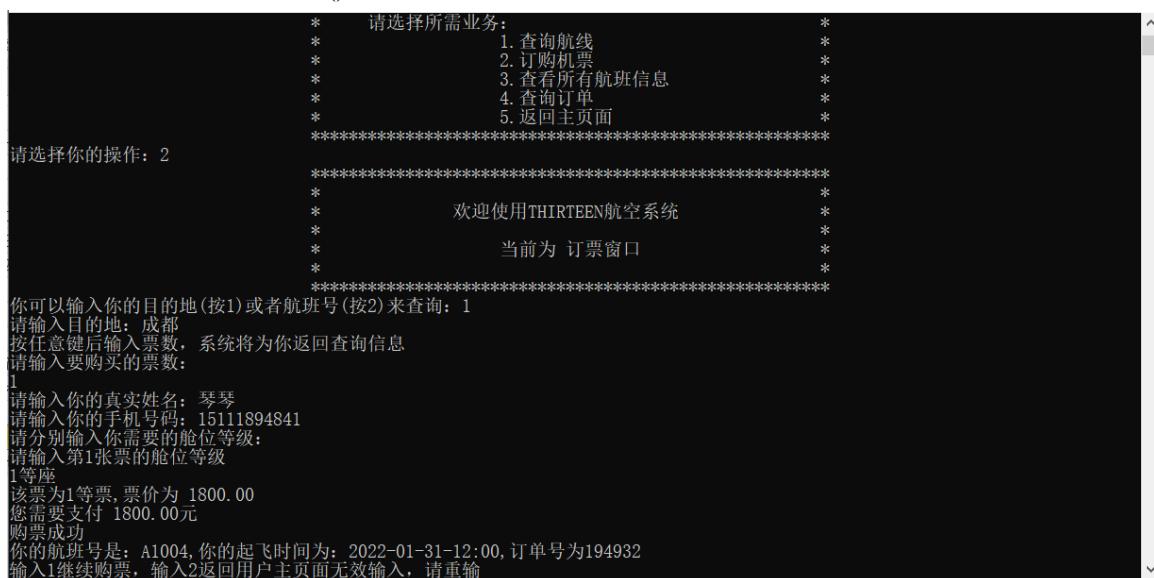


图 8 用户订票



图 9 输入航班号查询航线

(4) 当选择数字 3 时，会调用 `void displayAllInfo(Flight *F)` 函数，遍历所有航班，如果不为空，输出所有航班信息，如果为空就会输出抱歉没有航班信息，实现查询所有航班信息功能，如图 10 所示。



图 10 查看所有航班消息

(5) 当选择数字 4 时，会调用 `void finddingdan(flightList air)` 函数，输入航班号从头结点后一结点开始查询航班，当没有航班时，输出不存在航班，当查找到航班时，输入名字开始查询订单，如果查询到相应名字输出查询结果，如图 11 所示。

