第3节：动态测试

资料整理
火龙果软件
荐

目录

- 动态测试
- 动态测试目标
- 动态测试方法

资料整理
火龙果软件
荐

动态测试

- 软件测试分为静态测试和动态测试，本节关注动态测试：
  - 所谓动态测试，就是通过运行软件来检验软件的动态行为和运行结果的正确性；
  - 运行软件并非动态测试的目的，通过运行来检验软件是否正确才是动态测试的真正目的；
  - 动态测试包括三个基本要素：
    - 被测试程序；
    - 用以运行软件的数据，称为测试数据；
    - 软件需求规格说明；

资料整理
火龙果软件
荐

动态测试目标

- 一个成熟的动态测试方法必须回答如下问题：
  - 如何选择或产生测试数据？
  - 如何组织软件的测试运行？
  - 如何考察和记录软件动态运行的行为？
  - 如何判断软件动态行为的正确性？
  - 测试过程何时结束？
  - 如何通过软件测试结果分析软件性质？

资料整理
火龙果软件
荐

动态测试方法

- 按照选择或产生测试数据所根据的信息的来源，动态测试方法可以分为：
  - 以程序为基础的测试
  - 以需求和功能说明为基础的测试
  - 程序与需求相结合的测试
  - 以界面为基础的测试
- 按照选择或产生测试数据以及判断测试充分性的方法，动态测试方法可以分为：
  - 结构测试（白箱测试）
  - 功能测试（黑箱测试）

资料整理
火龙果软件
荐

结构测试

- 也称白箱测试或逻辑驱动测试；
- 把测试对象看成一个透明的盒子，允许测试人员利用程序内部的逻辑结构及有关信息，设计或选择测试用例，对程序中的所有逻辑路径进行测试。
- 旨在充分地覆盖软件的结构，并以软件中的某类成分是否都已得到测试为准则来判断软件测试的充分性

资料整理
火龙果软件
荐

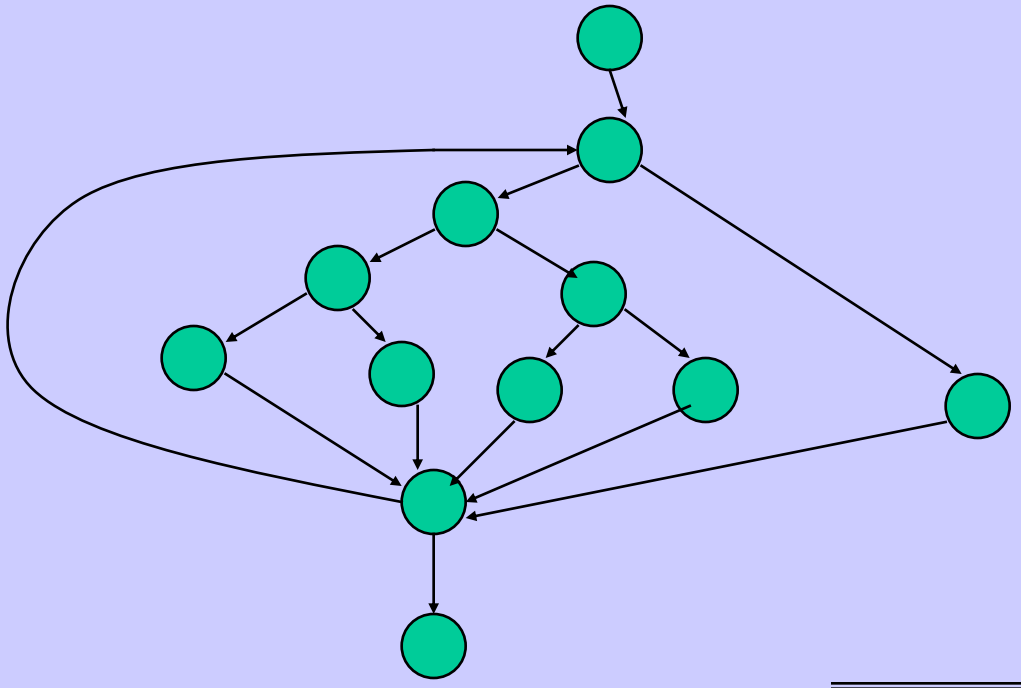
结构测试方法

- 对程序模块的独立的执行路径至少测试一次；
- 对所有的逻辑判定，取“真”或“假”的两种情况都至少测试一次；
- 在循环的边界和运行界限内执行循环体；
- 测试内部数据结构的有效性；
- 是一种穷举测试，工作量大，实际中很难展开；

