Software Quality Assurance and Testing (SQAT)

Black-box testing (pt.2)

dr. Joost Schalken-Pinkster
Windesheim University of Applied Science
The Netherlands

joost@schalken.me

The contents of these course slides is (in great part) based on:
Neil Taylor (2019) *Course presentations for Software Quality Assurance and Testing (SQAT)*, Aberystwyth University.

Equivalence Classes

Back to the Triangle Example

A Shape module has a feature to return a value to indicate if a triangle is one of the types “Equilateral”, “Isosceles” or “Scalene”

The feature takes three values which represent the lengths of the three sides for the triangle.

- `public static Triangle makeTriangle(Point a, Point b, Point c)`

- What should we test?
- Should we use input or output values to identify equivalence classes?
- What other information do we need to know?

Example Steps

Step 1: Possible equivalence classes:

- Isosceles
- Equilateral
- Scalene
- Not a triangle

Is there another class?

Steps 2 and 3: Gives us one test case per class

Test Case	a	b	c	Expected Output
TC1	(0.0, 0.0)	(0.0, 5.0)	(5.0, 2.5)	Isosceles
TC2	(0.0, 0.0)	(5.0, 0.0)	(2.5, 4.33)	Equilateral
TC3	(0.0, 0.0)	(0.0, 5.0)	(5.5, 5.5)	Scalene
TC4	(0.0, 0.0)	(5.0, 5.0)	(10.0,10.0)	Not a triangle

Is there another class?

What happens if the values of a, b or c are ‘out of range’?

- For this example, we will assume that the values for each of x and y coordinates should be between 0 and 200.
- Gives a new class or ‘A, B or C out of range’

Test Case	a	b	c	Expected Output
TC1	(0.0, 0.0)	(0.0, 5.0)	(5.0, 2.5)	Isosceles
TC2	(0.0, 0.0)	(5.0, 0.0)	(2.5, 4.33)	Equilateral
TC3	(0.0, 0.0)	(0.0, 5.0)	(5.5, 5.5)	Scalene
TC4	(0.0, 0.0)	(5.0, 5.0)	(10.0,10.0)	Not a triangle
TC5	(999.9,0)	(0.0, 5.0)	(5.0, 2.5)	Out of range

Weak, Normal, Strong, Robust?

Jorgensen (*Software Testing: A Craftsman's Approach*) identifies different approaches

- How complete is the selection of test cases?

Weak vs **Strong**

- **Weak** – Assuming a single fault – choose one value and select test cases for each equivalence partition
- **Strong** – Assuming multiple faults - choose test cases for the combination of equivalence partitions

Normal vs **Robust**

- **Normal** – select values from the 'valid' ranges
- **Robust** – select values from the 'invalid' ranges

Back to the Triangle Example

The values selected so far are:

Test Case	a	b	c	Expected Output
TC1	(0.0, 0.0)	(0.0, 5.0)	(5.0, 2.5)	Isosceles
TC2	(0.0, 0.0)	(5.0, 0.0)	(2.5, 4.33)	Equilateral
TC3	(0.0, 0.0)	(0.0, 5.0)	(5.5, 5.5)	Scalene
TC4	(0.0, 0.0)	(5.0, 5.0)	(10.0,10.0)	Not a triangle
TC5	(999.9,0.0)	(0.0, 5.0)	(5.0, 2.5)	Out of range

TC1 to TC4 represent Weak-Normal

Does TC5 represent Weak-Robust?

TC5 and Weak-Robust

TC5 just represents one test case for Weak-Robust

Think about the outputs –

- Could use A out of Range
- Would it be better to have A too low and A too high, instead?

