

摘 要

飞机在现代得生活中扮演者非常重要得角色。它能够快速得把人们送到自己想要去得地方,既快速,又方便。所以现在坐飞机时很普遍得.但就是都到机场去买票浪费时间,因此,航空订票系统应运而生。有了航空订票系统,用户可以在该系统进行飞机票得查询,订票,退票等操作。方便了大家

关键词 航空订票系统 、查询、订票、退票

目录

摘 要	I
1 需求分析	2
1、1 需求概述	2
1、2 需求环境	2
1、3 功能描述	2
2 概要设计	3
2、1 程序功能模块	3
2、2 程序流程图	3
2、3 课程设计得思想	3
3 详细设计	4
3、1 程序初始化	4
3、1、1 代码功能	4
3、1、2 功能实现代码	4
3、2 查询航班信息	6
3、2、1 代码功能	6
3、2、3 功能实现代码	7
3、3 订票模块	8
3、3、1 代码功能	8
3、3、2 功能实现代码	8
4 测试与运行	14
5 结 束 语	16
6 参 考 文 献	17
7 附 录	18

1 需求分析

1、1 需求概述

航线管理。每条航线所涉及得信息有:终点站名、航班号、飞机号、飞行周日(星期几)、乘员定额、余票量

客户管理。有关订票得客房信息(包括姓名、订票量、舱位等级(1, 2 与 3))以及等替补得客房名单(包括姓名、所需标量)。

系统实现主要操作操作与功能。系统实现得主要操作规程与功能如下:

- ① 查询航线。根据旅客提出得终点站名输出下列信息: 航班号、飞机号、飞行日期、余标额。

- ① 承办订票业务. 根据客户提出得要求（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; 若已满或者余票少于订票额, 则需要重新询问客户要求; 若需要, 可等待排队候补
- ② 承办退票业务。根据客户提供得情况（日期、航班）, 为客户输退票手续, 然后查询该航班就是否有人排队候补, 首先询问排在第一得客户, 若所退票额满足该客户得要求, 则为其订票手续, 否则依次询问其她排队候补得客户。

1、2 需求环境

本课程设计需要得设备为硬件要求与软件配置要求具体要求如下:

- ①硬件要求: 一台计算机。
- ②软件配置: WINDOWS 7 / VC++ 6.0。

1、3 功能描述

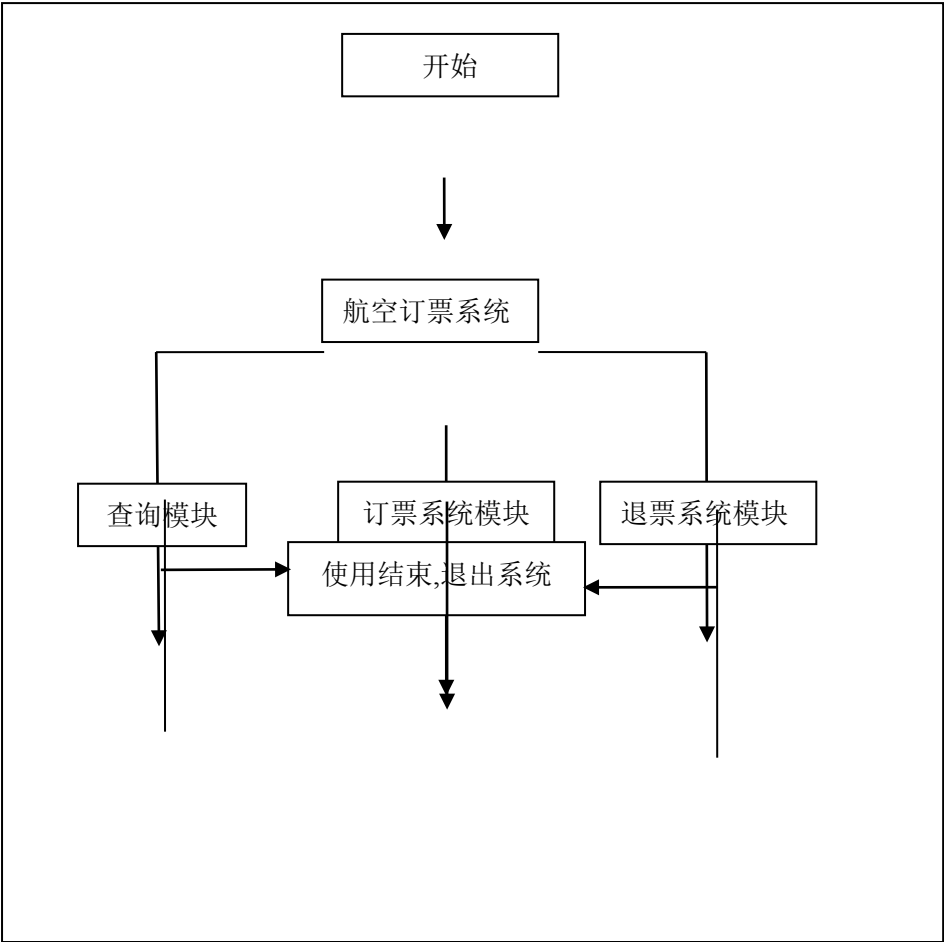
本次课程设计就是航空订票系统主要就是由查询航班信息, 订票还有退票三大主要功能。除此之外, 还有系统得界面等等。

