

软件工程课程设计I

项目开发报告

|  |  |
| --- | --- |
| 项目名称： | 剧院票务管理系统 |
| 团队名称： | 魔仙堡宝 |
| 院（系）： | 计算机学院 |
| 专业： | 软件工程 |
| 班级： | 软件工程1607 |
| 指导教师： | 张丽丽 |
| 起止时间： | 2017年06月13日至2017年06月21日 |
| 团队成员： | 李影 张妍 江婷婷 朱颖 |

|  |  |  |
| --- | --- | --- |
| **学号** | **姓名** | **角色** |
| 04163214 | 李影 | 成员 |
| 04163215 | 张妍 | 管理员 |
| 04163216 | 江婷婷 | 组长 |
| 04163217 | 朱颖 | 副组长 |

“剧院票务管理系统”项目开发报告

# 项目概述

## 项目背景

**（1）开发背景**：当今社会的计算机的快速发展，以及剧院的越来越受欢迎，很需要这么一个软件来方便剧院的管理，减少劳动力。

**（2）目标**：为中小规模的剧院（包含电影院，歌剧院，演唱会等）开发一个通用的票务管理软件，对剧院的演出厅，剧目，演出计划，售票，销售统计等售票相关业务实现全程计算机管理

## 团队分工

李影：整理框架(Studio、Seat、Ticket)、完成部分Queries、UI界面设计、ppt制作、ppt讲解、完成部分项目开发报告

张妍： Play全部、完成部分Schedule、完成部分Queries、UI界面设计、ppt制作

江婷婷：Account全部、SaleAnalysis全部、完成部分Queries、程序bug修改、完善程序功能、完成部分项目开发报告、完成部分用户使用手册

朱颖： Sale全部、完成部分Queries、UI界面设计、程序bug修改、完成部分用户使用手册

## 术语与缩写

|  |  |
| --- | --- |
| View(UI) | 界面层 |
| Service(Srv) | 业务逻辑层 |
| Persisence (Per) | 持久化层 |
| Studio | 演出厅 |
| Seat | 座位 |
| schedule | 演出计划 |
| Sale | 卖票 |
| Query | 查询 |

## 参考资料

[1]作者：王曙燕资料名称：C语言程序设计教程

出版社：人民邮电出版社

[2]作者.： Stephen Prata资料名称.: C Primer Plus

出版社: 人民邮电出版社

[3]作者： Kenneth A.Reck资料名称： C和指针

出版社：人民邮电出版社

# 系统实现

## 2.1开发环境

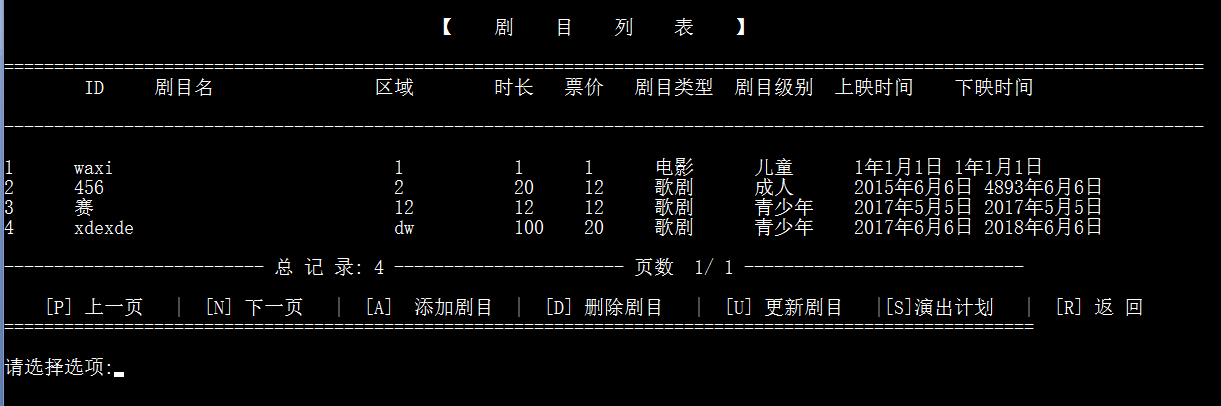
1. 处理机的型号、内存容量：Intel Core i5-6300HQ 2.30GHz 4.00GB
2. 操作系统的名称、版本号 ：Windows10家庭中文版
3. 开发工具 ：Eclipse（jdk1.8.0、MinGW）

