Lecture 1-B

Basic Concepts of Testing

1 

What is testing?

2 

Verification

- Debugging
  - Locating and fixing the error
- Testing
  - An attempt to reveal errors
- Proving
  - Proving the correctness of the program

3 

Testing Activities

- Define the objectives
- Design the test cases
- Generate the test cases
- Execute the test cases
- Analyze the test results

4 

Why Testing Objective?

5 

Verification

- Debugging
  - Locating and fixing the error
- *Testing*
  - An attempt to reveal errors
- Proving
  - Proving the correctness of the program

6 

Testing demonstrates the presence of faults

Testing does not demonstrate the absence of faults

7 

Testing Activities

- *Define the objectives*
- Design the test cases
- Generate the test cases
- Execute the test cases
- Analyze the test results

8 

Example

Suppose that we are asked to test the following program

Input A, B // A and B are integer variables

C = A - B

Output C

9

Example (continued)

- *Define the objectives*
  - *Correct Arithmetic Operator*
- Design the test cases
  - ……
- Generate the test cases
  - ……
- Execute the test cases
- Analyze the test results

10

Example (continued)

Possible alternatives to C = A – B are:

- C = A + B
- C = A * B
- C = A / B

11

Test Oracle

A mechanism or procedure to check whether the output for any input is correct or not

12

Example of Test Oracle

X**2 – 3*X + 2 = 0

Solutions for X are: 1 and 2

Test oracle is backward substitution and evaluation

13 

Example (continued)

- Define the objectives
- *Design the test cases*
  - ……
- Generate the test cases
  - ……
- Execute the test cases
- Analyze the test results

14 

Example (continued)

A – B =\=

- A + B
- A * B
- A / B

Why?

15

Example (continued)

- Define the objectives
- Design the test cases
- *Generate the test cases*
  - ……
- Execute the test cases
- Analyze the test results

16

Example (continued)

Generate concrete test cases, that is, find concrete values for A and B (which are integers) such that

A – B =\=

- A + B
- A * B
- A / B

17

Example (continued)

By constraint solving

By trial

- A=5 and B=0
- A=15 and B=3

18

Testing Activities

- Define the objectives
- Design the test cases
- Generate the test cases
- *Execute the test cases*
- Analyze the test results

19 

Testing Activities

- Define the objectives
- Design the test cases
- Generate the test cases
- Execute the test cases
- *Analyze the test results*
  - *Apparently* correct results

20 

Another Testing Objective

- *Define the objectives*
  - *Correct Arithmetic Operator*
- Design the test cases
- Generate the test cases
- Execute the test cases
- Analyze the test results

21 

Example (continued)

Possible alternatives to C = A – B are:

- C = B - A
- C = A - A
- C = B – B
- A = A – B
- B = A – B
- ……
- ……

22 

Example (continued)

Possible alternatives to [Input A, B] are:

- Input B, A
- ……..
- ……..

Possible alternatives to [Output C]

- Output A
- Output B
- …….

23

Another Testing Objective

- *Define the objectives*
  - *Correct Variable*
- *Design the test cases*
  - ……
- *Generate the test cases*
  - ……
- Execute the test cases
- Analyze the test results

24

Another Example

Input A, B // A and B are integer variables

C = A – B

D = C* B

Output D

25 

Example (continued)

Possible alternatives to C = A – B are:

- C = A + B
- C = A * B
- C = A / B

Possible alternatives to D = C * B

- D = C + B
- D = C – B
- D = C / B

26 

Scalability Problem

27

Scalability Problem

code-based method (white box)

28

Various Levels of Testing

- Unit test (Module test)
  - Each module is individually tested
- Integration test (incremental test)
  - A set of modules are tested collectively
- Systems test (evaluation test)
  - The entire system is tested
- Acceptance test

29 

Summary

30