目录

1.测试用例带给我们的便捷之处

2.测试用例标准

3.测试用例的基本要素

4.设计测试用例的方法

4.1.根据需求去设计测试用例

4.2.软件需求

5.具体的设计测试用例方法（黑盒测试设计测试用例的方法）

5.1等价类\*

5.2 边界值法\*

5.3 因果图法

5.3.1 恒等

5.3.2 与 ∧

5.3.3 或 ∨

5.3.4 非 ~

5.3.5 用因果图法设计测试用例的步骤

5.4 正交法

5.4.1 正交表的构成

5.4.2 正交表的性质

5.4.3 正交表设计测试用例的步骤

5.5 场景设计法\*

5.6 错误猜测法

1.测试用例带给我们的便捷之处

（1）使工作可重复，是自动化测试的基础

（2）评估需求的覆盖率

（3）用例的复用

（4）积累测试的方法思路以供后续借鉴

2.测试用例标准

（1）用例表述清楚、无二义性

（2）可操作性强

（3）用例的输入输出明确（一条用例只有一个预期结果）

（4）可维护性强

（5）对需求的覆盖率高

3.测试用例的基本要素

向测试系统发起的一组集合，测试平台，测试数据，测试步骤，预期结果等。

（测试方式，标题，重要性，优先级，功能模块等）

4.设计测试用例的方法

用户【需求】—产品经理【软件需求】—程序员【开发软件】—测试人员【测试功能】—【上线】

4.1.根据需求去设计测试用例

验证需求的正确性；

分析、细化需求，从需求中提炼功能模块，划分子功能，根据每一个子功能去设计测试用例；

即为以下两步：

（1）需求验证（合理性、正确性）

（2）进行测试用例的设计

举例：

用户需求：购买3000元以内的华为智能手机

测试点：

1.价格：3000元以内（小于等于）

2.手机类型：智能手机

3.品牌：华为

4.手机功能的测试（打电话、发短信、APP的安装使用、连网等）

4.2.软件需求

事件流

1.用户未收到激活邮件，可在登录界面录入电子邮件及密码后，再次发送激活邮件

事件测试

1.用户收到激活邮件 在登录界面输入电子邮件、密码，不会发送激活邮件，并且提示信息显示已发送

2.用户没有收到激活邮件 在登录界面输入电子邮件、密码， 会发送激活邮件

邮件测试: 内容、激活链接、排版、字体

2 每次发送的激活邮件，仅在发送邮件后24小时之内有效，超过24小时之后需要重新发送激活邮件

测试：

1.激活邮件发送，24小时之内（包括24小时）可以激活系统

2.激活邮件发送，24小时之后激活链接失效，无法激活系统

3.24小时之内激活系统，超过24小时之后再次点击激活链接，系统提示：该用户已激活系统

4.24小时之内激活系统，24小时之内再次点击激活链接，系统提示：该用户已激活系统

5.邮件内容（排版，日期，字体等）测试

5.具体的设计测试用例方法（黑盒测试设计测试用例的方法）

5.1等价类\*

思想：把输入（特殊情况下考虑输出）划分为若干个等价类，从每一个等价类当中选一个有代表性的测试用例进行测试，如果测试用例测试通过，那就称这个测试用例代表的等价类测试通过。

有效等价类：根据需求规格说明，有意义的输入数据构成的集合，称之为有效等价类

无效等价类：根据需求规格说明，不满足需求的集合

举例：

| 用户名 | 必填，录入用户名 | 6-15 | 字符类型A-Z，不区分大小写

根据长度划分：

有效等价类：6-15位

无效等价类：小于6位，大于15位

根据字符类型划分：

有效等价类：A-Z,a-z,大小写混合

无效等价类：汉字、特殊字符、标点符号、空格、A-Z或者a-z和其他字符混合

5.2 边界值法\*

针对输入输出的边界值进行测试的一种黑盒测试的方法。通常边界值分析法是作为对等价类划分法的补充。

举例：

输入框长度1-11：取边界值为1、11、12、0

5.3 因果图法

是一种简化版的逻辑图，能够直观的表明程序输入条件（原因）和输出动作（结果）之间的相互关系。是借助图形来设计测试用例的一种系统方法。

\*\*应用背景：\*\*被测试程序具有多种输入条件、程序的输出又依赖于输入条件的各种情况。

5.3.1 恒等

如果原因为真，那么结果必为真。

例如：动物园运来大熊猫，动物园一定有大熊猫。

5.3.2 与 ∧



只有两个原因都为真，那么结果为真。

例如：富豪，有车有房

（隐藏关系：有车无房，不是富豪；无车有房，不是富豪；无车无房，不是富豪；）—仅举例

5.3.3 或 ∨



两个原因中有一个为真时，结果就为真。

例如：富豪，有车有房

（隐藏关系：有车无房，是富豪；无车有房，是富豪；无车无房，不是富豪；）—仅举例

5.3.4 非 ~



只有原因为假，结果才为真。

例如：不起床，能捡到钱

5.3.5 用因果图法设计测试用例的步骤

（1）分析出所有的输入、输出

（2）找出输入输出之间的逻辑关系

（3）根据输入输出之间的关系画因果图

（4）根据因果图画判定表

（5）根据判定表设计测试用例

假设业务单据的处理规则为：“淘宝618活动，订单已提交，订单合计金额大于300或者有红包，则进优惠”