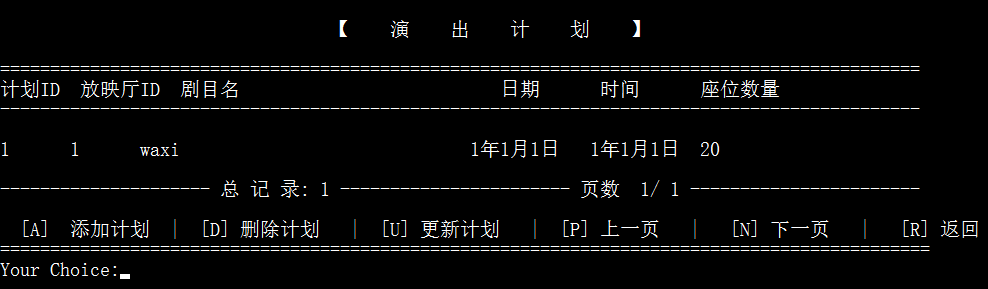
## 2.2系统功能

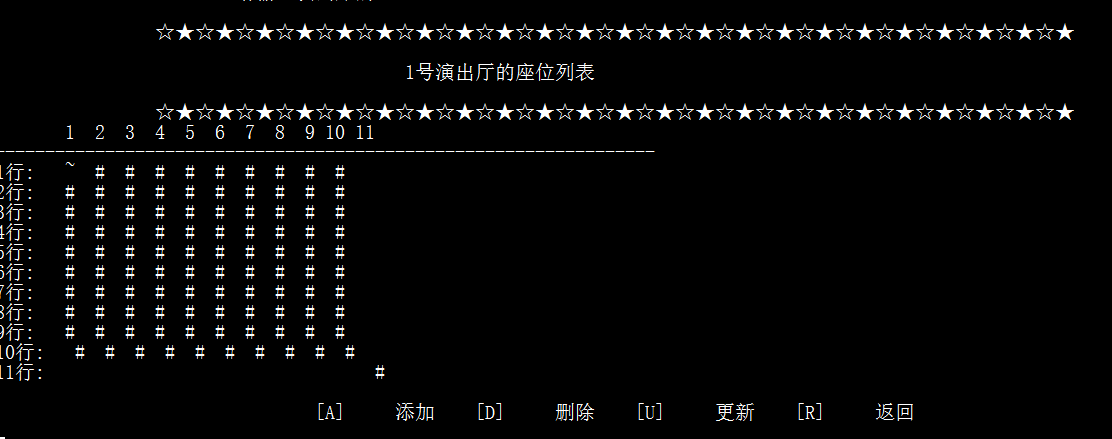
## 功能：剧院的演出厅，剧目，演出计划，售票，销售统计等售票相关业务

主菜单：

## 演出厅的增删改查：

剧目的增删改查：

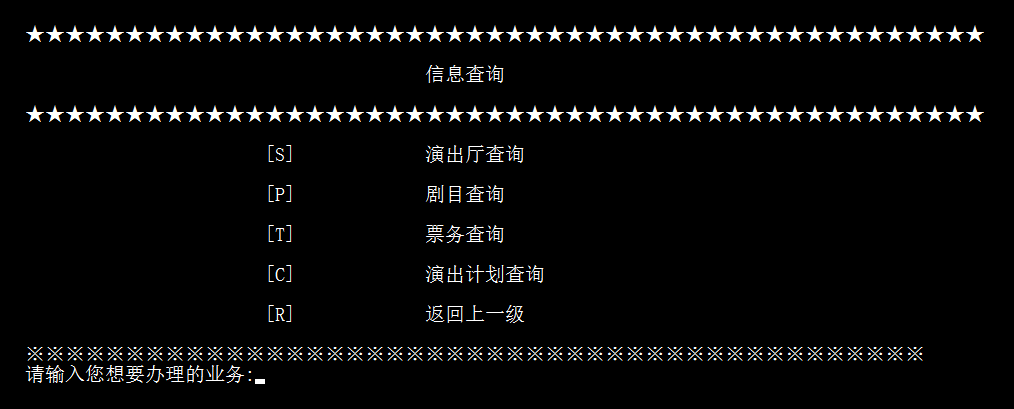
演出计划的增删改查：

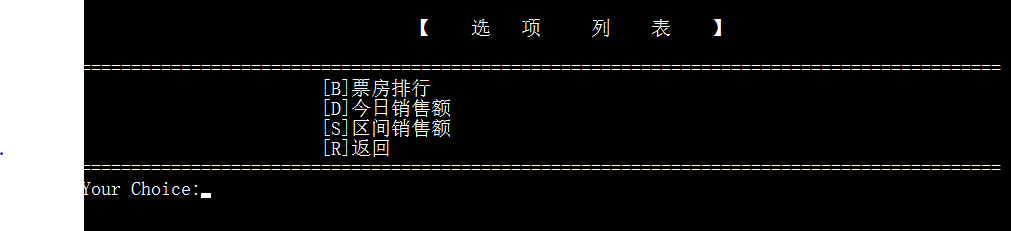
座位的增删改查：

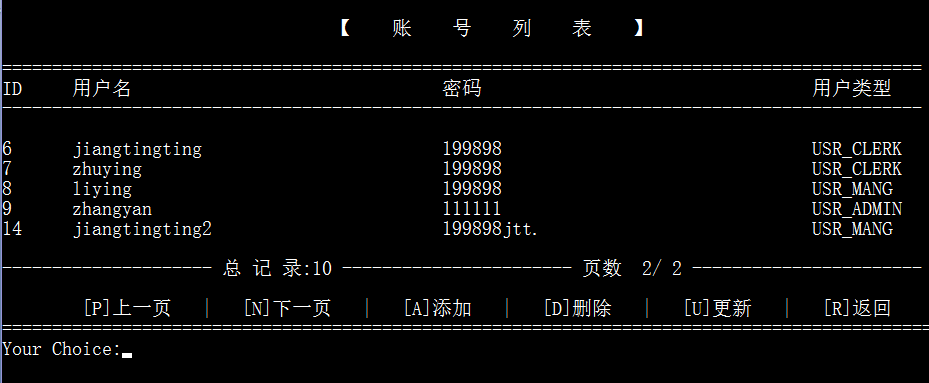
退票



购票：

查询：

数据分析：

账号的增删改查：

## 2.3团队开发工作总结

### 2.3.1团队组织管理

项目开始之前，组长先带领我们了解UI层，业务逻辑层，持久化层之间的关系，以及做项目的时候的思路。在项目进行的时候，组长每天都会给我们说明当天的任务，并在最后会进行检查验收。当我们遇到问题的时候组长会先让我们自己百度，查找书籍或者其他的方法解决，解决不了组长会给我们进行讲解。

### 2.3.2软件开发过程

（1）把设计说明书进行了浏览，进行成员角色分配，分工

（2）搭建环境

（3）在组长带领下对做项目的流程进行了了解（先从UI层开始找UI层里面的所调用的业务逻辑层的函数，在业务逻辑层里面找所调用的持久化层的函数，从持久化层开始一一完成，思路和递归函数很相似）

### 2.3.3难点问题解决

1. 遇到的问题：函数传参，以及接口的问题

解决方法：查找书籍，组员之间进行交流

(2)遇到的问题：指针和文件的知识掌握的不够导致在业务逻辑层和持久化层写代码的时候遇到困难

解决办法：百度，查找书籍对于指针和文件的知识进行了学习

(3)遇到的问题：枚举类型在输出的时候遇到问题

解决办法：为了让在输出的时候不是数字，利用switch语句进行判断，从而让其输出字符串

## 2.4经验与教训

**(1)收获**：

1. 不仅仅对以前所学的东西有所回顾而且还通过这次的课设任务让我们学习了很多新知识
2. 了解到大中型文件的多层结构，以及它们之间的关系
3. 懂得团队合作的重要性，这样可以提高工作效率

