

大作业：图书管理系统

ACM Class 2020



C

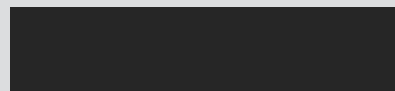
ONTENTS

01. 任务定义

02. 人员分工


03. 分数组成

04. 时间安排






PART. 01



任务定义



任务定义

- 作业目的：体验工程向软件开发过程
- 任务内容：实现一个图书管理系统
- 说明文档：<https://github.com/cmd2001/Bookstore-2020>

■ 基本要求

- 本次大作业需要使用C++实现一个网上书店管理系统，要求如下：
- 要求将数据存储**在文件中**
- 支持文档中给出的命令操作
- 通过给定的数据测试，不要求完成GUI
- 需提供命令行界面来操作该系统

交互模式

- 采用标准输入输出(stdin/stdout)
- 通过命令行进行直接操作
- 程序首次启动时(未初始化状态)，进行初始化：创建相关文件结构；自动新建超级管理员用户，用户名root，密码sjtu。
- 存在部分鲁棒性测试，对于非法操作，不要执行，输出「Invalid\n」(不含方)。

用户系统

- 用户分为三个角色：老板(超级管理员)、员工(管理员)、顾客(普通用户)。
- 老板：root账户，可以访问所有功能，可以进行用户管理。
- 员工：可以且仅可以访问与自己业务相关的功能、数据。
- 顾客：可以进行账户注册，查询书目以及书目购买，不能访问其他功能。
- 在权限上满足子集关系，即老板7>员工3>顾客1>未登录0。

访问权限

- 不同角色的数据访问：
- 老板：了解财务情况，了解员工情况，了解库存，管理各类用户，查看日志
- 员工：查看自己的操作记录，添加图书种类并进货，了解库存，修改书目信息
- 客户：查询图书，购买图书

用户指令

```
su [user-id] [passwd] #0:登录到某用户，从高权限用户登录到低权限不需填写密码
logout #1:返回到未登录状态
useradd [user-id] [passwd] [7/3/1] [name] #3:增加一个指定权限的用户，不能创建不小于自己权限的账户
register [user-id] [passwd] [name] #0:注册一个带有这些信息的权限1用户
delete [user-id] #7:删除某用户
passwd [user-id] [old-passwd(if not root)] [new-passwd] #1:root
不需要填写旧密码，其余账户需要
```

数据指令



- 为了简化指令模式，我们规定一个选定(select)机制(只能选择唯一图书)，修改和进货都在选中后进行。

```
select [ISBN] #3:选定ISBN为指定值的图书，若不存在则创建该ISBN的书并将其余信息留空
等待modify进行填写
modify -ISBN=[ISBN] -name=[name] -author=[author] -keyword=[keyword] -
price=[price] #3:参数可省略，也不要求输入顺序，会更新(替换而非添加)上次选中的书至
新的信息
import [quantity] [cost_price(in total)] #3:将上次选中的书以总共
[cost_price]的价格进[quantity]本
show -ISBN=[ISBN] -name=[name] -author=[author] -keyword=[keyword] #1:参
数可省略，也不要求输入顺序，将匹配的图书以ISBN号排序输出，需要注意该命令关键字项只支
持单关键字
show finance [time] #7:time项省略时，输出总的收入与支出;否则输出近[time]次进
货、卖出操作(分别算一次)的收入支出。
buy [ISBN] [quantity] #1:购买该ISBN号的图书[quantity]本
```



PART. 02

人员分工



人员分工

- 本次大作业通过流水线完成。
- 人员共有三种角色：文档、编码、测试。
- 每位同学在每个阶段会扮演不同角色。
- 每份作业一定由三位不同的同学完成。
- 对接方式为：文档->编码->测试。
- 人员对接方案将在一定时间节点后分步发布。（开盲盒）



详细解释


- 即：在本次大作业开始的一段时间，每位同学分别制订一份开发文档。
- 之后，开发文档会被交到另一位同学手中，由这位同学完成编码，通过OJ基础测试和鲁棒性测试A。
- 接下来，每位同学会与另一位负责测试的同学对接，对程序进行测试。测试后的程序将在之后尝试鲁棒性测试B。




PART. 03

分数组成





分数组成



- 本次大作业总分15分。
- 组成形式为：3+7+3+2。

Basic Task

- 文档：3分，该部分评分标准针对「文档同学」。
- 制订详细的开发文档，按此文档可正确实现全部功能。
- 评分方式：
- 1分：完成即可获得。
- 1分：助教评价。由助教评价你的最后一版文档完整度。
- 1分：同学评价。由依照你的文档(最后一版)编码的同学从指定方面进行评价。



Basic Task



- 编码：7分，该部分评分标准针对「编码同学」。
- 4分：完成程序，通过OJ基础测试，保证所有指令合法。
- 对于未通过所有测试的同学，我们将按照通过子任务(而不是测试点)百分比线性计算此部分分数。
- 3分：通过OJ鲁棒性测试A。该测试全部数据均会下发。
- 数据规模、时间限制、空间限制、磁盘限制、文件数量限制将于近期公布在OJ上。

Basic Task

- 测试：3分，该部分评分标准针对「测试同学」。
- 「测试同学」需要对「编码同学」的程序进行鲁棒性测试。
- 该分数中1分为参与度分，由「编码同学」对「测试同学」的参与度评分，如果积极参与测试则可获得。
- 2分为测试得分。展示日将在OJ上发布鲁棒性测试B，根据通过子任务百分比线性计算「测试同学」此部分分数。

Basic Task

- Code Review: 2分, 该部分分数针对编码同学。
- 在Code Review中正确展示测试中不涉及的三个功能: report finance、report employee、log, 即可得分。
- 进行小规模测试, 该部分数据(里数据)在之前阶段从未出现。不能通过会少量扣分。
- 若代码风格过于糟糕(封装不当)会少量扣分。
- 若代码与开发文档无法对应, 会造成大量扣分。

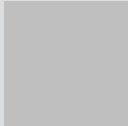

Bonus

- 与本次大作业15分无关的额外加分：(全部针对编码同学)
- 实现基于数字选择的用户界面 (0.5分)
- 实现Tab补全(Posix平台) (1.0分)
- 实现命令语法高亮(Posix平台) (1.0分)
- 实现GUI前端(Web也可) (1.0分)
- 实现日志式操作 (1.0分)
- 所有Bonus分数在期末直接加算入平时成绩(50分封顶)。



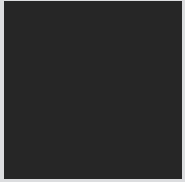
PART. 04

时间安排



■ 时间线

- Week 3: 发布图书管理系统 大作业
- Week 5: 文档DDL。检查文档，发布文档->编码对应名单。
- 注：因为Week 5(补课时间)在机考周，且Week 5(非补课时间)周四在十一期间，所以时间及形式可能有所调整。可能通过线下讨论、Zoom会议、腾讯会议、邮件形式进行。
- Week 7: 发布代码->测试对应名单。
- Week 8: 项目DDL。成果展示(完成Bonus的同学)+Code Review。



THANKS

Amagi_Yukisaki
2020.09.24

