

今日内容

- ▶ 黑盒测试过程
- ▶ 性能测试过程
- ▶ 测试标准
- ▶ 控制与发现问题
- ▶ 测试实验室的建设

黑盒测试过程

软件测试技术

- ▶ 黑盒测试（功能测试）
- ▶ 白盒测试（代码测试）
- ▶ 性能测试（负载测试）

黑盒测试的特点

▶ 优点

- 比较简单，与软件的内部实现无关；
- 从用户角度出发；
- 基于软件开发文档。

▶ 缺点

- 覆盖率较低，大概只能达到总代码量的30%；
- 自动化测试的复用性较低。

测试策略

- ▶ 通过测试
- ▶ 失败测试

测试用例

- ▶ 为达到最佳的测试效果或高效的揭露隐藏的错误而精心设计的少量测试数据
- ▶ 使用测试用例令软件测试的实施重点突出、目的明确
- ▶ 一个好的测试用例是在于它能发现至今未发现的错误？

测试用例的组成

- ▶ 测试设计说明
- ▶ 测试用例说明
- ▶ 测试程序说明

测试设计说明

- ▶ 提炼测试方法，明确指出设计包含的特性及其相关测试。如果要求完成测试还明确指出测试用例和测试程序，指定特性通过/失败的规则。

测试设计说明包含的要素

- ▶ 标识符
- ▶ 要测试的特性
- ▶ 方法
- ▶ 测试用例论证
- ▶ 通过/失败规则

测试用例追踪关系

测试定义标识	测试定义	用例标识	用例名称
CSTCDF055830219	1.1登录退出	CSTCCS055830178	1.1登录退出
CSTCDF055830113	1.2公共信息管理	CSTCCS055830179	1.2.1新增公共信息
		CSTCCS055830180	1.2.2查看公共信息
		CSTCCS055830181	1.2.3搜索公共信息
		CSTCCS055830182	1.2.4发布公共信息
CSTCDF055830114	1.3频率申请管理	CSTCCS055830183	1.3.1查看频率申请信息
		CSTCCS055830184	1.3.2搜索频率申请信息
		CSTCCS055830185	1.3.3频率申请统计
		CSTCCS055830186	1.3.4导出频率申请信息
		CSTCCS055830187	1.3.5审批频率申请信息
CSTCDF055830115	1.4干扰信息管理	CSTCCS055830188	1.4.1新增干扰信息

测试用例说明

- ▶ 编写用于输入的实际数值和预期结果。测试用例还明确指出使用具体测试用例产生的测试程序的任何限制。

测试用例说明包含的要素

- ▶ 标识符
- ▶ 测试项
- ▶ 输入说明
- ▶ 输出说明
- ▶ 环境要求
- ▶ 特殊要求
- ▶ 用例依赖性

测试用例示例

- 用例标识
- 用例名称
- 被测功能
- 用例目的
- 数据准备
- 测试步骤
- 预期结果
- 实际结果
- 测试人员
- 测试日期

测试用例序号		用例名称	节假日表维护	被测功能	国际收支统计监测系统 (总局) 节假日表维护
测试用例目的	测试节假日表功能是否符合用户要求				
数据准备	定义节假日规则。建立可以修改节假日表的系统用户				
测试步骤	a) 在系统用户界面中输入用户名、密码以及验证码 b) 点击确定，查看是否可以成功登录 c) 转到节假日表维护功能模块 d) 改变当前的节假日设置 e) 重新登录系统，查看节假日是否已变化 f) 发送节假日表到分支局 g) 检查分支局是否接收到数据，并对节假日表做相应修改				
期待输出结果	用户登陆正常，可以完成节假日表的修改和下发，分支局可以成功接收数据。				
实际输出结果					
测试人员 A		测试人员 B		审核人	
测试日期					

