源文件

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include<conio.h>

#define MaxSize 200

#include "SeqList.h"

SeqList queueList;

//ListInitiate(&queueList);

typedef struct queuecli //等候替补客户信息

{

int flinum; //航班号

char name[20]; //客户姓名

char id[20]; //客户ID

int demand; //需求数量

int level; //舱位等级

}QueuecliType;

typedef QueuecliType DataType;

int num1=0; //订票客户人数

int num2=0; //排队客户人数

struct client //订票客户信息

{

int flinum; //航班号

char name[20]; //客户姓名

char id[20]; //客户ID

int tickets; //订票数量

int level; //舱位等级

};

struct client client[1000];

void menu();

void scline();

void order();

void tuipiao();

void sccli();

void scque();

void back();

void scline();

void changescline();

void askfortime();

void tuipiao();

void changepassword();

void menu()

{

//int p;

char c;

printf("\n\n 航空订票系统菜单 \n");

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

printf("\n ◇ 1 查询航线\n");

printf("\n ◇ 2 修改航线\n");

printf("\n ◇ 3 查询订票客户名单\n");

printf("\n ◇ 4 查询等候替补客户名单\n");

printf("\n ◇ 5 订票\n");

printf("\n ◇ 6 退票\n");

printf("\n ◇ 7 修改密码\n");

printf("\n ◇ 8 退出\n");

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

printf("\n\n请选择功能选项（1-8）:[ ]\b\b\b ");

switch(c=getch())

{

case 49: scline();break;

case 50: changescline();break;

case 51: sccli();break;

case 52: scque();break;

case 53: order();break;

case 54: tuipiao();break;

case 55: changepassword();break;

case 56: back();break;

default:printf("\n\n\*\*\*\*\*输入错误！\*\*\*\*\*\n");system("cls");back();

}

}

void scline() //查询航线信息

{

int i;

int city;

printf("\n\n 航空订票系统->航线查询 \n");

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

shuru: printf("\n请选择终点站（1北京 2上海 3南京）:[ ]\b\b\b");

scanf("%d",&city);

if(city>0&&city<4)

{

printf("\n查询结果：\n");

printf("\n终点站：");

switch(city)

{

case 1: printf("北京\n\n");break;

case 2: printf("上海\n\n");break;

case 3: printf("南京\n\n");break;

}

printf("航班号 飞机号 飞机周日 起飞时间 到达时间 余票量\n");

for(i=0;i<8;i++)

{

if(fliinfo[i].city==city)

printf("\n %d %s %13s %s %s %d\n",fliinfo[i].flinum,fliinfo[i].planenum,fliinfo[i].week,fliinfo[i].starttime,fliinfo[i].endtime,fliinfo[i].overplus);

}

}

else

{

printf("\n\*\*\*\*\*输入错误！请重新选择！\*\*\*\*\*\n");

goto shuru;

}

back();

}

void changescline() //修改航线

{

int choice,line\_num;

printf(" \n");

printf("\* 删除航线-------0 \*\n");

printf("\* 增加航线-------1 \*\n");

printf("请选择: ");

choice =(char) getch();

printf("%c",choice);

if(choice=='0')

{

printf(" \n");

printf("请输入你要删除的航班号: \n");

scanf("%s",&line\_num);

printf("删除成功!\n");

}

else if(choice=='1')

{

int plane\_num,end\_place,total;

printf(" \n");

printf("请输入你要增加的航班号: ");

scanf("%s",&line\_num);

printf("请输入飞机号: ");

scanf("%s",&plane\_num);

printf("请输入目的地: ");

scanf("%s",&end\_place);

printf("请输入座位总数: ");

scanf("%d",&total);

printf("增加成功!\n");

}

}

void order() //订票函数

{

int m,n,level,op,levelmax,p,i; //,queueList

printf("\n\n 航空订票系统->订票 \n");

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

shuru: printf("请输入航班号（1-8）:[ ]\b\b\b");

scanf("%d",&m);

if(m>0&&m<9)

{

printf("\n请输入您要订票的数量:[ ]\b\b\b");

scanf("%d",&n);

printf("\n请输入您订票的舱位等级（1-3）:[ ]\b\b\b");

scanf("%d",&level);

switch(level)

