

Lecture 3

Metamorphic Testing - I

1

Metamorphic Testing

A Testing Technique
for
Generating Test Cases
&
Testing the Untestable Programs

Testable Programs

- A program is said to be testable if the output of any input can be verified

Example 1 - Testable Programs

- To compute $41^{1/7}$

Suppose the computed output is 1.7

How can we know whether this output is correct or not?

Example 1 - Testable Programs

$$1.7 ** 7 =?= 41$$

5

Example 2 - Testable Programs

- To solve the following system of linear equations

$$3x + 2y - z = 4$$

$$x - 2y - 2z = -9$$

$$2x + y + z = 7$$

Suppose the solutions $x=1$, $y=2$ and $z=3$

Example 2 - Testable Programs

$$3x + 2y - z = 4$$

$$x - 2y - 2z = -9$$

$$2x + y + z = 7$$

To validate the following equalities:

$$3*1 + 2*2 - 3 \quad =?= \quad 4$$

$$1 - 2*2 - 2*3 \quad =?= -9$$

$$2*1 + 2 + 3 \quad =?= 7$$

7

Example 3 - Testable Programs

- To find the values of x such that

$$x^{**67} + 3*(x^{**46}) - x^{**37} + 4.5 = 0$$

Suppose the solutions for x are: 2.17, 6.5, ..

Example 3 - Testable Programs

- $x^{**67} + 3*(x^{**46}) - x^{**37} + 4.5 = 0$

Suppose the solutions for x are: 2.17, 6.5, ..

To validate the following equalities:

- $2.17^{**67} + 3*(2.17^{**46}) - 2.17^{**37} + 4.5 = ? = 0$
- $6.5^{**67} + 3*(6.5^{**46}) - 6.5^{**37} + 4.5 = ? = 0$
-

9

Example 3 - Testable Programs

Are you happy with this validation process?

10

Test Oracle

- A mechanism or procedure against which the computed outputs could be verified

Test Oracle (continued)

- An inverse function
- A backward substitution
-
-
-

Untestable Programs

- A program is said to be untestable or non-testable if the outputs of some inputs cannot be verified

Example 1 – Non-Testable Programs

- A weather forecasting system which reports the amount of rain for a specific date
- A clinical x-ray system
- A self-driving car system
- An earthquake warning system
- Various simulation systems – buildings, nuclear reactors
-

Example 2 – Non-Testable Programs

- Compute the average for 10 million real numbers
- Compute the sum
- Find the maximum, minimum
- Compute sine function
-
-

Example 2 – Non-Testable Programs (continued)

- *sin* function
 - $\sin(0^\circ)=0$
 - $\sin(30^\circ)=0.5$
- Suppose the program returns:
 - $\sin(29.8^\circ)=0.51234$ incorrect
 - $\sin(29.8^\circ)=0.49876$ correct????

Example 3 Non-Testable Programs

Shortest path program $SP(G, a, b)$

where G is a graph, a is the starting node and b is the destination node

$SP(G, a, b)$ returns a path like:

$a - x - y - \dots - s - t - b$

Example 3 Non-Testable Programs (continued)

Find all possible paths from node a to node b

Check against all these possible path to see whether $SP(G, a, b)$: $a - x - y - \dots - s - t - b$ is the shortest

Example 3 Non-Testable Programs (continued)

Number of all possible paths from a to b

Example 3 Non-Testable Programs (continued)

Number of all possible paths from a to b

$$n! = n * (n-1) * (n-2) * \dots * 2 * 1$$

Where n denotes the number of nodes in G

Non-Testable Programs

- A program is said to be non-testable if the output of any input cannot be verified (or cannot be verified in practice)

Test Oracle Problem

- Absence of test oracle
- Test oracles are available but too expensive to be applied

Non-Testable Programs

interchangeably

Test Oracle Problem

Summary

Reference

F. T. Chan, T. Y. Chen, S. C. Cheung, M. F. Lau and S. M. Yiu, Application of Metamorphic Testing in Numerical Analysis, *Proceedings of the IASTED International Conference on Software Engineering*, 191-197, 1998.