测试程序说明

- ▶ 明确指出为实现相关测试设计而操作软件系统和试验具体测试用例的全部步骤。

测试程序说明包含的要素（一）

- ▶ 标识符
- ▶ 目的
- ▶ 特殊要求
- ▶ 程序步骤
- ▶ 日志
- ▶ 设置
- ▶ 启动
- ▶ 程序

测试程序说明包含的要素（二）

- ▶ 衡量标准
- ▶ 关闭
- ▶ 终止
- ▶ 重置
- ▶ 偶然事件

一个测试程序说明的例子

标识号：计算器程序

目的：这个程序描述执行加法测试用例必须步骤

特殊要求：本次测试不需要特殊的硬件和软件

程序步骤：

日志：测试员按测试要求记录程序执行过程，所有必须填写的项都必须填写，包括问题的记录。

设置：测试者必须安装Windows2000的干净副本，使用测试工具Tool-A和Tool-B等等。

启动：启动Windows2000，点击开始按钮，选择程序，选择附件、选择计算器。

程序：用键盘输入每个测试用例并比较结果。

衡量标准：...

测试用例的组成

- ▶ 测试设计说明
- ▶ 测试用例说明
- ▶ 测试程序说明

黑盒测试方法

- ▶ 等价类划分
- ▶ 边界值分析
- ▶ 因果图
- ▶ 场景法
- ▶ 错误推测法

等价类划分

等价类划分的办法是把程序的输入域划分成若干部分，然后从每个部分中选取少数代表性数据当作测试用例。

示例：三角形的等价划分

题干：根据下面给出的规格说明，利用等价类划分的方法，给出足够的测试用例。“一个程序读入3个整数，把这三个数值看做一个三角形的3条边的长度值。这个程序要打印出信息，说明这个三角形是不等边的、是等腰的、还是等边的。”

Step1: 分析三角形的特点

- ▶ 3条边分别为A, B, C。满足: $A > 0$, $B > 0$, $C > 0$, 且 $A + B > C$, $B + C > A$, $A + C > B$;
- ▶ 等腰需满足 $A = B$, 或 $B = C$, 或 $A = C$;
- ▶ 等边需满足 $A = B$, 且 $B = C$, 且 $A = C$;

Step2: 列出三角形的等价类列表

输入条件	有效等价类	无效等价类
是否三角形的 3 条边	$(A > 0)$, (1) $(B > 0)$, (2) $(C > 0)$, (3) $(A + B > C)$, (4) $(B + C > A)$, (5) $(A + C > B)$ (6)	$(A \leq 0)$, (7) $(B \leq 0)$, (8) $(C \leq 0)$, (9) $(A + B \leq C)$, (10) $(B + C \leq A)$, (11) $(A + C \leq B)$ (12)
是否等腰三角形	$(A = B)$, (13) $(B = C)$, (14) $(C = A)$ (15)	$(A \neq B)$ and $(B \neq C)$ and $(C \neq A)$ (16)
是否等边三角形	$(A = B)$ and $(B = C)$ and $(C = A)$ (17)	$(A \neq B)$, (18) $(B \neq C)$, (19) $(C \neq A)$ (20)

Step3: 设计三角形的测试用例

序号	【A, B, C】	覆盖等价类	输出
1	【3, 4, 5】	(1), (2), (3), (4), (5), (6)	一般三角形
2	【0, 1, 2】	(7)	不能构成三角形
3	【1, 0, 2】	(8)	
4	【1, 2, 0】	(9)	
5	【1, 2, 3】	(10)	
6	【1, 3, 2】	(11)	
7	【3, 1, 2】	(12)	
8	【3, 3, 4】	(1), (2), (3), (4), (5), (6), (13)	
9	【3, 4, 4】	(1), (2), (3), (4), (5), (6), (14)	
10	【3, 4, 3】	(1), (2), (3), (4), (5), (6), (15)	
11	【3, 4, 5】	(1), (2), (3), (4), (5), (6), (16)	非等腰三角形
12	【3, 3, 3】	(1), (2), (3), (4), (5), (6), (17)	是等边三角形
13	【3, 4, 4】	(1), (2), (3), (4), (5), (6), (14), (18)	非等边三角形
14	【3, 4, 3】	(1), (2), (3), (4), (5), (6), (15), (19)	
15	【3, 3, 4】	(1), (2), (3), (4), (5), (6), (13), (20)	

