

CS10102302

软件黑盒测试

叶晨 教授级高工

计算机科学与技术系 电子与信息工程学院

同济大学

教学提纲

1

黑盒测试方法

- 等价类与边界值
- 因果图与决策表
- 场景法
- 正交试验法

黑盒测试方法

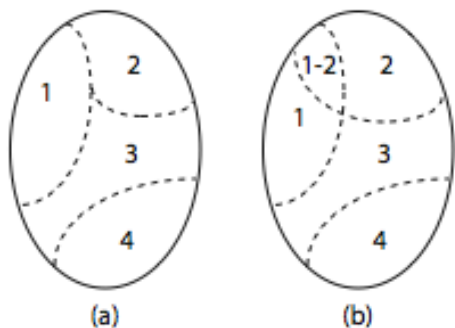
黑盒测试并不是白盒测试的替代品，而是用于辅助白盒测试发现其他类型错误。通常由独立测试人员根据用户需求文档来进行，但不一定要求用户参与。



等价类划分

等价类划分是将输入域划分成尽可能少的若干子域，在划分中要求每个子域两两互不相交，每个子域称为一个等价类。

等价类划分的原则是用同一等价类中的任意输入对软件进行测试，软件都输出相同的结果。因此，只需从每个等价类中选取一个输入作为测试用例，从而构成软件的完整测试用例集。

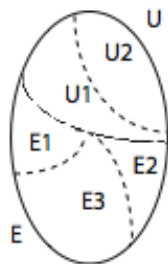


- (a) 图等价类之间互不相交，构成输入域的一个划分；
(b) 图不构成一个划分。
- 同一输入域的等价类划分可能不唯一。
- 对于相同的等价类划分，不同测试人员选取的测试用例集可能是不同的，测试用例集的故障检测效率取决于人员经验。

等价类类型

有效等价类是对规格说明有意义、合理的输入数据构成的集合。利用有效等价类，能够检验程序是否实现了规格说明中预先规定的功能和性能。

无效等价类是对规格说明无意义、不合理的输入数据构成的集合。利用无效等价类，可以发现程序异常处理的情况，检查被测对象的功能和性能的实现是否有不符合规格说明要求的地方。

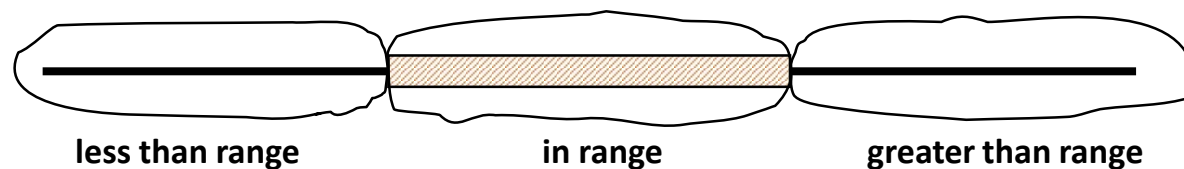


- E 表示所有正常和合法的输入
- U 表示所有异常和非法的输入

设计测试用例时要同时考虑这两种等价类，因为软件不仅要能接收合理的数据，也要能经受意外的考验。

变量的等价类

取值范围： 在输入条件规定了取值范围的情况下，可以确定一个有效等价类和两个无效等价类。



举例： 程序的输入参数 x 是小于100大于10的整数。

1个有效等价类： $10 < x < 100$

2个无效等价类： $x \leq 10$ 和 $x \geq 100$

变量的等价类

字符串：在规定了输入数据必须遵守的规则情况下，可确定一个有效等价类（符合规则）和若干个无效等价类（从不同角度违反规则）。

举例：姓名是长度不超过20的非空字符串，且只由字母组成，数字和其他字符都是非法的。

1个有效等价类：满足了上述所有条件的字符串

3个无效等价类：

- 空字符串
- 长度超过20的字符串
- 包含了数字或其它字符的字符串

变量的等价类

枚举：若规定输入数据是一组值（假定N个），并且程序要对每一个输入值分别处理，可确定N个有效等价类和一个无效等价类。

举例：某程序根据不同的学历分别计算岗位工资，其中学历可以是专科、本科、硕士、博士等四种类型。

4个有效等价类：专科、本科、硕士、博士

1个无效等价类：其他学历

如果该程序将专科、本科、硕士、博士按一种方式计算岗位工资，这时应该如何划分等价类？

变量的等价类

数组：数组是一组具有相同类型的元素的集合，数组长度及其类型都可以作为等价类划分的依据。

举例：假设某程序的输入是一个整数数组 `int oper[3]`

1个有效等价类：所有合法值的数组，如 `{-10,20}`

2个无效等价类：

- 空数组
- 所有大于期望长度的数组，如 `{-9,0,12,15}`

如果对数组元素有其他附加约束，例如数组`oper`元素的取值范围是`[-3,3]`，则需要增加相应的等价类。

变量的等价类

复合数据类型：复合数据类型是包含两个或两个以上相互独立的属性的输入数据，在进行等价类划分时需要考虑输入数据的每个属性的合法和非法取值。

举例：

```
struct student {  
    string name;  
    string course[100];  
    int grade[100];  
}
```