（1）分析输入输出：

输入：金额大于300，有红包，订单已提交

输出：有优惠，无优惠

（2）分析关系：

订单提交，金额大于300，有红包，优惠

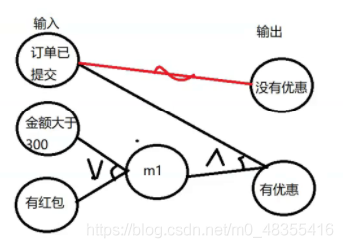
订单提交，金额大于300，无红包，优惠

订单提交，金额小于等于300，有红包，优惠

订单提交，金额小于等于300，无红包，无优惠

订单未提交，无关金额红包，无优惠

（3）画因果图



（4）判定表



（5）具体测试用例

订单已提交，金额大于300，有红包，有优惠

订单已提交，金额大于300，没有红包，有优惠

订单已提交，金额小于等于300，有红包，有优惠

订单已提交，金额小于等于300，没有红包，没有优惠

订单未提交，金额大于300，有红包，没有优惠

订单未提交，金额大于300，没有红包，没有优惠

订单未提交，金额小于等于300，有红包，没有优惠

订单未提交，金额小于等于300，没有红包，没有优惠

5.4 正交法

研究多因素多水平的一种设计方法，它是根据正交性，由试验因素的全部水平组合中挑选出部分有代表性的点进行试验，通过对这部分试验结果分析了解全面试验的情况，找出最优的水平组合。

