



介绍 

测验

评测

互测

BUG 修复 讨论

创建新讨论

# 

# 面向对象设计与构造第十三次作业

\*

第一部分: 训练目标

本次作业,将以一个图书馆模拟系统为例,锻炼同学们**对程序架构的设计和抽象能力,以** 及加强对 UML 图的绘制训练。

第二部分: 预备知识

需要同学们掌握 UML 图的相关知识,以及 StarUML 的使用方法。在这里给出一些参考资 料。

第四单元手册

StarUML软件课程组提供下载地址

同学亦可参阅OO工具链教程的文末部分:

OO工具链教程

# 第三部分: 题目描述

### 一、情景描述

在一所小型图书馆中,同学们借阅图书需要遵守一定的规章制度。

图书馆里的所有图书按照 "**类别号-序列号**" 的形式编制**书号**(同学们可以理解为ISBN国际 标准书号的简化形式,即一本书的书号是唯一的,它的所有副本的书号都是相同的)。图 书分A、B、C三类,每种类别可能包含多个序列号(即每一类可能有多本书可供借阅), 每个书号的书可能具有多个副本。其中 A **类书仅能在图书馆阅览,不能外借; B 类书一** 人同一时间仅能借一本书的一个副本; C 类书, 对于每一个书号, 一人同一时间仅能借其 一个副本。

图书馆存在着以下几类可能和同学们直接交互的管理员, 每类管理员有且仅有一位:

- 借还管理员: 负责同学借 B 类书籍的登记, 以及 B 类书的归还。
- **整理管理** · 允素将同学还到自助机和借还管理员外的书完时摆放到架上。以及将预









此外, 图书馆有一台目助机器。

A

\*

?

若一位同学来到图书馆借阅书籍,ta 首先前往 **自助机器** 处,查阅其需要的书目是否有剩 余。 

如果所需书籍目前有余本在架上,那么该同学会到对应类别的书架处取书。如果是A类书 籍,该同学将在图书馆完成阅读 (阅读时间忽略不计,即假设同学从书架取书后立即完 成阅读,将书放回);如果是 B 类书籍,该同学会先到 借还管理员 处登记,若符合借书 数目限制则成功借书,否则书本 **当即被留在借还管理员处** ,借书失败;如果是 C 类书 籍,该同学会直接到 自助机器 处刷卡借书,若符合借书数目限制则借书成功,否则书本 **当即被留在自助机器处** , 借书失败。

当图书被留在自助机器处或借还管理员处,且没有在整理日(下文详述)上架之前,是不 能被借阅的。

如果所需书籍目前无余本在架上, 那么该同学会到 **预定管理员** 处登记预定。对于 B、C 类书籍的本数限制: 若同学手中无 B 类书, 不管是否有预定, 仍可以借阅架上的或预定 B 类书; 若同学目前持有一本 B 类书籍的副本,则再预定 B 类书籍当即被拒绝; C 类书籍 的数目限制同理。特别地,任何时候同一人对于相同书号图书的预定 仅有第一次被接受 ,并且一天之内同一人仅允许预定最多三本书的副本,更多的预订 **不会被接受** 。(不接 受的记录 当即被弃置,即使该同学之后获得了某本成功预定的书籍副本,本次失败预定 也不会被填充讲预定清单里。)

同学在看书过程中可能会 损毁书籍 或 丢失书籍。如果出现损毁,会在还书时被检查出 (假设同学一定会在某时刻还书);如果出现丢失,假设同学会立即到借还管理员处登 记,并缴纳罚款。丢失的图书不在占据借阅数目限制的计数,例如丢失了一本 B 类书副 本的同学在缴纳罚款后可以继续借阅 B 类书。

当一位同学来到图书馆还书:如果是 B 类书籍,该同学到 借还管理员 处还书, 若有损毁 情况,将 **立即向借还管理员缴纳罚款**,然后(不管是否有损毁都)还书成功;如果是 C 类书籍, 该同学到 **自助机器** 处还书, 若被扫描出有损毁情况, 将 **立即到借还管理员处缴 纳罚款**, 然后 (不管是否有损毁都) 在系统上被认定还书成功。