边界值分析

边界值分析法是一种补充等价划分的测试用例设计技术，它不是选择等价类的任意元素，而是选择等价类边界的测试用例。

边界值设计遵守的原则

- 1) 如果输入条件规定了取值范围，应以该范围的边界内及刚刚超范围的边界外的值作为测试用例；
- 2) 若规定了值的个数，分别以最大、最小个数及稍小于最小、稍大于最大个数作为测试用例；
- 3) 针对每个输出条件使用前面的第1) 和2) 条原则；
- 4) 分析规格说明，找出其他的可能边界条件。

示例：边界条件缺陷的程序

```
1: Rem Create a 10 element integer array
2: Rem Initialize each element to -1
3: Dim data(10) As Integer
4: Dim I As Integer
5: For I=1 TO 10
6: data(i)=-1
7: Next i
8: End
```

边界问题会在哪儿呢？

data(1)=-1 data(2)=-1
data(3)=-1 data(4)=-1
data(5)=-1 data(6)=-1
data(7)=-1 data(8)=-1
data(9)=-1 data(10)=-1

data(0)=0

因果图法

因果图方法的思路是：从用自然语言书写的程序规格说明的描述中找出因（输入条件）和果（输出或程序状态的改变），通过因果图转换为判定表。

因果图法的设计步骤

- 1) 分析程序规格说明的描述中，哪些是原因，哪些是结果；
- 2) 分析程序规格说明的描述中语义的内容，并将其表示成连接各个原因与各个结果的“因果图”；
- 3) 由于语法或环境的限制，有些原因和结果的组合情况是不可能出现的。为表明这些特定的情况，在因果图上使用若干个特殊的符号标明约束条件；
- 4) 把因果图转换成判定表；
- 5) 为判定表中每一列表示的情况设计测试用例。

示例：自动售货机

产品说明书：有一个处理单价为1元钱的盒装饮料的自动售货机软件。若投入1元硬币，按下“可乐”、“雪碧”、或“红茶”按钮，相应的饮料就送出来。若投入的是2元硬币，在送出饮料的同时退还1元硬币。

Step1: 原因和结果

原因:

投入1元钱
投入2元钱
按可乐按钮
按雪碧按钮
按红茶按钮

结果:

退还1元钱
送出可乐饮料
送出雪碧饮料
送出红茶饮料

Step2: 因果图

输入条件（原因）

输出条件（结果）

投入1元硬币

投入2元硬币

按“可乐”按钮

按“雪碧”按钮

按“红茶”按钮

(21)

(22)

(23)

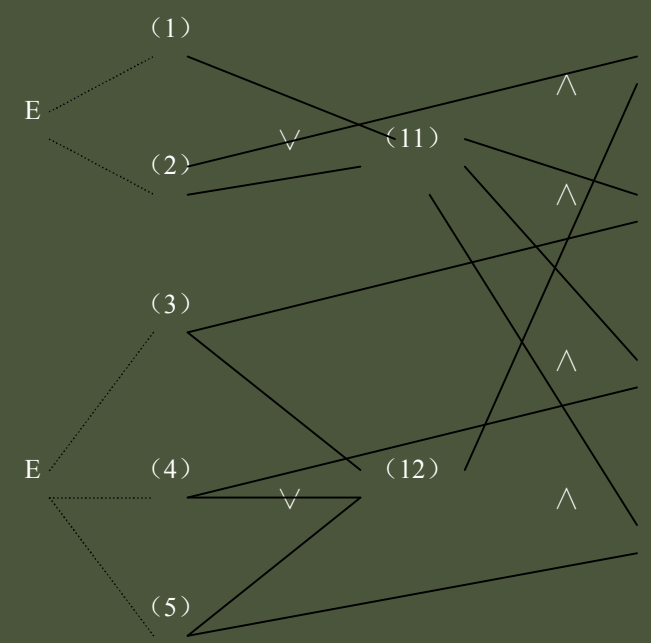
(24)

