



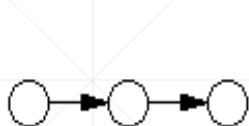
3.白盒测试（续）

- 基本概念
- 节点、边覆盖
- 路径覆盖
- 基本路径覆盖

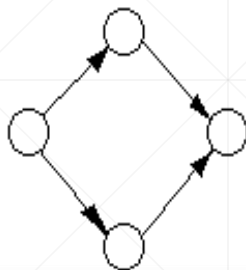
基本概念

控制流图覆盖测试：是将代码转变为控制流图（CFG），基于其进行测试的技术。

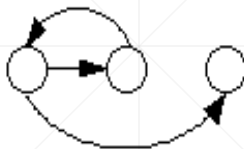
程序的控制流图



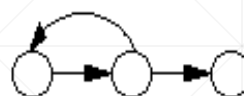
顺序结构



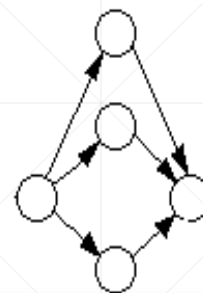
IF 选择结构



WHILE 重复结构



UNTIL 重复结构



CASE 多分支结构



结点：符号○，表示一个或多个无分支的PDL语句或源程序语句。

边：箭头，表示控制流的方向。

汇聚节点：在选择或多分支结构中，分支的汇聚处应有一个汇聚结点。

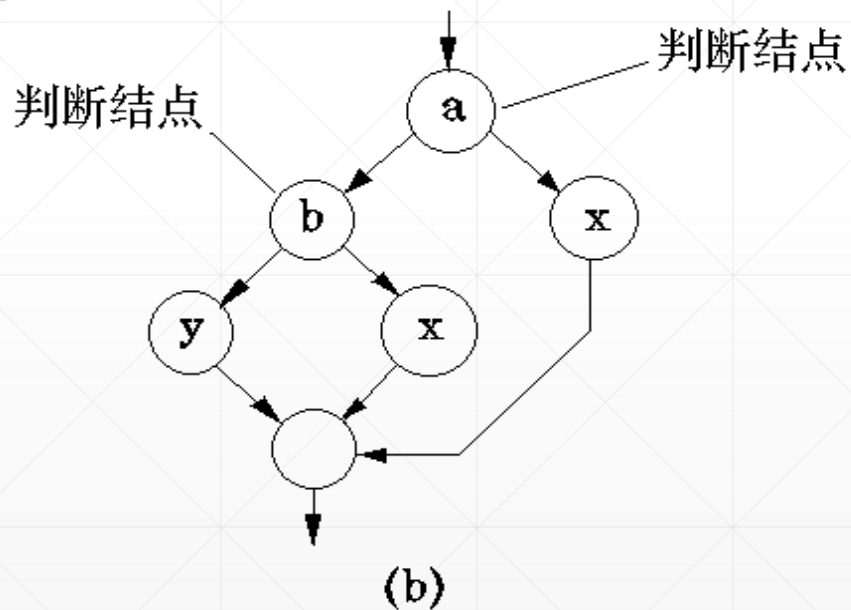
区域：边和结点圈定的区域。对区域计数时，图形外的区域也应记为一个区域。

基本概念

单条件嵌套：如果判断中的条件表达式是由一个或多个逻辑运算符 (OR, AND, NAND, NOR) 连接的复合条件表达式，则需要改为一系列只有单个条件的嵌套的判断。

例

```
·  
·  
·  
if  a OR b  
then procedure x  
else procedure y;  
·  
·  
·  
(a)
```