因素：试验中想要考察的变量（输入的变量）

水平：试验范围内，因素被考察的值（变量的取值）

5.4.1 正交表的构成

行数：正交表中行的个数，即试验次数

因素数：正交表中列的个数

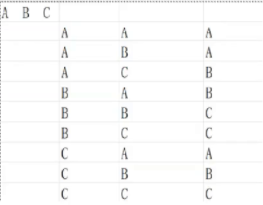
水平数：任何单个因素能够取得的值的最大个数。

行 = （水平数-1）\*因素数+1

5.4.2 正交表的性质

（1）每一列不同数据出现的次数一致

（2）任意两列不同的数据组合出现的次数一致



5.4.3 正交表设计测试用例的步骤

（1）确定所有因素（变量）

（2）确定每个因素有哪几个水平（变量的取值）

（3）确定因素数（正交表的列）

（4）根据正交表的性质，把变量的值映射到表中

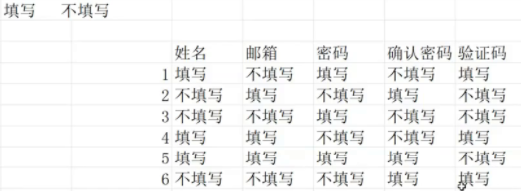
（5）写测试用例，正交表的每一行就是一个测试用例

（6）补充正交表中没有的但是你认为可疑且没有在表中出现的用例组合

练习：姓名、邮箱、密码、确认密码、验证码（只考虑每一项输入不输入两种情况）

（1）确定因素数：5；水平数：2；

（2）正交表的列=因素数=5；正交表的行=（2-1）\*5+1=6；

（3）

（4）写测试用例：

姓名填写，邮箱不填写，密码填写，确认密码不填写，验证码填写

姓名不填写，邮箱填写，密码不填写，确认密码填写，验证码不填写

姓名不填写，邮箱不填写，密码填写，确认密码不填写，验证码不填写

姓名填写，邮箱填写，密码不填写，确认密码不填写，验证码填写

姓名填写，邮箱不填写，密码填写，确认密码不填写，验证码填写

姓名不填写，邮箱不填写，密码不填写，确认密码填写，验证码不填写

正交表查询地址：http://support.sas.com/techsup/technote/ts723\_Designs.txt

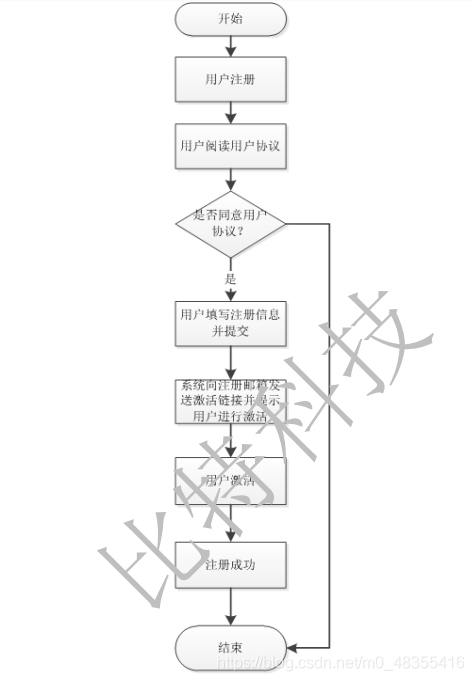
5.5 场景设计法\*

现在的软件几乎都是用事件触发来控制流程的，事件触发时的情景便形成了场景，而同一事件不同的触发顺序和处理结果就形成事件流。该方法可以比较生动地描绘出事件触发时的情景，有利于测试设计者设计测试用例，是测试用例更容易理解和执行。

典型的应用是是用业务流把各个孤立的功能点串起来，为测试人员建立整体业务感觉，从而避免陷入功能细节忽视业务流程要点的错误倾向。

案例：

以注册为例



5.6 错误猜测法

错误猜测法是经验丰富的测试人员喜欢使用的一种测试方法。

基于经验和直觉，找出程序中你认为可能出现的错误，有针对性地设计测试用例。经验可能来自于在对某项业务的测试较多，也可以来自于售后用户的反馈意见，或者从故障管理库中整理bug。梳理出产品以往哪些地方容易出现问题，问题越多的地方，潜在的bug也就越多。

以注册为例

1、校验中特殊字符空格的处理?

2、密码校验中的大小写？

3、姓名中的特殊字符？

4、密码发送是否明文

# 7.测试用例的粒度和评价

7.1 测试用例的粒度

粒度：指测试用例编写的详细程度

测试用例可以写得很简单，也可以写得很复杂。最简单的测试用例是测试的纲要，仅仅指出要测试的内容，如探索性测试中的测试设计，仅会指出需要测试产品的哪些要素、需要达到的质量目标、需要使用的测试方法等。而最复杂的测试用例就像飞机维修人员使用的工作指令卡一样，会指定输入的每项数据，期待的结果及检验的方法， 具体到界面元素的操作步骤，指定测试的方法和工具等。

(1)测试用例写得过于复杂或详细，会带来两个问题：一个是效率问题，另一个是维护成本问题。另外，测试用例设计得过于详细，留给测试执行人员的思考空间就比较少，容易限制测试人员的思维。

(2)测试用例写得过于简单，则可能失去了测试周例的意义。过于简单的测试用例设计其实并没有进行“设计”，只是把需要测试的功能模块记录下来而已，它的作用仅仅是在测试过程中作为一个简单的测试计划，提醒测试人员测试的主要功能包括哪些而已。测试用例的设计的本质应该是在设计的过程中理解需求，检验需求，并把对软件系统的测试方法的思路记录下来，以便指导将来的测试。

大多数测试团队编写的测试用例的粒度介于两者之间。而如何把握好粒度是测试用例设计的关键，也将影响测试用例设计的效率和效果。应该根据项目的实际情况、测试资源情况来决定设计出怎样粒度的测试用例。

> 主要考虑可以参考如下内容：

> 产品的质量要求

> 项目对用例的要求

> 测试时间和资源是否充分

7.2 测试用例的评价

测试用例设计出来了，如何提高测试用例设计的质量？

就像软件产品需要通过各种手段来保证质量一样，测试用例的质量保证也需要综合使用各种手段和方法。评审分为正式和非正式评审。

> 同行评审

> 用户检查

> 项目组评审

（1）测试用例的检查可以有多种方式 但是最敏捷的应当属临时的同行评审。同行评审，尤其是临时的同行评审，应该演变成类似结对编程一样的方式。从而体现敏捷的“个体和交互比过程和工具更有价值”，要强调测试用例设计者之间的思想碰撞，通过讨论、协作来完成测试用例的设计，原因很简单，测试用例的目的是尽可能全面地覆盖需求，而测试人员总会存在某方面的思维缺陷，一个人的思维总是存在局限性。因此需要一起设计测试用例。

（2）除了同行评审，还应该尽量引入用户参与到测试用例的设计中来，让用户参与评审，从而体现敏捷的“顾客的协作比合同谈判更有价值”这一原则。这里顾客的含义比较广泛，关键在于如何定义测试，如果测试是对产品的批判，则顾客应该指最终用户或顾客代表（在内部可以是市场人员或领域专家）；如果测试是被定义为对开发提供帮助和支持，那么顾客显然就是程序员了。

(3) 由测试负责人组织协调开展会议，用例编写人对用例进行讲解，参会人员有异议的当场提出。

案例：

某手机软件有用TF卡导出数据的功能，请写出测试此功能点的思路

