

数据构造与算法设计课程设计

【设计题目】 航空公司订票系统

【问题描述】

编写一个咸阳航空公司订票大厅的一个订票系统，可以在系统中录入并修改各个航班的具体情况，帮助乘客准确的查询各航班的最新信息(包括航班号、航班的起始时间和地点、票价)和各项业务的及时办理〔包括订票、退票等〕。该系统用可以用 C++ 的类编写。

【软件功能】

- 1.输入航班信息并保存到磁盘
- 2.浏览航班信息
- 3.修改航班信息并保存到磁盘
- 4.查找航班
- 5.删除航班信息
- 6.订票
- 7.退票

【算法思想】

1 先定义一个乘客根本信息的类包含乘客的姓名、证件号、订票数量及所乘的航班号〔注：每个航班有固定的人数限制，假设满额则提示其他可供选择的航班。

2 建立一个功能类，包括对航班信息和乘客订票信息的录入及修改的函数，其中订票信息采用单链表的顺序存储方式存储。另外还包括对信息的查询和修改的功能函数，乘客退票的函数。并将最终的改动结果通过 i/o 流写入磁盘文件保存。

3 编写主函数，对编写的航班系统进展全面测试，对用户界面采用人性化的菜单方式。

【类的设计】

定义一个航班的类

```
class plane
{
    friend class customer;//友元类

public:
    void input();    //插入航班信息
    void print();    //打印航班信息
    void save();     //保存航班信息
    void read();     //读取航班信息
    void search();   //查找航班信息
    void Delete();  //删除航班信息
    void book();    //订票
    void back();   //退票
    void *iugai(); //修改航班信息

private:
    int num;//航班号
    char start_place[20];//起点站
    char arrive_place[20];//终点站
    char time[10];//起降时间
```

```

        int count;//机票数量
    };

    定义一个顾客信息的类

    class customer
    {
        friend class plane;//友元类

    protected:

        Int id;//身份证号

        char name[10];//

        int customer_num;//订票数量

    public:

        void read1();//读取数据

        void save1();//保存数据

    };

