**飞机订票系统实验报告**

#### 一、实验题目

飞机订票系统设计

#### 二、实验内容

设计一个功能比较完全的飞机订票系统。

该系统的功能主要包括：

①用文件保存航班的信息，需要提供文件的写入的读出操作。

②航班信息的浏览功能，需要提供文件的显示操作。

③存储客户信息和客户订票信息，并能浏览和查询客户订票信息

④提供键盘式选择菜单以实现功能选择。

三、实验目的和要求

**1．目的**

做一个训练C语言的各种语法、文件的读出和写入等使用、结构体的使用、链表的应用等操作。

**2．输入输出的要求：**

①数字选择输入：根据光屏提示文字进行输入。

②输出多项式：通过读取文件信息以及用户写入文件的信息显示在系统的终端。

#### 四、程序的框架结构

1．主要数据类型定义：

对于飞机订票系统来说，主要分为客户的订票信息和各条航班的信息，客户订票信息包括：客户姓名、所选择航班信息、客户身份证号、所订票数、所选航班级别。因此可将每个客户的订票信息作为线性表中的一个元素，定义一个嗲有头节点的链表来做客户订票信息模块。

typedef struct BOOK

{

char Airnum[8]; //订票的航班号

char name[20]; //客户信息

char didentify[20]; //客户身份证

int order\_amount; //订票数

int grade; //航班级别

struct BOOK \*Pnext;

}\*PBookList,BookList;

对航班而言，需要：航班号、起始地、终止地、起始时间、终止时间、起飞日期、价格、航班类型、乘客定额、所余票数，以此来组成链表。

typedef struct airinfo

{

char Airno[8]; //航班号

char start[10]; //起始地点

char end[10]; //终止地点

char sche[10]; //起飞日期

char time1[10]; //起飞时间点

char time2[10]; //终止时间点

char mode[20]; //飞机类型

int price; //机票价格

int fixed;//乘客定额

int extratkt; //剩余票数

PBookList order;/\*指向乘员名单链表的头指针，通过这个指针就可以找到所有乘客的名单了\*/

}AirInfo,\*PAirInfo;

2．本程序包括以下四个模块：

①主程序模块：

main()

{

用户登录系统；

读取文件中所有航班信息；

读取客户订票信息系统；

选择{

1.显示各航线信息；

2.查询各航线信息；

3.订 票；

4.退 票；

5.新航班信息录入；

0.退 出 系 统；

}

打印用户所选的信息；

}

②用户登录模块：

void login\_user () //用户的登陆

{

打开用户文件；

将用户的姓名、年龄、性别、身份证号写入文件；

关闭文件；

}

③航班信息的显示：

void meau1(void)

{

输入需要显示信息的序号；

{

1. 显示所有航班的信息；
2. 显示所有订单客户；
3. 返回主菜单；

}

打印用户选择的显示信息；

}

④航线信息的查询：

void meau2() //查询航线信息的子菜单。

{

输入需要查询信息的序号；

{

1. 按航班号查询；
2. 按起始站查询；
3. 按终点站查询；
4. 按起飞日期查询；
5. 返回主菜单；

}

通过与文件信息内容的对比进行查询；

若存在该信息则进行打印该信息；

若不存在返回用户不存在的信息；

}

⑤订票模块：

void orderTicket() //用户订票信息的写入

{

打开订票信息的文件；

若文件不存在，返回error，程序终止；

写入入订票客户的姓名、身份证号、所订航班号、订票数量、航班级别；

查询有没有该用户输入的航班信息；

输出客户的座位号(通过航班总额度和剩余票数计算)；

打印该条航班的信息；

该航班的剩余票更新；

}

⑥退票模块：

void refund\_ticket()

{

输入客户的姓名；

查询订票信息中有没有该客户姓名；

若有，通过链表去掉该订票信息，并更新余额票数；

若没有，返回错误信息；

}

⑦新航班信息的录入：

void newAirMsg()

{

打开航班信息的文件；

将一个航班的所有信息：航班号、起始地、终止地、起始时间、终止时间、起飞日期、价格、航班类型、乘客定额、所余票数，写入航班信息的文件；

打印该航班的信息；

关闭航班信息的文件；

}

3．各程序模块之间的层次关系。

主程序模块

用户登录模块

查询各航线信息m

订票模块

显示各航线信息 nm

航班录入m

退票模块m

#### 五、源代码

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

void login\_user();

void meau1(void); //打印航线信息模块；

void printAllofAirMsg(); //打印所有航线

void printfOneofAirMsg(int i); //打印选中的某条航线；

void meau2(void); // 航线查询模块

void searchbyAirno(); //以飞机型号查询飞行信息；

void searchbystartPlace(); //以飞机起始地查询

void searchbyEndPlace(); //以飞机终止地查询

void searchbySche(); //以飞机飞行日期查询

void newAirMsg(); //新航线的录入

#define MAXSIZE 20

#define MAX 80

#define BOOKMAX 100

struct User //用户

{

char userName[20]; //用户姓名

char passWord[18]; //用户密码

char sex[5]; //性别

char didentify[20]; //身份证号

int is\_admin; 1

struct User \*next;

} user;

typedef struct BOOK

{

char Airnum[8];

char name[20];

char didentify[20];

int order\_amount;

int grade;

struct BOOK \*Pnext;