退还1元硬币

送出“可乐”饮料

送出“雪碧”饮料

送出“红茶”饮料



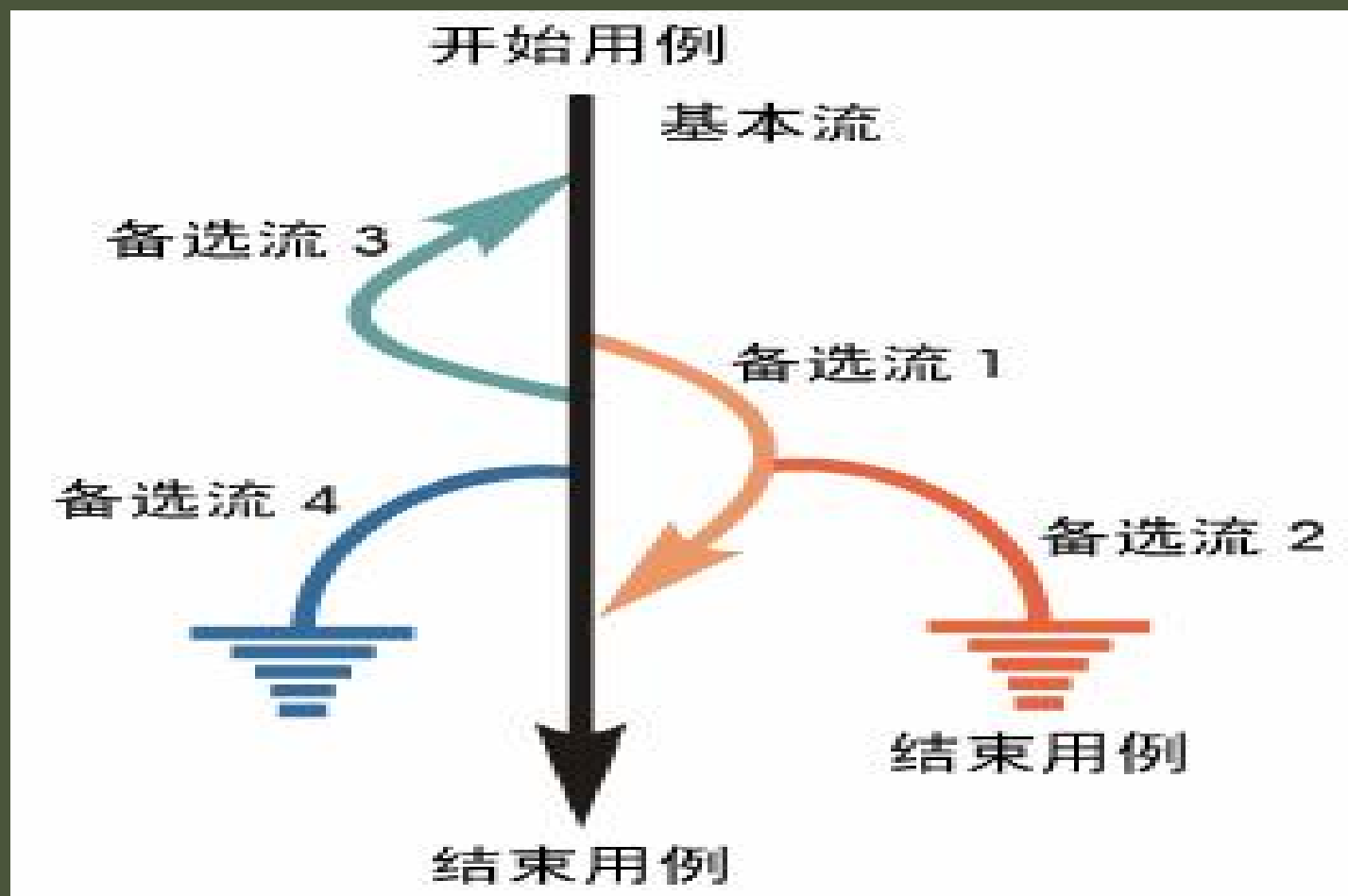
Step3: 测试用例

			1	2	3	4	5	6	7	8	9	10	11
输入	投入 1 元硬币	(1)	1	1	1	1	0	0	0	0	0	0	0
	投入 2 元硬币	(2)	0	0	0	0	1	1	1	1	0	0	0
	按“可乐”按钮	(3)	1	0	0	0	1	0	0	0	1	0	0
	按“雪碧”按钮	(4)	0	1	0	0	0	1	0	0	0	1	0
	按“红茶”按钮	(5)	0	0	1	0	0	0	1	0	0	0	1
中间 结点	已投币	(11)	1	1	1	1	1	1	1	1	0	0	0
	已按钮	(12)	1	1	1	0	1	1	1	0	1	1	1
输出	退还 1 元硬币	(21)	0	0	0	0	1	1	1	0	0	0	0
	送出“可乐”饮料	(22)	1	0	0	0	1	0	0	0	0	0	0
	送出“雪碧”饮料	(23)	0	1	0	0	0	1	0	0	0	0	0
	送出“红茶”饮料	(24)	0	0	1	0	0	0	1	0	0	0	0

