**基于Linux的多用户航班购买系统**

项目背景

现在人们的日常出行都会搭乘交通工具，其中就包括火车、高铁、飞机等等，搭乘这些交通工具就需要先买票才能乘坐，所以就需要一个能够让用户能够自助买票的系统。网上电子售票系统, 我做的这个多用户航班购买系统除了能够实现用户的自助购票的相关基础操作，管理员可以对航班进行查询、录入、删除等操作。

目录

1、环境要求…………………………………………………………………………………………………………（1）

2、项目功能…………………………………………………………………………………………………………（1）

3、项目技术点概述…………………………………………………………………………………………… （2）

4、需求分析…………………………………………………………………………………………………………（3）

5、项目实施平台以及技术分析 ……………………………………………………………………… （3）

6、系统框架…………………………………………………………………………………………………………（4）

7、项目细节分析…………………………………………………………………………………………………（5）

8、购票操作分析…………………………………………………………………………………………………（6）

**1、环境要求**

项目运行：Linux操作系统、

开发工具: Visual Studio Code

**2、项目功能**

项目的功能主要可以分成两个部分：用户部分和管理员部分。

用户部分：用户可以选择登录、注册、找回密码功能。当用户登录成功后可以使用：

功能1航班列表，查看所有航班；

功能2查询航班，通过匹配目的地、出发日期、机型、价格查询符合条件的航班；

功能3快速查询，通过输入航班号快速查询该航班；

功能4订购航班，用户可以购买多趟航班，但同一航班只能购买一张票；

功能5已购航班，显示用户以购买的航班；

功能6退票改签，退票时需要扣除相应的手续费，改签则不需要；

功能7查询余额，查询账户的余额；

功能8充值账户，有相应的充值优惠活动；

功能9修改信息，可以修改用户的相关信息；

功能10取票服务，打印用户已购买的航班；

功能11打印发票，打印用户个人购买航班的发票；

功能14退出登录，会给服务器发送提醒。

用户注册就需要用户输入相应的个人信息来完成注册，

如果用户忘记密码需要找回密码就需要输入一些个人信息和手机短信验证 码来进行验证，验证成功后就可以修改密码。

管理员部分：管理员可以注册多个账户,但需要邀请码。

登录成功后可以使用：

功能1航班列表，查看所有航班；

功能2查询航班，通过匹配目的地、出发日期、机型、价格查询符合条件的航班；

功能3快速查询，通过输入航班号快速查询该航班；

功能4录入航班，输入航班的相关信息完成航班的录入；

功能5删除航班，删除航班时对购买了该航班的用户进行退款操作；

功能6已删航班，显示已删除的航班, 通过逻辑删除来实现；

功能7退出登录。

**3、项目技术点概述**

该项目运用到的技术主要有：字符串的相关操作，如：控制字符串的输出格式，字符串的分割，字符串的比较；文件IO的相关操作，如：文件的创建，文件的打开和关闭，文件的读和写；链表的相关操作，如：初始化头节点，节点数据域和指针域的赋值，节点的插入，节点的删除，链表的销毁；线程的相关操作，如：线程的创建和退出；互斥锁的使用；网络编程的相关操作，如：套接字的创建，绑定IP地址、端口号、协议到套接字上，设置监听套接字，接收和发送数据；

**4、需求分析**

可以实现普通用户和管理员登录；普通用户可以查看所有航班信息、通过指定信息查询航班信息、购票、查看已购买的票、退票、改签、查询余额、充值、修改个人信息、取票、打印发票、用户之间聊天、发朋友圈等功能；管理员可以查看所有航班信息、通过指定信息查询航班信息、录入航班、删除航班、查看已删除航班信息等功能。