2 概要设计

2、1 程序功能模块

由需求分析知，本次课程设计就是航空订票系统其主要由三大模块构成，即查询模块（这部分由我负责），订票系统模块与退票系统模块构成。

2、2 程序流程图



该程序开始
进
用
信息得
与退票

这两个结构体分别用
息，比如顾客得姓名，
大增强。

3 详细设计

3、1 程序初始化

3、1、1 代码功能

这一块还是由我负责。我初始化了五个航班得基本信息。在初始化程序中，主要就是航班得基本信息。比如航班号，出发时间与剩下得机票数目等等基本信息。通过运用指针与链表来进行对初始化信息得书写。在初始化时将三个重点得信息分别初始化为：

①广州	航班号 005	飞机号 PZH113	机票数 120
②天津	航班号 004	飞机号 PZH121	机票数 120
③北京	航班号 003	飞机号 PZH133	机票数 120
④上海	航班号 002	飞机号 PZH132	机票数 120
⑤成都	航班号 001	飞机号 PZH122	机票数 120

3、1、2 功能实现代码

```
void InitLine() //初始化航线信息
{
    airline *p,*q; // 航线
    L=(airline *) malloc(sizeof(airline));
    L->booked=NULL;
    L->wait=NULL;
    L->next=NULL;
    strcpy(L->end_addr, "00000000"); // 终点站
    strcpy(L->line_num, "000"); //航班号
    strcpy(L->plant_num, "0000000"); //飞机号
    L->day=L->left=L->total=0;
    q=L;
    p=(airline *) malloc(sizeof(airline)); //广州
    p->booked=NULL;
    p->wait=NULL;
    strcpy(p->end_addr, "广州");
    strcpy(p->line_num, "005");
    strcpy(p->plant_num, "PZH113");
```

```

p->day=5;
p->left=120;
p->total=120;
p->next=q->next;
q->next=p;
p=(airline *)malloc(sizeof(airline)); // 天津
p->booked=NULL;
p->wait=NULL;
strcpy(p->end_addr,"天津");
strcpy(p->line_num,"004");
strcpy(p->plant_num,"PZH121");
p->day=4;
p->left=120;
p->total=120;
p->next=q->next;
q->next=p;
p=(airline *)malloc(sizeof(airline)); // 北京
p->booked=NULL;
p->wait=NULL;
strcpy(p->end_addr,"北京");
strcpy(p->line_num,"003");
strcpy(p->plant_num,"PZH133");
p->day=1;
p->left=120;
p->total=120;
p->next=q->next;
q->next=p;
p=(airline *)malloc(sizeof(airline)); // 上海
p->booked=NULL;
p->wait=NULL;
strcpy(p->end_addr,"上海");
strcpy(p->line_num,"002");
strcpy(p->plant_num,"PZH132");
p->day=2;