对复合数据类型中的每个元素进行等价类划分，再将这些等价类进行组合，最终形成对软件整个输入域的划分。

等价类组合

测试用例生成：测试对象通常有多个输入参数，如何对这些参数等价类进行组合测试，来保证等价类的覆盖率，是测试用例设计首先需要考虑的问题。

所有有效等价类的代表值都集成到测试用例中，即覆盖有效等价类的所有组合。任何一个组合都将设计成一个有效的测试用例，也称**正面测试用例**。

无效等价类的代表值只能和其他有效等价类的代表值（随意）进行组合。因此，每个无效等价类将产生一个额外的无效测试用例，也称**负面测试用例**。

**等价类的组合将产生数以百计的测试用例，
如何有效地减少测试用例的数目？**

选取等价类的测试用例

- 由所有代表值组合而成的测试用例按使用频率（典型的使用特征）进行排序，并按照这个序列设置优先级，这样就能仅对相关的测试用例（典型的组合）进行测试。
- 优先考虑包含边界值或者边界值组合的测试用例。
- 将一个等价类的每个代表值和其他等价类的每个代表值进行组合来设计测试用例（即双向组合代替完全组合）。
- 保证满足最小原则：一个等价类的每个代表值至少在一个测试用例中出现。
- 无效等价类的代表值不与其他无效等价类代表值进行组合。

选取等价类的测试用例

考虑是否对无效数据进行测试，可以将等价类测试分为：

- **标准等价类测试**：不考虑无效数据值，测试用例使用每个有效等价类中的一个值。
- **健壮等价类测试**：主要是考虑无效等价类，即对有效输入，测试用例从每个有效等价类中取一个值；对无效输入，一个测试用例有一个无效值，其他值均取有效值。

说明：在有些情况下，一些输入数据无法输入到被测软件中。例如软件的 GUI 过滤掉一些无效输入，或者对强类型的语言没有必要考虑无效的输入，这时只需要进行标准等价类测试即可。

举例：判断三角形类型

输入三个整数 a 、 b 、 c ，分别作为三角形的三条边，现通过一个程序判断这三条边构成的三角形类型，包括等边三角形、等腰三角形、一般三角形（特殊的还有直角三角形）以及构不成三角形。

现在要求输入的三个整数 a 、 b 、 c 必须满足以下条件：

条件1: $1 \leq a \leq 100$

条件4: $a < b + c$

条件2: $1 \leq b \leq 100$

条件5: $b < a + c$

条件3: $1 \leq c \leq 100$

条件6: $c < a + b$

请使用等价类划分方法，设计该程序的测试用例。

举例：判断三角形类型

等价类划分：

①按输入取值划分

$\{0, >0, <0\}$ 或 $\{0, 1, >1, <0\}$

②按输入的几何特性划分

$\{\text{等腰且非等边三角形, 等边三角形, 不等边三角形, 非三角形}\}$

分析：多数情况下，等价类是根据输入域划分的，但并非不能从输出域反过来定义等价类，事实上这对于三角形问题是最简单的划分方法。

$R1 = \{ \langle a, b, c \rangle : \text{边 } a、b、c \text{ 组成等边三角形} \}$

$R2 = \{ \langle a, b, c \rangle : \text{边 } a、b、c \text{ 组成等腰三角形} \}$

$R3 = \{ \langle a, b, c \rangle : \text{边 } a、b、c \text{ 组成一般三角形} \}$

$R4 = \{ \langle a, b, c \rangle : \text{边 } a、b、c \text{ 不能组成三角形} \}$

举例：判断三角形类型

标准等价类测试用例

序号	测试用例描述	输入参数			期望输出
		a	b	c	
1	$a > 0, b > 0, c > 0$ $a + b > c, b + c > a, a + c > b$ $a = b = c$	10	10	10	等边三角形
2	$a > 0, b > 0, c > 0$ $a + b > c, b + c > a, a + c > b$ $a = b \neq c$ 或 $b = c \neq a$ 或 $a = c \neq b$	10	10	5	等腰三角形
3	$a > 0, b > 0, c > 0$ $a + b > c, b + c > a, a + c > b$ $a \neq b \neq c$	3	4	5	一般三角形
4	$a > 0, b > 0, c > 0$ $a + b \leq c$ 或 $b + c \leq a$ 或 $a + c \leq b$	4	1	2	非三角形

举例：判断三角形类型

健壮等价类测试用例

序号	测试用例描述	输入参数			期望输出
		a	b	c	
1-4	有效等价类同前面（略）				
5	$a < 0, b > 0, c > 0$	-1	5	5	a 值越界
6	$a > 0, b < 0, c > 0$	5	-1	5	b 值越界
7	$a > 0, b > 0, c < 0$	5	5	-1	c 值越界
8	$a > 100, b > 0, c > 0$	101	5	5	a 值越界
9	$a > 0, b > 100, c > 0$	5	101	5	b 值越界
10	$a > 0, b > 0, c > 100$	5	5	101	c 值越界

边界值分析

边界值分析是对输入或输出的边界值进行测试的一种方法，它通常作为等价类划分法的补充，这种情况下的测试用例来自等价类的边界。

- 首先确定边界情况。通常输入或输出等价类的边界就是应该着重测试的边界情况。
- 选取正好等于、刚刚大于或刚刚小于边界的值作为测试数据，而不是选取等价类中的典型值或任意值。

实践表明：大多数故障往往发生在输入定义域或输出值域的边界上，而不是在其内部。因此，针对各种边界情况设计测试用例，通常会取得很好的测试效果。

边界值分析

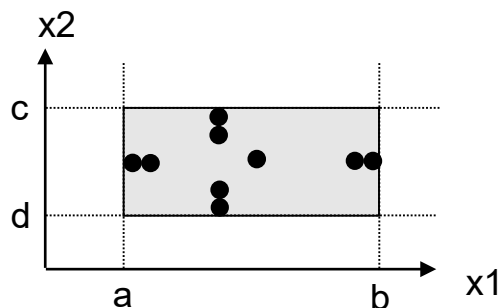
输入项	边界值	测试用例的设计思路
字符	起始 - 1个字符 结束 + 1个字符	假设一个文本输入区域允许输入1到255个字符，输入1个和255个字符作为有效等价类；输入0个和256个字符作为无效等价类，这几个数值都属于边界条件值。
数值	最小值 - 1 最大值 + 1	假设某软件要求输入5位十进制整数值，可以使用10000作为最小值、99999作为最大值；然后使用刚好小于5位和大于5位的数值作为边界条件。
空间	小于空余空间一点 大于满空间一点	例如在用U盘存储数据时，使用比剩余磁盘空间大一点（几 KB）的文件作为边界条件。

