Structural Based Testing Strategies

Control Flow Testing

Ira A. Fulton Schools of Engineering

Arizona State University

Objective

Objective

Develop test cases to achieve control flow coverage

Code Coverage

- It is important to analyze code coverage obtained by executing requirement’s based test cases

- Code coverage can be assessed in terms of:
  - Control flow
  - Data flow

- Failure to obtain coverage may be due to:
  - Undocumented requirements contained in the code
  - Dead code
  - Incomplete test cases for a requirement

Control Flow Coverage Levels

- Statement coverage
- Decision coverage
- Decision / Condition coverage
- Multiple condition coverage

Statement Coverage

- Develop test cases such that every statement is executed at least once.

```
if a < 10 or b > 5
  then
         x := 50
  else
         x := 0;
if w = 5 or y > 0
  then
         z := 48
  else
         z : = 5;
```

Decision Coverage

Develop test cases such that each branch is traversed at least once.

- What are examples of branch statements?
- Does decision coverage satisfy statement coverage?
- Does statement coverage satisfy decision coverage?

Decision / Condition Coverage

- Develop test cases such that each condition in a decision takes on all possible outcomes at least once and each decision takes on all possible outcomes at least once

Multiple Condition Coverage

- Develop test cases such that all combinations of conditions in a decision are tested

Binary Search Example

```
inputs:   table, num, key
outputs: found, loc
start := 1;
end := num;
found := false
while start <= end and not found
     middle := (start + end) / 2
     if key > table [middle]
          then start := middle + 1
          else if key = table [middle]
                    then found := true
                           loc := middle
                    else end := middle -1
```

Summary