(**2)教训***：*

1. 懂得基础知识的重要性（学东西要学的比较扎实不能似懂非懂的样子）
2. 学习东西真的不能停留在学校所发的书本上，还需要自己通过其他的的渠道去学习一些知识

# 3成员开发总结

**3.1李影个人总结**

### 3.1.1开发任务

**职责***：*成员

**开发任务**：管理演出厅，设置座位，查询演出

### 3.1.2开发成果

1. 开发的模块：管理演出厅，设置座位，查询演出
2. 写的文档：用户手册里面自己开发的模块

（3）贡献：1.完成自己的任务

2.UI界面设计

3. ppt制作，ppt讲解

4.课程设计报告

### 3.1.3模块开发中难点问题及解决

（1）遇到的问题：搭建环境，Eclipse的环境的时候不知道为什么其中的Mi即便是安装完成了之后还是不嫩

解决问题的方法：其中对mingw进行了多次的卸载安装

（2）遇到的问题：对于文件和指针的知识掌握不够，不能够熟练应用

解决问题的方法：查阅书籍，小组之间进行交流

（3）遇到的问题：对于项目之间的关系了解不够

解决问题的方法：百度，向老师求助，小组之间相互交流

（4）遇到的问题：对于函数的传参遇到问题

解决问题的方法：看其中的函数所调用的函数进行一一检查，查看其中传参情况

（5）遇到的问题：UI方面的对齐问题

解决方法：其中可以给设置负值，从而达到对齐的效果

### 3.1.4模块测试



**

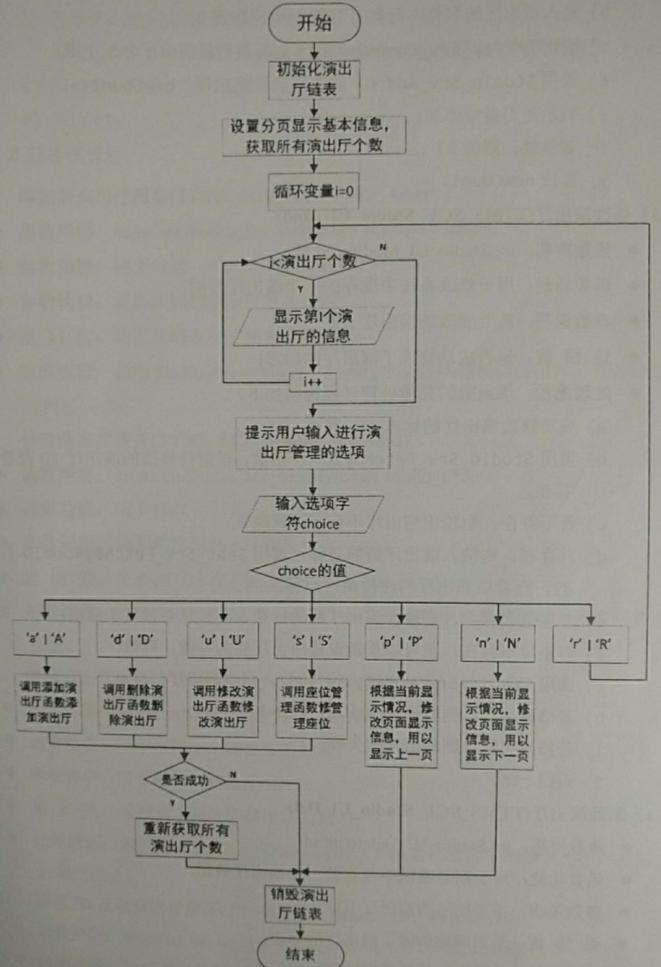
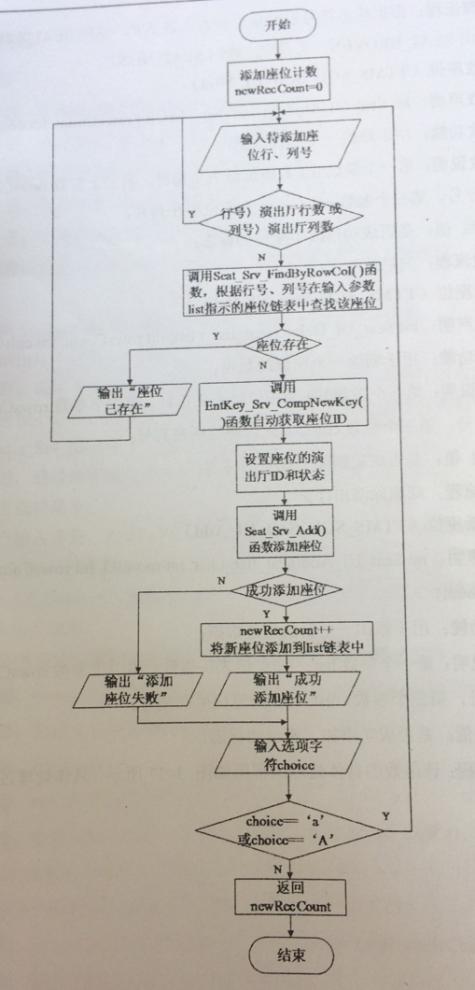
在这里展示了其中演出厅的界面以及添加的演出厅的成功之后的界面。以及展示了其中的演出厅中所实现的功能。

### 3.1.5收获与教训

1. 收获：对以前所学的东西进行了复习，对文件的的知识有更多的了解，初识于项目的进行的思路。比如说其中对于文件的操作，以及其中的判断文件是否存在（access函数根据其中的返回值来判断，返回1就是存在文件，返回0就是不存在这个文件）；其中做出来闪烁的效果（利用的就是sleep和cls函数）

（2）教训：对基础知识掌握的不够好，在以后的学习中应该把基础知识掌握好，避免在以后的应用过程中又翻书，查资料浪费时间

经过这次的课程设计我主要实现的就是管理演出厅，自己写UI层，业务逻辑层，持久化层，其中自己写UI层遇到的问题挺多的，有的不知道该怎么写程序，因为那个流程不是很清楚，需要参考其他的用例。其中业务逻辑层还可以其中主要用的就是持久化层的函数，通过调用持久化层的函数的返回值来实现业务逻辑层的功能。最后就是持久化层，其中持久化层主要就是对文件的操作其中包括文件的打开，查看文件是否存在，给文件中写东西，查看是否读到了文件的最后等操作。这个持久化层就是完成的是大量数据的操作。通过这3层的操作让我明白了其中这三层的关系，关系就是UI层调用业务逻辑层，业务逻辑层调用持久化层，其中在程序进行的时候就是从持久化层返回给业务逻辑层，业务逻辑层返回给ui层，最后将结果返回给用户。演出厅和座位管理流程图如下：



通过这次的课程设计，让我认识到自己的不足，很多最基础的东西有些都已经忘记了，对链表和文件的东西掌握的不够好，不能够很好的利用。自己的知识面还是太窄了，以至于在这次的课程设计中用了很大的时间来查阅资料和询问同学。以后还是不仅仅只是看学校发的书，还需要去看一些课外的书来扩充自己的知识面。

通过这次课程设计也让我明白了团队合作的重要性，说实话刚开始的时候挺迷的，找不到方向，在组长的带领下才对这次课程设计有了自己的想法。其中组长也会每天给我们分配任务，每天检查我们实际完成的情况。这次课程设计的任务量也有点大，正是通过大家的合作才让我们在短短的一周半完成任务。**3.2张妍个人总结**