损毁的书会 **立即被送到后勤处修复**,且可以忽略图书修复时间,即认为书在后勤处 **立刻** 被修好,并且暂留在后勤处。

整理管理员每三天整理一次待上架书籍。即若上一次整理是 2023-01-01 ,则下一次整理 在 2023-01-04, 并且在当日开馆前完成整理。整理管理员 首先收集 借还管理员、自助机 器、后勤处的所有书本 ,然后将在预定清单上的书按数量送到 预定管理员 处,最后将其 余书摆放上架。紧接着,预定管理员会在 **按照预定的先后顺序** 通知相应同学来取书,可 以假设同学会立即到馆取走书籍。 特别地, 若某同学借到一本 B 类书籍的副本,则该同 学此前对于任何 B 类书籍的预定, 将从此刻起被取消。









- 1. 设计程序架构, 绘制并提交 UML 类图。
- 2. 根据所设计的架构进行编程实现,并提交程序代码。
- - 在这里,我们建议同学们按照"**设计程序架构->绘制 UML 类图->编程实现**"的顺序完成。 作业,以达到对于程序架构的设计和抽象能力的训练。同时请注意,你的程序架构应和 UML 类图保持一致,更多评测要求详见本指导书第四部分。
- ?
- 另外,尽管本次作业没有要求对书的状态的状态图绘制和对同学与管理员之间交互的时序 图绘制的测评,但仍然强烈建议同学在设计程序架构阶段,利用状态图和时序图梳理题目 描述中书可能的状态 (例如: 在书架上、在借还管理员处、已丢失等) 和各个实体之间的 交互,从而准确把握业务逻辑,做好设计,便于程序实现。

# 第四部分: 评测标准

#### 类图

评测分为三部分:

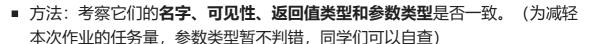
- 1. 对 UML 类图正确性的检查, 检查项目有:
  - 。 类图中的所有元素,**除了**以下元素之外,其余元素的 name 字段不能为空。
    - direction 为 return 的 UMLParameter
    - UMLAssociation
    - UMLAssociationEnd
    - UMLGeneralization
    - UMLInterfaceRealization
  - 。 不能含有重名的类。
  - 。 类不能有重名的属性。
  - 。 不能有循环继承。
  - 任何一个类或接口不能重复继承另外一个类或接口。
  - 。 接口的所有属性和方法均需要为public。
- 2. 对程序正确性的检查, 仍采用和先前作业相同的传统方式。关于输入输出描述以及数 据规模, 详见本指导书第五部分。
- 3. 对 UML 类图和程序一致性的检查,检查项目有:
  - 。 类图与程序能互相找到名字相同的类。











- 类图中对于 接口(Interface) 的要求与上述类(Class)的要求相同。
- 另外, 类图中每个枚举类与代码中的枚举类需要一致
  - 所定义的**枚举项**要求与代码中相应枚举类所定义**枚举项**名称相同,且是——对 应
- 2 4. 关于UML 类图和程序一致性的检查,有以下说明:
  - 。 对于所提交的UML类图和所提交程序
    - 设类图中的设计单元集合为MC,程序中的实现单元集合为CC
    - "单元"包括class和interface
  - 对于MC中任意一个元素Ma
    - Ma中属性个数或方法个数不少于1
    - CC中一定存在一个元素Ca与Ma相对应,且Ma中属性数目不少于Ca中属性数目 的60%,同时Ma中方法数目不少于Ca中方法数目的60%
  - 。 对于CC中的任意一个元素Ca
    - Mc中必然存在一个元素Ma相对应
  - 对于MC中任意两个元素Ma和Mb的关系Rab
    - CC中必然有对应的两个元素Ca和Cb,且Ca和Cb同样具有相同的关系Rab,反 之亦然

#### 5. 提示与注意:

- 请同学们将画好的UML图命名为 uml.mdj , 并确保类图对应的UMLModel的name 字段为 Model , 提交至仓库目录最外层, 否则会导致评测失败。
- 完成作业时应先做好相应的设计,在完成UML图绘制后再实现具体代码。当然,代 码变化后是可以回头更新UML图的,但绝不应完成代码后再匆匆补上UML图。
- 为了保证同学们写代码时的灵活性,在实现具体代码时,可以根据需要在代码中实 现不属于UML图中的**属性**和**方法**,但UML图中的属性和方法应在程序中进行实 现,具体检查方法请见前文说明。**注意,将人性化设计作为偷鸡的方式不被允许** 的,请同学们认真完成作业。
- debug时可以通过阅读mdi代码检查各字段是否符合预期
- 对于类图中的方法参数,请保证其在mdi文件中的先后顺序与程序中的一致。另 外,每一个方法的RETURN类参数应有且仅有一个。