```

```

p->left = 120;
p->total=120;
p->next=q->next;
q->next=p;
    p = (airline *) malloc (sizeof (airline)); //成都
p->booked=NULL;
p->wait=NULL;
strcpy (p->end_addr, "成都");
strcpy(p->line__num, "001");
strcpy (p->plant__num, "PZH122");
p->day=3;
p->left=120;
p->total=120;
p->next=q->next;
q->next=p;
}

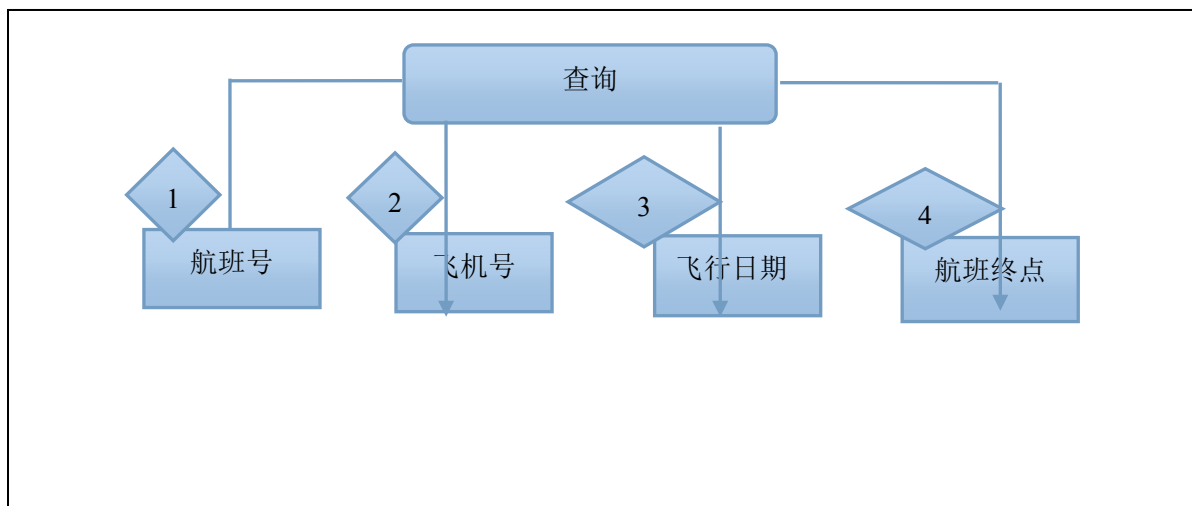
```

3、2 查询航班信息

3、2、1 代码功能

此模块主要由我负责，在此模块我实现了查询航班的功能。当进入查询功能时，通过输入数字用户可以查询航班得基本信息，比如目的地，剩下得机票数目，出发得时间等等基本信息。如果输入得数字在界面中没有显示则会提示用户没有这次航班。在这个模块我使用了个 `switch` 语句来实现对查询方式得选择。通过判断输入得数字进入到相对应得查询方式。如果输入得信息有误得话，系统将会提示用户没有这次航班，需要重新输入。

3、2、2 查询模块流程图



3、2、3 功能实现代码

```
void search ()                                //航班查询系统
{
    int i;
    int flag=0;
    airline *p;
    PR ("请输入查询航班的方式: \n");
    PR ("航班号---->1\n");
    PR ("飞机号---->2\n");
    PR ("飞行日期-->3\n");
    PR ("航班终点-->4\n");
    SC ("%d", &i);
    for (;!flag;)
        switch(i)                            //switch 语句来进行航班得查询
        方式
        {
            case 1: p=search_line();flag=1; break;
            case 2: p=search_plane(); flag=1;break;
            case 3: p=search_day();flag=1;break;
            case 4: p=search_addr ();flag=1; break;
            default:PR (" 操作错误, 请重新输入\n"); break;
        }
    if(p)
    {
        PR (" 终点站                航班号    飞机号    飞行周日
        余票量\n");
        PR ("%20s%10s%10s%8d%8d\n", p->end_addr, p->line_num, p->
        plane_num, p->day, p->left);
        PR (" 就是否订票(确定请按1) " );
        SC ("%d", &i);
        if(i==1)
            book ();
    }
    else
```

```
PR(" 没有该航班\n");}
```