### 3.2.1开发任务

**职责***：*管理员

**开发任务**：管理剧目，安排演出计划

**3.2.2开发成果**

（1）开发的模块：管理剧目，安排演出计划

（2）写的文档：用户手册里面自己开发的模块

（3）贡献：1.完成自己的任务

2优化UI

3.课程设计报告

4.ppt制作，

5.作品演示

### 3.2.3模块开发中难点问题及解决

（1）遇到的问题：定义指针的时候没有分配内存

解决问题的方法：查阅资料，和小组成员进行交流，以后再使用指时都会先给它分配地址。

（2）遇到的问题：取址符用法错误程序崩。

解决问题的方法：查资料百度问同学老师，记住正确的使用方法。

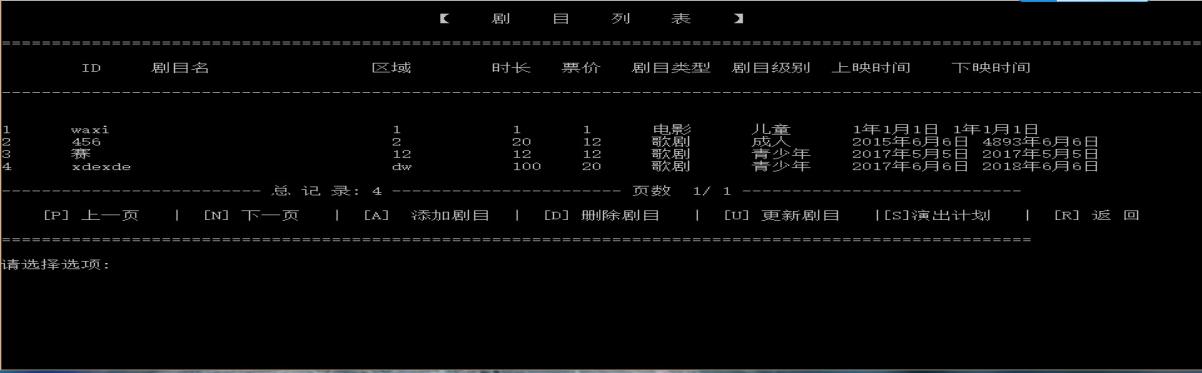
（3）遇到的问题：文件因写法不规范而引起程序崩

解决问题的方法：查资料百度问同学老师，记住正确的使用方法。

（4）遇到的问题：程序总是有好多不知道的错误

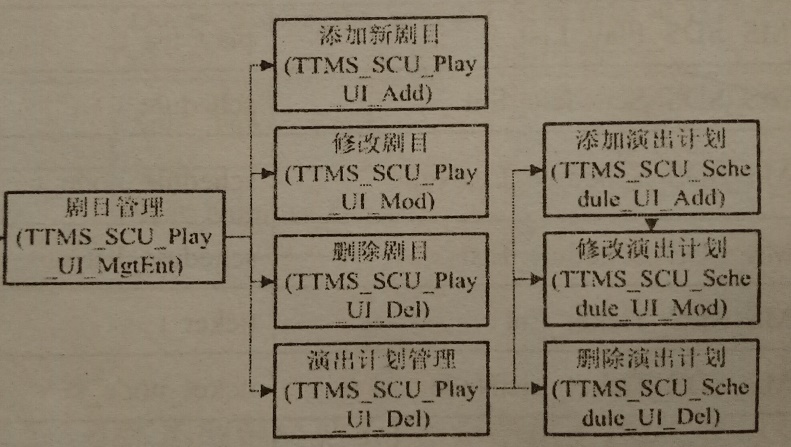
解决问题的方法：删掉inline。

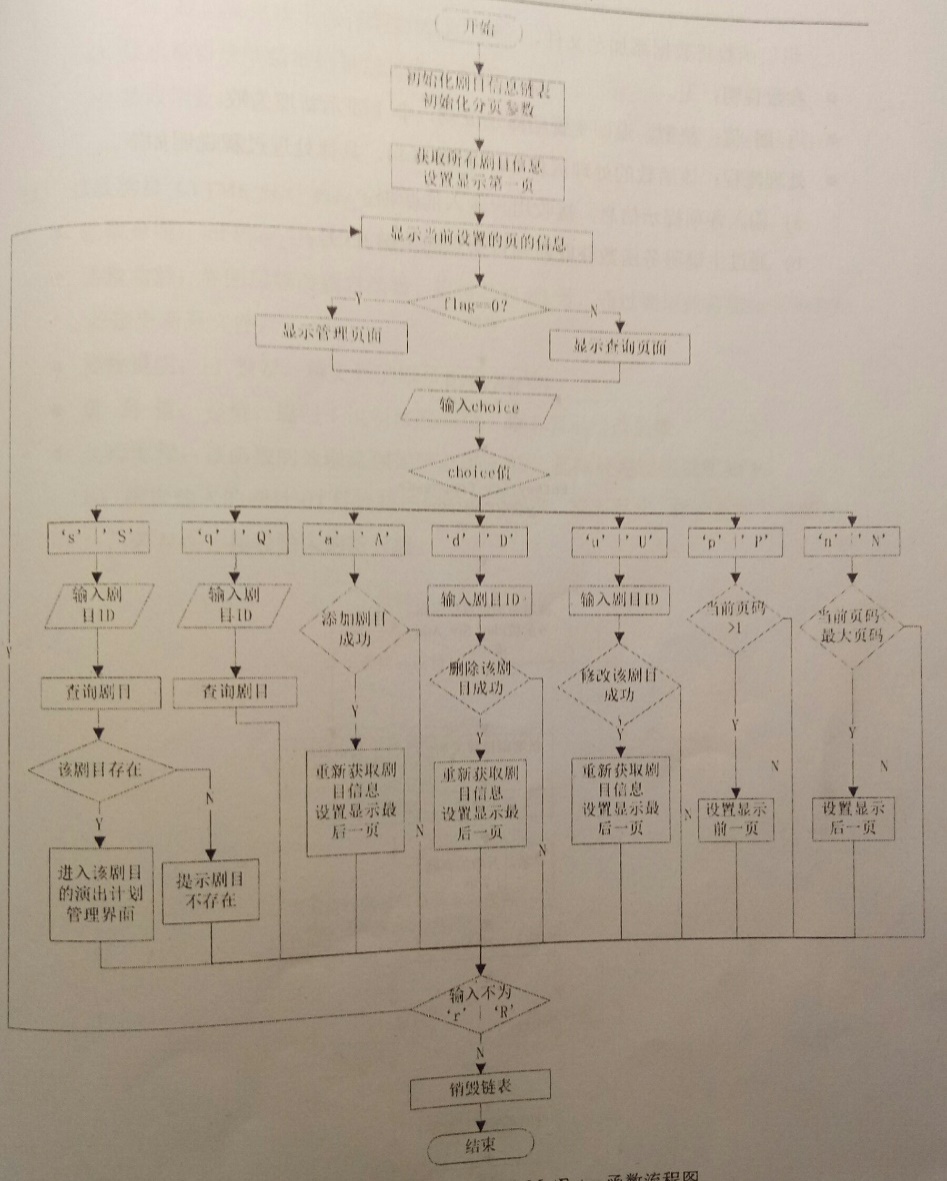
### 3.2.4模块测试



### 3.2.5收获与教训

刚准备开始做的时候还是有点怕的，毕竟以前从未写过这么大的程序，拿到书之后看了一遍后大致对这个程序有了些许认识，在学习了List.h中的知识后我才开始构思并完成我的任务，Play和schedule的部分流程图如图如下：





通过我这次的课程设计不仅仅让我学到了很多新的知识，其中有我们刚开始才学不久的文件里面的一些新的东西，还掌握了链表里面的东西。对于其中的3层之间的关系，实质上来说其中就是ui调用业务逻辑，业务逻辑调用持久化。就像递归函数一样。

通过我这次的课程设计不仅仅让我学到了很多新的知识，其中有我们刚开始才学不久的文件里面的一些新的东西，还掌握了链表里面的东西。对于其中的3层之间的关系，实质上来说其中就是ui调用业务逻辑，业务逻辑调用持久化。就像递归函数一样。还有很重要的一点，

我因为之前写的程序都是些小程序只是些简单的函数，所以不知道怎么和别人配合，而因为这次程序比较大需要队友们相互配合，所以在完成我自己的函数块时我需要和队友们交流配合，这就促使我学会了同队友们配合，我想这在我以后的学习工作中对我都有很大的帮助。

还有因为程序很大所以免不了要出现一些问题，我以前的代码风格非常混乱，导致我在检察时需要花费很多的时间去厘清我到底写了些什么，所以在一次次的修改中我渐渐改变了我的代码风格，现在虽不是太整齐但也还算条理清晰；再者因为我以前没有写注释的习惯，所以队友们在开始的时候并不清楚我程序的功能，哪怕是我有时也会忘，所以渐渐的我也开始学会了给程序中添加一些必要的注释，所以这次课设不但让我学到了很多的知识更让我学到了许多平时根本学不到的东西，极大的改善了我的代码风格，还有以前总是不注意那些细小的知识点，现在需要用到的东西多了就不可避免的会用到他们，问题也就暴露出来了，我也将其改正，学到了不少东西。

还有因为这次任务是小组合作完成的，所以再写程序的时候大家都是在一起的，有了问题也是大家一起讨论，不知不觉间就从她们身上学到了很多东西，从思考方式到编写方式她们都给了我很多的灵感和新思路，真的是太谢谢她们了，总的来说真的是获益良多。

**3.3江婷婷个人总结**

### 3.3.1开发任务

职责：组长

开发任务：

（1）理解TTMS系统框架，各个用例功能之间的穿插，以及单个用例三层的互相调用和任务分配。

（2）理解Common文件夹中对于链表和分页器的操作有关宏定义及函数的实现，并熟悉如何使用它们。

（3）完成SaleAnalysis全部代码、Account全部代码、Schedule部分和Sale部分的补充和改Bug。

### 3.3.2开发成果

（1）Account

1. 实现匿名登录和账号登录两种登录方式，并在登录后分配不同用户使用权限。匿名用户直接进入主菜单，账号用户需要进入登录界面验证身份。

2. 实现登录账号密码的控制输入，登录界面账号实现控制无法键入空白字符等非数字字母标点符号以外的输入（不显示非正常输入），并且控制在6-29个字节，否则无法成功输入（按回车键无效）。密码实现回显’ \* ’号，确保密码安全性。同账号输入控制输入字符种类及位数。三次登录失败自动退出程序。