}\*PBookList,BookList;

typedef struct airinfo

{

char Airno[8];

char start[10];

char end[10];

char sche[10];

char time1[10];

char time2[10];

char mode[20];

int price;

int fixed;//乘客定额

int extratkt;

PBookList order;/\*指向乘员名单链表的头指针，通过这个指针就可以找到所有乘客的名单了\*/

}AirInfo,\*PAirInfo;

typedef struct

{

AirInfo s[MAX];

int acount;

} SSList;

//订单客户总体信息

SSList L;

//航班信息的结构体变量L，生成s[MAX]个Airinfo结构体大小的数组

typedef struct

{

BookList book[BOOKMAX];

int book\_acount;

} SbookList;

SbookList B;

//订票信息的结构体变量，生成book[BOOKMAX]个客户的订票信息

void login\_user() //用户的登陆

{

FILE \*p;

if((p=fopen("客户信息.txt","a+"))==NULL)

{

printf("文件打开错误！");

exit(1);

}

printf("登陆");

printf("姓名：");

scanf("%s",user.userName);

printf("\n");

getchar();

printf("年龄；");

scanf("%s",user.passWord);

printf("\n");

getchar();

printf("性别：");

gets(user.sex);

printf("\n");

printf("身份证号码：");

gets(user.didentify);

printf("\n");

fprintf(p,"\n");

fputs(user.userName,p); fprintf(p," ");

fputs(user.passWord,p); fprintf(p," ");

fputs(user.sex,p); fprintf(p," ");

fputs(user.didentify,p);

fclose(p);

}

void readinfo(SSList \*L) //航班信息文件的读取

{

FILE \*fp;

if((fp=fopen("航班信息.txt","r"))==NULL)

{

printf("航班文件不存在\n");

exit(1);

}

else

{

int i=0;char tep[500];

int retake;

L->acount=0;

while(!feof(fp))

{

fgets(tep,500,fp);//读取第一行说明

retake=fscanf(fp,"%s %s %s %s %s %s %s %d %d %d",

L->s[i].Airno,L->s[i].start,L->s[i].end,L->s[i].sche ,

L->s[i].time1 ,L->s[i].time2,L->s[i].mode,&L->s[i].price,

&L->s[i].fixed,&L->s[i].extratkt);

if(retake!=10)

{

printf("读取文件出错1\n");

fclose(fp);

exit(1);

}

i++;

L->acount++;

}

}

fclose(fp);

}

void readinfo\_book(SbookList \*book) //订票客户信息的读取

{

FILE \*fp;

if((fp=fopen("订票信息.txt","r"))==NULL)

{

printf("订票信息.txt文件不存在 \n");

exit(1);

}

else

{

int i = 0; char title[500];

B.book\_acount = 0;

int retake;

while(!feof(fp))

{

fgets(title,500,fp);

if(fscanf(fp,"%s %s %s %d",B.book[i].Airnum,B.book[i].name,B.book[i].didentify,&B.book[i].grade)!=4)

{

printf("读取文件出错！\n");

fclose(fp);

exit(1);

}

i++;

B.book\_acount++;

}

}

fclose(fp);

}

void printfOneofAirMsg(int i) //输出某一条特定的航班信息

{

printf("航班号 起点 终点 飞行日 起飞时间 到达时间 机型 价格 成员定额 余票量\n");

printf("%s\t%s %s %s %s %s %s %d %d %d \r\n",L.s[i].Airno,L.s[i].start,L.s[i].end,L.s[i].sche,

L.s[i].time1,L.s[i].time2,L.s[i].mode,L.s[i].price,L.s[i].fixed,L.s[i].extratkt);

}

PBookList insertlink(PBookList head,char Airnum[],int amount,char name[],int grade,char ID[]) //客户订票信息链表

{

PBookList p1,new1;

p1=head;

new1= (PBookList)malloc(sizeof(BookList));

if(new1==NULL)

{

printf("\n动态内存分配失败，请重新分配\n");

return NULL;

}

strcpy(new1->Airnum,Airnum);

strcpy(new1->name,name);

new1->order\_amount=amount;

new1->grade=grade;

strcpy(new1->didentify,ID);

new1->Pnext=NULL;

if(head==NULL)

{

head=new1;

new1->Pnext=NULL;

}

else

{

head=new1;

new1->Pnext=p1;

}

return head;

}

void InitOrder()

{

PAirInfo info;

info=L.s;

int i,j;

for(i=0;i<B.book\_acount;i++)

{

for(j=0;j<L.acount;j++)

{

if(strcmp(B.book[i].Airnum,info[j].Airno)==0)

{

printf("第%d个航班线路匹配信息：\n",i+1);

info[j].order = insertlink(info[j].order,B.book[i].Airnum,B.book[i].order\_amount,

B.book[i].name,B.book[i].grade,B.book[i].didentify);

printfOneofAirMsg(j);

}

}

}

}

void printAllofAirMsg(void) //所有航班信息的输出

{

int i=0;

for(i=0;i<L.acount;i++)

{

printf("航班号 起点 终点 飞行日 起飞时间 到达时间 机型 价格 成员定额 余票量\n");

printf("%s %s %s %s %s %s %s %d %d %d\r\n",L.s[i].Airno,L.s[i].start,L.s[i].end,L.s[i].sche,

L.s[i].time1,L.s[i].time2,L.s[i].mode,L.s[i].price,L.s[i].fixed,L.s[i].extratkt);

