



# 黑盒测试

目录页 CONTENTS PAGE

01

等价类划分法

02

边界值测试法

03

决策表

04

错误猜测

05

场景法

06

正交实验法

07

行为建模

08

综合应用



## 第一节 等价类划分



## 第二节 边界值测试



## 第三节 决策表

---



## 第四节 错误猜测

## 定义

基于经验和直觉推测程序中可能存在的各种错误, 针对这些错误设计相应的测试用例

常作为一种补充测试用例的设计方法

## Steps

错误猜测设计法是一个在很大程度上凭直觉列出程序中可能出现错误的列表和容易出错

基于该列表构造测试用例





## 第五节 因果图



当程序输入之间有关系，例如：约束关系、组合关系时，用等价类划分和边界值分析是很难描述的，测试效果难以保障。

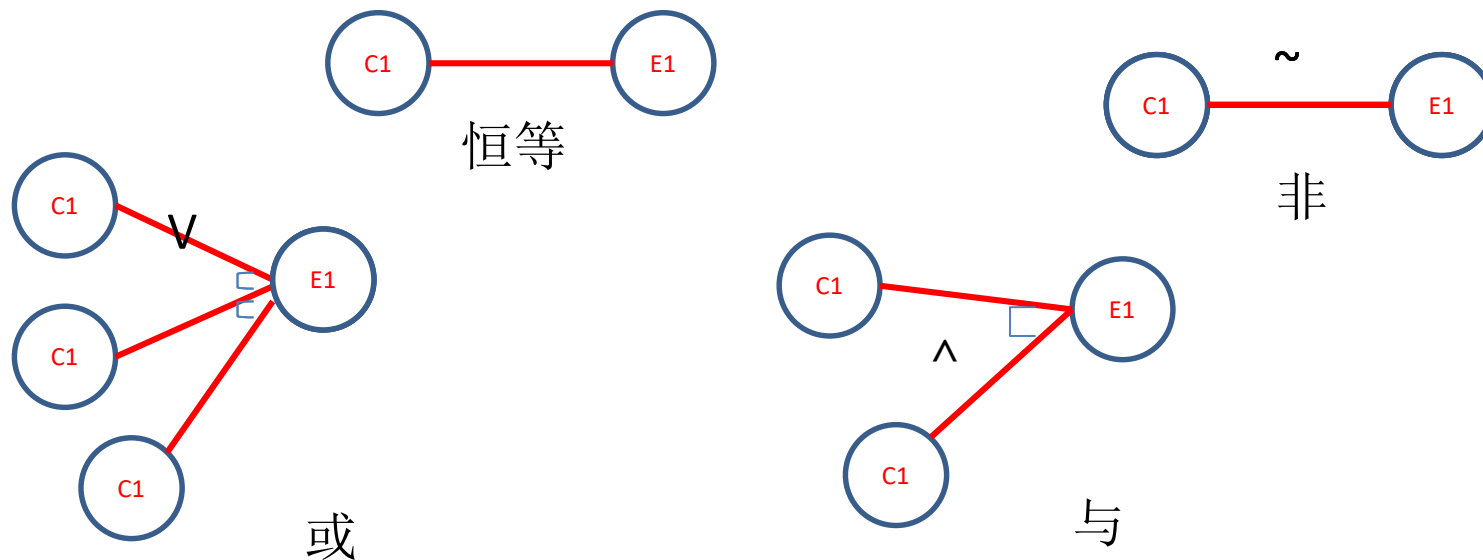
因此必须考虑使用一种适合于描述对于多种条件的组合，产生多个相应动作的测试方法——因果图法。