3. 实现权限判断后的系统账号增删改功能及分页显示所有账号功能。添加账号时实现密码二次输入判断是否一致，及输入的控制。

（2）SaleAnalysis

1. 实现票房排行功能，通过剧目出票数进行链表的插入排序，遍历输出。

2. 实现查询今日销售额功能，根据输入的售票员账号（经理权限）或自身（售票员）输出今日销售额。

3. 实现查询区间销售额功能，给定时间区间显示对象的销售总额。

（3）Sale、Schedule、Ticket部分的穿插

1. 实现了添加演出计划时自动生成票的功能，创建票的记录，在选完演出计划后可查看票，列出票的属性，包括ID、票价、状态。

2. 实现出票购票，实现售票员通过查询剧目到查询演出计划再到查看演出票，最后输入需要的座位行号列号进行买票，买票成功后打印票信息。

（4）完成用户手册和课设报告

### 3.3.3模块开发中难点问题及解决

问题1：对实践指导书专业度和厚度和框架文件分层的恐惧感。

解决：小组大部分成员之前没有接触过对于我们来说的这么一个大工程，书上很多专业名词都很陌生，包括主键、分页器的使用还有多文件多层次都突然提升了很大一个层次，但是通过一起静下心来理清思路，去看框架的源码，最后还是弄清了这些，打败了恐惧感，建立起了自信心。

问题2：前期对程序框架认识不够透彻，对程序运行过程不是很了解，导致部分有关票的功能没有在计划时间内实现。

解决：在发现没有票时，重新去看了Ticket部分的代码，理清Ticket与Sale、Schedule部分的关系，修改了Sale和Schedule部分有关票的代码，最后实现了买票、出票的功能。

问题3：环境问题，安装过程中小组成员有过很多问题，在安装成功后作为一个陌生的IDE，去学习如何使用它以及一些快捷操作花费了将近半周的时间，期间也因为各种问题无法专心于代码，而是不停地去弄工具。

解决：通过老师和同学的帮助，以及百度解决办法，在课设的第三天各成员解决了IDE的问题，开始专心敲代码，并且懂得eclipse跳转函数及查找修改的操作，更加有助于之后的代码完成。

问题4：在文件操作中的删除记录操作，使用框架给出的rename()方法实现不了，perror()报出说文件权限不够。

解决：fopen()来创建并打开临时文件，遍历写入，达到和rename()创建临时文件一样的效果。

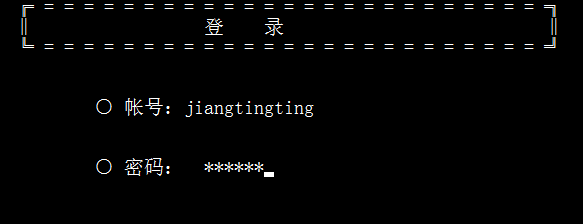
问题5：删除记录时会出现最后一条记录被重写的现象。

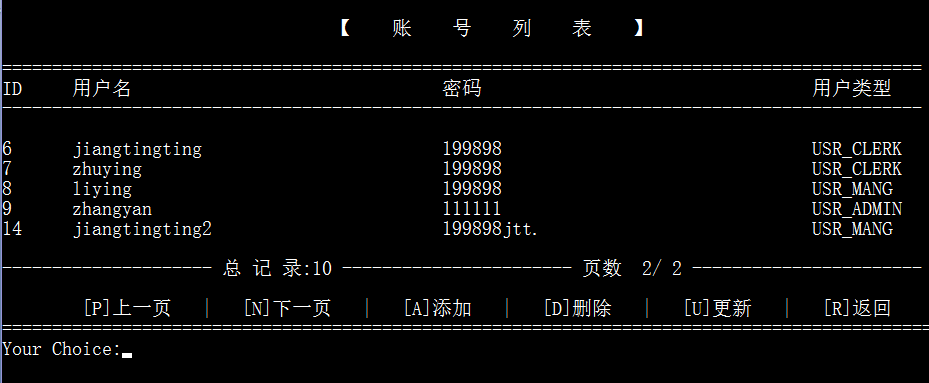
解决：通过百度，发现问题在于feof判断是否到达文件末尾和fread读记录的顺序问题，在feof之前fread一次就解决了这个问题。

问题6：对未初始化分配内存的指针进行间接访问，报错，或是因为循环控制条件不当造成对空指针访问，导致程序崩掉。

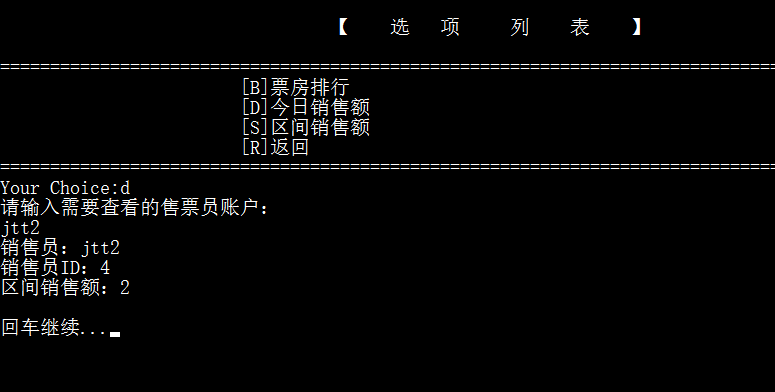
解决：在接触过几次这些错误之后就更加注意对指针的初始，并且在调试过程中若程序崩掉，首先就会去找是不是访问了空指针。