节点、边覆盖

节点覆盖

对图中的每个节点，至少要有
一条测试路径访问该节点
显然，节点覆盖=语句覆盖



边覆盖

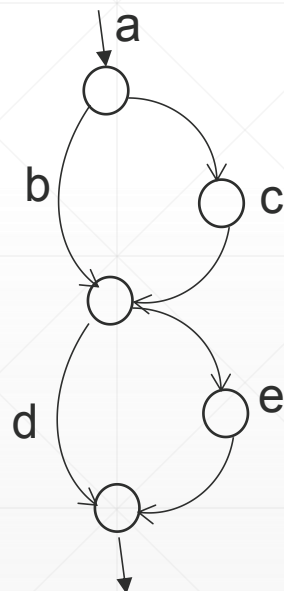
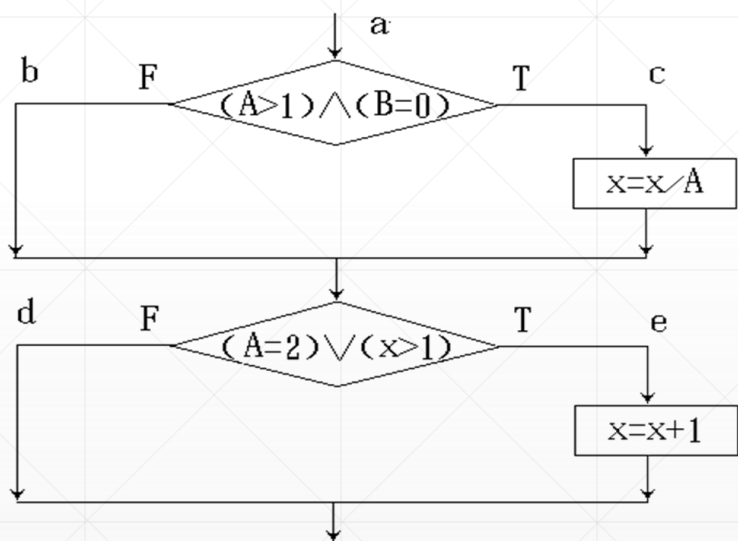
对图中每一个可到达的长度小
于(无边图)等于1 的路径，中至
少存在一条测试路径覆盖。
显然，边覆盖包含节点覆盖，
且边覆盖也可以实现分支覆盖。



路径覆盖

路径覆盖测试：就是设计足够的测试用例，覆盖程序中所有可能的路径

例



测试用例

覆盖路径

- | | |
|------------------------|----------|
| 【(2, 0, 4), (2, 0, 3)】 | ace (L1) |
| 【(1, 1, 1), (1, 1, 1)】 | abd (L2) |
| 【(1, 1, 2), (1, 1, 3)】 | abe (L3) |
| 【(3, 0, 3), (3, 0, 1)】 | acd (L4) |