```

【存储构造设计】

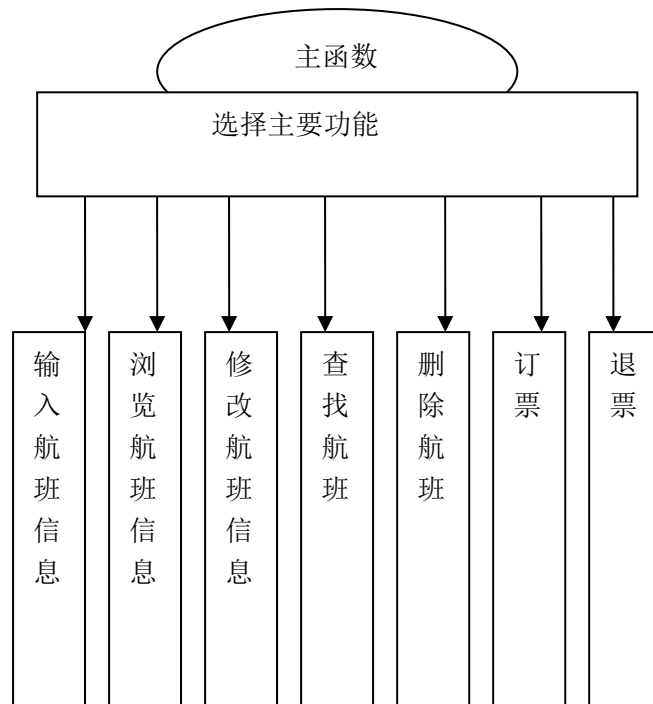
采用顺序存储构造，利用动态申请空间实现信息的存储。

【模块划分及调用关系】

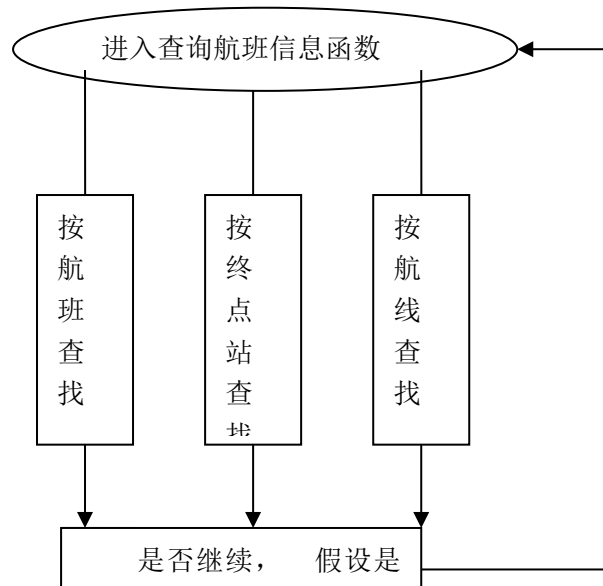
定义了两个类 class plane 和 class customer，分别实现顾客及航班的信息操作。在两个类的后面分别定义了类中的各个函数。另外在主函数中分别调用函数，实现各个功能。

【模块流程图】

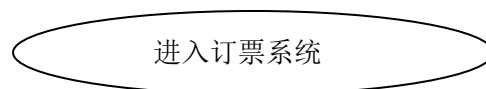
1.主函数 main()



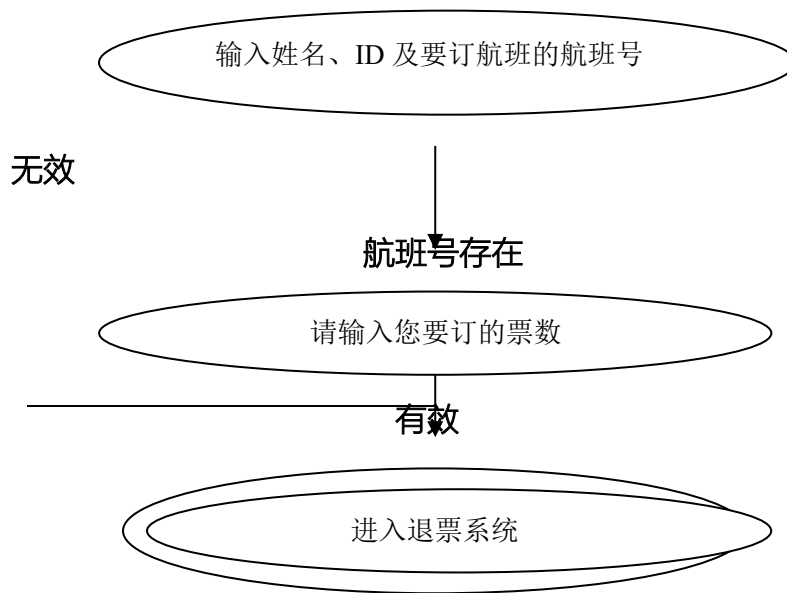
2. void search()//查询函数



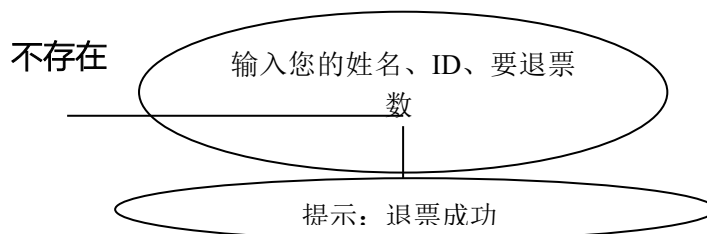
3. void book()//订票函数



航班不存在



4. void back()//退票函数



【界面设计】

采用人性化的菜单提示语，让用户很容易的操作；

【用户手册】

用户只需要在vc环境下就可以操作,具体步骤可见菜单提示就可以顺利操作。

另外，再参加、删除、修改航班信息时必须以管理员的身份进去，所以必须

要有密码，在这里初始密码是：*in*i0701.

