Path Testing – Generating Test Cases

Chapter 9

- **What is the control graph – DD-path graph for the following?**

```
if a < b then c = a+b ; d = a*b
         else c = a*b ; d = a+b
if c < d then x = a+c ; y = b+d
         else x = a*c ; y = b*d
```

Generating Test Cases

- The key question is how to make the path execute, if possible.
  - Generate input data that satisfy all the conditions on the path.
- Key concepts in generating test cases
  - Input vector
  - Predicate
  - Path condition
  - Predicate interpretation
  - Path predicate expression
  - Generating test input from path predicate expression

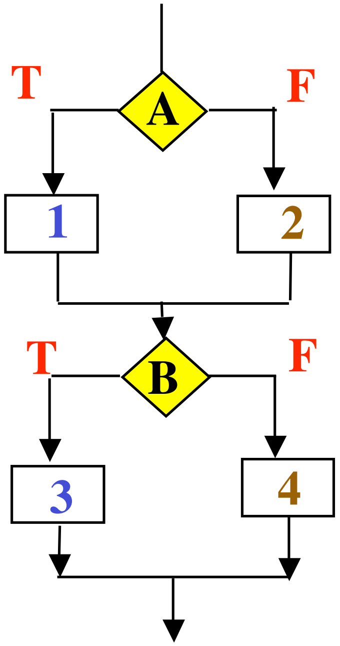
Input Vector

- An input vector is a collection of all data entities read by the routine whose values must be fixed prior to entering the routine.
- Members of an input vector can be as follows.
  - Input arguments to the routine
  - Global variables and constants
  - Files
  - Contents of registers (in Assembly language programming)
  - Network connections
  - Timers

Predicate

- Predicate
  - A predicate is a logical function evaluated at a decision point.
  - Example
    - In the following **a < b** and **c < d** are predicates

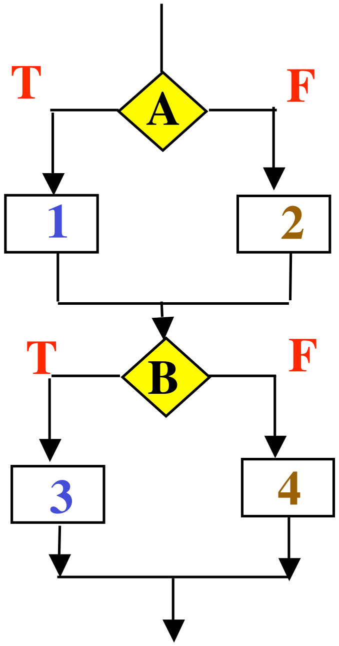
```
if a < b then c = a+b ; d = a*b
          else c = a*b ; d = a+b
if c < d then x = a+c ; y = b+d
          else x = a*c ; y = b*d
```



Path Predicate

- Path predicate
  - A path predicate is the set of predicates associated with a path.
  - Example
    - In the following **a < b = true** and **c < d = false** is a path predicate for path A 1 B 4

```
if a < b then c = a+b ; d = a*b
          else c = a*b ; d = a+b
if c < d then x = a+c ; y = b+d
          else x = a*c ; y = b*d
```



Predicate Interpretation

- A path predicate may contain local variables.
- Local variables play no role in selecting inputs that force a path to execute.
- Local variables can be eliminated by a process called **symbolic execution**.
  - Symbolically substituting operations along a path in order to express the predicate solely in terms of the input vector and a constant vector.
- A predicate may have different interpretations depending on how control reaches the predicate.

Path Predicate Expression

- An interpreted path predicate is called a path predicate expression.
- A path predicate expression has the following attributes.
  - It is void of local variables.
  - It is a set of constraints in terms of the input vector, and, maybe, constants.
  - Path forcing inputs can be generated by solving the constraints.
  - If a path predicate expression has no solution, the path is infeasible.

Path Predicate Generating Input Values

```
if a < b then c = a+b ; d = a*b
          else c = a*b ; d = a+b
if c < d then x = a+c ; y = b+d
          else x = a*c ; y = b*d
```

- Path predicate: $a < b$ = true & $c < d$ = false
- Substitute for c and d:
  $a < b$ = true & $a + b < a * b$ = false
  --> $a < b$ & $a + b \geq a * b$
- Solve for a and b: $a = 0$ & $b = 1$
  Solutions are not unique
- We have feasible path, since a solution exists.
- Can have infeasible paths, if there is no solution to the constraints

Can have decision table

	A1B3	A1B4	A2B3	A2B4
A < B	T	T	F	F
C < D	T	F	T	F
A value	2	0	1	5
B value	5	1	0	2

Paths **A1B3** and **A2B4** give statement coverage
Or paths **A1B4** and **A2B3** give statement coverage

Selecting paths

- A program unit may contain a large number of paths.
  - Path selection becomes a problem. Some selected paths may be infeasible.
  - Apply a path selection strategy:
    - Select as many short paths as possible.
    - Choose longer paths.
  - Make an effort to write code with fewer/no infeasible paths.