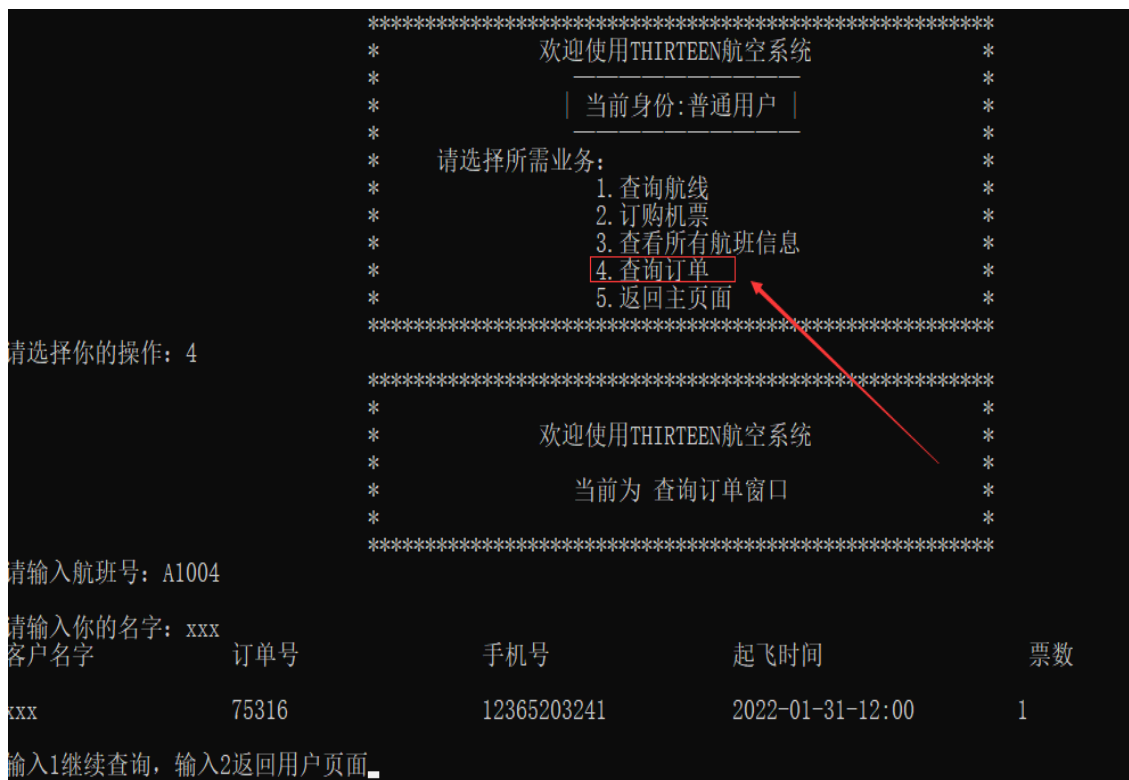


图 11 查看订单消息

1.2 管理员方面

(1) 通过管理员身份登录, 输入密码 (admin) 进入到管理员登录界面, 如图 12 所示。当密码输错 3 次时, 会调用 mixmima() 函数, 会提示退出系统重登。



图 12 管理员主界面

(2) 当选择数字 1 时, 会调用 void addFlight(flightList &air) 函数, 输入对应的信息, 设置了验证码, 由随机数生成, 输入验证码成功后, 航班信息才能添加成功, 如图 13 所示。

```

*****
*                               欢迎使用THIRTEEN航空系统                               *
*                               | 当前身份:管理员 |                               *
*                               |                               |                               *
*      请选择所需业务:                               *
*      1. 增加航线                               *
*      2. 删除航线                               *
*      3. 查看航班的客户订单                               *
*      4. 查看航班候补客户订单                               *
*      5. 查看所有航班                               *
*      6. 退票                               *
*      7. 返回主页面                               *
*****
请选择你的操作: 1

*****
*                               欢迎使用THIRTEEN航空系统                               *
*                               | ！！高级权限，请谨慎操作！！ |                               *
*                               |                               |                               *
*                               |                               |                               *
*                               |                               |                               *
*****

请输入终点站名: 南京
请输入航班号: A1005
请输入飞机号: D333
请输入起飞时间(格式为cccc-ww-jj-hh:aa):2021-12-15-12: 00
票价(精确到小数点后两位): 800.00
请输入乘员定额: 6
请输入余票数量: 4

```

图 13 添加航班信息

(3) 当选择数字 2 时，会调用 void deleflight(flightList &air)函数，进行删除航班功能，如图 14 所示。

```

*****
*                               欢迎使用THIRTEEN航空系统                               *
*                               | 当前身份:管理员 |                               *
*                               |                               |                               *
*      请选择所需业务:                               *
*      1. 增加航线                               *
*      2. 删除航线                               *
*      3. 查看航班的客户订单                               *
*      4. 查看航班候补客户订单                               *
*      5. 查看所有航班                               *
*      6. 退票                               *
*      7. 返回主页面                               *
*****
请选择你的操作: 2

*****
*                               欢迎使用THIRTEEN航空系统                               *
*                               | ！！高级权限，请谨慎操作！！ |                               *
*                               |                               |                               *
*                               |                               |                               *
*                               |                               |                               *
*****

请输入你要删除的航班号: A1005

删除成功！删除航班成功，输入1继续删除，输入2返回管理员页面

```

图 14 删除航线

(4) 当选择数字 3 时，会调用 void displayClientInfo(flightList air)函数，输入航班号，如果找到对应航班号将会输出该航班所有客户订单，如图 15 所示。



图 15 查看航班的客户订单

(5)当选择数字 4 时,会调用 void 当选择数字 3 时,会调用 void displayClientInfo(flightList air) 函数,输入航班号,如果找到对应航班号将会输出该航班所有客户订单。
displaywaitClientInfo(flightList air)函数查看航班候补客户订单,与查看航班客户订单方法类似,如图 16 所示。