{

case 1:{levelmax=fliinfo[m-1].levelone;op=fliinfo[m-1].opone;}break;

case 2:{levelmax=fliinfo[m-1].leveltwo;op=fliinfo[m-1].optwo;}break;

case 3:{levelmax=fliinfo[m-1].levelthree;op=fliinfo[m-1].opthree;}break;

}

if(n<=op)

{

client[num1].flinum=m; //登记订票客户信息

client[num1].tickets=n;

client[num1].level=level;

printf("\n请输入您的姓名： ");

scanf("%s",&client[num1].name);

printf("\n请输入您的身份证号码： ");

scanf("%s",&client[num1].id);

num1=num1+1;

printf("\n\n订票成功！\n\n");

printf("您的预订票舱位等级：%d\n",level);

printf("\n您的座位号： ");

for(i=1;i<=n;i++)

printf("%d ",(levelmax-op)+i);

switch(level)

{

case 1: fliinfo[m-1].opone=fliinfo[m-1].opone-n;break;

case 2: fliinfo[m-1].optwo=fliinfo[m-1].optwo-n;break;

case 3: fliinfo[m-1].opthree=fliinfo[m-1].opthree-n;break;

}

fliinfo[m-1].overplus=fliinfo[m-1].opone+fliinfo[m-1].optwo+fliinfo[m-1].opthree;

}

else

{

printf("\n\*\*\*\*\*对不起！余票量少于您的订票额！\*\*\*\*\*\n\n");

printf("是否登记排队候补（1 是 2 否 ）:[ ]\b\b\b");

scanf("%d",&p);

}

back();

}

else

{

printf("\n\*\*\*\*\*输入错误！请重新选择！\*\*\*\*\*\n");

goto shuru;

}

}

void tuipiao() //退票函数

{

char a[20],b[20];

int c,p,t; //opl,i,j,

int flag=0; //查看是否有退票人记录的标志

int flag2=0; //查看是否有等候替补客户预定已退票的标志

//QueuecliType k;

printf("\n 航空订票系统->退票 \n");

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

printf("\n请输入您的姓名： ");

scanf("%s",&a);

printf("\n\n请输入您的身份证号码： ");

scanf("%s",&b);

printf("\n\n请输入您要退票的航班号（1-8）:[ ]\b\b\b");

scanf("%d",&c);

printf("\n\n您的订票信息：\n");

printf("\n航班号 订票量 舱位等级\n");

for(t=0;t<num1;t++)

if(c==client[t].flinum&&strcmp(b,client[t].id)==0)

{

printf("%d %d %d\n",c,client[t].tickets,client[t].level);

flag=1;

}

if(flag==0)

{

printf("\n\n没有您的订票记录！\n");

back();

}

else printf("\n\n您确定要退票？（1是 2否）:[ ]\b\b\b");

scanf("%d",&p);

if(p==1)

{

printf("\n\n★★★★★退票成功！★★★★★\n\n");

}

if(p==2)

{

printf("已取消退票\n");

back();

}

}

void changepassword() //修改密码

{

int re\_pass\_1,re\_pass\_2;

char pass;

printf("注意:密码不能超过8位!\n");

printf("请输入新密码：");

scanf("%s",&re\_pass\_1);

printf("请再输入一次: ");

scanf("%s",&re\_pass\_2);

if(re\_pass\_1==re\_pass\_2)

{

pass=re\_pass\_1;

printf("密码修改成功!请记住.\n");

}

else

{

printf("你两次输入的密码不一致!\n");

}

}

void sccli() //查询订票客户名单

{

int p,i;

int flag=0;

printf("\n\n\n 航空订票系统->查询订票客户名单 \n");

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

printf("\n请输入要查询的航班号（1-8）:[ ]\b\b\b");

scanf("%d",&p);

if(p>0&&p<9)

{

printf("\n\n航班号：%d\n\n",p);

printf("订票客户名单：\n");

printf("\n姓名 订票量 舱位等级\n");

for(i=0;i<num1;i++)

if(p==client[i].flinum)