**因果图法**——就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，它适合于检查程序输入条件的各种情况的组合。

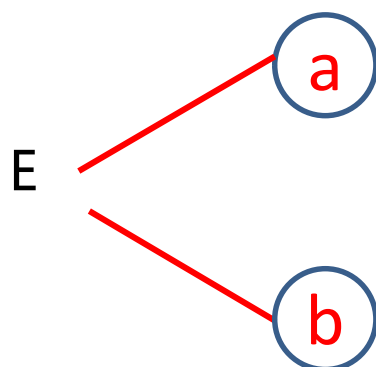
如果开发项目在设计阶段就采用的决策表，就不必再画因果图，可以直接利用决策表设计测试用例。

## ■ 因果图法的原理

- 因果图：以直线连接左右结点，左结点表示输入状态（因），右结点表示输出状态（果）。



- 在实际问题中，输入状态之间可能存在某些依赖关系，称为约束。

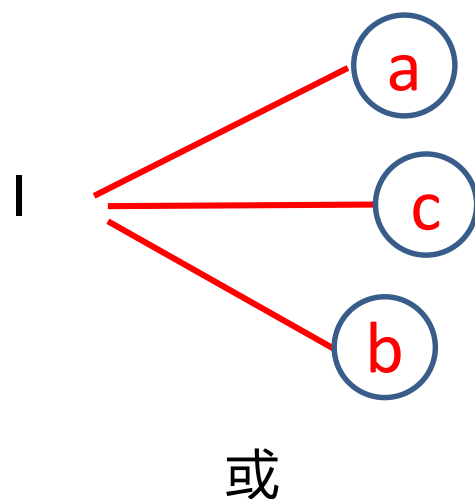


异或

（输入条件）约束的含义：

E (Exclusive or 异或) : a和b中最多只能有一个为1，即a和b不能同时为1。

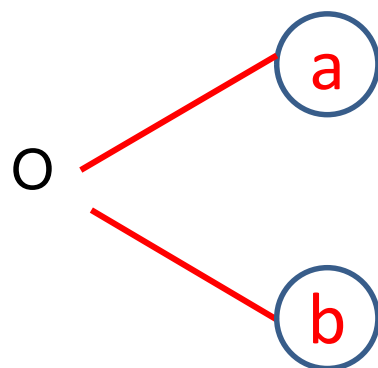
- 在实际问题中，输入状态之间可能存在某些依赖关系，称为约束。



（输入条件）约束的含义：

I（In或）：a、b和c中至少有一个为1，即a、b和c不能同时为0

- 在实际问题中，输入状态之间可能存在某些依赖关系，称为约束。

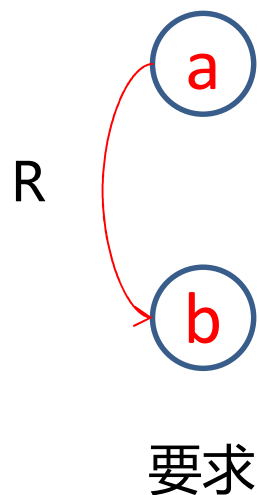


唯一

（输入条件）约束的含义：

O（Only唯一）：a和b必须有一个，且仅有一个为1.

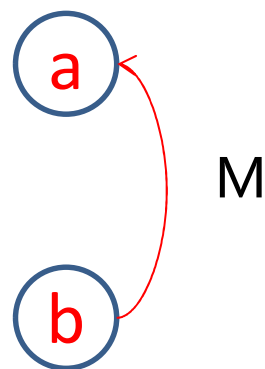
- 在实际问题中，输入状态之间可能存在某些依赖关系，称为约束。



（输入条件）约束的含义：

R（Request要求）：a是1时，  
b必须是1，

- 在实际问题中，输入状态之间可能存在某些依赖关系，称为约束。



强制

（输出条件）约束的含义：

M（强制）：如果结果a是1，  
则结果b强制为0



设计步骤：

1) 确定软件规格中的原因和结果。

分析规格说明中哪些是原因（即输入条件或输入条件的等价类），哪些是结果（即输出条件），并给每个原因和结果赋予一个标识符。

设计步骤：

2) 确定原因和结果之间的逻辑关系。

分析软件规格说明中的语义，找出原因与结果之间、原因与原因之间对应的关系，根据这些关系画出因果图。

设计步骤：

**3)** 确定因果图中的各个约束。

在因果图上用一些记号表明有些原因与原因之间、原因与结果之间不可能出现的组合情况，即约束或限制条件。

**4)** 把因果图转换为决策表。

**5)** 根据决策表设计测试用例。

- 因果图法的测试运用

- 程序的规格说明要求：输入的第一个字符必须是#或\*，第二个字符必须是一个数字，在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。