感谢您的使用！！

1、程序上机调试报告

【语法错误及其排除】

两个类的公有函数中，在循环嵌套调用时总是受阻，不能执行。所以我把两

个类都定义为相互的有元类。这样问题就解决了。

【算法错误及其排除】

在订票与退票函数中，要核对用户信息如姓名、ID 等，开场是自己定义一个函数进展字符间的比拟，在调用时总是出现语法错误，最后想到可以用一个函数库中的函数 `strcmp(char,char)`.把问题简单化，而且语法错误也解决了。

2、程序测试结果

【测试数据】

说明：测试数据都是输入数据，见输出结果。

【输出结果】

【程序性能评价】

该程序可以查询航班的信息、修改信息，查询顾客的信息，订票、退票等，界面采用人性化的提示，可以让用户很容易的操作。另外在修改信息、添加航班信息时必须要以管理员的身份进入即必须要有密码才能进入。保证了系统的平安。

【性能改良方向】

该系统在操作中要是不按照菜单的有关提示操作，而进展非法操作，可能会导致系统的死循环。在这点该系统还有待改良。

【收获及体会】

事实总比想象的难，刚开场编写这个程序的时候思路比拟清晰。可是编写的时候遇到了很多难题，譬如将顾客的信息及航班的信息保存到文件中，不知道用那种方式更简便，最后仔细思考，阅读课本资料，选择了用流的形式输出到文件中，感觉用起来特别方便简洁。完成了程序的编写后又对该程序进展了大量的修改，调试，使程序更加简明，并添加了不少提示用户操作的语句，使用户使用

该系统时更加方便。对自己的这个程序还是比拟满意的，可以说已经超出了自己当初的设想。从编写这个程序的过程中不但发现了许多自己没学明白的地方，而且锻炼了自己的耐心。刚调试时有很多错误，我逐个仔细的修改、排除，最终成功的完成了调试。进过这次实习，我更加深刻的体会到，作任何事情要仔细，要静下心来认真思考，只要努力了，必定会有回报。

3、源程序代码

```
#include<iostream.h>

#include<string.h>

#include<stdio.h>

#include<iomanip.h>

#include<stdlib.h>

#define N 1000 //宏定义

class customer;//申明顾客信息类

//-----定义航班信息类-----

class plane

{

    friend class customer;//有元类

public:

    void insert();    //参加航班信息

    void print();    //输出航班信息

    void save();    //保存航班信息

    void read();    //读取航班信息
```

```

void search();    //查找航班信息

void Delete();   //删除航班信息

void book(); //订票

void back();  //退票

void *iugai();   //修改航班信息

private:

    int num;//航班号

    char start[10];//起点站

    char arrive[10];//终点站

    char time[20];//起降时间

    int plane_num;//机票数量

}s[N];

//-----定义顾客信息类-----

class customer

{

    friend class plane;//有元类

protected:

    int id;//身份证号

    char name[20];//

    int pshu;//用户订票数量

public:

    void read1();//读取数据

```

```

        void save1();//保存数据

    }yh[N];

    int i,j;//全局变量

    int m=0,y=0;

    //宏定义一个输出

    #define PRINT

    s[i].num<<setw(13)<<s[i].start<<setw(13)<<s[i].arrive<<setw(13)<<s[i].time<<setw(13)<

    <s[i].plane_num<<endl    //定义输出格式

    //-----主函数-----

    void main()

    {

        plane abc;//定义一个航班类的对象

        int j;

        do

        {

            cout<<"          管理员密码: *in*i0701          "<<endl;

            cout<<"          "<<endl;

            cout<<"  -----欢迎进入航空订票菜单-----"<<endl; //主界面

            cout<<"|=====|"<<endl;

            endl;

            cout<<"  |          1.新增航班          |"<<endl;