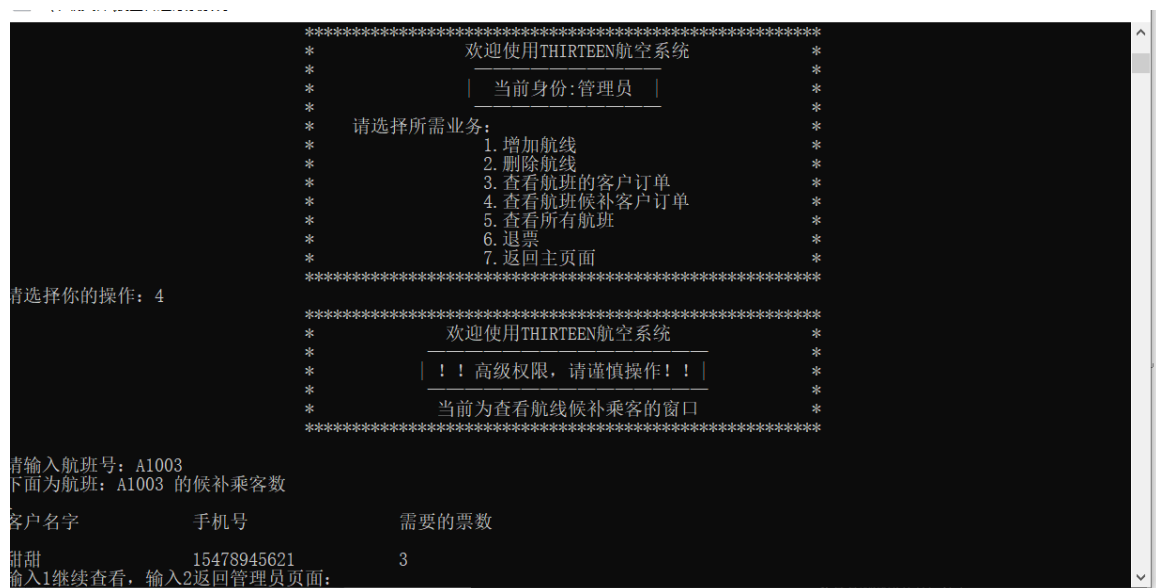


图 16 查看航班候补客户名单

(6)当选择数字 5 是,会调用 void displayAllInfo1(Flight *F)函数,可以查看所有航班的信息,如图 17 所示。

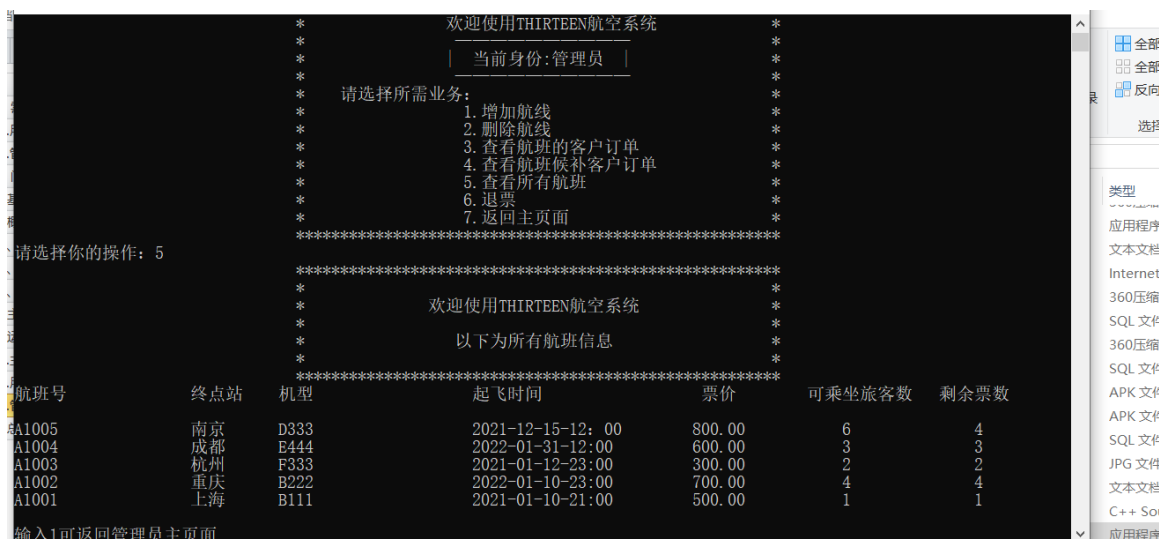


图 17 查看航班信息

(7) 当选择数字 6 时，会调用 `int refund(flightList &air)` 函数，输入对应的信息进行退票，退票完成后通过调用 `void show(flightList &air, waitNode *&Q)` 函数，将之前用户提交的候补订单进行处理，循环查找满足当前航班余票数的候补乘客订单，并询问是否接受此次订票请求，操作成功会将对应信息发送给客户，如图 18, 19 所示。



图 18 为客户退票(无候补客户时的情况)

```
*
*      3. 查看航班的客户订单      *
*      4. 查看航班候补客户订单    *
*      5. 查看所有航班            *
*      6. 退票                    *
*      7. 返回主页面              *
*
*****

请选择你的操作: 6

*****
*      欢迎使用THIRTEEN航空系统    *
*      | ！！高级权限，请谨慎操作！！ | *
*      当前为退票窗口              *
*
*****

请输入你的名字: qqg
请输入航班号: A1001
请输入你要退订的票数: 1

退票成功！退款金额已经成功原路返回您的账号
现在票数为1
现在在检索乘客  www... 请稍后...
现在在检索乘客  ppp... 请稍后...
该候补乘客满足条件！

他的名字是 ppp ，他的手机号码是 12032023055 ，他的订票数是 1

请与他进行联系！！（他的订票数可以在余票范围内进行更改）
*****
*      余票为:1      *
*****

请询问其新的票数需求，和舱位需求
他是否接受此次订票请求?(y/n)y
请输入他的订票数: 1
请分别输入客户需要的舱位等级:
请输入第1张票的舱位等级
2