资料整理
火龙果软件
荐

举例：穷尽结构测试-1

- 一段程序对嵌套的IF语句执行20次，共有$5^{20}\approx10^{14}$条通路，若每执行一次需1毫秒，则需3000年。



资料整理
火龙果软件
荐

举例：穷尽结构测试-2

- 在测试阶段不可行，为了节省时间和测试效率，就必须从极大的可用测试用例中精心地挑选少量的测试数据，使得采用这些测试数据能够达到最佳的测试效果，能够高效的把隐藏的错误暴露出来；
  - 逻辑覆盖
  - 基本路径测试

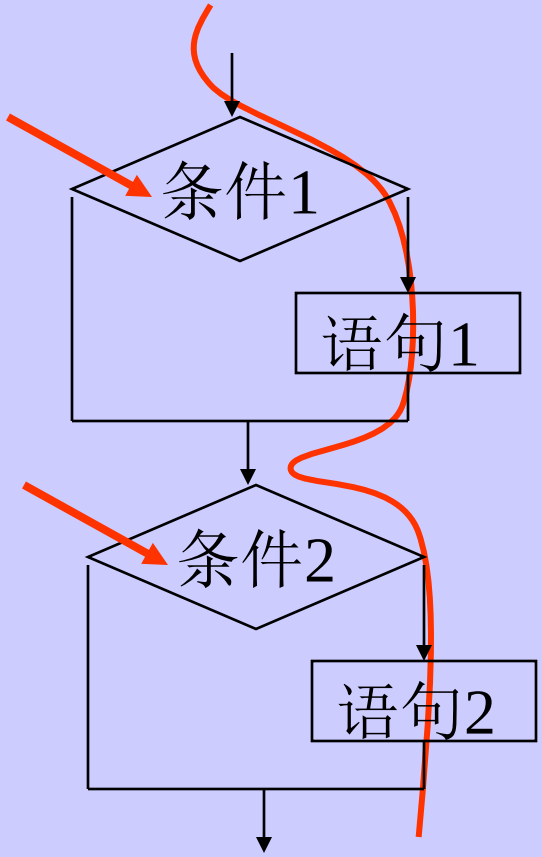
资料整理
火龙果软件
荐

逻辑覆盖

- 以程序内部的逻辑结构为基础设计测试用例的技术；
- 要求测试人员对程序的逻辑结构有清楚的了解，甚至要求能够掌握源代码的所有细节；
- 考虑的重点是图中的判定框（与选择或循环有关），根据覆盖测试的目标不同进行分类：
  - 语句覆盖
  - 判定覆盖
  - 条件覆盖
  - 判定-条件覆盖
  - 条件组合及路径覆盖

资料整理
火龙果软件
荐

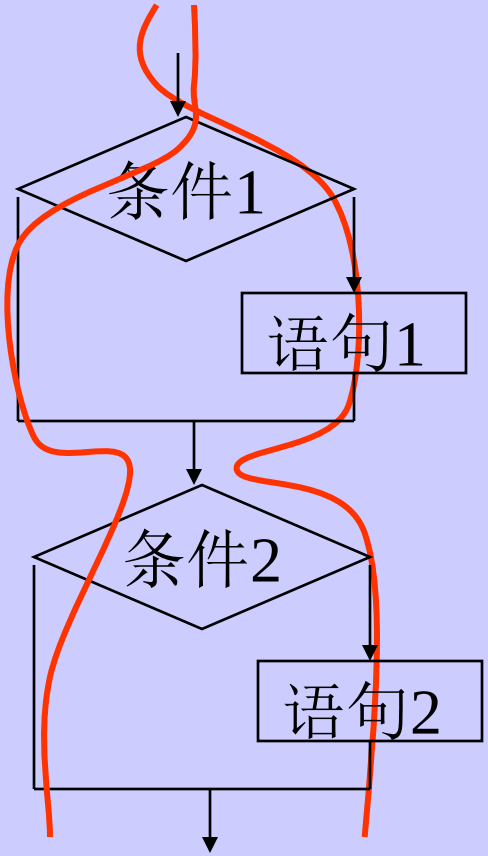
语句覆盖



- 设计若干个测试用例，运行被测试程序，使得每条可执行语句至少执行一次；
- 只是覆盖了可执行的语句，对于逻辑运算出现的错误不能发现，因此语句覆盖是最弱的逻辑覆盖准则，效果有限，必须与其他方法交互使用；

