

*FOR INTERNAL SCRUTINY (date of this version: 17/2/2016)*

UNIVERSITY OF EDINBURGH  
COLLEGE OF SCIENCE AND ENGINEERING  
SCHOOL OF INFORMATICS

**SOFTWARE TESTING**

**Tuesday 1<sup>st</sup> April 2014**

**00:00 to 00:00**

**INSTRUCTIONS TO CANDIDATES**

**Answer QUESTION 1 and ONE other question.**

**Question 1 is COMPULSORY.**

**All questions carry equal weight.**

**CALCULATORS MAY NOT BE USED IN THIS EXAMINATION**

Year 3 Courses

Convener: ITO-Will-Determine

External Examiners: ITO-Will-Determine

**THIS EXAMINATION WILL BE MARKED ANONYMOUSLY**

1. Consider the following method that returns the grade for a given score.

```
public string get_grade(int score)
{
    string grade = "";
    if (score >=0 && score <10) {
        grade = "E";
    }
    else if (score >=10 && score <20) {
        grade = "D";
    }
    else if (score >=20 && score <30) {
        grade = "C";
    }
    else if (score >=30 && score <40) {
        grade = "B";
    }
    else if (score >=40) {
        grade = "A";
    }
    return grade;
}
```

- (a) Draw the control flow graph for the `get_grade` method. Label each of the edges in the control flow graph with a number and each of the blocks with a letter. [4 marks]
- (b) Write tests with input and expected output for this method. Using the labels on the edges in the flow graph, write down the sequence of edges traversed by each of the tests. [4 marks]
- (c) Evaluate the fraction of edges covered by the tests (Covered/Total). If any edges are uncovered, write additional tests to cover them. [2 marks]
- (d) Suppose the developer has incorrectly written the condition with `||` instead of `&&` in the first `if` statement as `if score >=0 || score <10`. Will the tests you developed in the previous step reveal this fault? Explain your answer. [3 marks]
- (e) Which coverage criteria is both practical and rigorous for testing complex boolean expressions? For the expression `((x1 && y1) || z1)`, develop tests that achieve this coverage criteria. Assume `x1`, `y1`, `z1` are all input boolean variables. Each of your tests should assign values to these input boolean variables and evaluate the expression. [6 marks]

- (f) For the `find_maximum` method given below ( $n$  is number of elements in parameter `a[]`), write tests with input and expected output that achieves “All DU pairs” coverage for variables `c`, `max` and `index`. Write out the  $\langle D, U \rangle$  pairs using the line numbers shown below for the three variables and show coverage achieved. [6 marks]

```
1  int find_maximum(int a[], int n) {
2  int c, max, index;
3
4  max = a[0];
5  index = 0;
6
7  for (c = 1;
8         c < n;
9         c++) {
10     if (a[c] > max) {
11         index = c;
12         max = a[c];
13     }
14 }
15
16 return index;
17 }
```

2. You have to conduct functional testing of a dialog with three sets of radio buttons on it, representing travel needs within UK. The first set of radio buttons specifies a destination, by country within UK — England, Scotland, Wales, or Northern Ireland; the second notes the choice of an aisle or window seat; and the third affords the choice of coach, business, or economy class. When the user makes her selections and presses "OK" to close the dialog, the application will take on a different path of execution for each combination of given options in given states.
- (a) How many tests do you need to test all possible combinations of input choices in the dialog? [2 marks]
  - (b) Which functional testing technique will you apply to bring down the number of combinations and why? [5 marks]
  - (c) Apply this functional testing technique to the given problem and show the resulting test suite. [8 marks]
  - (d) Give an example of a combination of inputs that your test suite will miss and state why. (5)
  - (e) Given a specification: a function that takes as input an integer *inp1*, and returns half the value of *inp1* if *inp1* is even, and *inp1* otherwise. For this function specification, identify representative value classes for the input *inp1* with a brief explanation. [5 marks]

3. (a) For the class diagram in Figure 1, draw the use/include relation to represent potential interactions between classes to be verified through inter-class testing. [5 marks]