边界值分析

基本思想：故障往往出现在程序输入变量的边界值附近

- 边界值分析法是基于可靠性理论中称为“单故障”的假设，即有两个或两个以上故障同时出现而导致失效的情况很少。
- 对程序中的每个变量重复：每次保留一个变量，让其余的变量取正常值，被保留的变量依次取 min、min+、nom、max- 和 max。

举例：程序F有输入变量 x_1 ($a \leq x_1 \leq b$) 和 x_2 ($c \leq x_2 \leq d$)



$\langle x1_{nom}, x2_{min} \rangle$	$\langle x1_{nom}, x2_{min+} \rangle$	$\langle x1_{nom}, x2_{nom} \rangle$
$\langle x1_{nom}, x2_{max} \rangle$	$\langle x1_{nom}, x2_{max-} \rangle$	$\langle x1_{min}, x2_{nom} \rangle$
$\langle x1_{min+}, x2_{nom} \rangle$	$\langle x1_{max}, x2_{nom} \rangle$	$\langle x1_{max-}, x2_{nom} \rangle$

举例：判断三角形类型

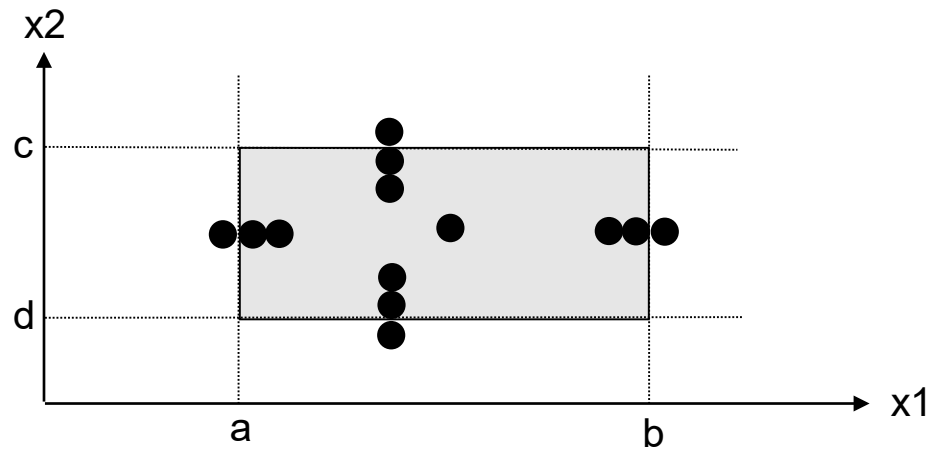
序号	测试用例描述	输入参数			期望输出
		a	b	c	
1	$a > 0, b > 0, c > 0$ $a + b > c, b + c > a, a + c > b$ $a = b = c$	60	60	60	等边三角形
2	$a > 0, b > 0, c > 0$ $a + b > c, b + c > a, a + c > b$ $a = b \neq c$ 或 $b = c \neq a$ 或 $a = c \neq b$	60	60	1	等腰三角形
3		60	60	2	
4		50	50	99	
5		60	1	60	
6		60	2	60	

举例：判断三角形类型

序号	测试用例描述	输入参数			期望输出
		a	b	c	
7	$a > 0, b > 0, c > 0$ $a + b > c, b + c > a, a + c > b$ $a = b \neq c$ 或 $b = c \neq a$ 或 $a = c \neq b$	50	99	50	等腰三角形
8		1	60	60	
9		2	60	60	
10		99	50	50	
11	$a > 0, b > 0, c > 0$ $a + b \leq c$ 或 $b + c \leq a$ 或 $a + c \leq b$	50	50	100	非三角形
12		50	100	50	
13		100	50	50	

健壮性测试

健壮性测试是作为边界值分析的一个简单的扩充，它除了对变量的5个边界值分析取值外，还需要增加一个略大于最大值（max+）以及略小于最小值（min-）的取值，检查超过极限值时系统的情况。



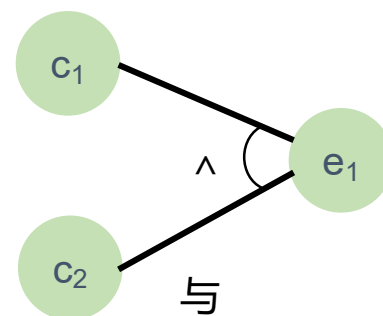
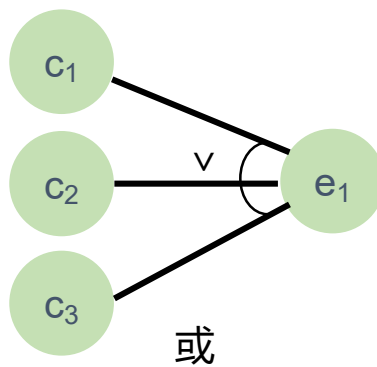
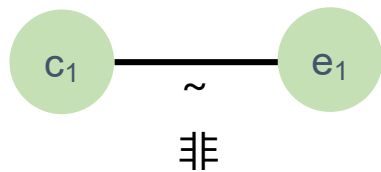
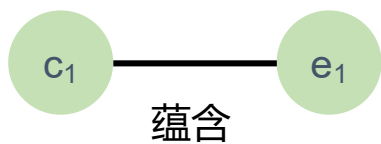
因果图