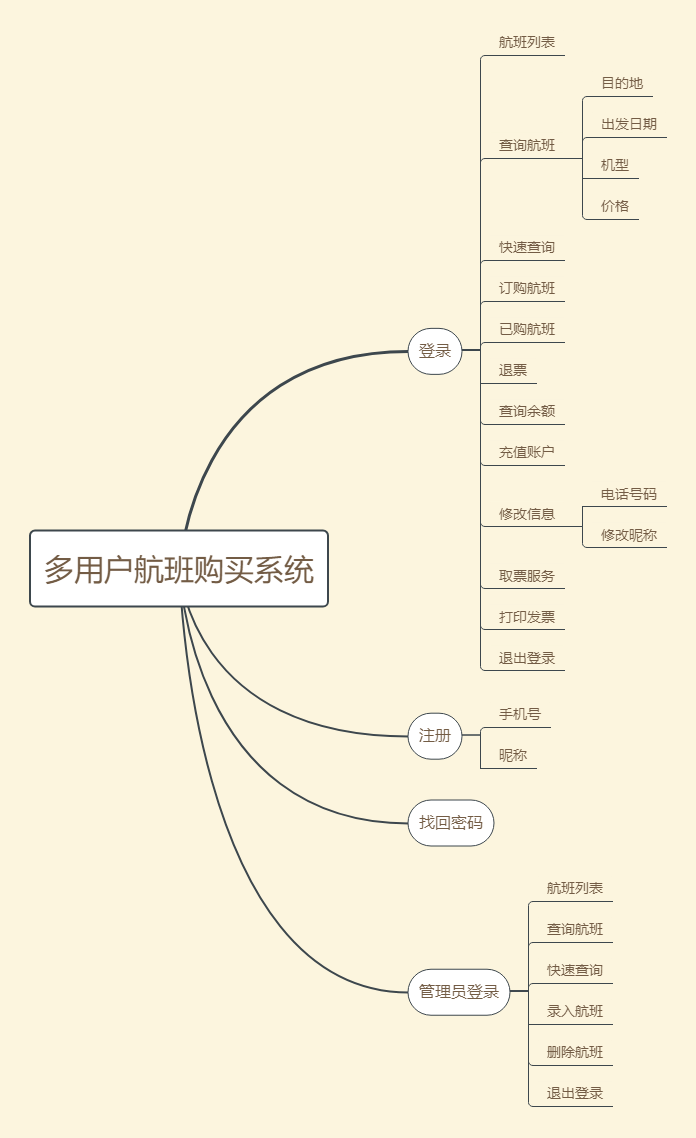
**5、项目实施平台以及技术分析**

该项目是在Linux平台下，基于C语言实现的，一开始服务器先运行，就会先创建链表，就需要用到链表的相关操作，然后把文件中的数据读取出来，就需要文件IO的相关操作，然后通过字符串的分割把信息分别赋值到链表节点的数据域中，最后把节点插入到链表中。

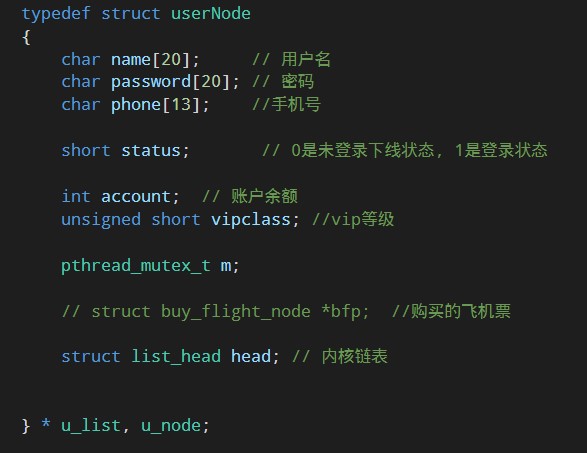
客户端可以不断向服务器发送数据，服务器端需要不断接收多个客户端的信息，因此就需要用到多任务并发，多任务并发有多线程和多进程，由于多线程消耗的资源比多进程要少得多，所以本系统使用的的多线程，通过一个线程对应一个客户端来接收客户端的信息并做出相应的操作。同时由于是服务器与客户端之间进行通信就需要用到网络编程的相关操作。

