Software Testing Methods

Assignment 1

Cem Kaner  J.D., Ph.D.
Florida Institute of Technology
Fall 2002
Note to the Instructor

- I used this in an earlier version of the course, when we worked through the Myers triangle in the first lecture.
- At the start of the first lecture, students would attempt the Myers triangle and hand in their notes. I graded Pass/Fail where the criterion for Fail was did-not-submit or did-not-attempt-the-question. The main value of the hand-in was to give me insight into what the students knew at the start of the course.
- I then lectured on domain testing using the current introductory program (add two numbers) in much the same way as the domain testing notes present it in the current slides. As soon as we got to the point where we talked about a risk-based way of thinking about domain testing, I handed out this assignment. In that version of the course, this happened about Day 2 or Day 3 (2 lecture sessions per week, 15 week course).
- Note that the assignment asks the student to state a risk first, and THEN to state the test that addresses that risk. This is actually important. When I give the assignment with the test case first, students give me a test and then a weak risk description, often something equivalent to “It fails the test.” I get answers of that low quality from so many students in the first assignment that I assign few points to this assignment and give detailed feedback.
- Finally, I provide a rubric. I gave this to the students AFTER they handed in their answers last time. Next time I teach the course, I’ll try handing it to students with the assignment to see if it helps them perform better.
- I’m not sure how or whether I can use this assignment with the latest version of the course slides, but I’ll probably create something like it, run into the same problems (students have a learning curve before they can state risks and relate test cases to them) and find a similar rubric useful.
## Assignment 1

Let’s take one more crack at Myers’ exercise (see last slide for a reminder of the exercise we did in class). I’d like a table like this from you that lists 5 good test cases:

<table>
<thead>
<tr>
<th>Test</th>
<th>Risk</th>
<th>Test Case</th>
<th>Why this test is powerful</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Assignment for Next Tuesday

- **Test**—fill in the test case number
- **Risk**—be specific about the error you are trying to detect. For example (from 1,1,2), the risk is that the program might accept a “triangle” that has side 1 + side 2 = side 3.
- **Test case**—be specific about the inputs that you’ll feed the program. For example, 1,1,2
- **Why this test is powerful**—A test is powerful, compared to other tests, if it is more likely to expose a failure than they are. If you test for a specific type of failure (the risk), use a powerful test—one that is at least as likely to expose that failure as any other. To fill in this part of the answer, explain why you think the test you chose is powerful. It might be helpful to provide examples of similar tests that are less powerful than this one. If you think that this test is equivalent to many others (no more, no less powerful), say “Equivalent” and list 2 or 3 other tests that you think are equivalent to this one. This is an easy answer—but if I can easily spot a better test against the same risk, you won’t get credit for saying “equivalent”.
- **Expected result**—What should the program do? You might respond, “Error Message.”
Assignment

- Feel free to work in groups
- If you work with others, make sure to name them.
- If you work in a group, it is OK for the group to hand in one collective answer that you all co-sign. (Just provide your name, I don’t need your student number.) However, if you co-sign it, you have to be able to explain EVERY test case that you have on the page.
- I expect better work from a group than from individuals and I will mark accordingly.
- If you work in a group, I expect 5 tests from each of you. So, if there are two of you in the group, the assignment should contain 10 tests.
- Please don’t submit 100 tests. Pick 5 good ones per person. Prioritizing among possible tests is one of the important skills of good testers.
Grading Rubric for the Assignment

Risk:
- 4 - Clear statement of plausible risk, related to power (i.e. the stated risk relates well to the answer to “Why this test is powerful”)
- 3 - Clear statement of plausible risk, not related to power or Less clear or less plausible but related to power
- 2 - Less clear or less plausible, but related to power
- 1 - Generic risk statement. Vague, not related to power
- 0 - Implausible or generic to the point of no value. Not a risk.

Test Case:
- 1 - Values for the sides of the triangle are provided, consistent with the risk statement, and are either correct (they describe a valid triangle of the right type) or are incorrect in the “right” way
- 0 - Values for the sides of the triangle are not provided or are incorrect

Power
- 4 - States the principle(s) under which this is more powerful than others and gives a persuasive example
- 3 - Good comparison examples but weak explanation otherwise. A sufficiently strong explanation but without examples. Must indicate this is better than the others.
- 2 - The test can detect an error, indicate how or why and provide some comparison to other tests. Some indication that this is a good test.
- 1 - Shows that the test can detect the error identified in the risk.
- 0 - No clear linkage between the power discussion and the risk.

Expected Result:
- 1 - Expected result provided and correct.
- 0 - Expected result not provided or incorrect
Instructor’s notes

- Soon after the students handed in their first analysis of Myers’ triangle (the in-class exercise), I gave them feedback on their answers. The next slide, “Notes on the Exercise” is typical of my feedback.

- Along with those notes, I gave them Myers’ answer and worked through examples with them.
Notes on the Exercise

Several classes of issues were missed by most students. For example:

- Few students checked whether they were producing valid triangles. (1,2,3) and (1,2,4) cannot be the lengths of any triangle.

  Knowledge of the subject matter of the program under test will enable you to create test cases that are not directly suggested by the specification. If you lack that knowledge, you will miss key tests. (This knowledge is sometimes called “domain knowledge”, not to be confused with “domain testing.”)

- Few students checked non-numeric values, bad delimiters, or non-integers. Only half of you checked zero.

- No one tested at MaxInt.
Myers’ Answer

- Test case for a valid scalene triangle
- Test case for a valid equilateral triangle
- Three test cases for valid isosceles triangles (a=b, b=c, a=c)
- One, two or three sides has zero value (5 cases)
- One side has a negative
- Sum of two numbers equals the third (e.g. 1,2,3) is invalid b/c not a triangle (tried with 3 permutations a+b=c, a+c=b, b+c=a)
- Sum of two numbers is less than the third (e.g. 1,2,4) (3 permutations)
- Non-integer
- Wrong number of values (too many, too few)
Myers’ Triangle Exercise
(Reminder from class)

The program reads three integer values from a card. The three values are interpreted as representing the lengths of the sides of a triangle. The program prints a message that states whether the triangle is scalene, isosceles, or equilateral.

From Glen Myers, *The Art of Software Testing*

- Write a set of test cases that would adequately test this program.
- Please write your name on your answer so that we can hand it back to you. Hand it in when you are done.