因果图是一种利用图解法分析输入的各种组合情况，从而设计测试用例的方法，适合于检查程序输入条件的各种组合情况。

- 等价类划分和边界值分析方法都是着重考虑输入条件，但没有考虑输入条件的各种组合、输入条件之间的相互制约关系。这样虽然各种输入条件可能出错的情况已经测试到了，但多个输入条件组合起来可能出错的情况却被忽视了。
- 如果在测试时必须考虑输入条件的各种组合，则可能的组合数目将是天文数字，因此必须考虑采用一种适合于描述多种条件的组合、相应产生多个动作的形式来进行测试用例的设计，这就需要利用因果图。

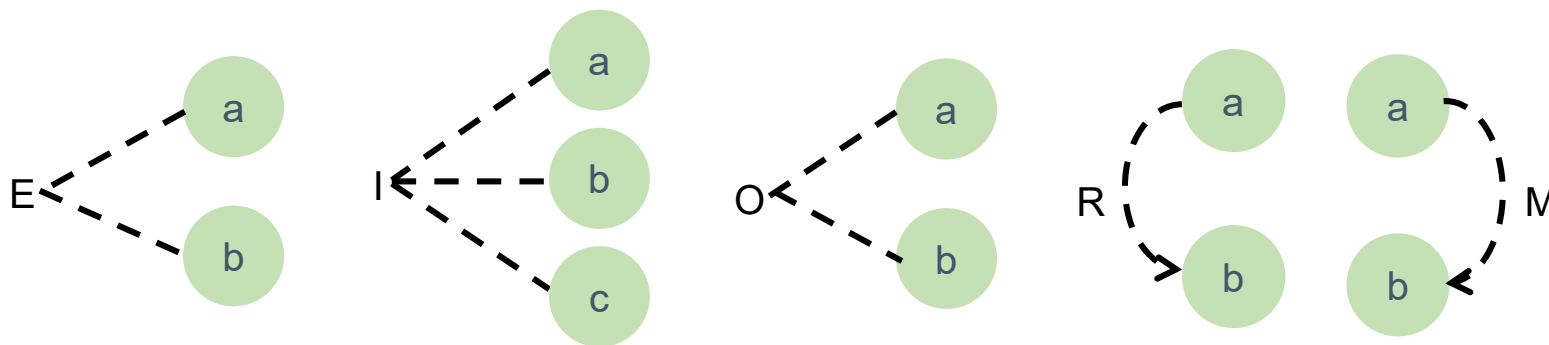
因果图

基本符号：用于表示四种因果关系



因果图

约束符号：用于表示约束关系



E约束（异）：a 和b中最多有一个可能为1，即a和b不能同时为1。

I 约束（或）：a、b、c中至少有一个必须为1，即a、b、c不能同时为0。

O约束（唯一）：a和b必须有一个且仅有一个为1。

R约束（要求）：a是1时，b必须是1，即a为1时，b不能为0。

M约束（强制）：若结果a为1，则结果b强制为0。

因果图

采用因果图法设计测试用例：

- ①根据程序规格说明书描述，分析并确定原因（输入条件）和结果（输出结果或程序状态的改变），画出因果图；
- ②将得到的因果图转换为决策表；
- ③为决策表中每一列所表示的情况设计一个测试用例。

因果图方法的优点：

- 考虑输入情况的各种组合以及各个输入情况之间的相互制约关系；
- 可以帮助测试人员按照一定的步骤，高效率地开发测试用例；
- 因果图法是将自然语言规格说明转化成形式语言规格说明的一种严格的方法，可以指出规格说明中存在的完整性二义性。

因果图

举例：某程序输入的第一个字符必须是 # 或 *，第二个字符必须是一个数字，此情况下进行文件的修改；如果第一个字符不是 # 或 *，则给出信息N，如果第二个字符不是数字，则给出信息M。



因果图

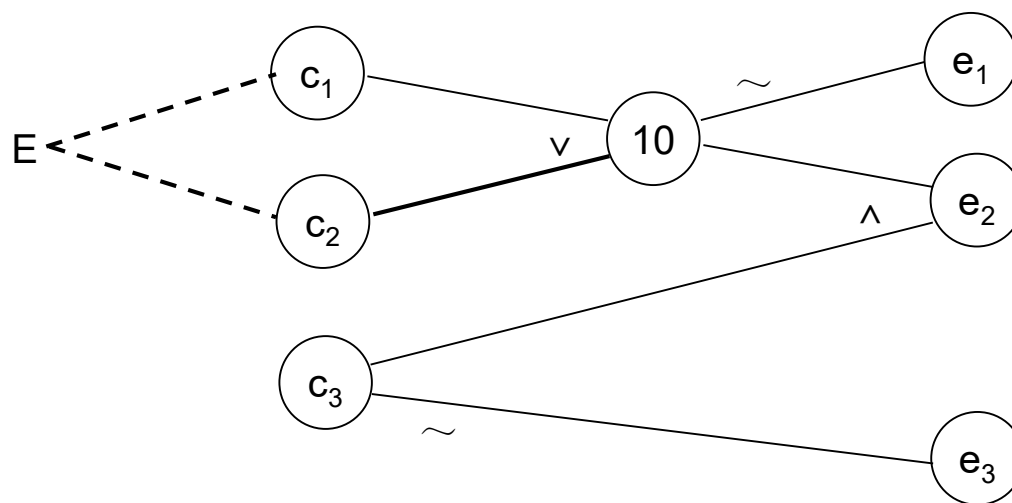
举例：某程序输入的第一个字符必须是 # 或 *，第二个字符必须是一个数字，此情况下进行文件的修改；如果第一个字符不是 # 或 *，则给出信息N，如果第二个字符不是数字，则给出信息M。