场景法

- ▶ 用例场景是通过描述流经用例的路径来确定的过程，这个流经过程要从用例开始到结束遍历其中所有基本流和备选流。

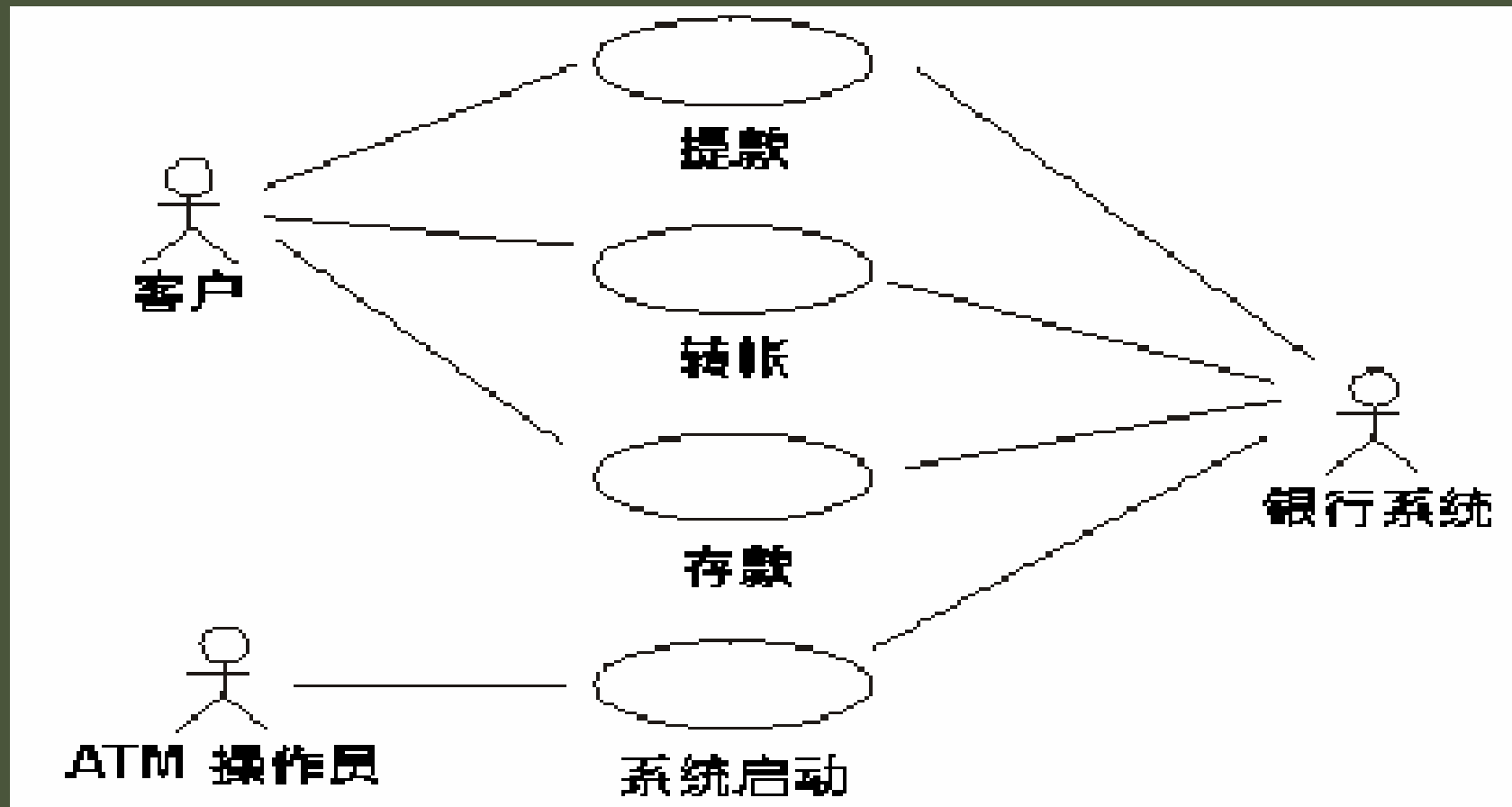
用例场景



用例场景描述

- ▶ 场景1 基本流
- ▶ 场景2 基本流 备选流1
- ▶ 场景3 基本流 备选流1 备选流2
- ▶ 场景4 基本流 备选流3
- ▶ 场景5 基本流 备选流3 备选流1
- ▶ 场景6 基本流 备选流3 备选流1 备选流2
- ▶ 场景7 基本流 备选流4
- ▶ 场景8 基本流 备选流3 备选流4

示例：ATM（业务模型）



Step1: ATM基本流（一）

- ▶ 步骤1: 准备提款- 客户将银行卡插入ATM 机的读卡机;
- ▶ 步骤2: 验证银行卡- ATM 机从银行卡的磁条中读取帐户代码, 并检查它是否属于可以接收的银行卡;
- ▶ 步骤3: 输入PIN - ATM 要求客户输入PIN 码;
- ▶ 步骤4: 验证帐户代码和PIN - 验证帐户代码和PIN 以确定该帐户是否有效以及所输入的PIN 对该帐户来说是否正确;
- ▶ 步骤5: ATM 选项- ATM 显示在本机上可用的各种选项。在此事件流中, 银行客户通常选择“提款”;

Step1: ATM基本流 (二)

- ▶ 步骤6: 输入金额- 要从ATM 中提取的金额;