资料整理
火龙果软件
荐

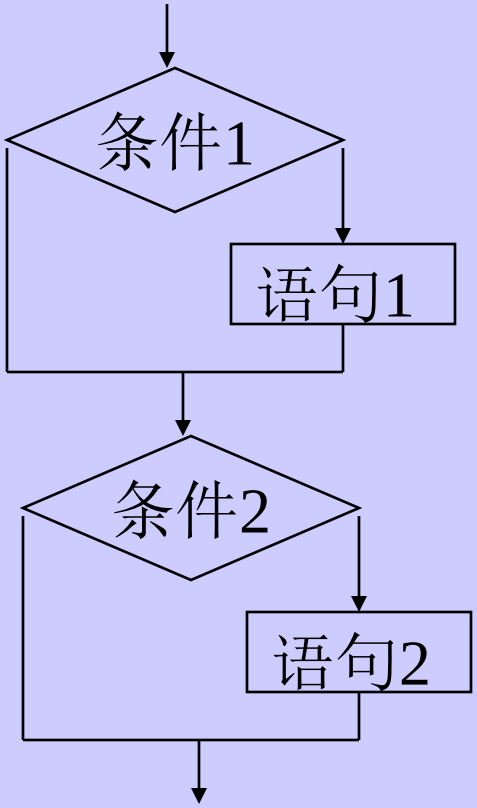
判定覆盖



- 设计若干个测试用例，运行被测试程序，使得程序中每个判断的取真或取假的分支至少执行一次；
- 测试用例的取法不唯一，判定覆盖比语句覆盖稍强，但是应当注意，只是判定覆盖仍然不能查处在判断的条件中存在的错误；

资料整理
火龙果软件
荐

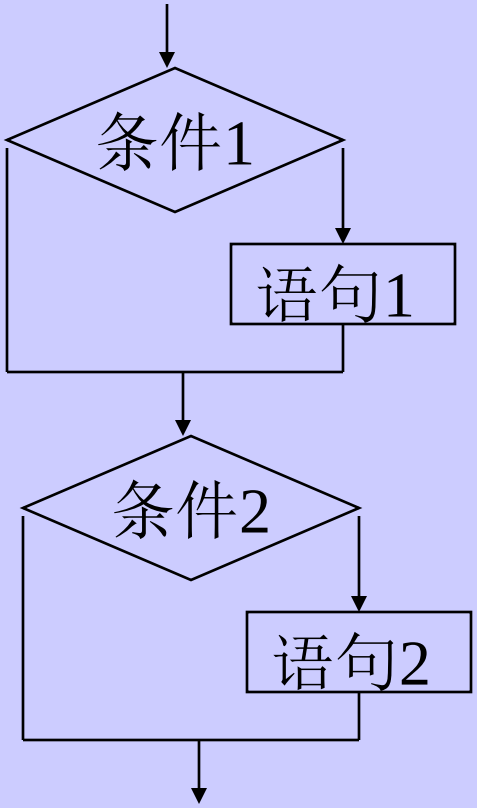
条件覆盖



- 设计若干个测试用例，运行被测试程序，使得程序中每个判断的每个条件的可能的取值至少执行一次；
- 条件1有哪些些情况以及条件2有哪些些情况
- 比前两者更强大，但是只对条件进行判断，没有兼顾对分支进行判断；