第一步：分析程序规格说明中的原因和结果

原因	结果
c1：第一个字符是#	e1：给出信息N
c2：第一个字符是*	e2：修改文件
c3：第二个字符是一个数字	e3：给出信息M

因果图

第二步：画出因果图



其中：编号为10的中间结点是导出结果的进一步原因。

因果图

第三步：将因果图转换成如下所示的决策表

		1	2	3	4	5	6	7	8
条件	c1	1	1	1	1	0	0	0	0
	c2	1	1	0	0	1	1	0	0
	c3	1	0	1	0	1	0	1	0
	10			1	1	1	1	0	0
动作	e1							√	√
	e2			√		√			
	e3				√		√		√
	不可能	√	√						
测试用例				#3	#A	*6	*B	A1	GT

因果图

第四步：根据决策表中的每一列设计测试用例

测试用例编号	输入数据	预期输出
1	#3	修改文件
2	#A	给出信息M
3	*6	修改文件
4	*B	给出信息M
5	A1	给出信息N
6	GT	给出信息N和信息M

决策表

适用情况： 在一些数据处理问题当中，某些操作的实施依赖于多个逻辑条件的组合，即针对不同逻辑条件的组合值，分别执行不同的操作。

条件桩	条件项
动作桩	动作项

规则

条件桩：列出问题的所有条件

条件项：针对条件桩给出的条件列出所有可能的取值

动作桩：列出问题规定的可能采取的操作

动作项：指出在条件项的各组取值情况下应采取的动作

将任何一个条件组合的特定取值及相应要执行的动作称为一条规则。
在决策表中贯穿条件项和动作项的一列就是一条规则。

决策表

测试用例设计：

- 确定条件桩和动作桩；
- 确定规则的个数。假如有n个条件，每个条件有两个取值（0，1），故有 2^n 种规则；
- 填入条件项和动作项，得到初始的决策表；
- 简化合并相似的规则，得到优化的决策表；
- 每列规则设计一个测试用例。

规则		规则 1	规则 2	规则 p
条件桩	条件项 1				
	条件项 2				
				
	条件项 m				
动作桩	动作项 1				
	动作项 2				
				
	动作项 n				

举例：判断三角形类型

输入三个整数 a 、 b 、 c ，分别作为三角形的三条边，现通过一个程序判断这三条边构成的三角形类型，包括等边三角形、等腰三角形、一般三角形（特殊的还有直角三角形）以及构不成三角形。

现在要求输入的三个整数 a 、 b 、 c 必须满足以下条件：

条件1： $1 \leq a \leq 100$

条件4： $a < b + c$

条件2： $1 \leq b \leq 100$

条件5： $b < a + c$

条件3： $1 \leq c \leq 100$

条件6： $c < a + b$

请使用决策表方法，设计该程序的测试用例。

举例：判断三角形类型

规则		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
条件桩	C1: a、b、c构成三角形？																
	C2: a=b?																
	C3: a=c?																
	C4: b=c?																
动作桩	A1: 非三角形																
	A2: 不规则三角形																
	A3: 等腰三角形																
	A4: 等边三角形																
	A5: 不符合逻辑																

举例：判断三角形类型

规则		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
条件桩	C1: a、b、c构成三角形?	Y	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	N	N	N	N
	C2: a=b?	Y	Y	Y	Y	N	N	N	N	Y	Y	Y	Y	N	N	N	N
	C3: a=c?	Y	Y	N	N	Y	Y	N	N	Y	Y	N	N	Y	Y	N	N
	C4: b=c?	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N
动作桩	A1: 非三角形									Y	Y	Y	Y	Y	Y	Y	Y
	A2: 不规则三角形								Y								
	A3: 等腰三角形				Y		Y	Y									
	A4: 等边三角形	Y															
	A5: 不符合逻辑		Y	Y		Y											

举例：判断三角形类型

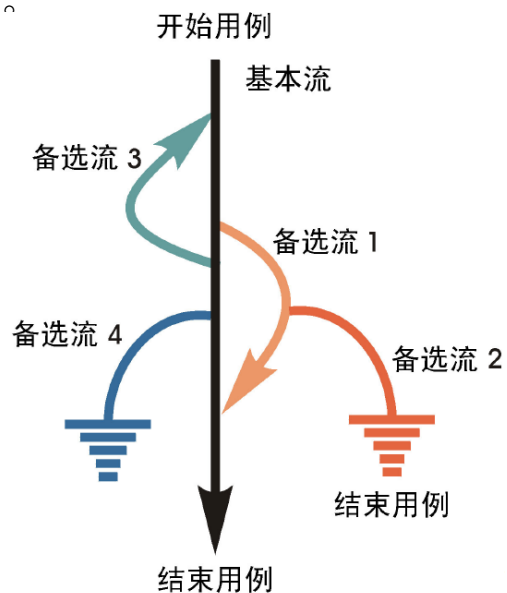
规则		1	2	3	4	5	6
条件桩	C1: a、b、c构成三角形?	Y	Y	Y	Y	Y	N
	C2: a=b?	Y	Y	N	N	N	-
	C3: a=c?	Y	N	Y	N	N	-
	C4: b=c?	Y	N	N	Y	N	-
动作桩	A1: 非三角形						Y
	A2: 不规则三角形					Y	
	A3: 等腰三角形		Y	Y	Y		
	A4: 等边三角形	Y					

举例：判断三角形类型

序号	测试用例描述	期望输出
1	C1、C2、C3、C4 全部满足	等边三角形
2	C1和C2 满足, C3和C4 不满足	等腰三角形
3	C1和C3 满足, C2和C4 不满足	等腰三角形
4	C1和C4 满足, C2和C3 不满足	等腰三角形
5	C1 满足, C2、C3和C4 不满足	不规则三角形
6	C1 不满足	非三角形