接收到的数据需要需要进行分析处理，就需要用到字符串的比较。执行用户的命令，如果是查询航班则需要用到链表的遍历，买票则是节点的插入，退票则是节点的删除，同时需要把信息同步到文件中保存起来就需要用到文件IO的相关操作，并且访问临界资源前需要用互斥锁先上锁，等操作完成之后再解锁。

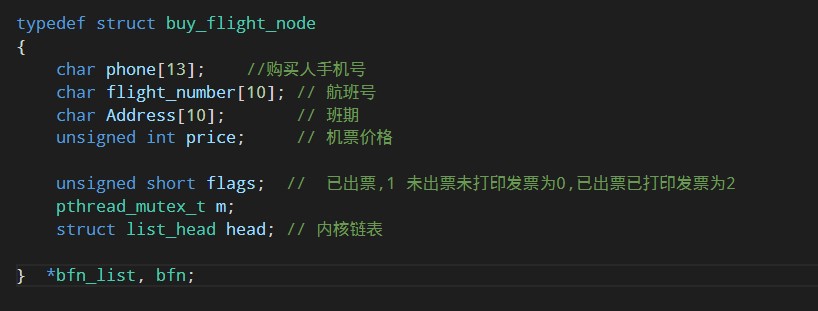
**6、系统框架**



**7、项目细节分析**



上图是本项目的用户信息结构体，其中name用于存储用户的名字，passwd用于存储用户的密码，phone用于存储用户的手机号码（用户找回密码是需要用到，因为通过短信验证码来找回密码的），t，account用于存储用户的账户余额，status用于表示该用户是否上线,vipclass代表用户vip等级（1表示VIP，0表示普通用户）