资料整理
火龙果软件
荐

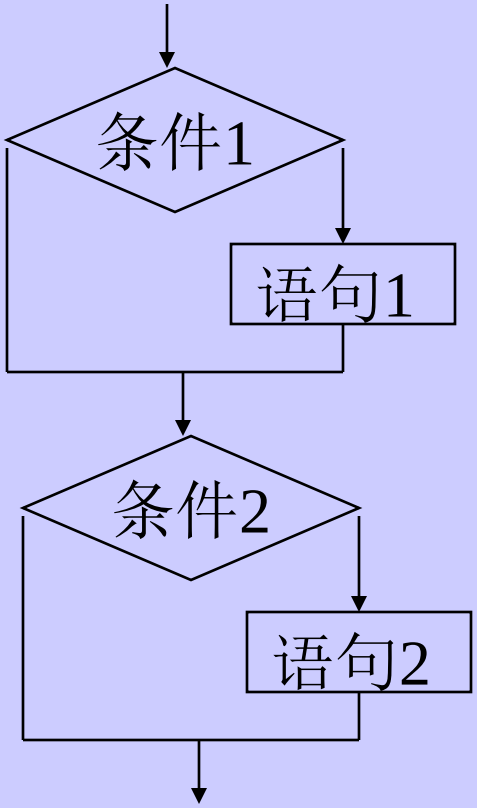
判定-条件覆盖



- 设计若干个测试用例，运行被测试程序，使得各个判断的各种条件组合至少执行一次；
- 注意：此法往往某些条件会掩盖另一些条件，这样，逻辑表达式中的错误不一定能够查的出；

资料整理
火龙果软件
荐

条件组合及路径覆盖



- 设计足够多的测试用例，运行被测试程序，使得每个判断的所有可能组合至少执行一次；
- 设计足够多的测试用例，运行被测试程序，覆盖所有的可能的路径；

资料整理
火龙果软件
荐

基本路径测试

- 把覆盖的路径数目压缩到一定的限度内，程序的循环体最多只执行一次。
- 在程序流程图的基础上，分析控制结构的环路复杂性，导出基本可执行路径集合，来设计测试用例的方法；设计出的测试用例要保证在测试中，程序的每一条可执行语句至少要执行一次；

资料整理
火龙果软件
荐

基本路径测试特征

- 满足结构测试的最低要求；
- 有利于安排循环测试；
- 循环测试是基本路径测试的一个重要部分；
- 从程序的执行路径而不是单纯的角度来测试程序，有利于对程序中包含的循环结构进行更有效的测试；

资料整理
火龙果软件
荐

功能测试

- 也称黑箱测试；
- 依据软件的需求规格说明书，检查程序的功能是否满足符合用户要求
- 这种方法不需要考虑程序内部的逻辑结构和内部特性，完全把测试对象看成一个黑盒子；
- 通常，黑盒测试在软件开发和测试过程的后期进行；

资料整理
火龙果软件
荐

功能测试内容

- 程序是否按照需求完整正确地实现了功能；
- 程序的接口是否正确有效；
- 程序输出的数据或外部数据库访问是否正确；
- 程序性能是否符合需求；
- 程序初始化和终止是否正确；
- 是一种穷尽性测试，实际中无法实施

资料整理
火龙果软件
荐

举例：穷尽性功能测试

- 例：一个程序需3个整型的输入数据，若计算机的字长为16位，则每个数据可能取的值有2^{16}个，3个数的排练组合共有

$$2^{16} \times 2^{16} \times 2^{16} = 2^{48} \approx 3 \times 10^{14}$$（种）

若每执行一次需1毫秒，则需1万年。

必须设法使之可行

资料整理
火龙果软件
荐

功能测试方法

- 等价类划分
- 边界值分析
- 错误推测
- 因果图
- 功能图

资料整理
火龙果软件
荐

等价类划分

- 等价类法是功能测试的一种方法
  - 等价类是指某个输入域的子集，在该子集中，各个输入数据对于揭露程序中的错误都是等效的，测试某等价类的代表值就是对这一类的其他值的测试；
  - 对全部数据进行划分，形成若干个等价类，在每个等价类中取一个数据作为测试的输入条件，利用这种有代表的测试数据，能够取得比较好的测试效果，同时大大缩减了测试用例的数目；

资料整理
火龙果软件
荐

等价类

- 有效等价类：对于程序规格说明来说是合理的、有意义的输入数据构成的集合。利用这些数据可以检验程序是否实现了规格说明书预先规定的功能和性能；
- 无效等价类：对于程序规格说明来说是不合理的、无意义的输入数据构成的集合。利用这些数据可以检查程序是否有不符合规格说明书预先规定的功能和性能；

资料整理
火龙果软件
荐

等价类划分的步骤

- 划分等价类：把输入数据划分成若干个有效等价类和无效等价类；
- 确定测试用例：确定等价类之后，建立等价类表，列出所有划分的等价类；

资料整理
火龙果软件
荐

划分等价类原则 -1

- 如果输入条件规定了取值范围，或值的个数，则可以确定一个有效等价类和两个无效等价类
  - 比如：输入条件规定“…成绩可以从0-100分之间…”
  - 则，有效等价类是[0，100]，无效等价类是<0或>100