{

printf("%s %d %d\n",client[i].name,client[i].tickets,client[i].level);

flag=1;

}

if(flag==0)

printf("\n\*\*\*\*\*无记录！\*\*\*\*\*\n");

}

else printf("\n\n\*\*\*\*\*输入航班号错误！\*\*\*\*\*\n\n");

back();

}

void scque() //查询等候替补客户名单

{

int p,i;

int flag=0;

printf("\n\n\n 航空订票系统->查询排队客户名单 \n");

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

printf("\n\n请输入要查询的航班号（1-8）:[ ]\b\b\b");

scanf("%d",&p);

if(p>0&&p<9)

{

printf("\n\n航班号：%d\n\n",p);

printf("等候替补客户名单：\n");

printf("\n姓名 需求量 舱位等级\n");

for(i=0;i<num2;i++)

if(p==queueList.list[i].flinum)

{

printf("%s %d %d\n",queueList.list[i].name,queueList.list[i].demand,queueList.list[i].level);

flag=1;

}

if(flag==0)

printf("\n\*\*\*\*\*无记录！\*\*\*\*\*\n");

}

else printf("\n\n\*\*\*\*\*输入航班号错误！\*\*\*\*\*\n\n");

back();

}

void back()

{

printf("\n\n《《《《《《任意键返回菜单页！》》》》》》\n");

getch();

system("cls");

menu();

}

void main()

{

printf("\n 欢迎进入航空订票系统！ \n");

menu();

}

头文件SeqList.h

//#include<stdio.h>

//#include<stdlib.h>

struct fliinfo //航班信息

{

int city;

int flinum;

char planenum[10]; //飞机型号

char week[10]; //星期几飞行

int maxnum; //成员定额

int overplus; //余票量

int levelone; //一等舱座位数量

int opone; //一等舱余票量

int leveltwo; //二等舱座位数量

int optwo; //二等舱余票量

int levelthree; //三等舱座位数量

int opthree; //三等舱余票量

char starttime[7]; //起飞时间

char endtime[7]; //到达时间

}fliinfo[8]={{1,1,"737","MONDAY",120,120,10,10,20,20,90,90,"09:50","12:15"},{1,2,"K320","WEDNESDAY",120,120,10,10,20,20,90,90,"15:00","17:30"},{1,3,"737","FRIDAY",120,120,10,10,20,20,90,90,"13:25","15:55"},{2,4,"757","TUESDAY",120,120,10,10,20,20,90,90,"10:15","11:45"},{2,5,"737","SATURDAY",120,120,10,10,20,20,90,90,"14:45","16:00"},{2,6,"K320","THURSDAY",120,120,10,10,20,20,90,90,"20:10","21:30"},{3,7,"737","WEDNESDAY",120,120,10,10,20,20,90,90,"11:30","12:30"},{3,8,"737","SUNDAY",120,120,10,10,20,20,90,90,"17:25","18:30"}};

struct wait

{

char name[10];

int demand[20];

int level[5];

int flinum;

};

typedef struct

{

struct wait list[MaxSize];

int size;

}SeqList;

void ListInitiate(SeqList \*L)

{

L->size=0;

}

int ListLength(SeqList L)

{

return L.size;

}

int ListInsert(SeqList \*L,int i,struct wait x)

{

int j;

if(L->size>=MaxSize)

{

printf("顺序表已满无法插入！\n");

return 0;

}

else if(i<0||i>L->size)

{

printf("参数i不合法！\n");

return 0;

}

else

{

for(j=L->size;j>i;j--)L->list[j]=L->list[j-1];

L->list[i]=x;

L->size++;

return 1;

}

}

int ListDelete(SeqList \*L,int i,struct wait \*x)

{

int j;

if(L->size<=0)

{

printf("顺序表已无数据可删！\n");

return 0;

}

else if(i<0||i>L->size-1)

{

printf("参数i不合法！\n");

return 0;

}

else

{

\*x=L->list[i];

for(j=i+1;j<=L->size-1;j++)L->list[j-1]=L->list[j];

L->size--;

return 1;

}

}