

第8章因果图

# 上一章回顾

* 边界值的基本概念
* 边界条件类型
* 边界值的选择

# 课堂提问

* 内点
* 上点
* 离点

# 本章学习目标

* 能够掌握因果图设计测试用例

# 本章学习方法

* 运用

# 本章主题

* 因果图的基本概念
* 因果图的需要掌握的知识
* 因果图法设计测试用例的步骤
* 因果图实例
* 因果图适合场合

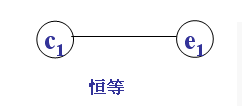
# 因果图的基本概念

* 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，它适合于检查程序输入条件的各种组合情况

# 因果图的需要掌握的知识

* 原因和结果之间的关系（NOT、AND、OR、

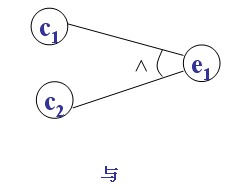
NAND、NOR、恒等）

**恒等：如果原因为真，那么结果必定为真。**

**例如：动物园运来大熊猫，动物**

**园一定有大熊猫**

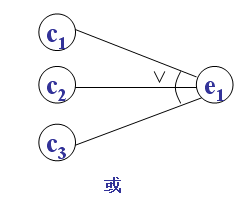
# 原因和结果之间的关系



**与：只有2个原因都为真，那么结果为真**

**例如：北京姑娘，必须有车且有房**

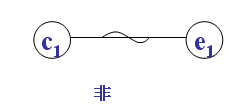
# 原因和结果之间的关系

**或：2个原因中有一个为真时， 结果就为真。**

**例如：长沙姑娘，你有车或者有**

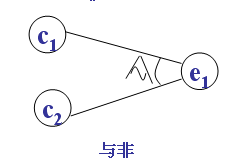
**房**

# 原因和结果之间的关系



**非：只有原因为假，结果才为真。例如：你不好好学习，找到好工 作**

# 原因和结果之间的关系

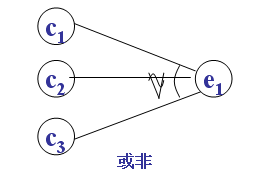


**与非：先与后非。(只要有一个 为假，结果就为真，2个都为真， 结果为假)**

**例子：有爸爸，有妈妈，不是一**

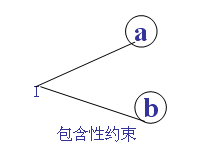
**个幸福的孩子**

# 原因和结果之间的关系



**或非：先或后非（只要有1个为真，结果为假；两个为假，结果为真）**

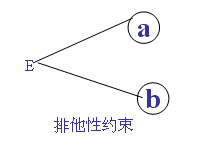
# 因果图的需要掌握的知识

* 原因和原因的关系

**各个原因中总有一个为真。即可以同时为真，但不可以同时为假。**

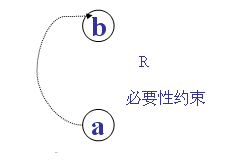
# 因果图的需要掌握的知识

* 原因和原因的关系



**各个原因之间不能同时为真，但可以同时为假（例如：小明不是11班学生，也不是12班学生）**

# 因果图的需要掌握的知识

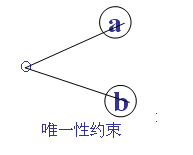
* 原因和原因的关系

**必要性约束：当原因a为真时，原因b必须同时为真； 但是原因b为真时，原因a既可以为真，也可以为假。**

**例如： 2个原因--你是妈妈， 你结婚了**

# 因果图的需要掌握的知识

* 原因和原因的关系

**唯一性约束：有且只有原因a和原因b中的一个为真。非此即彼，不存在第三种情况。例如：男女**

# 输出条件约束类型

**结果与结果的关系：掩码标记（结果约束）：如果结果b为真，那么结果a一定为假，如果结果b为假，则结果a的状态不定。例如：2个结果—你变得难看了、你变得漂亮了**

因果图法设计测试用例的步骤

* 分析软件规格说明描述中, 那些是原因(即输入条件或输入条件的等价类),那些是结果(即输出条件), 并给每个原因和结果赋予一个标识符。
* 分析软件规格说明描述中的语义，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。
* 由于语法或环境限制, 有些原因与原因之间,原因与结果之间的组合情况不可能出现，为表明这些特殊情况, 在因果图上用一些记号表明约束或限制条件。
* 把因果图转换为判定表。
* 把判定表的每一列拿出来作为依据,设计测试用例。

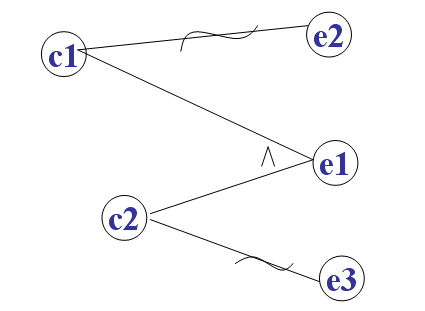
# 因果图示例

* 需求：
* 一胖妞减肥，第一条件必须是每餐只能吃 一个苹果或者一块牛肉，第二个条件必须 每天锻炼1个小时，在此情况下则可以每天减肥1斤。但如果第一个条件不满足，则出现经常晕倒情况；如果第二个条件不满足， 则只能减1俩肉