}

}

PAirInfo findByAirno() //通过航班号查询航班信息

{

PAirInfo info ,a;

char Airnumber[10];

int sel= 0;;

int i=0;

info=L.s;

a=L.s;

printf("===============================\n");

printf("== 为您列出航班号供您选择： =\n");

int j;

for(j=0;j<L.acount;j++)

{

printf("%d %s\n",j,a[j].Airno);

}

printf("===============================\n");

//让用户选择一个序列号，根据序列号进行选择

printf("请输入要选择航班的序号:");

scanf("%d",&sel);

if(sel>=L.acount)

{

printf("输入错误\n");

return;

}

else

{

strcpy(Airnumber,info[sel].Airno);

}

while(i<L.acount)

{

if(strcmp(Airnumber,info[i].Airno)==0)

{

printf("==================================================================\n");

printf("您选择的航班具体信息如下：\n\n");

printfOneofAirMsg(i);

return &info[i];

}

i++;

}

if(i>=L.acount)

{

printf("\n\n");

printf("sorry,we did not find Msg!\n\n");

}

return NULL;

}

void printOrderCustomByAirNum1() //订票信息的查询

{

PBookList p;

PAirInfo info;

info = findByAirno();

p=info->order;

if(p!=NULL)

{

printf("===============================\n");

printf("===============================\n\n\n");

printf("航班号 姓名 身份证 订票数量 座位等级\n");

while(p)

{

printf("%s\t%s\t %s \t%d %d\n",p->Airnum,p->name,

p->didentify,p->order\_amount,p->grade);

p=p->Pnext;

}

printf("===============================\n\n\n");

printf("===============================\n");

}

else

{

printf("\n\n");

printf("!!!!!该航线没有客户信息!!!!!\n");

printf("\n\n");

}

}

void searchbyAirno() //通过航班号查询航班信息

{

PAirInfo info ,a;

char Airnum01[10];

int i=0,sel;

info=L.s;

a=L.s;

printf("===============================\n");

printf("== 为您列出航班号供您选择： =\n");

int j;

for(j=0;j<L.acount;j++)

{

printf("%d %s\n",j,a[j].Airno);

}

printf("===============================\n");

printf("请输入要选择航班的序号:");

scanf("%d",&sel);

if(sel>=L.acount)

{

printf("输入错误\n");

return;

}

else

{

strcpy(Airnum01,info[sel].Airno);

}

while(i<L.acount)

{

if(strcmp(Airnum01,a[i].Airno)==0)

break;

i++;

}

if(i>=L.acount)

{

printf("\n\n");

printf("对不起，该航线未找到！\n");

printf("\n\n");

}

else

{

printf("================================================================================\n");

//printf("航班号 起点 终点 飞行日 起飞时间 到达时间 机型 价格 乘员定额 余票量\n");

printfOneofAirMsg(i);

}

}

void searchbystartPlace() //通过起始地查询航班信息

{

PAirInfo info ,a;

char StartPlace[10];

int i=0,sel;

info = L.s;

a=L.s;

printf("===============================\n");

printf("== 为您列出所有航班的起点站供您选择： =\n");

scanf("%d",&sel);

if(sel>=L.acount)

{

printf("输入错误\n");

return;

}

else

{

strcpy(StartPlace,info[sel].start);

}

while(i<L.acount)

{

if(strcmp(StartPlace,a[i].start)==0)

break;

i++;

}

if(i>=L.acount)

{

printf("\n\n");

printf("对不起，该航线未找到!");

printf("\n\n");

}

else

{

printf("================================================================================\n");

//printf("航班号 起点 终点 飞行日 起飞时间 到达时间 机型 价格 乘员定额 余票量\n");

printfOneofAirMsg(i);

}

}

void rewrite(SSList L) //航班信息的写入

{

FILE \*fp;

int i;

if((fp = fopen("航班信息p.txt","w+"))==NULL){

printf("航班文件不存在\n");

exit(1);

}

else

{

fprintf(fp, "航班号\t起点\t终点\t飞行日\t起飞时间\t到达时间\t机型\t价格\t乘员定额\t余票量\n");

for (i = 0; i < L.acount; i++)

{

fprintf(fp, "%s\t%s\t%s\t%s\t%s\t%s\t%s\t%d\t%d\t%d \n", L.s[i].Airno, L.s[i].start, L.s[i].end, L.s[i].sche, L.s[i].time1, L.s[i].time2, L.s[i].mode,

&L.s[i].price,&L.s[i].fixed,&L.s[i].extratkt);

}

}

fclose(fp);

}

void refund\_ticket() //用户退票

{

PAirInfo info;

int grade;

PBookList p1,p2=NULL,head;

char cusname[10];

if((info=findByAirno())==0)

return;

head = info->order;

p1=head;

printf("请输入您的姓名：");

scanf("%s",cusname);

while(p1!=NULL)

{

if(strcmp(cusname,p1->name)==0)

break;

p2=p1;

p1=p1->Pnext;

}

if(p1==NULL)

{

printf("对不起，您没有订过票\n");

return;

}

else

{

if(p1==head)

{

head = p1->Pnext;

printf("hello\n");

}

else

{

p2->Pnext=p1->Pnext;

}

info->extratkt+=p1->order\_amount;

grade = p1->grade;

printf("%s成功退票！\n",p1->name);

free(p1);

}

info->order=head;