- ▶ 步骤7: 授权-ATM 通过将卡ID、PIN、金额以及帐户信息作为一笔交易发送给银行系统来启动验证过程;
- ▶ 步骤8: 出钞- 提供现金;
- ▶ 步骤9: 收据- 打印收据并提供给客户。ATM 还相应地更新内部记录;
- ▶ 步骤10: 返回银行卡- 银行卡被返还。

Step2: ATM备选流（一）

备选流 1 - 银行卡无效	在基本流步骤 2 中 - 验证银行卡，如果卡是无效的，则卡被退回，同时会通知相关消息。
备选流 2 - ATM 内没有现金	在基本流步骤 5 中 - ATM 选项，如果 ATM 内没有现金，则“提款”选项将无法使用。
选流 3 - ATM 内现金不足	在基本流步骤 6 中- 输入金额，如果 ATM 机内金额少于请求提取的金额，则将显示一则适当的消息，并且在步骤 6 - 输入金额处重新加入基本流。
选流 4 - PIN 有误	在基本流步骤 4 中- 验证帐户和 PIN，客户有三次机会输入 PIN。如果 PIN 输入有误，ATM 将显示适当的消息；如果还存在输入机会，则此事件流在步骤 3 - 输入 PIN 处重新加入基本流。 如果最后一次尝试输入的 PIN 码仍然错误，则该卡将被 ATM 机保留，同时 ATM 返回到准备就绪状态，本用例终止。
选流 5 - 帐户不存在	在基本流步骤 4 中 - 验证帐户和 PIN，如果银行系统返回的代码表明找不到该帐户或禁止从该帐户中提款，则 ATM 显示适当的消息并且在步骤 9 - 返回银行卡处重新加入基本流。