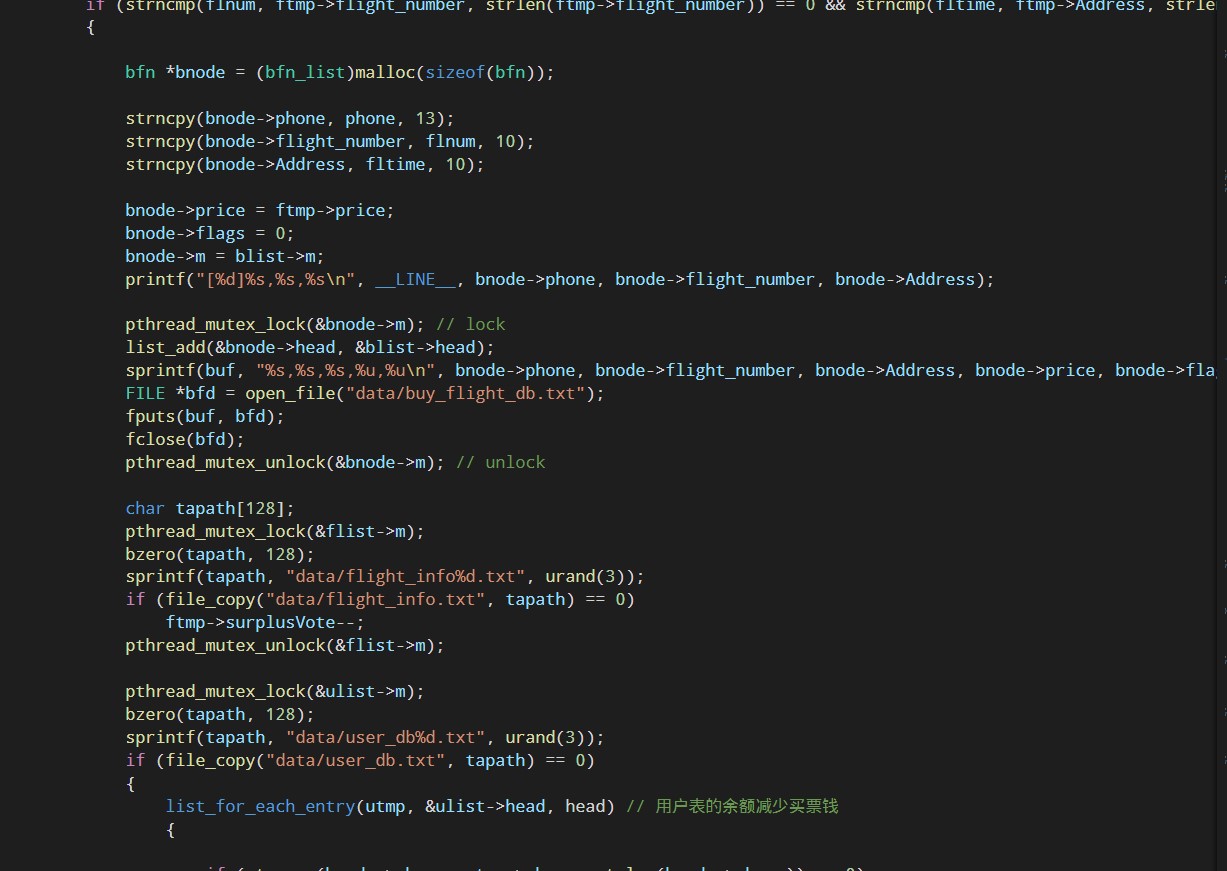


head是用户已购买航班号链表的头节点，该链表的节点如上图，存储了用户购买的航班号和购买该航班所花的钱，当用户需要显示当前用户已购买的航班时，服务器就会遍历航班列表找到该航班号，并打印该航班的相应信息。

存储用户购买该航班时所花的钱，是为了在用户进行退票和改签时金额操作的方便和准确，例如，当用户是VIP的话，他购买航班是享受八折优惠的，如果他进行退票操作当然也是退他当时买这张票所花的钱，防止多退钱情况。另一种情况，用户在不是会员的时候全价买了该票，然后通过充值变成了VIP，这时他去进行退票操作也是退他当时买这张票所花的钱，防止少退钱情况。由此可以看出，存储用户购买航班时所花的价钱对服务器进行相应的金额操作是十分重要的。

**8、购票操作分析**

购票需要考虑到的方面很多，例如：用户的输入的航班是否已经购买过了（系统设置了用户只能购买同一航班的一张票），如果没有购买过，该航班是否存在，如果存在，该航班的余票是否大于0，如果有余票则需要判断用户的账户余额是否足够购买该航班（就需要考虑用户是否为VIP，VIP的话有打折），如果用户的账户余额足够购买该航班则，进行购买但在购买时也要判断这时的余票是否为0（有可能用户想要购买的票别其他人先买走了，造成余票为0），如果为0则购票失败，否则的话就购票成功，购票成功还需要进行相应的文件保存和数据的更新操作。



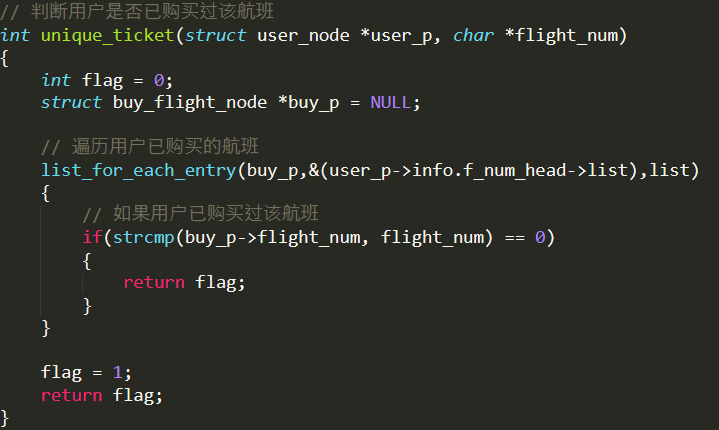
用户选择购票时先打印相关的提示信息，比如当前的航班列表，方便用户查看和购买（如上图）。