### 3.3.4模块测试

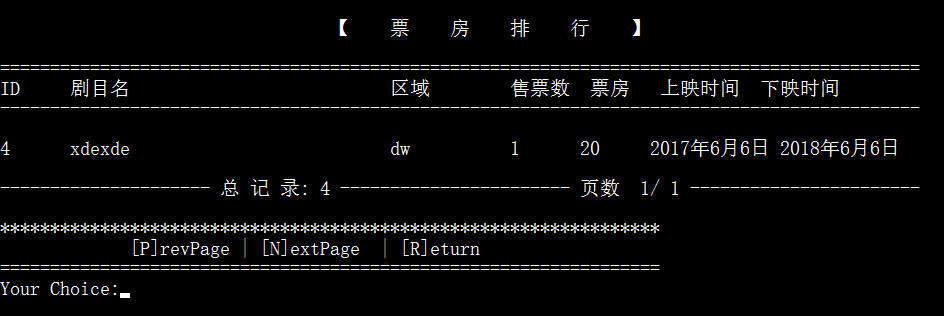
账号登录界面：

账号列表及账号操作选项：

订票成功并打印：

查看销售额：

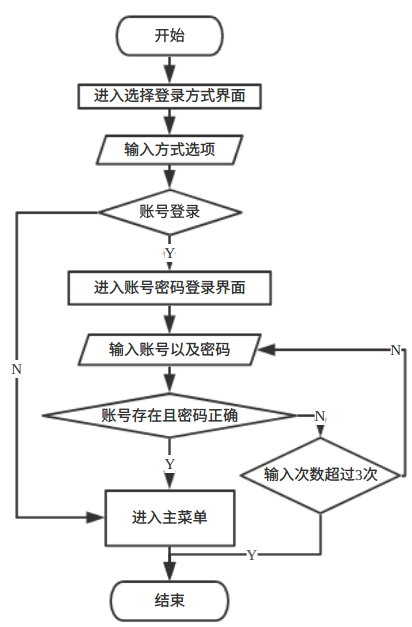
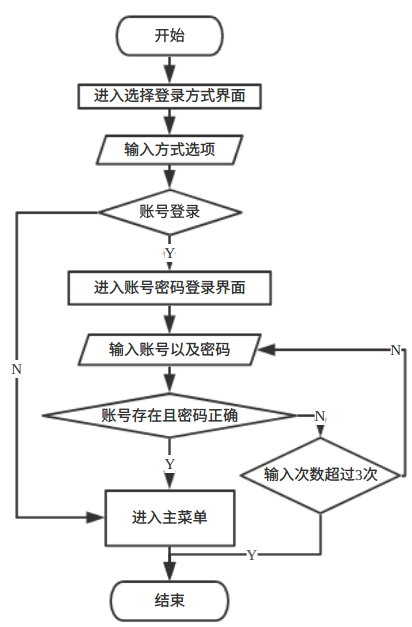
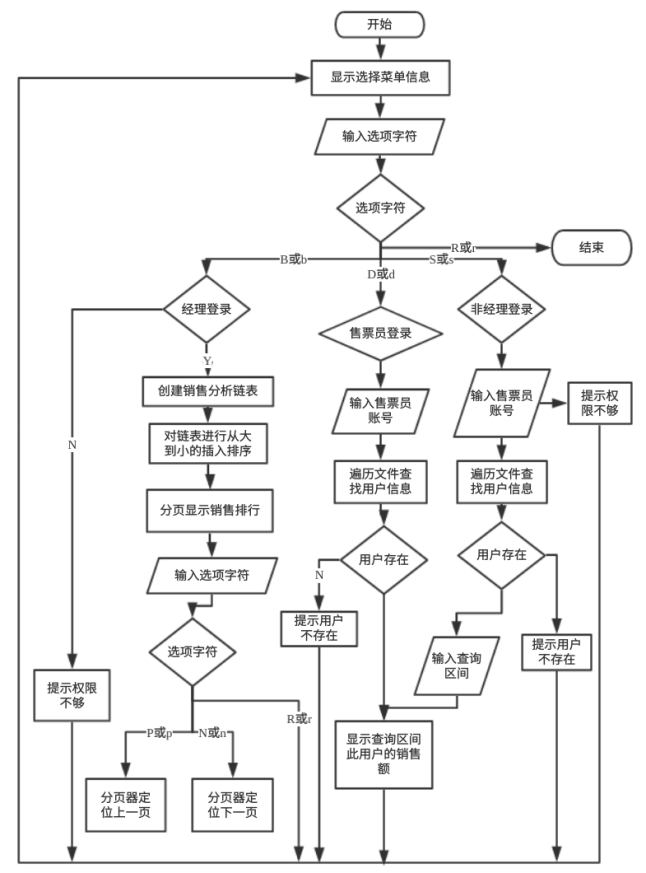
票房排行：

输出有点问题，记录数、票房和售票数都正常

### 3.3.5收获与教训

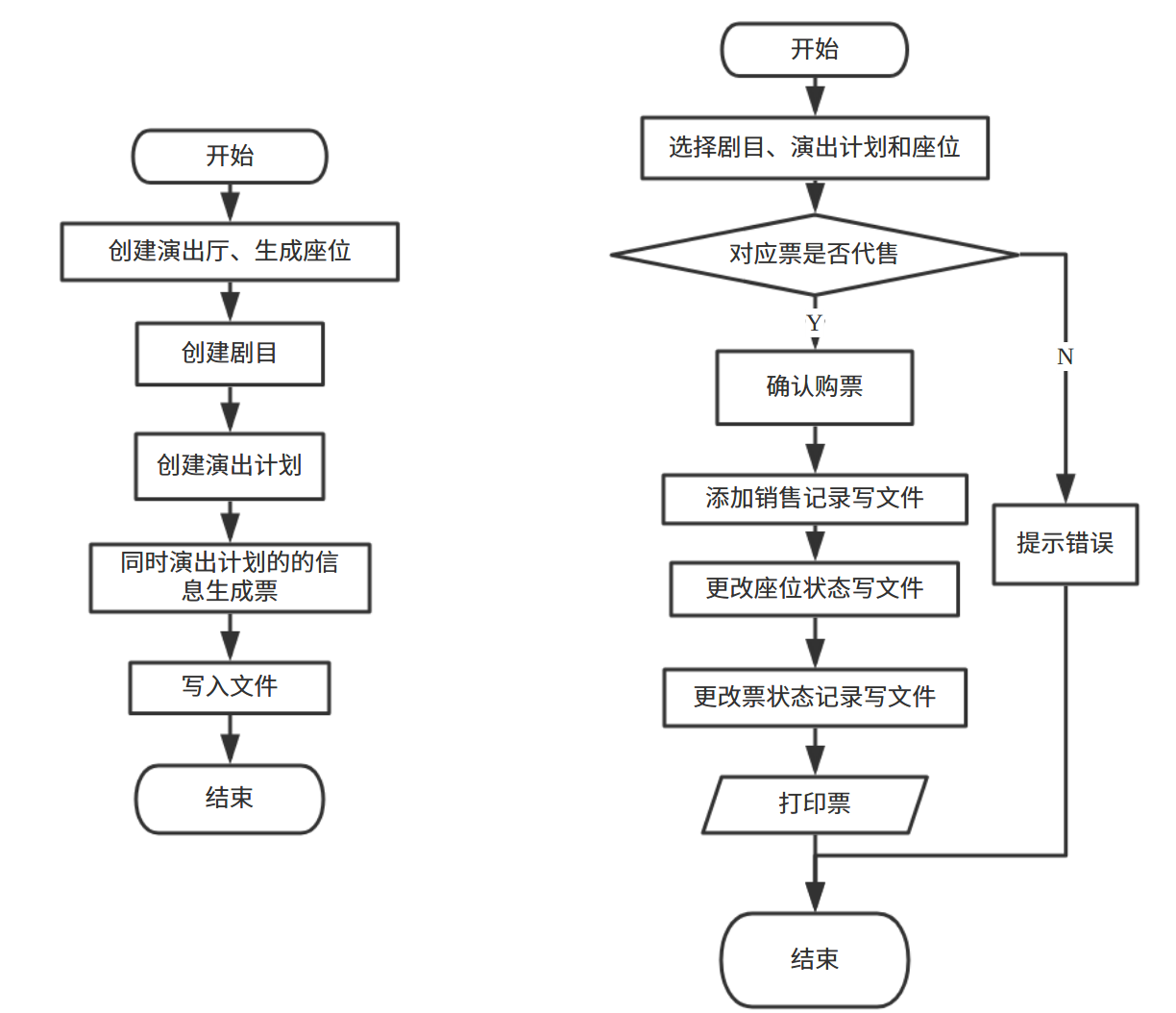
大学一年以来，第一次接触这样一个在我们看起来特别专业的任务，为期一周半，团队分工完成，框架层次明确的TTMS。最后虽然不完美，但还是挺满意的，最重要的还是在这次课程设计的过程中，让我学到了很多，收获了很多。

拿到实践指导时有点吃惊，没有想到会这么专业。之前在实验室的要求下也有试写过学生管理，但是看到这么层次分明的框架，心里还是有些忐忑，不知道最后完成的效果会怎么样。于是我也是在搭建好环境之后马上去尝试了Account和SaleAnalysis部分，理清框架三层的调用关系，熟悉Common中函数的运， Account部分的登录流程和SaleAnalysis部分流程图如下：



在完成这两个相对简单的用例时，除了文件上的一些小知识点的学习，更多的是感受到了原来没有接触过的技术给我带来的收获。比如结构体定义分页器的使用，操作链表节点使用宏定义再去调用避免了类型导致的代码重复，主键自动分配ID，分三层次互相调用方便维护系统使系统层次逻辑清晰。这些原来没有接触过的技术都给人眼前一亮的感觉，并且每一项都很贴切实际，都很方便。也希望在以后的编程中包括学习其他语言时，我也能在适当的地方熟练地应用这些技术。

一个团队，虽然技术还不够硬，但是每天写日志，最后写报告和用户手册，团队成员分工合作，让我有一种身处公司的感觉。我们四个人一起熬夜，一起商讨问题，互相帮助，互相体谅。身为组长应当承担更多的工作任务，但是组员每次看到我很累的时候，也会去陪我熬夜，帮我分担很多，我们四个就像一个家庭一样，也正是因为有这样的互相体谅和团结，才能完成的比较好。比如在售票部分会牵扯其他很多用例，而不同的用例又是由不同成员负责的，这就考验了团队配合和交流，最终理清思路实现功能，票的生成和购票的流程图如下图所示：