资料整理
火龙果软件
荐

划分等价类原则-2

- 如果输入条件规定了输入值的集合，或者是规定了“必须如何”的条件，而且程序对这组可能的值作为相同的处理，这时可确定一个有效等价类或无效等价类；
  - 比如在C语言中规定对变量标志符必须“…以字母开头的…串”，则有效等价类是所有以字母开头的串，其他则为无效等价类

资料整理
火龙果软件
荐

划分等价类原则-3

- 如果输入是一个BOOL变量，则可以确定一个有效等价类和一个无效等价类；

资料整理
火龙果软件
荐

划分等价类原则-4

- 如果规定输入数据是一组值，而且程序对每个组进行分别处理，则可为每个输入值确定一个有效等价类，此外针对这组值确定一个无效等价类（所有不允许的输入值集合）
  - 比如有效等价类是：教授、副教授、讲师和助教，无效等价类是不符合以上组合的输入值

资料整理
火龙果软件
荐

划分等价类原则-5

- 如果规定了输入数据必须遵守的规则，则可以确定一个有效等价类和若干多个无效等价类；
  - 比如：C语言中规定“一个语句必须以分号结束”，这时可确定一个有效等价类“以分号结束”，若干个无效等价类“以：结束”、“以‘’结束”等

资料整理
火龙果软件
荐

确定测试用例

- 为每一个等价类指定一个唯一编号；
- 设计一个新的测试用例，使其尽可能多地覆盖未被覆盖的有效等价类，重复这些过程，直到所有的有效等价类都被覆盖为止；
- 设计一个新的测试用例，使其尽可能多地覆盖未被覆盖的无效等价类，重复这些过程，直到所有的无效等价类都被覆盖为止；

资料整理
火龙果软件
荐

边界值分析

- 边界值分析是功能测试的一种方法
  - 等价类划分的一种补充，是一种有效的方法，常与等价类划分结合使用；
  - 大量的测试工作表明：软件错误大多发生在输入输出的边界上，一针对各种边界边界条件设计测试用例，使查处更多错误的可能性更大；
  - 首先确定边界情况，通常输入等价类与输出等价类的边界就是应着重测试的边界情况，选取正好等于、刚刚大于、刚刚小于边界的值作为测试数据
  - 当边界条件复杂时，要找出合适的测试用例需要经验积累、耐心考虑；

资料整理
火龙果软件
荐

错误推测

- 错误推测是功能测试的一种方法
  - 依靠经验和直觉推测程序中可能存在的错误，从而有针对性地编写检查这些错误的例子；
  - 基本思想：列举出程序中所有可能的错误和容易发生错误的特殊情况，根据它们选择测试用例；
  - 有一些在模块测试中常见的错误，作为经验的总结，常常用来作为错误推测的依据；此外对于在程序中容易出错的情况，也有一些经验总结出来；

资料整理
火龙果软件
荐

因果图

- 因果图是功能测试的一种方法
  - 使用一种适合于描述对于多种条件的组合产生的多个动作形式来考虑测试用例
  - 利用因果图可以最终生成判定表，它适合检查程序输入条件的各种组合情况；

资料整理
火龙果软件
荐

因果图基本步骤

- 分析软件规格说明书，判定哪些是因（输入条件的等价类）哪些是果（输出条件）并给每个原因和结果赋予一个标志符；
- 分析软件规格说明书中的语义，找出原因与结果之间、原因与原因之间对应的是什么关系？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。为了表明这些特殊情况，在因果图上用一些记号表明约束条件或限制条件；
- 把因果图转换成判定表；
- 把判定表的每一列拿出来作为依据设计测试用

资料整理
火龙果软件
荐

功能图

- 功能图是功能测试的一种方法
  - 对程序功能的形式化说明，功能图由状态图和逻辑模型组成；
  - 在状态图中，起始点称为初始状态，活动的终止点称为终止状态；输入相应的数据，系统产生对应的数据输出并转换到后继状态；
  - 逻辑功能模型表示状态的输入条件和输出条件之间的对应关系，测试用例描述了测试执行的一系列状态转换和每一状态的输入/输出条件；

资料整理
火龙果软件
荐

本节小结

- 动态测试的定义、目标
- 结构测试（白盒测试）定义及方法
- 功能测试（黑盒测试）定义及方法

资料整理
火龙果软件
荐