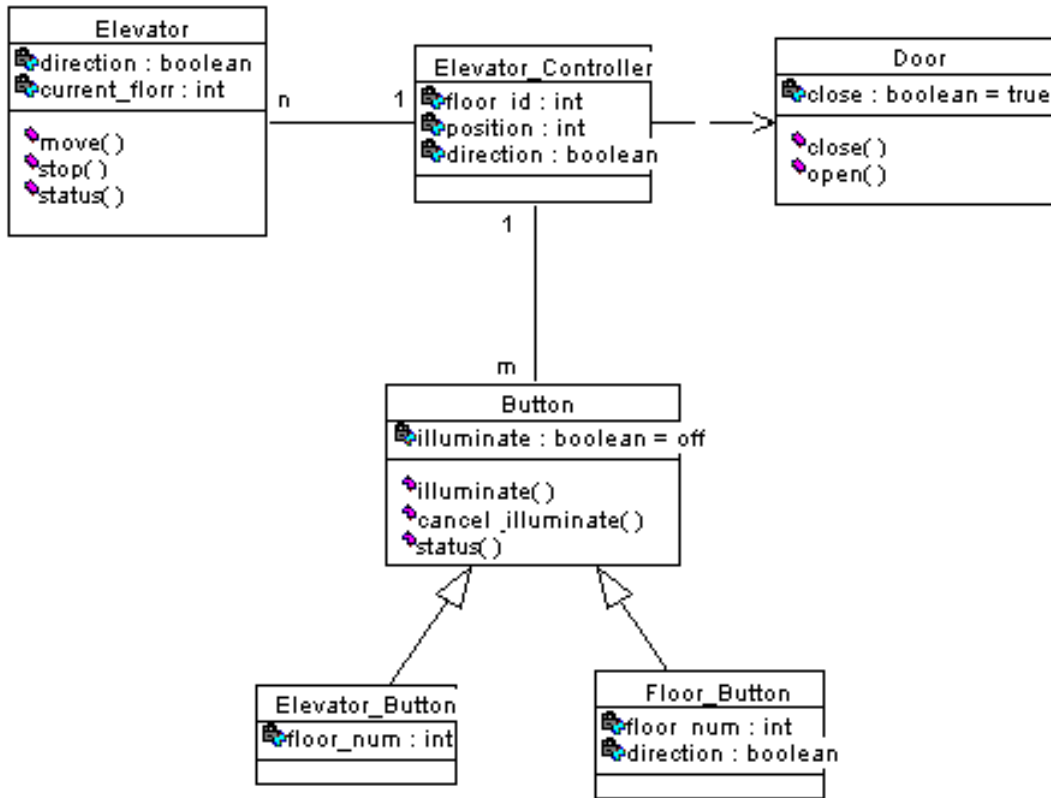


Figure 1: Class diagram for an elevator

- (b) Imagine a grandmother who will only approve you dating her grandchild if you are older than 25 and younger than 40 and either rich or really smart. If that grandmother was a programmer and could convince applicants to answer honestly, her program might look a bit like this:

```
ShallowGrandmother.java
1 import java.util.Scanner;
2
3 public class ShallowGrandmother {
4     public static void main( String[] args ) {
5         Scanner keyboard = new Scanner(System.in);
6         int age;
7         double income, smart;
8         boolean allowed;
9
10        System.out.print( "Enter your age: " );
11        age = keyboard.nextInt();
12
13        System.out.print( "Enter your yearly income: " );
14        income = keyboard.nextDouble();
15
16        System.out.print( "How smart are you, on a scale from 0.0 to 10.0? " );
17        smart = keyboard.nextDouble();
18
19        allowed = ( age > 25 && age < 40 && ( income > 50000 || smart >= 8.5 ) );
20
21        System.out.println( "Allowed to date my grandchild? " + allowed );
22    }
23 }
```

Inputs to the program are 'age', 'income', 'How smart are you?', and output is the boolean 'allowed'.

- i. For the `ShallowGrandmother` class above, derive 6 mutations that perform expression/operand modifications. Lines in the method above are shown with the line number. To show each mutation, simply state the line number being mutated and show the line with the modification. [6 marks]
  - ii. Derive tests that reveal these modifications (one at a time, assuming each mutation results in a new program). Please show expected and actual output for each of the tests. [6 marks]
  - iii. Do the tests revealing the mutations achieve condition coverage of the code, show your answer. If they do not achieve condition coverage, add tests for full condition coverage. [3 marks]
- (c) Assume you have 1000 regression tests and your resources only allow you to run 200 of them. How will you choose the regression tests to run with the limited resources? Discuss the different ways of choosing regression tests. [5 marks]