}

void searchbyendPlace() //通过终点站查询航班信息

{

PAirInfo info,a;

char endplace[10];

int i=0,sel;

info = L.s;

a=L.s;

printf("===============================\n");

printf("== 为您列出所有终点站供您选择： =\n");

int j;

for(j=0;j<L.acount;j++)

{

printf("%d %s\n",j,a[j].end);

}

printf("===============================\n");

printf("请输入要选择航班的序号:");

scanf("%d",&sel);

if(sel>=L.acount)

{

printf("输入错误\n");

return;

}

else

{

strcpy(endplace,info[sel].end);

}

while(i<L.acount)

{

if(strcmp(endplace,a[i].end)==0)

break;

i++;

}

if(i>=L.acount)

{

printf("\n\n");

printf("对8起，该航线未找到\n");

printf("\n\n");

}

else

{

printf("================================================================================\n");

printfOneofAirMsg(i);

}

}

void searchbySche() //通过时间查询航班信息

{

PAirInfo info,a;

char sche[10];

int i=0,sel;

info = L.s;

a=L.s;

printf("===============================\n");

printf("== 为您列出所有飞行日供您选择： =\n");

int j;

for(j=0;j<L.acount;j++)

{

printf("%d %s\n",j,a[j].sche);

}

printf("===============================\n");

printf("请输入要选择飞行日的序号:");

scanf("%d",&sel);

if(sel>=L.acount)

{

printf("输入错误\n");

return;

}

else

{

strcpy(sche,info[sel].sche);

}

while(i<L.acount)

{

if(strcmp(sche,a[i].sche)==0)

break;

i++;

}

if(i>=L.acount)

{

printf("\n\n");

printf("对8起，该航线未找到！\n");

printf("\n\n");

}

else

{

printf("================================================================================\n");

//printf("航班号 起点 终点 飞行日 起飞时间 到达时间 机型 价格 乘员定额 余票量\n");

printfOneofAirMsg(i);

}

}

void meau2() //查询航线信息的子菜单。

{

int subsel;

while(1)

{

printf("============================\n");

printf("= 航线信息查询子菜单 =\n");

printf("============================\n");

printf("\* 1.按航班号查询 \*\n");

printf("\* 2.按起点站查询 \*\n");

printf("\* 3.按终点站查询 \*\n");

printf("\* 4.按时间查询 \*\n");

printf("\* 5.返回主菜单 \*\n");

printf("============================\n");

printf("请 选 择(1-5):\n");

scanf("%d",&subsel);

switch(subsel)

{

case 1:searchbyAirno();break;

case 2:searchbystartPlace();break;

case 3:searchbyendPlace();break;

case 4:searchbySche();break;

case 5:return;

default:printf("输入错误\n");break;

}

}

}

void orderTicket() //用户订票信息的写入

{

PAirInfo info;

int amount;

int grade;

char name[10];

char ID[21];

FILE \*p;

info=L.s;

if((info=findByAirno())==0)

return;

if((p=fopen("订票信息.txt","a+"))==NULL)

{

printf("ERROR!\n");

exit(0);

}

printf("请输入你订票所需要的数量:");

scanf("%d",&amount);

if (amount > info->fixed)/\*若客户订票额超过乘员定票总额，退出\*/

{

printf("\n对不起，您输入的票的数量已经超过乘员定额!");

return;

}

if(amount<=info->extratkt)

{

fprintf(p,"\n");

fputs(info->Airno,p);

fprintf(p," ");

printf("请输入您的姓名: ");

getchar();

gets(name);

fputs(name,p);

fprintf(p," ");

printf("请输入客户的身份证号码: ");

gets(ID);

fputs(ID,p);

printf("请输入%s客户的舱位等级: ", name);

scanf("%d",&grade);

getchar();

fprintf(p," ");

fprintf(p,"%d", amount);

info->order= insertlink(info->order,info->Airno,amount,name,grade,ID);

int i;

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

{

printf("%s的座位号是：%d\n",name,(info->fixed)-(info->extratkt)+i+1);

}

info->extratkt-=amount;

rewrite(L);

printf("\n=========订票成功========\n");

printf("========祝您旅途愉快！======\n");

}

fclose(p);

}

void meau1(void) //航线信息输出子菜单

{

int subsel;

while(1)

{

printf("============================\n");

printf("= 显示各航线信息的子菜单 =\n");

printf("============================\n");

printf("\* 1.显示所有航线 \*\n");

printf("\* 2.显示订单客户 \*\n");

printf("\* 3.返回主菜单 \*\n");

printf("============================\n");

printf("请 选 择(1-3):\n");

scanf("%d",&subsel);

switch(subsel)

{

case 1:printAllofAirMsg();break;

case 2:printOrderCustomByAirNum1();break;

case 3:return;

default:printf("输入错误\n");break;

}

}

}

void newAirMsg() //新航班信息的录入；

{

PAirInfo info;

char Airno[10];

char start[10];

char end[10];

char sche[20];

char time1[10];

char time2[10];

char mode[8];

int price;

int fixed;//乘客定额

int extratkt;

FILE \*p;

info = L.s;

if((p=fopen("航班信息.txt","a+"))==NULL)

{

printf("error!");

exit(0);

