1．为一个名为quadratic-solver的函数设计黑盒测试集。quadratic-solver接受三个浮点数据(a，b，c)，这三个数表示一个二次方程，形式为ax²+bx+c=O，计算并显示解。

等价类划分表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入及外部条件 | 有效等价类 | 等价类编号 | 无效等价类 | 等价类编号 |
| 数据的类型 | 浮点数 | 1 | 非数字字符 | 6 |
| a | 不为0.0的任意浮点数 | 2 | 0.0 | 7 |
| 超出浮点数范围的数字 | 8 |
| b | 任意浮点数 | 3 | 超出浮点数范围的数字 | 9 |
| c | 任意浮点数 | 4 | 超出浮点数范围的数字 | 10 |
| b\*b-4\*a\*c | b\*b-4\*a\*c>=0 | 5 | b\*b-4\*a\*c<0 | 11 |

有效等价类设计测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 输入数据 | | | 预期输出 | 覆盖范围（等价类编号） |
| a | b | c | x |
| 1 | 1.0 | -2.0 | 1.0 | 1.0 | 1，2，3，4，5 |

无效等价类设计测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 输入数据 | | | 预期输出 | 覆盖范围（等价类编号） |
| a | b | c | x |  |
| 1 | abc | 3.5 | 4.7 | 输入无效 | 6 |
| 2 | 0.0 | 5.2 | 3.0 | 输入无效 | 7 |
| 3 | -10^200 | 7.0 | 2.0 | 输入无效 | 8 |
| 4 | 3.2 | 10^250 | 3.7 | 输入无效 | 9 |
| 5 | 5.5 | 2.7 | 10^300 | 输入无效 | 10 |
| 6 | 2.0 | 1.0 | 3.0 | 输入无效 | 11 |

2．在饮料自动销售系统中，顾客从机器的前端钱币口投入钱币，然后选择想要的饮料。钱币到达钱币记录仪，记录仪更新自己的选择。正常时记录仪通知分配器分发饮料到机器前端，但可能饮料已售完，也可能用完了找给顾客的零钱而无法销售饮料。先写出“买到饮料”的场景、“饮料已售完”的场景以及“机器没有合适的零钱”的场景，然后根据场景，用UML分别表示出“买到饮料”、“饮料已售完”以及“机器没有合适的零钱”的顺序图。



3．某图书管理系统有以下功能：

(1) 借书：输入读者借书证。系统首先检查借书证是否有效，若有效，对于第一次借书的读者，在借书文件上建立档案。否则，查阅借书文件，检查该读者所借图书是否超过10本，若已达10本，拒借，未达10本，办理借书（检查该读者目录并将借书情况登入借书文件）。

(2) 还书：从借书文件中读出与读者有关的记录，查阅所借日期，如果超期（3个月）作罚款处理。否则，修改库存目录与借书文件。

(3) 查询：可通过借书文件、库存目录文件查询读者情况、图书借阅情况及库存情况，打印各种统计表。

①用结构化方法画出系统顶层图、0层图，写出数据字典。

②画出该系统的初始化模块结构图。

③用面向对象方法画出该系统有哪些主要的类。

顶层图



0层图



数据字典：

|  |  |  |
| --- | --- | --- |
| **数据流名** | **数据流定义** | **注释** |
| 借书号 | 借书号={读者ID} |  |
| 借书请求 | 借书请求={图书ID} |  |
| 还书请求 | 还书请求={图书ID} |  |
| 查询请求 | 查询请求={图书ID} |  |
| 处罚信息 | 处罚信息={图书ID+处罚金额} |  |
| 查询结果 | 查询结果={图书ID+图书名+是否借出} | 在0层图中为“借阅信息”流 |
| 操作读者信息 | 操作读者信息={读者ID} |  |
| 修改借书信息 | 修改结束信息={读者ID+图书ID+借阅时间} | 修改包括新建、修改、删除 |
| 返回借书信息 | 返回借书信息={读者ID+图书ID+借书时间} |  |

|  |  |  |
| --- | --- | --- |
| **数据存储名** | **记录定义** | **注释** |
| 借阅数据库 | 文件名：借阅数据库  记录定义：借阅数据={读者ID+图书ID+借书日期}  读者ID：8为数字  图书ID：13位字符串  借书日期：日期型 | 主键为：读者ID+图书ID |
| 读者数据库 | 文件名：读者数据库  记录定义：读者数据={读者ID }  读者ID：8为数字  文件组织：以读者ID为关键字递增排列 | 未涉及用户名 |
| 图书数据库 | 文件名：图书数据库  记录定义：图书数据={图书ID+图书名}  图书ID：13位字符串  图书名：50位字符串  文件组织：以图书ID为关键字递增排列 |  |





注：本设计十分丧尸，搜索图书是否借阅、某人借阅图书本数均通过条件搜索图书借阅表完成，是一种很低效的设计，但完成了所要求的全部条件