# 确定原因和结果

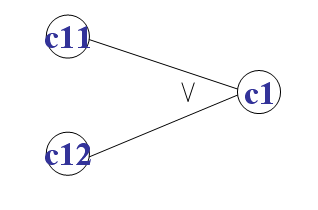
|  |  |  |
| --- | --- | --- |
| 原因 | c1 | 每餐吃一个苹果或者一  个块牛肉 |
| c2 | 每天锻炼1个小时 |
| 结果 | e1 | 每天减肥1斤 |
| e2 | 经常晕倒 |
| e3 | 减肥1俩 |

确定因果逻辑关系

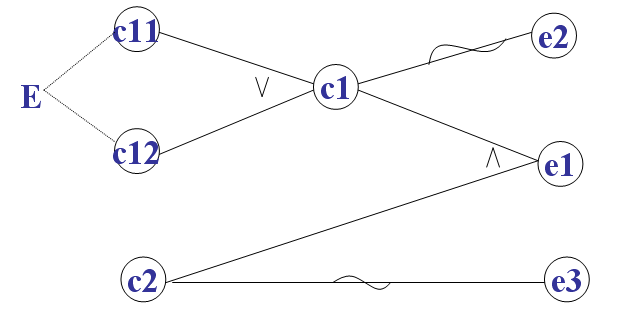




* 根据需求描述，原因c1还可以细分为2个原因：第一是吃一个苹果（c11），或者是吃一块牛肉（c12）。因此原因c1其实也可以看作成结果。



# 确定约束关系



根据因果图画决策表

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 原  因 | c11 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| c12 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| c2 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 结  果 | e1 |  |  |  |  |  |  |  |  |
| e2 |  |  |  |  |  |  |  |  |
| e3 |  |  |  |  |  |  |  |  |



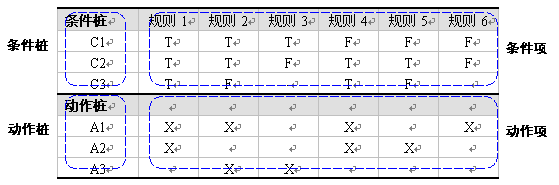
* 根据原因分析结果：分析每一种状态对应的结果，并根据约束关系，去掉不可能出现的状态

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | | 8 | | |
| 原  因 | c11 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | | | 1 | | |
| c12 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | | | 1 | | |
| c2 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | | | 1 | | |
| 结果 | e1 | 0 | 0 | 0 | 1 | 0 | 1 | 无此 | | | 无此 | | |
| e2 | 1 | 1 | 0 | 0 | 0 | 0 |  | 可  能 |  |  | 可  能 |  |
| e3 | 0 | 0 | 1 | 0 | 1 | 0 |

# 设计测试用例

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | | 8 | | |
| 原  因 | c11 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | | | 1 | | |
| c12 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | | | 1 | | |
| c2 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | | | 1 | | |
| 结  果 | e1 | 0 | 0 | 0 | 1 | 0 | 1 | 无  此 | | | 无  此 | | |
| e2 | 1 | 1 | 0 | 0 | 0 | 0 |  | 可  能 |  |  | 可  能 |  |
| e3 | 0 | 0 | 1 | 0 | 1 | 0 |
| 测试用例 | |  |  |  |  |  |  |  | | |  | | |





**规则:任何一个条件组合的特定取值及其相应要执行的操作.在判定表中贯穿条件项和动作项的一列就是一条规则.显然,判定表中列出多少组条件取值,也就有多少条规则,既条件项和动作项有多少列**

**条件桩（Condition Stub）:列出了问题的所有条件.通常认为列出的条件的次**

**序无关紧要.**

**动作桩（Action Stub）:列出了问题规定可能采取的操作.这些操作的排列顺**

**序没有约束.**

**条件项（Condition Entry）:列出针对它左列条件的取值.在所有可能情况下的真假值.**

**动作项（Action Entry）:列出在条件项的各种取值情况下应该采取的动作.**

# 因果图适合场合

* + 因果图适用于检查程序输入条件的各种组 合情况。利用因果图设计测试用例需要考 虑输入条件与输出结果之间的因果关系， 而这些因果关系很难从需求规格说明书中 直接得出，一般软件系统的因果关系非常 的复杂，所以利用因果图设计用例时费时、费力。

# 因果图事例

* + 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。”

