Heriot-Watt University

School of Mathematical and Computer Sciences

Software Design (F28SD2)

An Exercise in Dynamic & Static Analysis

Andrew Ireland

February 27, 2012

1 Introduction

The aim of this exercise is to give you a deeper understanding of the strengthens and limitations of coverage and complexity metrics. Here you will focus in particular on the **Condition/Decision** coverage metric as presented in lectures, along with **Cyclomatic Complexity**.

2 Programming task

Your first task is to implement a *triangle classification* program. The specification for the program is given below:

Input: A file called `sizes.dat` containing a single integer N, followed by N integer triples. All integers are separated by spaces.

Example (input):
```
3
3 4 5
6 1 6
5 1 2
```

Behaviour: Determine whether or not each of the N integer triples defines a valid triangle, *i.e.* where each integer denotes the size of one of the sides of a triangle.

If the input does define a triangle then classify it according to the length of its sides as *scalene* (no sides equal), *isosceles* (two sides equal), or *equilateral* (all sides equal).

In addition, if a triangle is defined then classify it according to the largest angle as *acute* (less than $90°$), *obtuse* (greater than $90°$), or *right* (exactly $90°$).

Output: Write to an output file, *i.e.* `results.dat`, each of the integer triples provided as input together with their classification. In the case where the input does not define a triangle then display on standard output the message `NOT A TRIANGLE`.

Example (output):
```
3 4 5   Scalene    Right
6 1 6   Isosceles  Acute
5 1 2   NOT A TRIANGLE
```

Your program **should be** implemented in Java, preferably as a single method. You may use predefined classes, *e.g.* `Math`.

3 Dynamic analysis task

Your second task involves selecting 3 decision points within your code, each of which contain at least 2 conditions. For each of your selected decision points, you are required to use the Condition/Decision metric to:

- specify test cases using the Condition/Decision coverage metric;
- generate test data for your specifications, giving expected results;
- execute your code using the generated test data, giving actual results;
- if your expected and actual test results differ, then explain the difference(s) and how you would eliminate them.

4 Static analysis task

Your final task is to produce a flow graph and compute the cyclomatic complexity of your triangle classifier program (method).

5 Deliverable

The source code for your program should be emailed to `a.ireland@hw.ac.uk` by **12noon on Thursday, March 29th 2012. Please include your surname in the subject header of the email.** For the same deadline, you should also submit a report via the coursework box outside EM 1.24. Your report should include:

D1: Your, i) Condition/Decision test case specification, ii) test data, iii) expected and iv) actual results.

D2: Your code, with your selected decision points highlighted clearly as well as the statements that are executed by your test cases.

D3: Your flow graph and cyclomatic complexity for your triangle classifier program (method).

This assignment counts for 10% of the final assessment of this course. **Note that this is an individual project which means that your solution MUST be your own work.**