场景法

场景法通过运用场景来对系统的功能点或业务流程的描述，从而提高测试效果的一种方法。



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

举例：ATM取款

用例描述	
基本流	成功提款
备选2	ATM内没有现金
备选3	ATM内现金不足
备选4	PIN有误（还有输入机会）
备选5	PIN有误（没有输入机会）
备选6	账户不存在 / 账户类型有误
备选7	账户余额不足

举例：ATM取款

场景设计

场景1：成功提款	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（没有输入机会）	基本流、备选流5
场景6：账户不存在 / 账户类型有误	基本流、备选流6
场景7：账户余额不足	基本流、备选流7

举例：ATM取款

测试用例设计

序号	场景	PIN	账号	取款金额	账面金额	ATM现金	预期结果
1	场景1：成功提款	V	V	V	V	V	成功提款
2	场景2：ATM里没有现金	V	V	V	V	X	提款选项不可用，用例结束
3	场景3：ATM里现金不足	V	V	V	V	X	警告信息，返回基本流相应步骤，重新输入金额
4	场景4：PIN有误 (还有不止一次机会)	X	V	n/a	V	V	警告信息，返回基本流相应步骤，重新输入PIN
5	场景4：PIN有误 (还有一次机会)	X	V	n/a	V	V	警告信息，返回基本流相应步骤，重新输入PIN
6	场景5：PIN有误 (不再有输入机会)	X	V	n/a	V	V	警告信息，卡被没收，用例结束
						

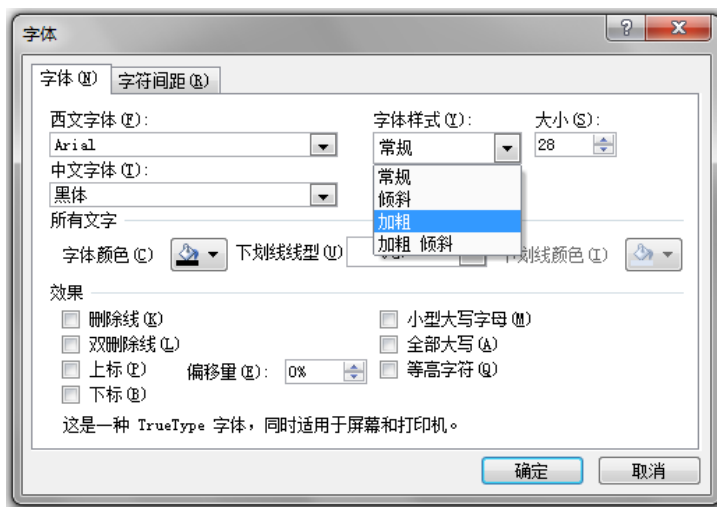
测试用例设计的挑战

测试用例设计的挑战



输入参数组合

等价类划分和边界值分析等方法没有考虑被测程序由于不同输入变量之间的组合引起故障的可能性，而且输入变量的组合规模非常大。



- ① 西文字体
- ② 中文字体
- ③ 字体样式
- ④ 大小
- ⑤ 字体颜色
- ⑥ 效果
- ⑦

平台参数组合

大多数软件都会设计成在多种环境下工作，多种平台参数的组合造就了环境的多样化。在每种环境中，各参数各自对应一个特定的值，这些特定值的集合称为测试配置。

举例：某网站需要支持

- 浏览器： IE6, IE7, IE8, IE9, Firefox14, Opera11, Safari5, Chrome18
- 插件： RealPlayer, MediaPlayer, None
- 客户端操作系统： Win2000, WinXP, Win2003, Win7, Win8, MacOS
- Web服务器软件： IIS, Apache, Weblogic
- 服务器端操作系统： WinNT, Win2003, Linux

这种测试配置的数量极其庞大，彻底的测试 $8 \times 3 \times 6 \times 3 \times 3 = 1296$ 是不可能的。

组合测试的策略

组合太多：测试输入、平台、路径等所有组合的规模太大

测试人员可以采取的常用策略：

- 尝试测试所有输入的组合，延期项目，可能是失去产品市场
- 选择一些容易设计和执行的测试用例，难以发现质量问题
- 罗列所有的组合，再随机选择其中的子集进行测试
- 采取合适的测试技术，选择能发现大部分缺陷的子集进行测试

建议：选用**组合设计技术**，从参数组合的完全集中选择一个较小的子集，其目的是发现因参数组合而引起的故障。

组合测试设计技术



- 模型是由一组参数及其对应的值组成，输入空间或环境建模并不互斥，可根据被测程序要求同时对二者或其中之一建模。
- 组合设计由一个 $N \times k$ 的矩阵表示成一个组合对象，其中 N 行中的每一行对应至少一次测试执行， k 列中的每一列对应一个参数。
- 每个参数组合可以生成一个或多个测试用例，每个测试用例由输入变量的值及其预期的输出组成。

常用的正交矩阵

标准正交矩阵：所有因素的取值集合相同

$$L_4(2^3)$$

$$L_8(2^7)$$

$$L_9(3^4)$$

$$L_{16}(4^5)$$

混合正交矩阵：每个因素的取值集合不相同

$$L_8(4 \times 2^4)$$

$$L_{12}(3 \times 2^3)$$

$$L_{12}(3 \times 2^4)$$

$$L_{16}(2^6 \times 4^3)$$

<http://www.research.att.com/~njas/oadir>

常用的正交矩阵

$L_4(2^3)$

	1	2	3
1	1	1	1
2	1	2	2
3	2	1	2
4	2	2	1

$L_8(2^7)$

	1	2	3	4	5	6	7
1	1	1	1	1	1	1	1
2	1	1	1	2	2	2	2
3	1	2	2	1	1	2	2
4	1	2	2	2	2	1	1
5	2	1	2	1	2	1	2
6	2	1	2	2	1	2	1
7	2	2	1	1	2	2	1
8	2	2	1	2	1	1	2