- 1. 将原因和结果分开。

- 原因：

- » c1——第一个字符是#

- » c2——第一个字符是\*

- » c3——第二个字符是一个数字

- 结果：

- » e1——给出信息N

- » e2——修改文件

- » e3——给出信息M

# 因果图法

— 20 —

1.绘制因果图。其中编号为10的中间节点是导出结果的进一步原因。

原因:

c1——第一个字符是#

c2——第一个字符是\*

c3——第二个字符是一个

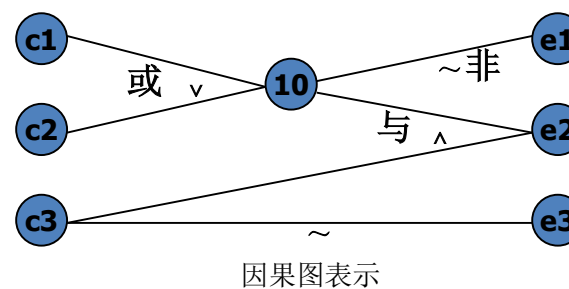
数字

结果:

e1——给出信息N

e2——修改文件

e3——给出信息M



# 因果图法

— 21 —

2.画出约束条件。因为c1和c2不可能同时为1，即第一个字符不可能既是#又是\*，在因果图上可对其施加E约束，得到具有约束的因果图。

原因：

c1——第一个字符是#

c2——第一个字符是\*

c3——第二个字符是一个

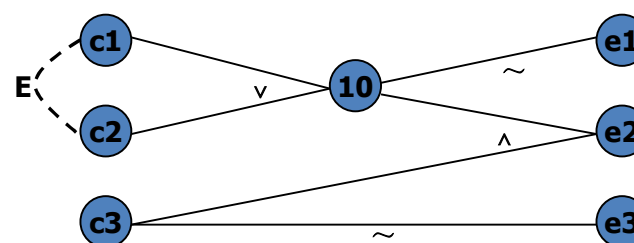
数字

结果：

e1——给出信息N

e2——修改文件

e3——给出信息M



具有E约束的因果图表示

## 3.将因果图转换成决策表

根据因果图建立的决策表

规则 选项	1	2	3	4	5	6	7	8
条件: c1								
c2								
c3								
10								
动作: e1								
e2								
e3								
不可能								



4. 设计测试用例。根据上一步中的决策表，最左面的两列c1和c2同时为1是不可能的，需要排除。所以，可设计出6个测试用例：

- 1: 输入数据——#3 预期结果——修改文件
- 2: 输入数据——#A 预期结果——给出信息M
- 3: 输入数据——\*6 预期结果——修改文件
- 4: 输入数据——\*B 预期结果——给出信息M
- 5: 输入数据——A1 预期结果——给出信息N
- 6: 输入数据——GT 预期结果——给出信息M和N

例如，有一个处理单价为 5 角钱的饮料自动售货机，软件测试用例的设计规格说明如下：

若投入 5 角钱或 1 元钱的硬币，押下〔橙汁〕或〔啤酒〕的按钮，则相应的饮料就送出来。若售货机没有零钱找，则一个显示〔零钱找完〕的红灯亮，这时在投入 1 元硬币并押下按钮后，饮料不送出来而且 1 元硬币也退出来；若有零钱找，则显示〔零钱找完〕的红灯灭，在送出饮料的同时退还 5 角硬币。”