由于对于 UML 图的检查以及对于设计的评判较为复杂灵活,本评测仅为判断同学们设计 合理的必要条件。即通过评测的 UML 图和程序在此次作业中即被认为通过,但不代表同 学的设计和实现没有其它隐藏问题,也不代表就已经是十分优秀的设计。鼓励同学们在完













#### 一、程序输入

程序输入由两部分组成:图书馆初始书目信息和学生借还书动态信息。

输入第一行为一个正整数 n, 代表图书馆中共有 n 本不同的图书。

接下来 n 行每行对应一本书的详细信息,格式为"**类别号-序列号 副本数**",例如 "B-0001 10" 代表 B 类下序号为 0001 的书 (即书号为B-0001的书) 馆藏 10 个副本。序列号为四 ? 位数字。输入保证不存在相同的"类别号-序列号",不同类别号下可以存在相同的序列。

号,每个书号的书最多 10 个副本。

接下来的一行为一个正整数 m, 代表有 m 条学生借还书动态信息。

接下来 m 行每行对应一条动态信息,格式为"[YYYY-mm-dd] <学号> <操作> <类别号-序列号>", 代表在日期为 [YYYY-mm-dd] 的当天, 学生 <学号> 对图书 <类别号-序列号 > 讲行 <操作>。

#### 具体的 <操作> 种类:

• borrowed: 借书。若程序运行正确,保证最开始图书馆拥有该书目。

• smeared: 损毁书。若程序运行正确,保证此时该学生已拥有该书。

• lost: 丢书。若程序运行正确,保证此时该学生已拥有该书。

• returned:还书。若程序运行正确,保证此时学生已拥有该书。

为简化逻辑,输入数据保证同学在一次借书过程中,损毁、丢失两种违规行为至多发生一 种。

对于借还书动态信息,输入保证已按时间先后顺序排序。也即同一日期的全部信息,可理 解为按照输入出现的顺序依次从早到晚发生。 图书馆在 [2023-01-01] 开馆, 在该日期开馆 前,整理管理员第一次整理了所有书本。

1≤n≤100, 1≤m≤100。 所有事件在 [2023-01-01] 至 [2023-12-31] 之间发生,保证日期 合法。

# 二、程序输出

当程序运行时发生表格中出现的场景时,要求输出相应信息。每条信息占一行。

场景	输出	
学生查询书籍信息时 输出	"[YYYY-mm-dd] <学号> queried <类别号-序列号> from < 服务部门>"	

























	<版分引 」>
学生登记预定时输出	"[YYYY-mm-dd] <学号> ordered <类别号-序列号> from <服务部门>"
学生被收取赔偿金时 输出	"[YYYY-mm-dd] <学号> got punished by <服务部门>"
学生成功还书时输出	"[YYYY-mm-dd] <学号> returned <类别号-序列号> to <服务部门>"
书籍被修复时输出	"[YYYY-mm-dd] <类别号-序列号> got repaired by <服务部 门>"

其中, <服务部门>要求程序输出该操作具体由图书馆何部门负责,选项有:

• self-service machine: 自助机器。

• borrowing and returning librarian: 借还管理员。

• arranging librarian:整理管理员。 • ordering librarian: 预定管理员。

• logistics division: 后勤处。

注: "成功借书"包括学生成功取到预定书籍的情况。 要求输出按照时间先后顺序排序,同 一日期的信息输出顺序也应与事件发生的先后顺序相同。

#### 三、样例

输入	输出	说明
3	[2023-01-05] 21987640 queried A-	展示了借阅三
A-0000 3	0000 from self-service machine	类图书的情况
B-0000 2	[2023-01-05] 21890631 queried C-	
C-0000 2	0000 from self-service machine	
5	[2023-01-05] 21890631 borrowed C-	
[2023-01-05]	0000 from self-service machine	
21987640 borrowed	[2023-01-06] 21987640 queried B-	
A-0000	0000 from self-service machine	
[2023-01-05]	[2023-01-06] 21987640 borrowed B-	
21890631 borrowed	0000 from borrowing and returning	
C-0000	librarian	
[2023-01-06]	[2023-01-10] 21890631 returned C-	
21987640 borrowed	0000 to self-service machine	











课程团队



















[2023-01-13] 21237643 borrowed B-

[2023-01-20] 21237643 returned B-

0000 to borrowing and returning

0000 from ordering librarian





关于



librarian

0000

0000

[2023-01-20]

21237643 returned B-