# 分析

* + 条件的组合，适用因果图
* 条件桩
* 投入5角 c1
* 投入1元 c2
* 按下橙汁c3
* 按下啤酒c4
* 有零钱找c5
* 无零钱找c6
* 动作桩
* [零钱找完]的红灯灭，退还5角钱，送出橙汁e1
* [零钱找完]的红灯灭，退还5角钱，送出啤酒e2
* [零钱找完]的红灯灭，送出橙汁e3
* [零钱找完]的红灯灭，送出啤酒e4
* [零钱找完]的红灯亮，退出1元钱，饮料不送出来e5

# 判定表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **原因** | | | | | | |
| **投5角** | **T** | **T** | **F** | **F** | **F** | **F** |
| **投1元** | **F** | **F** | **T** | **T** | **T** | **T** |
| **按橙汁** | **T** | **F** | **T** | **F** | **T** | **F** |
| **按啤酒** | **F** | **T** | **F** | **T** | **F** | **T** |
| **有零钱** | **-** | **-** | **T** | **T** | **F** | **F** |
| **无零钱** | **-** | **-** | **F** | **F** | **T** | **T** |
| **结果** |  |  |  |  |  |  |

支付宝

* 支付宝个人认证中，分为两部分：
  + 个人身份认证和银行卡认证。这两者都通过后，认为个人认证成功。
  + 个人身份认证需要提交个人基本信息及身份证复印件。
  + 银行卡认证分为两种：提现认证和充值认证。

提现认证的流程是：

用户提交正确的银行帐号——>支付宝给用户的银行卡中随机打款——>用户确认金额，认证成功。

充值认证的流程是：

用户提交正确的银行帐号——>充值——>充值完成——>网银反馈，认证成功。

# 分析

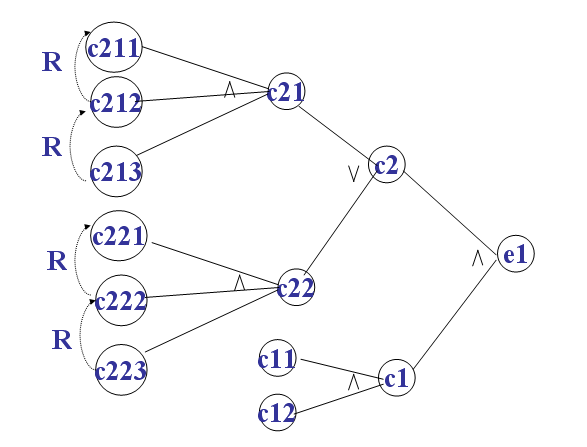
* + 原因一：身份认证成功
* 身份认证成功也是一个中间结果，它也有2个原因，提交基本信息成功和提交身份证成功。
  + 原因二：银行卡认证成功，包含2个原因：充值认证成功和提现认证成功。这2种原因也可以看做是中间结果，产生结果的原因在需求中可以也能明显看出来，不再赘述。
  + 一个结果：个人认证成功。

注意：为了简便起见，我们假设个人信息提交和身份证件提交成功后，身份认证则成功，忽略人工审核过程。



|  |  |  |
| --- | --- | --- |
| 原因 | c11 | 个人基本信息提交成功 |
| c12 | 个人身份证件提交成功 |
| 原因 | c221 | 充值认证的银行帐号提交成功 |
| c222 | 充值成功 |
| c223 | 网银反馈成功 |
| 原因 | c211 | 提现认证的银行帐号提交成功 |
| c212 | 支付宝打款成功 |
| c213 | 用户确认成功 |
| 中间结果 | c21 | 银行卡提现认证成功 |
| c22 | 银行卡充值认证成功 |
| 中间结果 | c1 | 身份认证成功 |
| c2 | 银行卡认证成功 |
| 结果 | e1 | 个人认证成功 |





# 象棋

分析中国象棋中走马的实际情况（下面未注明的均指的是对马的说明）

* + 如果落点在棋盘外，则不移动棋子；
  + 如果落点与起点不构成日字型，则不移动棋子；
  + 如果落点处有自己方棋子，则不移动棋子；
  + 如果在落点方向的邻近交叉点有棋子（绊马腿），则不移动棋子；
  + 如果不属于1-4条，且落点处无棋子，则移动棋子
  + 如果不属于1-4条，且落点处为对方棋子(非老将)，则移动棋子并除去对方棋子；
  + 如果不属于1-4条，且落点处为对方老将，则移动棋子， 并提示战胜对方，游戏结束。

# 分析

根据分析明确原因和结果

* + 原因:
* 落点在棋盘上
* 落点与起点构成日字；
* 落点处为自己方棋子；
* 落点方向的邻近交叉点无棋子；
* 落点处无棋子；
* 落点处为对方棋子（非老将）；
* 落点处为对方老将。
  + 结果：
* 不移动棋子；
* 移动棋子；
* 移动棋子，并除去对方棋子
* 移动棋子，并提示战胜对方，结束游戏

# 确定原因和结果

总结

* + 因果图的基本概念
  + 因果图的需要掌握的知识
  + 因果图法设计测试用例的步骤
  + 因果图实例
  + 因果图适合场合
  + 预习第九章