然后请求用户输入要购买的航班号，等接收到用户输入的信息后，先判断用户是否以购买过该航班（flag=1代表没买过，flag=0代表买过），如果没有购买过则遍历航班信息链表，检测该航班是否存在，如果找到了，则判断余票，如果余票小于1，则代表该航班已售罄，则打印相应信息（如下图）。





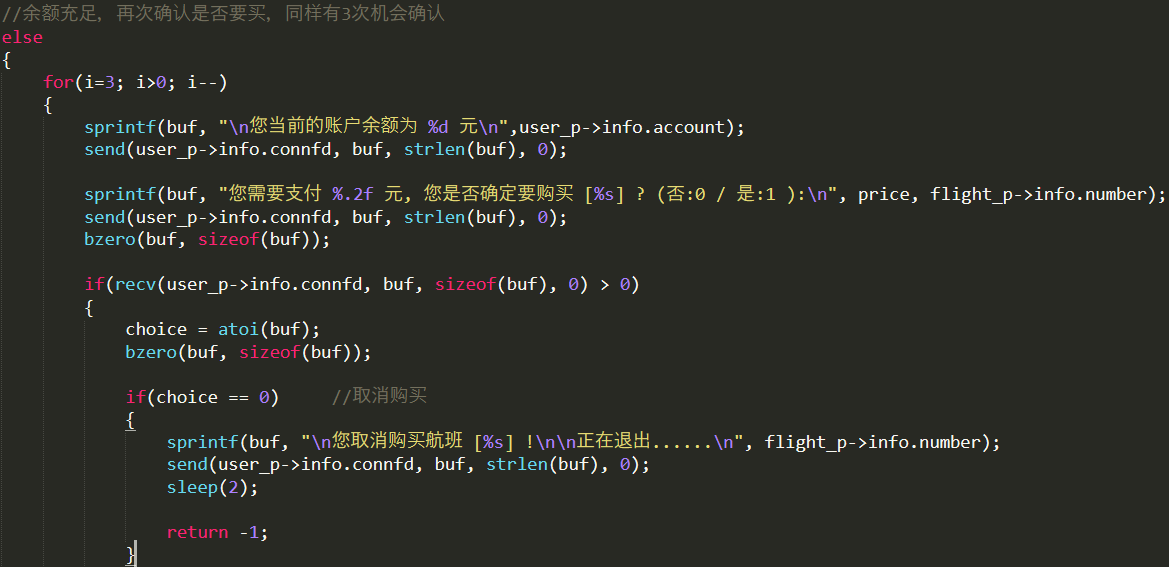
如果余票充足，则继续进行购票操作，下面的是另外一些情况的提示信息，如：没有找到该航班，该航班以购买过，不能再购买了（如上图）。



前面的购票判断用户是否购买过该航班，就是用上图的函数进行判断的，遍历用户的已购买航班链表，如果找到了（买过）就返回flag=0，否则就返回flag=1。



买票前要先判断用户的账户余额是否足够购买这张票，这就需要先计算当前用户购买该航班所要话地钱（VIP的话买票八折），然后在判断如果账户余额不足的话打印相应的提示信息。



如果账户余额充足，则询问是否确定要购买该航班，如果取消购买则打印相应的提示信息并退出。



如果确认购买则进行，继续进行确认购票操作，根据接下来的购票操作的返回值来判断购票是否成功，成功（ret=0）和失败（ret=-1）都打印相应的提示信息。



确认购票操作，先判断航班余票是否为0（代表航班被别人先付款购买了），如果为0则返回-1，如果不为0，则可以进行购买，购买时，需要将航班的余票减一，然后同步到文件中，然后要扣除用户购买该航班所花的钱，接着把该航班的信息尾插到用户已购买航班链表上，并且同步到文件中，这样才算购票成功。