常用的正交矩阵

$L_9(3^4)$

	1	2	3	4
1	1	1	1	1
2	1	2	2	2
3	1	3	3	3
4	2	1	2	3
5	2	2	3	1
6	2	3	1	2
7	3	1	3	2
8	3	2	1	3
9	3	3	2	1

常用的正交矩阵

$L_{16}(4^5)$

	1	2	3	4	5
1	1	1	1	1	1
2	1	2	2	2	2
3	1	3	3	3	3
4	1	4	4	4	4
5	2	1	2	3	4
6	2	2	1	4	3
7	2	3	4	1	2
8	2	4	3	2	1
9	3	1	3	4	2
10	3	2	4	3	1
11	3	3	1	2	4
12	3	4	2	1	3
13	4	1	4	2	3
14	4	2	3	1	4
15	4	3	2	4	1
16	4	4	1	3	2

常用的正交矩阵

$L_8(4 \times 2^4)$

	1	2	3	4	5
1	1	1	1	1	1
2	1	2	2	2	2
3	2	1	1	2	2
4	2	2	2	1	1
5	3	1	2	1	2
6	3	2	1	2	1
7	4	1	2	2	1
8	4	2	1	1	2

常用的正交矩阵

$L_{12}(3 \times 2^3)$

	1	2	3	4
1	1	1	1	1
2	1	1	2	1
3	1	2	1	2
4	1	2	2	2
5	2	1	1	2
6	2	1	2	2
7	2	2	1	1
8	2	2	2	1
9	3	1	1	2
10	3	1	2	1
11	3	2	1	1
12	3	2	2	2

常用的正交矩阵

$L_{12}(3 \times 2^4)$

	1	2	3	4	5
1	1	1	1	1	1
2	1	1	1	2	2
3	1	2	2	1	2
4	1	2	2	2	1
5	2	1	2	1	1
6	2	1	2	2	2
7	2	2	1	2	2
8	2	2	1	2	2
9	3	1	2	1	2
10	3	1	1	2	1
11	3	2	1	1	2
12	3	2	2	2	1

常用的正交矩阵

$L_{16}(2^6 \times 4^3)$

	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	2	2	1	2	1	2	1	3	3
3	1	2	2	2	2	1	3	1	3
4	2	1	2	1	2	2	3	3	1
5	1	1	2	2	2	2	1	4	4
6	2	2	2	1	2	1	1	2	2
7	1	2	1	1	1	2	3	4	2
8	2	1	1	2	1	1	3	2	4
9	2	2	1	1	2	2	4	1	4
10	1	1	1	2	2	1	4	3	2
11	2	1	2	2	1	2	2	1	2
12	1	2	2	1	1	1	2	3	4
13	2	2	2	2	1	1	4	4	1
14	1	1	2	1	1	2	4	2	3
15	2	1	1	1	2	1	2	4	3
16	1	2	1	2	2	2	2	2	1

基于正交矩阵的测试用例设计

应用正交矩阵设计测试用例的步骤:

- ① 识别测试对象的参数（即因素），确定每个参数的取值个数（即水平）
- ② 选择一个合适的正交矩阵
- ③ 将参数的取值映射到正交矩阵中
- ④ 把每一行的各因素的水平组合构造为一个测试用例
- ⑤ 加上一些你认为可疑但没有在矩阵中出现的组合

举例：Powerpoint打印测试

Powerpoint 打印功能的需求如下：

打印范围：	全部、当前幻灯片、给定范围
打印内容：	幻灯片、讲义、备注页、大纲视图
打印颜色/灰度：	彩色、灰度、黑白
打印效果：	幻灯片加框和幻灯片不加框

说明：

- 使用完全组合方法将产生 $3 \times 4 \times 3 \times 2 = 72$ 种情况，显然这种测试方法将会导致很大的测试工作量。
- 使用正交实验法可以大大降低测试用例数目，并保持较好的覆盖率。

举例：Powerpoint打印测试

步骤1：识别测试对象的参数，确定每个参数的取值个数

水平 / 因素	A 打印范围	B 打印内容	C 打印颜色/灰度	D 打印效果
1	全部 (A1)	幻灯片 (B1)	颜色 (C1)	幻灯片加框 (D1)
2	当前幻灯片 (A2)	讲义 (B2)	灰度 (C2)	幻灯片不加框 (D2)
3	给定范围 (A3)	备注页 (B3)	黑白 (C3)	
4		大纲视图 (B4)		

步骤2：选择一个合适的正交矩阵

- 表中的因素数 = 4，表中至少有4个因素的水平数 ≥ 2
- 行数取最少的一个，最少行数为 $2 \times (3-1) + (4-1) + (2-1) + 1 = 9$
- 最后选中正交表公式： $L_{16}(4^5)$

举例：Powerpoint打印测试

$L_{16}(4^5)$

	1	2	3	4	5
1	1	1	1	1	1
2	1	2	2	2	2
3	1	3	3	3	3
4	1	4	4	4	4
5	2	1	2	3	4
6	2	2	1	4	3
7	2	3	4	1	2
8	2	4	3	2	1
9	3	1	3	4	2
10	3	2	4	3	1
11	3	3	1	2	4
12	3	4	2	1	3
13	4	1	4	2	3
14	4	2	3	1	4
15	4	3	2	4	1
16	4	4	1	3	2