If we use ‘too high’, then three test cases exist, with one input value changed to 0 each time

Test Case	a	B	c	Expected Output
TC5	(999.9,0.0)	(0.0, 5.0)	(5.0, 2.5)	A too high
TC6	(0.0,0.0)	(999.9,0)	(5.0, 2.5)	B too high
TC7	(0.0,0.0)	(0.0, 5.0)	(999.9,0)	C too high

To complete the set, you would also have three more test cases for ‘too low’, where the 0 is replaced by 201 (or another invalid value above 200)

Alternative Equivalence Classes

- The equivalence classes for the triangle are based on the expected outputs
- Jorgensen suggests that we can consider equivalence classes based on the inputs.
- Is there an advantage to this?

ab = ac = bc

ab = ac; ab ≠ bc

ab = bc; ab ≠ bc

bc = ac; ab ≠ bc

ab≠ac; ab≠ac; ab≠bc

ab ≤ ac + bc

ac ≤ ab + bc

cb ≤ ab + ac

Example: Interest Calculation

What test cases would you identify to test a method for calculating interest on a bank account?

- **The bank pays 0% interest on balances under 1000 RMB, 1% interest on balances between 1000 RMB and 5000 RMB, and 2% interest over 5000 RMB**

Reminder:

- Identify the equivalence classes
- Design test cases for valid equivalence classes
- Design tests cases for invalid equivalence classes

Example: Next Date

What equivalence partitions and test cases would you select for this method?

Why is this example harder?

Given a date between January 1st 1880 and December 31st 2099, the method returns the next date. For example, if the input date is 30 June 2012, it returns 1st July 2012.

Reminder:

- Identify the equivalence classes
- Design test cases for valid equivalence classes
- Design tests cases for invalid equivalence classes

Adding values to a sorted list

What test cases would you identify to test a method for adding an object to an ordered list of objects?

What is different about this example?

Reminder:

- Identify the equivalence classes
- Design test cases for valid equivalence classes
- Design tests cases for invalid equivalence classes

Pre-conditions

- Adding values to the sorted list means that we are not just thinking about the values we pass to the method
- We are also thinking about the state of the list after the previous data input into the list.
- We are thinking about the pre-conditions needed to get ready to run a test

Boundary Value Analysis

Boundary Value Analysis

Boundary Value Analysis focuses on the boundary of the inputs for the test cases.

The reason for focusing on the boundary value testing is that errors do occur near the extreme values of an input variable

- E.g. conditions (< instead of <=), counters

Boundary Value Analysis

For any input variable, create tests cases with values that are:

- At the minimum (min)
- Just above the minimum (min + 1)
- At a nominal value
- Just below the maximum (max – 1)
- At the maximum (max)

Using Boundary Value Analysis

Boundary Value Analysis works well when the variables are independent and represent bounded physical quantities

- E.g. good for temperatures, air speed, etc.
- Helps us to explore what happens when values are outside the range

Are the boundaries useful for dates?

- Date has three related values, but they are not independent
  - Day, Month and Year
- There are boundaries with such values, e.g. start and end of the months and years

Triangle Example

We will use the same Triangle example:

- Six inputs: (Xa, Ya), (Xb, Yb) and (Xc, Yc)
- 1 <= Xa <= 200 1 <= Ya <= 200
- 1 <= Xb <= 200 1 <= Yb <= 200
- 1 <= Xc <= 200 1 <= Yc <= 200

What boundary values could we select for Xa?

Xa = 1, 2, 100, 199, 200

What values could we select for Ya, Xb, Yb, Xc and Yc?

What other values should we consider?

Robust Testing

- As well as the values in the valid (normal) range, we can test values which are just outside the valid range

- What happens when a value moves outside the range? Will it continue to work? Will it generate an error or exception?

Example: Interest Calculation

What boundary value test cases would you identify to test a method for calculating interest on a bank account?

- **The bank pays 0% interest on balances under 1000 RMB, 1% interest on balances between 1000 RMB and 5000 RMB, and 2% interest over 5000 RMB**

Example: Next Date

What boundary value test cases would you select for this method?

Given a date between January 1st 1880 and December 31st 2099, the method returns the next date. For example, if the input date is 30 June 2012, it returns 1st July 2012.

Adding values to a sorted list

What boundary value test cases would you identify to test a method for adding an object to an ordered list of objects?

Summary

- What is black-box testing?
- Techniques to help specify test cases based on the specification
- The use of Equivalence Partitions
- The use of Boundary Value Analysis

Any Questions?