在组员们差不多完成首先分配的任务后，便开始去修改error、warning，在能够成功运行，却会因为各种原因导致程序崩掉，我们又开始去调试修改bug。遇到问题，从起初的烦躁到最后的欣慰，我和组员们的态度转换了很多，因为我们逐渐明白了，遇到问题去解决，会让我学到更多，了解更多。去逃避，只会让人愈发消极。在改代码时，我们会先独立找错误，再相互找错误，在此过程中读程序的能力也显著提高。在之前，让她们读五十行代码就很吃力了，但是现在，可能随便就是几百行了。

作为组长，我也知道自己的管理和组织能力不如有些其他组组长，导致了一些问题的出现。在以前，我更喜欢一个人去编程完成任务，不喜欢去合作去交流。但是面对这样一个较大型的程序，合作是必须的。起初地工作分配并不是很合理，导致组员各自工作量差别比较大，对此我感到特别内疚。再有就是对组员们进度的了解不够，对她们实现地模块也没有去按时的指导。也不是说要时时刻刻跟进，需要的只是在前几天去看一看，之后她们也就会自己修改了，这一点我也做的不够好，在最后也因为这个重写了很多代码。最后就是拖延心理，我们组本身进度还是比较快的，但是周末我的电脑出了点问题拿去修了，没有去监督组员们继续完成自己的计划，我自己也因为有些小骄傲而浪费了两三天，要不然我们的TTMS可能可以更加完善。在最后一天晚上整合程序和界面时发现时间不够，导致组员们辛苦写的界面没有完全应用上，这都是我当组长的不足。

我们小组是四个女生组成的，女生的优势在于自觉，平常学习时我们都比较踏实，基础比较牢固。遇到问题时也能够在互相帮助下去自己解决问题，而不是做伸手党或是直接放弃。这些也为我们为期一周半且困难重重的课程设计打下了好的开端。所以，基础是前提，都是一学期的C语言学习，学习态度决定了编程水平，在今后的学习过程中，我也会去脚踏实地地走好每一步，抱着对编程的热爱和自信，不断地去汲取知识，提升自己。

TTMS综合了我们学过的几乎所有知识，在实践中对知识进行了强化，在运用中查缺补漏，了解自己的知识漏洞，去及时补上。认真完成的人都会觉得这次课程设计让自己的C语言不再仅限于单个知识点的小程序，而是真正实际的运用。程序面对的也不再是评测系统，而是真正的用户，我们需要去考虑界面的美观程度，提示用户如何使用，并且控制用户的输入等等实际问题。也让我深深感受到了编程的趣味，自己一步一步写出来的程序，一步一步修改，到运行，真的很有成就感。累也是肯定的，一周半的时间，大概都有将近一半的时间小组成员集体熬夜，特别是最后两天，几乎通宵了。虽然很多时间花在了无用功上，但是这些时间换来的代码量，无疑不让我们每一个人提升了许多。

这次课设，收获不仅仅在于知识方面，更是团队合作能力，还有对我自己组织管理能力的提升，让我提前体验到开发的过程。希望在今后的学习过程中能够更加踏实，走好每一步。

## 3.4朱颖个人总结

### 3.4.1开发任务

**职责***：*副组长

**开发任务**：查询演出票，售票

### 3.4.2开发成果

（1）开发的模块：查询演出票，售票

（2）写的文档：用户手册里面自己开发的模块

（3）贡献： 1.完成自己的任务

2.每天检查组内成员的任务完成度并且汇总

3. 打分

### 3.4.3模块开发中难点问题及解决

（1）遇到的问题：函数传参遇到问题

解决问题的方法：通过查阅书籍理清楚思路

（2）遇到的问题：UI层的对齐问题

解决问题的方法：利用负值可以实现对齐

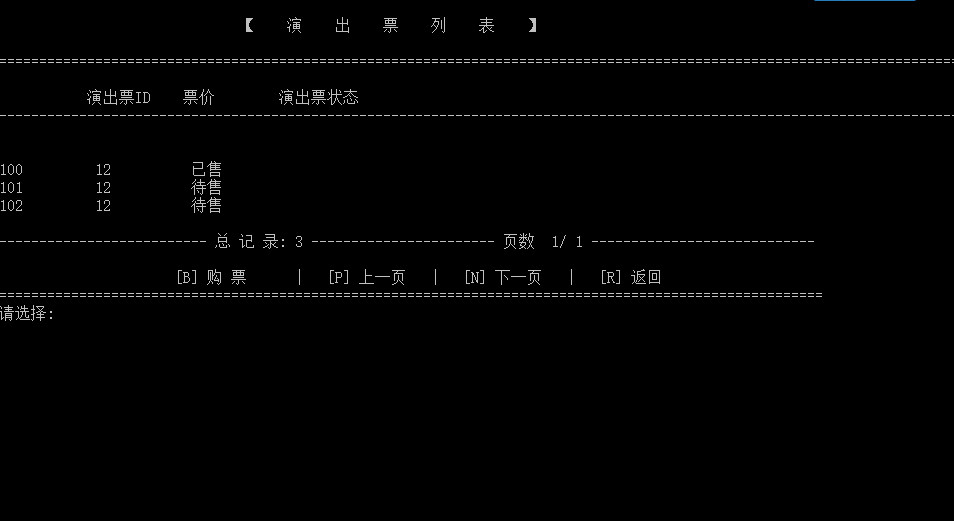
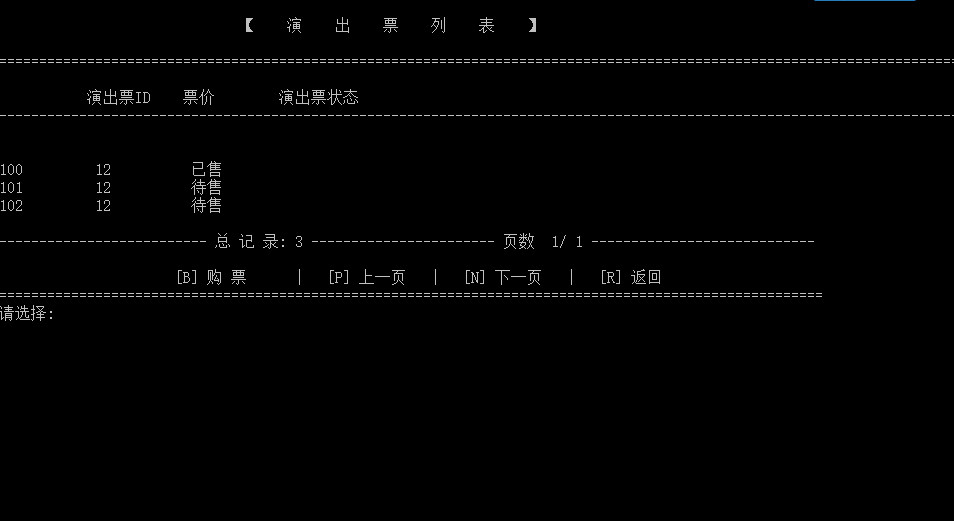
（3）遇到的问题：做出闪烁的效果