}

getchar();

printf("请输入航班的航班号：\n");

gets(Airno);

printf("请输入起始地点：\n");

gets(start);

printf("请输入终点站：\n");

gets(end);

printf("请输入起飞日期：\n");

gets(sche);

printf("请输入起飞时刻：\n");

gets(time1);

printf("请输入落机时刻：\n");

gets(time2);

printf("请输入飞机机型：\n");

gets(mode);

printf("请输入飞机价格：\n");

scanf("%d",&price);

printf("请输入乘客定额：\n");

scanf("%d",&fixed);

printf("请输入票数的余额：\n");

scanf("%d",&extratkt);

getchar();

fprintf(p,"\n");

fputs(Airno,p);

fprintf(p," ");

fputs(start,p);

fprintf(p," ");

fputs(end,p);

fprintf(p," ");

fputs(sche,p);

fprintf(p," ");

fputs(time1,p); fprintf(p," ");

fputs(time2,p); fprintf(p," ");

fputs(mode,p); fprintf(p," ");

fprintf(p,"%d",price); fprintf(p," ");

fprintf(p,"%d",fixed); fprintf(p," ");

fprintf(p,"%d",extratkt);

fclose(p);

}

int main()

{

login\_user();

int sel;

readinfo(&L);

readinfo\_book(&B);

InitOrder();

while (1)

{

printf("\n\n\n\n");

printf("\t欢迎进入航班订票系统\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 航空订票客运系统 \*\n");

printf("\* 1.显示各航线信息 \*\n");

printf("\* 2.查询各航线信息 \*\n");

printf("\* 3.订 票 \*\n");

printf("\* 4.退 票 \*\n");

printf("\* 5.新航班信息录入 \*\n");;

printf("\* 0.退 出 系 统 \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请 选 择(0-5):\n");

scanf("%d", &sel);

switch (sel)

{

case 1:meau1(); break;//选择菜单1

case 2:meau2(); break;//选择菜单2

case 3:orderTicket(); break;//订票

case 4:refund\_ticket(); break;//退票

case 5:newAirMsg();break;//新航线录入

case 0:return;//退出程序

default:printf("输入错误\n"); break;

}

}

return 0;