Step2: ATM备选流（二）

选流 6 - 帐面金额不足	在基本流步骤 7 - 授权中，银行系统返回代码表明帐户余额少于在基本流步骤 6 - 输入金额内输入的金额，则 ATM 显示适当的消息并且在步骤 6 - 输入金额处重新加入基本流。
选流 7 - 达到每日最大的提款金额	在基本流步骤 7- 授权中，银行系统返回的代码表明包括本提款请求在内，客户已经或将超过在 24 小时内允许提取的最多金额，则 ATM 显示适当的消息并在步骤 6 - 输入金额上重新加入基本流。
选流 x - 记录错误	如果在基本流步骤 10 - 收据中，记录无法更新，则 ATM 进入“安全模式”，在此模式下所有功能都将暂停使用。同时向银行系统发送一条适当的警报信息表明 ATM 已经暂停工作。
选流 y - 退出	客户可随时决定终止交易（退出）。交易终止，银行卡随之退出。
选流 z - “翘起”	ATM 包含大量的传感器，用以监控各种功能，如电源检测器、不同的门和出入口处的测压器以及动作检测器等。在任一时刻，如果某个传感器被激活，则警报信号将发送给警方而且 ATM 进入“安全模式”，在此模式下所有功能都暂停使用，直到采取适当的重启/重新初始化的措施。

Step3: ATM (场景)

场景 1 - 成功的提款	基本流	
场景 2 - ATM 内没有现金	基本流	备选流 2
场景 3 - ATM 内现金不足	基本流	备选流 3
场景 4 - PIN 有误 (还有输入机会)	基本流	备选流 4
场景 5 - PIN 有误 (不再有输入机会)	基本流	备选流 4
场景 6 - 帐户不存在/帐户类型有误	基本流	备选流 5
场景 7 - 帐户余额不足	基本流	备选流 6

注：为方便起见，备选流3和6（场景3和7）内的循环以及循环组合未纳入上表。

Step4: ATM (测试用例矩阵)

TC (测试用例) ID 号	场景/条件	PIN	帐号	输入的金 额 (或选 择的金 额)	帐面 金额	ATM 内的金 额	预期结果
CW1.	场景 1 - 成功的提款	V	V	V	V	V	成功的提款。
CW2.	场景 2 - ATM 内没有现金	V	V	V	V	I	提款选项不可用, 用例结束
CW3.	场景 3 - ATM 内现金不足	V	V	V	V	I	警告消息, 返回基本流步骤 6 - 输入金额
CW4.	场景 4 - PIN 有误 (还有不止一次输入机会)	I	V	n/a	V	V	警告消息, 返回基本流步骤 4, 输入 PIN
CW5.	场景 4 - PIN 有误 (还有一次输入机会)	I	V	n/a	V	V	警告消息, 返回基本流步骤 4, 输入 PIN
CW6.	场景 4 - PIN 有误 (不再有输入机会)	I	V	n/a	V	V	警告消息, 卡予保留, 用例结束

Step5: ATM (测试用例数据)

TC (测试用例) ID 号	场景/条件	PIN	帐号	输入的金 额或 选择的 金额	帐面金 额	ATM 内 的金额	预期结果
CW1.	场景 1 - 成功的提款	4987	809 - 498	50.00	500.00	2,000	成功的提款。帐户 余额被更新为 450.00
CW2.	场景 2 - ATM 内没有现金	4987	809 - 498	100.00	500.00	0.00	提款选项不可用， 用例结束
CW3.	场景 3 - ATM 内现金不足	4987	809 - 498	100.00	500.00	70.00	警告消息，返回基 本流步骤 6-输入金 额
CW4.	场景 4 - PIN 有误 (还有不 止一次输入机会)	4978	809 - 498	n/a	500.00	2,000	警告消息，返回基 本流步骤 4，输入 PIN
CW5.	场景 4 - PIN 有误 (还有一 次输入机会)	4978	809 - 498	n/a	500.00	2,000	警告消息，返回基 本流步骤 4，输入 PIN
CW6.	场景 4 - PIN 有误 (不再有 输入机会)	4978	809 - 498	n/a	500.00	2,000	警告消息，卡予保 留，用例结束

测试方法选择的综合策略

- ▶ 选择等价类划分使无限测试变为有限测试；
- ▶ 在任何情况下都必须使用边界值分析方法；
- ▶ 用错误推测法再追加一些测试用例；
- ▶ 对照程序逻辑，检查已设计出的测试用例的逻辑覆盖程度。如果没有达到要求的覆盖标准，应当再补充足够的测试用例；
- ▶ 如果程序的功能说明中含有输入条件的组合情况，则一开始就可选用因果图法。

黑盒测试方法

- ▶ 等价类划分
- ▶ 边界值分析
- ▶ 因果图
- ▶ 场景法
- ▶ 错误推测法