3、3 订票模块

3、3、1 代码功能

在此模块可以根据提示可以从航班号，飞行日期与终点站来进行订票。

如果订票时余票不足所需要得票得数目时就可以进行排队购票选择。

3、3、2 功能实现代码

```
void book()•    // 订票系统
{
    int i;
    int flag=0;
    airline *p;
    customer *cst, *c;
    PR("请输入查询航班得方式: \n");•
    PR(" 航班号——> 1\n");
    PR("终点站——> 2\n");
    PR(" 飞行日期——> 3\n");
    for(;;! flag;)
    {
        SC("%d", &i);
        switch(i)                                //运用一个 switch 语句进行航班查询
        {
            case 1: p=search_line(); flag=1; break;
            case 2: p=search_addr(); flag=1; break;
            case 3: p=search_day(); flag=1; break;
            default: PR("操作错误, 请重新输入\n"); break;
        }
    }
    if(p)
    {
        PR(" 终点站                航班号    飞机号    飞行周日
        余票量\n");
        PR("%-20s%10s%10s%8d%8d\n", p->end_addr, p->line__
        num, p->plant__num, p->day, p->left);
        cst=(customer *) malloc(sizeof(customer));
        PR("请输入订票数量:");
        SC("%d", &cst->num);
        PR("请输入舱位等级 (1/2/3)");
        SC("%d", &cst->level);
        PR("请输入您得名字");
        SC("%s", cst->name);
```

```

    if(cst->num<=p->left)
    {
        cst->next=p->booked;
        p->booked=cst;
        p->left=p->left-cst->num;
        PR("订票成功, 座位号为%d~%d\n", p->total-p->left-
cst->num+1, p->total-p->left);
    }
    else
// 当需要得票数大于剩余得票数时进行询问是否进行排队购票
    {
        PR("余票不足, 是否排队等候\n 是 (y) \n 否 (n) \n");
        SC("%d", &flag);
        if(flag)
        {
            if(!p->wait)
            {
                cst->next=p->wait;
                p->wait=cst;
            }
            else
            {
                c=p->wait;
                for(; c->next; c=c->next);
                cst->next=c->next;
                c->next=cst;
            }
        }
    }
    else
        PR("没有该航班\n");
guide();
}

```

排队订票时得排队订票系统:

```

void waited(airline *p) //排队订票系统
{
    int flag;
    customer *q, *q0, *h;
    if(!p->wait) return;
    q=(customer *)malloc(sizeof(customer));
    q->level=0;
    q->num=123;
    strcpy(q->name, "00000");
}

```

```

    q->next=p->wait;
    h=q;
    for (; p->left>0 && q->next;)
    {
        if (q->next->num < p->left)
        {
            PR(" %s, 您正在排队定%d张票, 现有票, 就是否定\n 就是 (y)\n 否 (n)\n ", q->next->name, q->next->num);
            SC ("%d", &flag);
            if (flag)
            {
                q0=q->next;
                q->next=q->next->next;
                q0->next=p->booked;
                p->booked=q0;
                p->left=p->left-q0->num;
                PR(" 排队订票成功\n\n ");
            }
            else
            {
                q=q->next;
            }
        }
    }
    p->wait=h->next;
}

```

3、4 退票系统模块

3、4、1 代码功能

在此模块, 用户可以根据系统得提示进行退票操作. 当进行退票操作时需要进行信息得核对, 如果正确, 打印出信息, 再询问就是否退票。

3、4、2 功能实现代码

```

void refund () // 退票系统
{
    int flag;
    char a[20];
    airline *p;
    customer *c, *cl;
    p=search_line ();
    c=p->booked;
    if(!p)
    {

```

```

    PR ( " 没有该航班\n\n" );
    guide();
return;
}
PR ( " 请输入您的姓名");
SC ( " %s", a);
if ( ! p-> booked-> name)          //进行信息的查找判断是否有用户的信息
{
    PR ( " 对不起, 没有找到您的信息\n\n" );
    guide ();
    return ;
}
if ( ! strcmp (a, p-> booked-> name)) //核对完信息后输出要退票乘客的信息
{
    PR ( "退票信息\n姓名/终点站/票数\n%s/%s / %d\n", p-> booked->
name, p-> end_addr, p-> booked-> num);
    PR ( " 确认要退票\n就是 (y)\n否 (n) \n" );
    scanf ( " %d", &flag);
    if(flag)
    {
        p-> booked=p-> booked->next;
        p->left=p-> left+c-> num;
        free (c);
        PR("退票成功\n\n" );
        waited ( p );
        guide();
    }
}
else
{
    for( ; c->next; c=c->next)
        if ( ! strcmp (a, c-> next->name))
            break;
    if ( ! c-> next )
    {
        PR ( " 对不起, 没有找到您的信息\n\n" );
        guide();
        return ;
    }
    else
    {

```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。

如要下载或阅读全文，请访问：

<https://d.book118.com/778117052100006055>