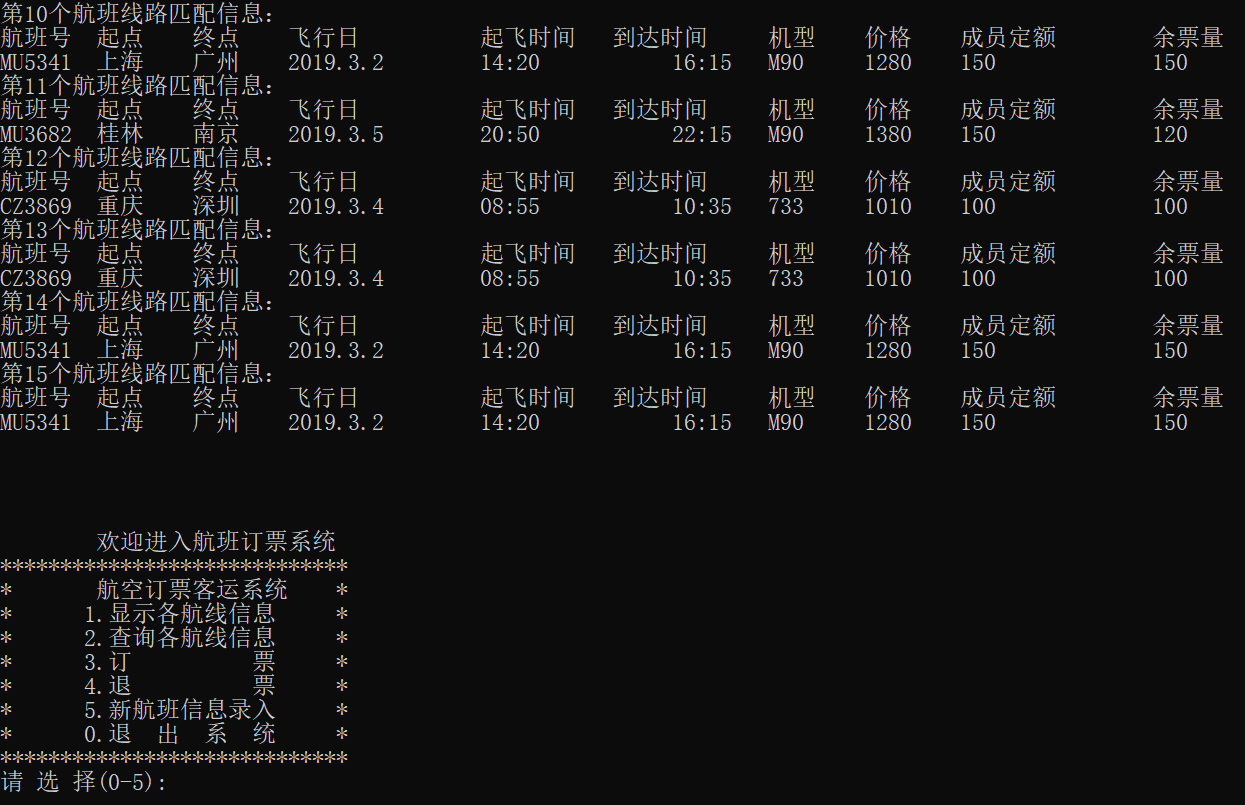
}

#### 六、实验结果

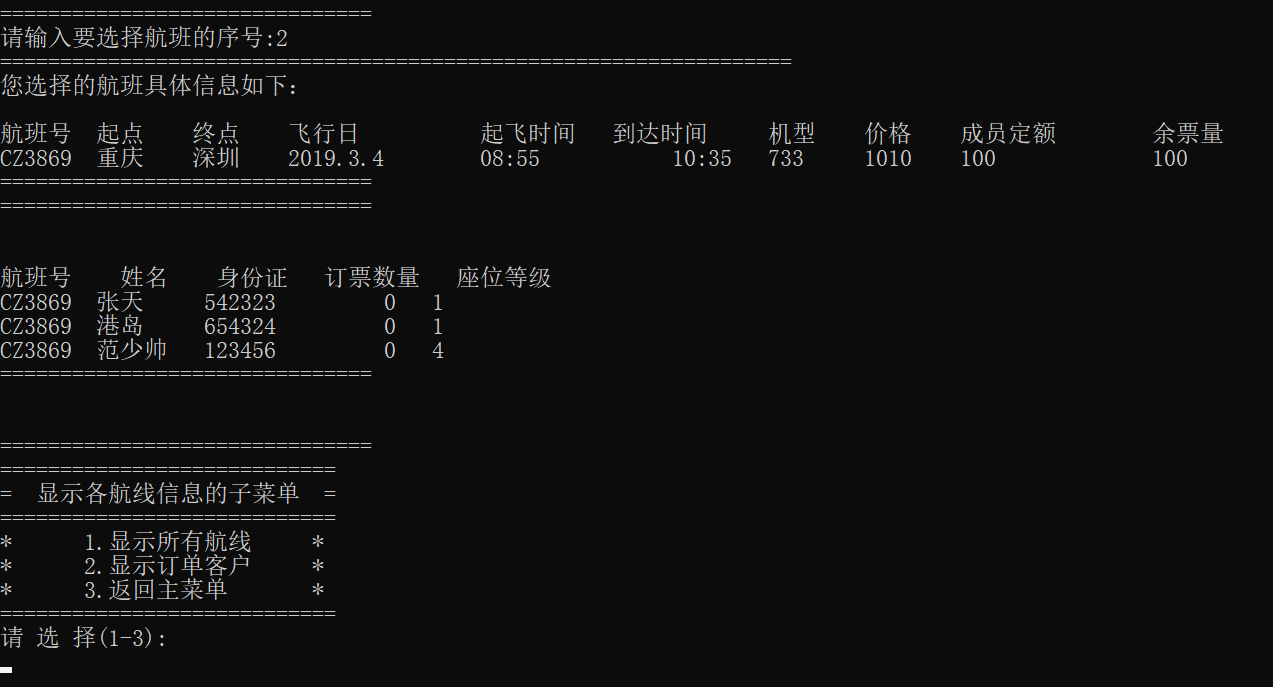
**1．使用说明：**

①按提示的数字进行输入；