该票为2等票,票价为 750.00
总费用为 750.00元
购票成功
该乘客的起飞时间为: 2021-01-10-21:00, 订单号为313204

!!!操作成功!!!
订单号和所需费用已经发送到客户手机 12032023055 中!!! 输入1返回主页面，输入2返回管理员页面请输入: _
```

图 19 为客户退票(有候补客户时的情况)

2. 系统改进设想

这个程序有很多地方可以改进，例如：

- (1) 数据全部存储在内存中，可以考虑编写在文本文件中，运行打开，这样可以保重数据不会丢失。
- (2) 航线信息只能通过修改代码来修改，在这里可以编写一个函数来修改航线信息并将其修改内容保存在文档中。
- (3) 在功能上还可以添加查看座位号。
- (4) 在该程序中我们没有实现修改乘客订单信息的功能，可以进行相对应的改进，在管理员的模块中加入修改信息的功能。

七 总结

本次实验是这学期最后一次小组实验，我们小组设计的是航空订票系统，通过本次实验，让我们更加了解了链队列的存储结构及其算法的实现，了解了链队列的特点以及链队列抽象出的数据类型的定义，熟练掌握了链队列基本算法如何实现，包括如何实现建立，查找、插入、删除、订票和退票等操作。

我们小组完成的航空订票系统实现了以下功能，首先进入登录界面，在这里我们可以选择用户登录或者管理员登录，如果我们三次输错密码，将返回主页面进行重新登录。当我们选择普通用户登录时，我们可以输入目的地进行航线的查询，查询目的地或者航班号来订购机票，查看所有航班的信息以及输入航班号查看订单。当我们选择管理员身份登录时，可以增加航线，输入航班号可以进行删除航线、查看航班的客户订单和查看航班候补客户订单，查看所有航班以及实行退票。这便是我们系统的所有实现功能，虽然在这过程中，我们会遇到很多问题，但是经过一系列的查找资料以及询问同学也都一一解决了。

实验就是把理论运用于实际，使我们了解到平时也应该注重老师讲的知识，也增强了我们的自学能力，以及程序设计的基本技能、解决实际问题的能力，为以后打下良好的基础。学习是一个厚积薄发的过程，平时在学的过程中掌握得多，理解多，练得多，真正到自己动手做某实验的时候就不会那么棘手，因此在以后学习过程中应该注重实践，而不只是只关注课本上的零碎知识点。通过本次课程设计，我们也认识到了合作学习的重要性，小组成员分工合作，才有了最后的实验成果。