```

```

        cout<<"    |          2.浏览航班信息          |"<<endl;

        cout<<"    |          3.修改航班信息          |"<<endl;

        cout<<"    |          4.查找航班信息          |"<<endl;

        cout<<"    |          5.删除航班            |"<<endl;

        cout<<"    |          6.订票                |"<<endl;

        cout<<"    |          7.退票                |"<<endl;

        cout<<"    |          0.退出                |"<<endl;

        cout<<"

|=====|"<<endl;

        cout<<"请选择(0~7):";

        cin>>j;

        switch(j)

        {

        case 1:abc.insert();//调用插入函数

                break;

        case 2:abc.print();//调用打印函数

                break;

        case 3:abc.*iugai();//调用修改函数

                break;

        case 4:abc.search();//调用查找函数

                break;

        case 5:abc.Delete(); //调用删除函数

```

```

        break;

    case 6:abc.book();//调用订票函数

        break;

    case 7:abc.back();//调用退票函数

        break;

    case 0:;

        break;

}

}while(j!=0);    //判断完毕

cout<<"谢谢使用，再见！ \n";

}

//-----主函数完毕-----

//定义类中各个函数

void plane::insert();//打印函数

{

    char f[]="*in*i701";    //设置密码

    char y;

    cout<<"请输入密码:";

    cin>>f;    //读取密码

    if(strcmp(f,"*in*i0701")==0)

    {

```

```

cout<<"-----\n";

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

{

    cout<<"航班号    出发地点    到达地点    起降时间    满载人数"<<endl;

    cout<<"|-----|"<<endl;

    cin>>s[i].num>>s[i].start>>s[i].arrive>>s[i].time>>s[i].plane_num;

    cout<<"|-----|"<<endl;

    m++;

    cout<<"第"<<m<<"个信息已经输入，是否继续(y/n)"";

    cin>>y;

    cout<<endl;

    if(y=='y')

        continue;

    else

    {

        save();//将构造体信息存盘

        cout<<"刚刚存盘的信息是："<<endl;

        print();//输出输入的航班信息

        break;

    }

}

}

```

```

else

    cout<<"密码错误,您不能使用此项功能, 强制推出!!!"<<endl;

}

void plane::save()//保存函数

{

    FILE *fp,*fp1;//定义文件指针

    if((fp=fopen("chen.t*t","wb"))==NULL)//翻开文件并判断是否出错

    {

        cout<<"创立文件失败!"<<endl;//出错提示

        getchar();

        return;

    }

    if((fp1=fopen("hao.t*t","wb"))==NULL)//翻开文件并判断是否出错

    {

        cout<<"创立文件失败!"<<endl;//出错提示

        getchar();

        return;

    }

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

        if(fwrite(&s[i],sizeof(class plane),1,fp)==0)//向文件写入数据, 并判断是否出错

            cout<<"向文件输入数据失败!"<<endl;

    fprintf(fp1,"%d",m);

```

```

fclose(fp);//关闭文件

fclose(fp1);//关闭文件

}

//-----

void plane::read()//从文件读取信息

{

    FILE *fp,*fp1;//定义文件指针

    if((fp=fopen("chen.t*t","rb"))==NULL)//翻开文件，并判断是否出错

    {

        cout<<"出现错误，请检查文件是否存在!!";//出错提示

        getchar();

    }

    if((fp1=fopen("hao.t*t","rb"))==NULL)//翻开文件并判断是否出错

    {

        cout<<"创立文件失败!\n\n";//出错提示

        getchar();

        return;

    }

    fscanf(fp1,"%d",&m);

    fclose(fp1);//关闭文件

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

    {

```

```

        fread(&s[i],sizeof(plane),1,fp);//从文件中读取信息

    }

    fclose(fp);//关闭文件

}

//-----

void plane::print()//输出函数

{

    char w[10];

    read();//调用读取文件函数

    cout<<"航班号      起始站      终点站      时间      机票数\n";

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

    {

        cout<<PRINT;//打印信息

    }

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

    cout<<"///请按任意键返回主菜单///" <<endl;

    cin>>w;//控制返回主菜单

}

//-----

void plane::search()//查询函数

{

    char name1[10];

```

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/327156024125006106>