程序运行界面如下：





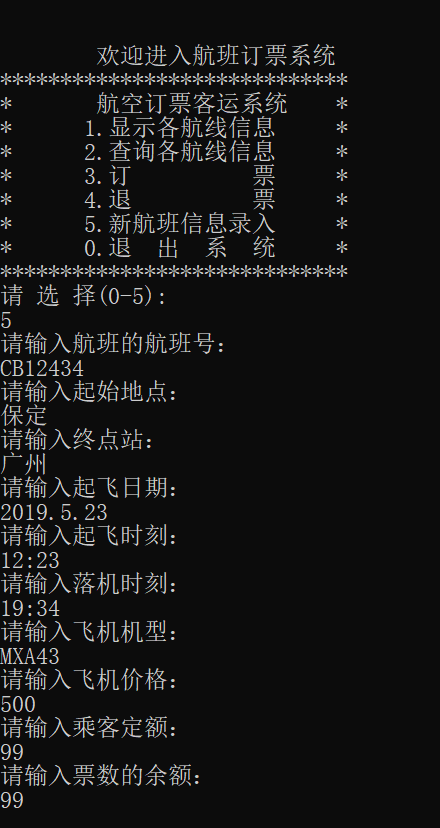
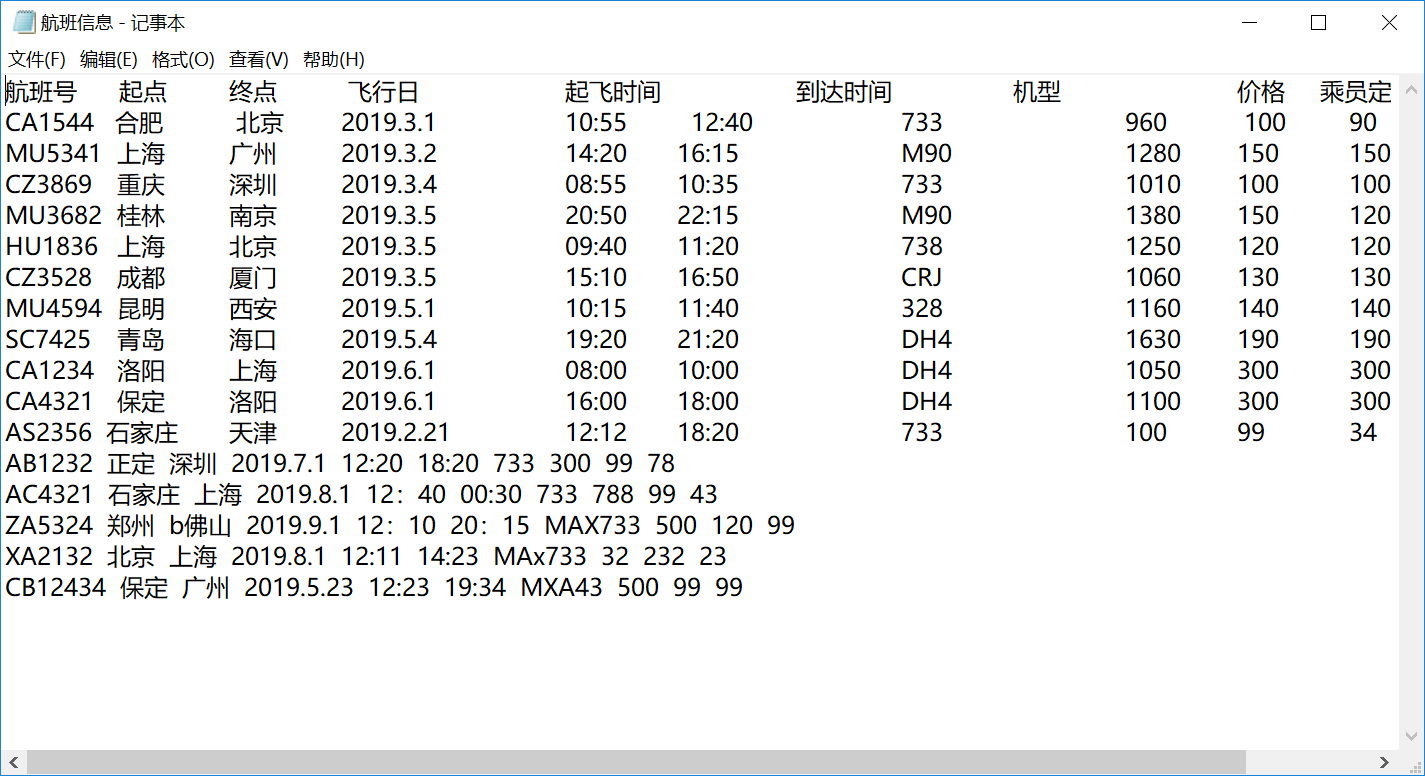