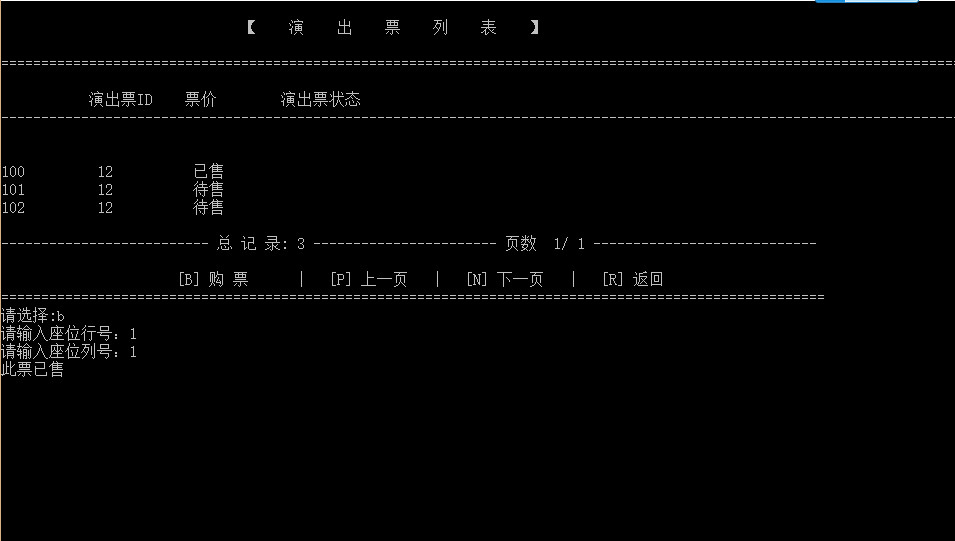
解决问题的方法：百度查到其中应该利用的是刷屏函数和sleep函数来实现其中的闪烁的效果

（4）遇到的问题：其中的在链表的时候建立的时候遇到问题

解决问题的方法：通过再次返回去查看其中有的时候会忘记申请内存

**3.4.4 模块测试**

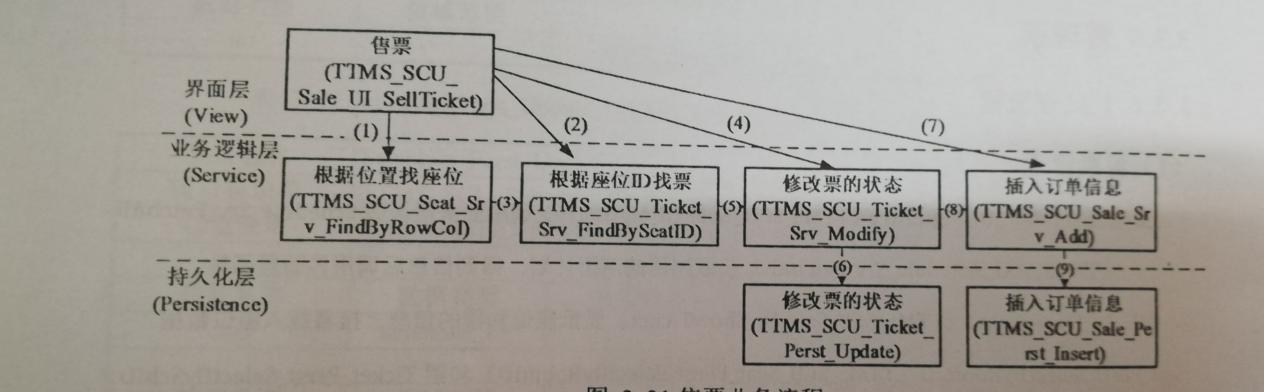
**



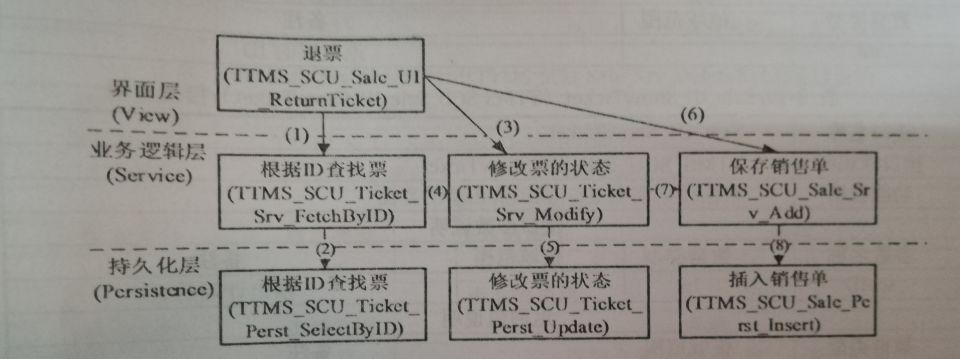
### 3.4.5收获与教训

刚拿到课程设计指导书，在组长的带领下我们搭建了eclipse环境，并粗略的阅读了这本书，在脑海里有了大致的框架。

这是售票的思路流程



这是退票的思路流程



通过这次课程设计， 我从中学到了很多东西。首先从专业知识方面来说 ，第一，通过老师给我们的框架，我知道了一个项目基本分三层，分别是持久化层，业务逻辑层和界面层。对于持久化层来说，一般是对文件进行一些操作，例如，读，写等之类的。对于业务逻辑层来说，一般是调用持久化层的一些东西，它的位置很关键，在整个项目中起到承上启下的作用，将界面层和持久化层联系起来。对于界面层来说，这是用户可以直观感受到的，所以在实现它的时候我们不仅仅考虑到了性能功能等方面的东西，与此同时我们也尽力做到美观，让用户有直观明了的体验，可能做得还不尽如人意，但这也是我们这次课程设计的一个小亮点吧 。第二，这次课程设计，把我们课堂上老师所讲的知识通过实践串了起来，平时很少用文件来做题练习之类的，但是这次的课程设计主要就是对链表及文件的一些操作，通过这次练习，使我将链表文件等的操作有了更为深刻的认识，或者是更加熟悉了它们的操作。第三，由于课堂上的时间紧张，有好多小的知识点老师来不及讲解，但是在这次实际操作中却又遇到了，所以我们不得不去通过各种方法去弄明白这个知识点，在这个过程中我也学到了以前不知道的新知识。总之这次的课程设计从知识方面不仅巩固了旧知识，同时也掌握了很多以前不知道的新东西，收获非凡。

其次，在软实力方面我也有所提升。第一，当我拿到课程设计的指导书时，我就发现书上的好多内容都是知识盲区。当时我就很疑惑，老师上课讲的东西我明明已经听懂了，可为什么还是什么都不会，此时我意识到仅仅靠课堂上老师讲师远远不够的，同时我们需要更多的实践，正如“纸上得来终觉浅，绝知此事要躬行”一样，想要更加熟练，透彻地掌握任何一门知识都少不了实践这一环节，所以通过这次的课程设计让我明白了，我们不仅仅要弄明白课堂上的知识，更要通过这些理论的东西去实践。第二，在实际的操作过程中不可避免的出现了很多问题，我们通过各种手段，例如询问老师，与同学交流，查找资料等方法解决掉，在这个过程中，我学到了一些学习方法，尤其是自学方面，从发现问题到解决问题，这都是莫大的收获，就像“授人以鱼不如授人以渔”一样的道理。对我来说，这方面可能影响我更加地深刻一些。第三，此次的课程设计我们通过小组合作的形式完成，在这个过程中，我知道了团队合作的重要性，组长通过用例来给我们分工，每个人都涉及了三个层的实践，有的部分有些重合，所以当出现了问题我们在一起商讨解决的办法，或者当自己思路堵塞的时候，通过参考小组成员的方法，从而找到了思路，或者交换两种不同的方法，这样我们就学到了更多的东西。

当然，在这个过程中我也走了不少弯路，在刚开始时，由于自己没好好看书，导致有些东西没连上，思路混乱，浪费了很多时间，还有就是对于对于传参的问题，当想让自己程序跑起来时，才发现有好多error，这时候才意识到时自己传参出了问题，对于这个问题，我觉得是自己基础知识还不够牢靠，正所谓“磨刀不误砍柴工”，得现有大致的思路才应该着手写代码，这样在以后的·进度中才能顺利进行。