举例：Powerpoint打印测试

步骤3：将参数的取值映射到正交矩阵中

	1	2	3	4	5
1	A1	B1	C1	D1	1
2	A1	B2	C2	D2	2
3	A1	B3	C3	3	3
4	A1	B4	4	4	4
5	A2	B1	C2	3	4
6	A2	B2	C1	4	3
7	A2	B3	4	D1	2
8	A2	B4	C3	D2	1
9	A3	B1	C3	4	2
10	A3	B2	4	3	1
11	A3	B3	C1	D2	4
12	A3	B4	C2	D1	3
13	4	B1	4	D2	2
14	4	B2	C3	D1	4
15	4	B3	C2	4	1
16	4	B4	C1	3	2

举例：Powerpoint打印测试

	1	2	3	4	
1	A1	B1	C1	D1	
2	A1	B2	C2	D2	
3	A1	B3	C3	D1	
4	A1	B4	C1	D2	
5	A2	B1	C2	D1	
6	A2	B2	C1	D2	
7	A2	B3	C2	D1	
8	A2	B4	C3	D2	
9	A3	B1	C3	D2	
10	A3	B2	C3	D1	
11	A3	B3	C1	D2	
12	A3	B4	C2	D1	

去掉

去掉

举例：Powerpoint打印测试

步骤4：设计测试用例

用例编号	PPT-ST-FUNCTION-PRINT-001
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 全部的幻灯片，有颜色，加框
重要级别	高
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“全部”；3. 打印内容选择“幻灯片”；4. 颜色/灰度选择“颜色”；5. 选中“幻灯片加框”复选框；6. 点击“确定”。
预期输出	打印出全部幻灯片，有颜色且已加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-002
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 全部的幻灯片为讲义，灰度，不加框
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“全部”；3. 打印内容选择“讲义”；4. 颜色/灰度选择“灰度”；5. 点击“确定”。
预期输出	打印出全部幻灯片，灰度且不加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-003
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 全部的备注页，黑白，加框
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“全部”；3. 打印内容选择“备注页”；4. 颜色/灰度选择“黑白”；5. 选中“幻灯片加框”复选框；6. 点击“确定”。
预期输出	打印出全部备注页，黑白且已加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-004
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 全部的大纲视图，有颜色
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“全部”；3. 打印内容选择“大纲视图”；4. 颜色/灰度选择“颜色”；5. 点击“确定”。
预期输出	打印出全部大纲视图，有颜色且不加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-005
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 当前幻灯片，灰度，加框
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“当前幻灯片”；3. 打印内容选择“幻灯片”；4. 颜色/灰度选择“灰度”；5. 选中“幻灯片加框”复选框；6. 点击“确定”。
预期输出	打印出当前幻灯片，灰度且已加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-006
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 当前幻灯片为讲义，有颜色
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“当前幻灯片”；3. 打印内容选择“讲义”；4. 颜色/灰度选择“颜色”；5. 点击“确定”。
预期输出	打印出当前幻灯片为讲义，有颜色且不加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-007
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 当前幻灯片的备注页，灰度，加框
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“当前幻灯片”；3. 打印内容选择“备注页”；4. 颜色/灰度选择“灰度”；5. 选中“幻灯片加框”复选框；6. 点击“确定”。
预期输出	打印出当前幻灯片的备注页，灰度且加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-008
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 当前幻灯片的大纲视图，黑白
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“当前幻灯片”；3. 打印内容选择“大纲视图”；4. 颜色/灰度选择“黑白”；5. 点击“确定”。
预期输出	打印出当前幻灯片的大纲视图，黑白且不加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-009
测试项目	测试 Powerpoint 打印功能
测试标题	打印 Powerpoint 文件 A 给定范围的幻灯片，黑白，不加框
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“幻灯片”；3. 打印内容选择“幻灯片”；4. 颜色/灰度选择“黑白”；5. 点击“确定”。
预期输出	打印出给定范围的幻灯片，黑白且不加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-010
测试项目	测试 Powerpoint 打印功能
测试标题	打印Powerpoint文件A给定范围的幻灯片为讲义，黑白，加框
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“幻灯片”；3. 打印内容选择“讲义”；4. 颜色/灰度选择“黑白”；5. 选中“幻灯片加框”复选框；6. 点击“确定”。
预期输出	打印出给定范围的幻灯片为讲义，黑白且加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-011
测试项目	测试 Powerpoint 打印功能
测试标题	打印Powerpoint文件A给定范围幻灯片为备注页，有颜色
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“幻灯片”；3. 打印内容选择“备注页”；4. 颜色/灰度选择“颜色”；5. 点击“确定”。
预期输出	打印出给定范围幻灯片的备注页，有颜色且不加框。

举例：Powerpoint打印测试

用例编号	PPT-ST-FUNCTION-PRINT-012
测试项目	测试 Powerpoint 打印功能
测试标题	打印Powerpoint文件A给定范围幻灯片的大纲视图，灰度，加框
重要级别	中
预置条件	Powerpoint 文件 A 已被打开，电脑主机已连接有效打印机
输入	文件 A：D:\test.ppt
操作步骤	<ol style="list-style-type: none">1. 打开打印界面；2. 打印范围选择“幻灯片”；3. 打印内容选择“大纲视图”；4. 颜色/灰度选择“灰度”；5. 选中“幻灯片加框”复选框；6. 点击“确定”。
预期输出	打印出给定范围的幻灯片的大纲视图，灰度且加框。

Reference

- 清华大学国家级精品课程《软件工程》

- ✓ https://www.icourses.cn/sCourse/course_3016.html

- ✓ https://www.xuetangx.com/course/THU08091000367/5883555?channel=learn_title

- 【黑盒测试用例设计方法详解，通俗易懂】

- ✓ https://www.bilibili.com/video/BV1vk4y1h7GS/?share_source=copy_web&vd_source=0564a7a68c171ecceef424b200481fe8&t=13