基本路径覆盖

基本路径测试：将覆盖的路径数压缩到一定限度内，程序中的循环体最多只执行一次。

1

- 绘制程序控制流图

2

- 分析控制构造的环路复杂性

3

- 导出基本可执行路径集合

4

- 设计测试用例的，保证在测试中，程序的每一个可执行语句至少要执行一次。
-

基本路径覆盖

程序的环路复杂性：程序基本路径集中的独立路径条数，这是确保程序中每个可执行语句至少执行一次所必需的测试用例数目的上界。



独立路径：从控制流图来看，一条独立路径是至少包含有一条在其它独立路径中从未有过的边的路径。



计算方法： $V(G) = e - n + 2$ 。

其中， e 为图中边的数目； n 为节点数目。

基本路径覆盖

确定线性独立路径的基本集合

从源节点（控制流图的入口点）开始，一直走到汇节点

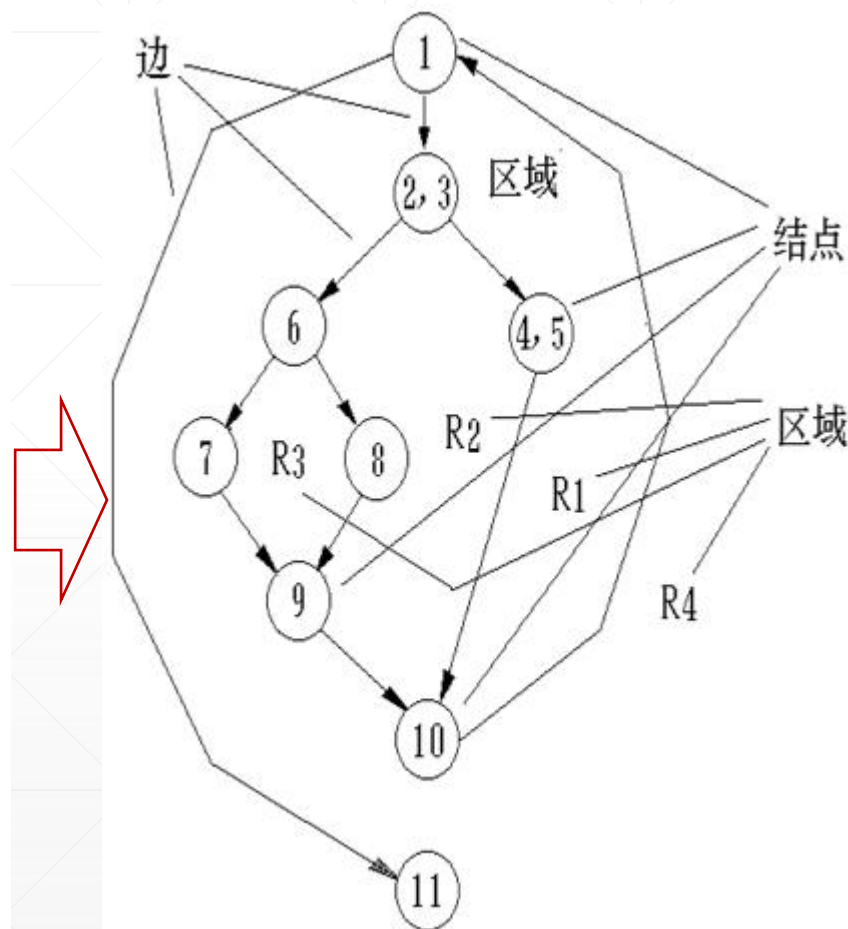
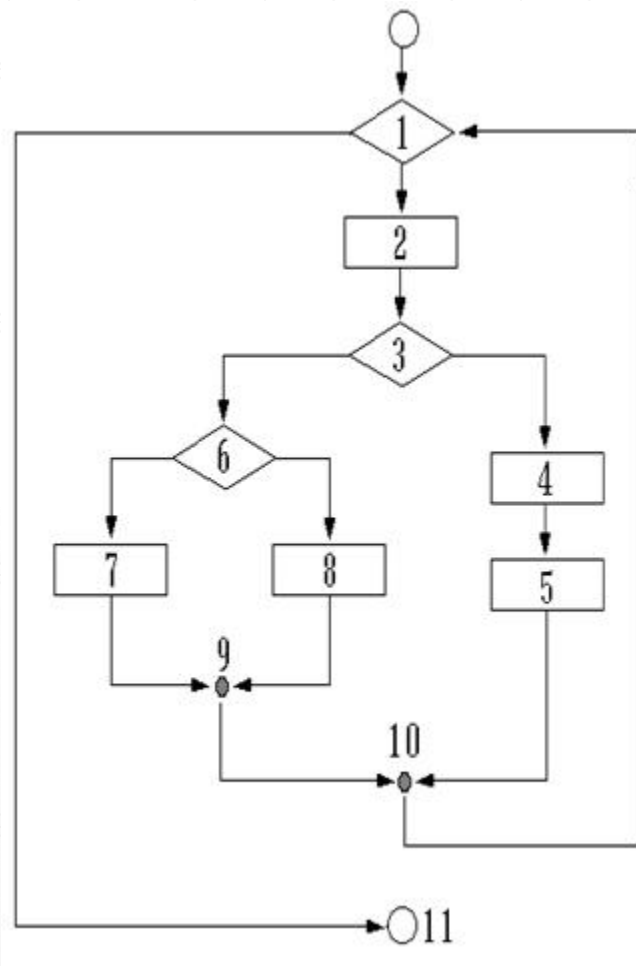
按控制流图回溯到基线路径，替换“翻转线”

在判断节点上原来选择的路径。即当遇到节点的出度大于等于 2 时，

重复以上过程，直到得到的路径数目等于 $V(G)$

基本路径覆盖

例



计算程序环路复杂性：

$$V(G)=e(11)-n(9)+2=4$$

确定基本路径级：

path1 : 1 - 11

path2 : 1 - 2 - 3 - 4 - 5 - 10 - 1 - 11

path3 : 1 - 2 - 3 - 6 - 8 - 9 - 10 - 1 - 11

path4 : 1 - 2 - 3 - 6 - 7 - 9 - 10 - 1 - 11

基本路径集：path1 , path2 , path3 , path4

基本路径覆盖

导出测试用例

确保基本路径集的每一条路径的执行。

选择测试用例

根据判断结点给出的条件，选择合适用例以保证某一条路径可以被测试到

测试结果比较

测试执行后，与预期结果进行比较。

注意事项

非孤立的独的路径可以是另一条路径测试的一部分。



授课教师：吴祖峰

电子邮箱：wuzufeng@uestc.edu.cn