**2．其他测试数据：**

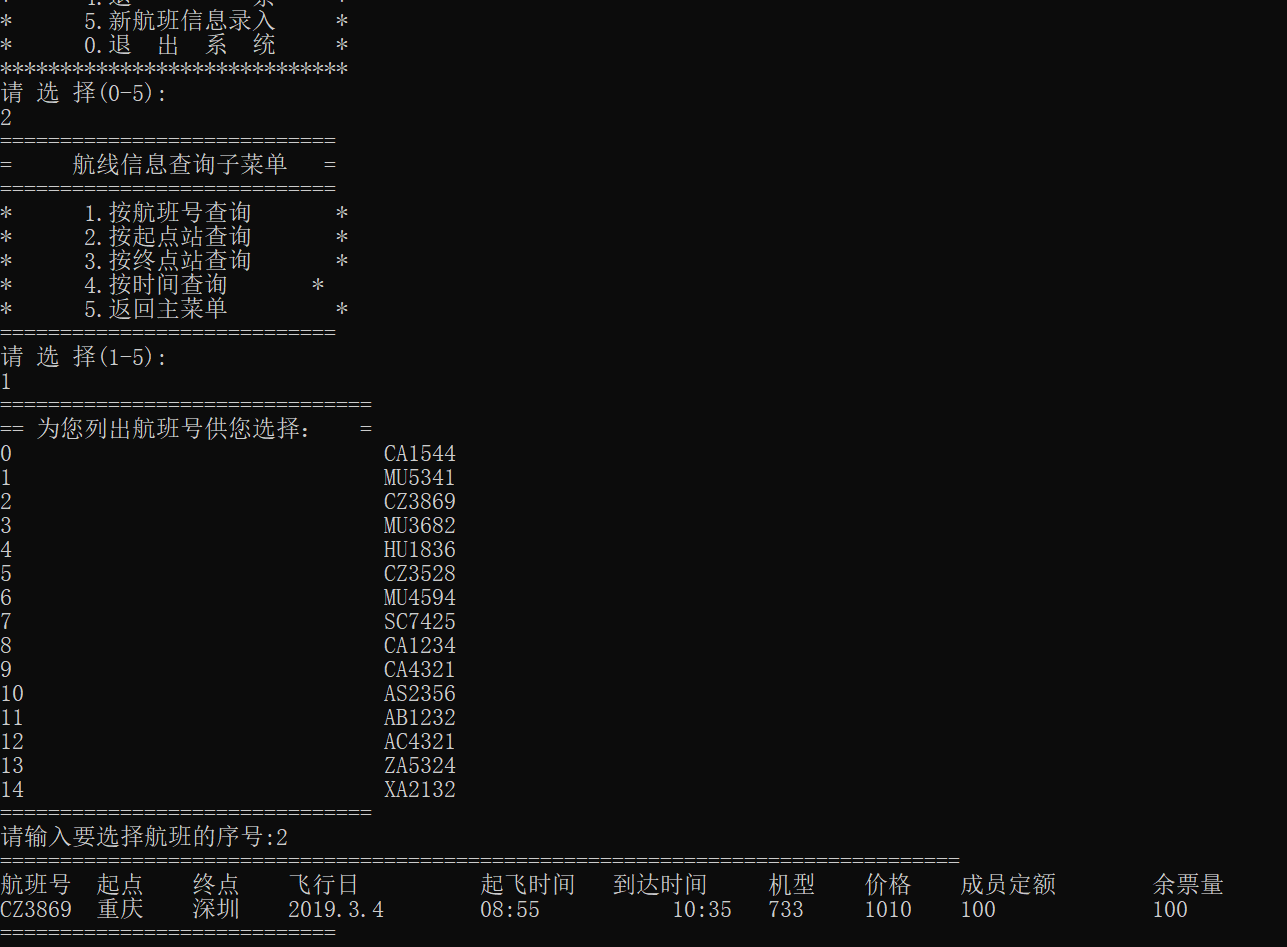
①订票系统：



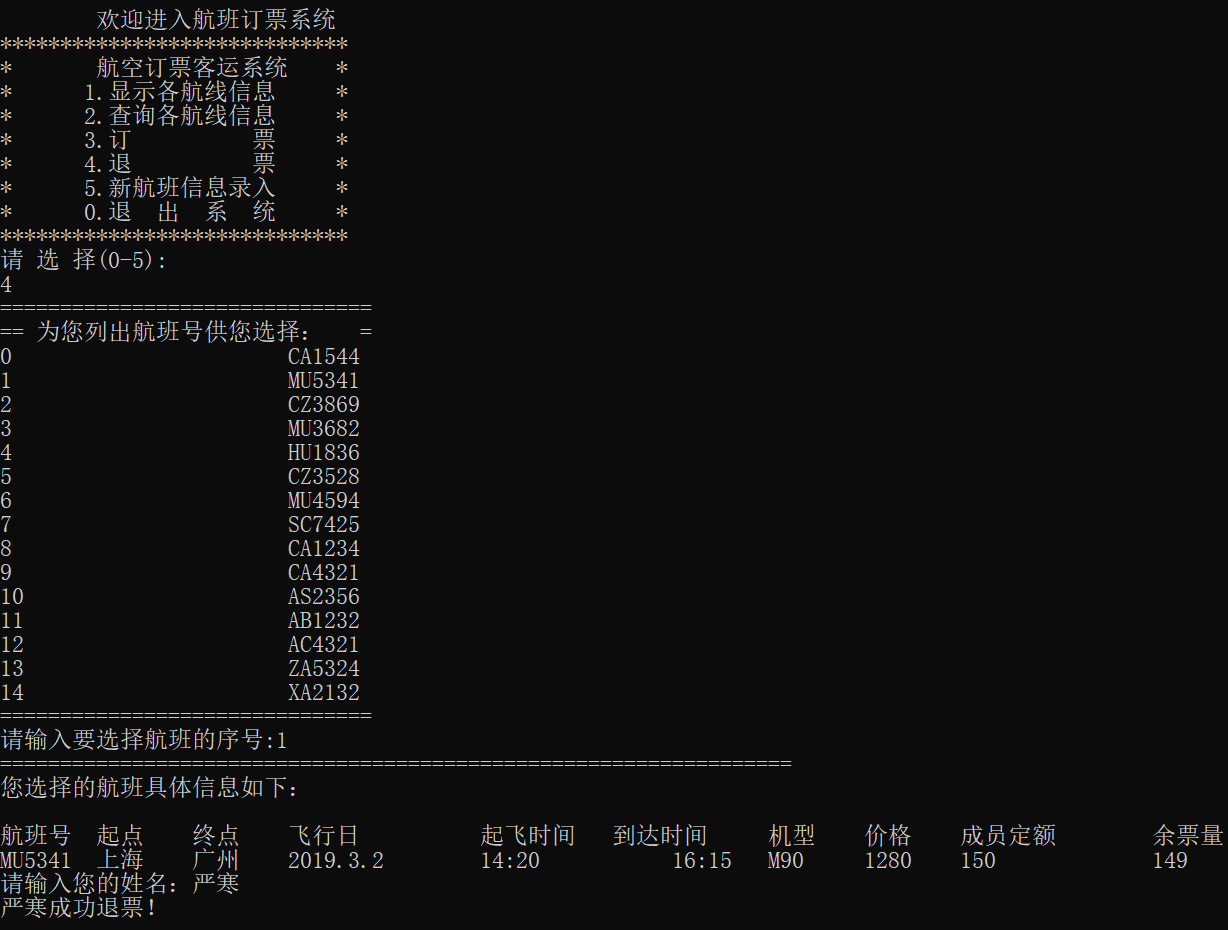
②航班录入模块：



③航班查询板块：



④航班退票板块：



#### 七、遇到的问题及解决方法

**1、遇到的问题：**

①系统代码较长，稍有马虎，容易出现Bug。

②文件的写入时格式不正确，读出时容易出错。

③写入文件信息较多时，稍有不注意，容易漏掉某个信息，导致最终产生叠加效应。

**2、解决方法：**

①在写较大类型的代码时，可以写完一部分代码后进行编译，或在中间部分插入输出语句，进行局部检验。

②在对文件进行操作时，应该注意文件内容的格式，尤其是一些回车、空格之类的分隔符；

③写入文件时应该先对文件信息的内容进行校对，进行统一操作。

其次，在整个系统写完之后应该进行自身查漏，从用户的角度出发，增加代码的健壮性。比如在系统提示输入数字时，应该同时对用户输入字母、文字之类的内容进行排查，防止因为某个错误输入而使整个系统崩溃或返回错误值。