University of Glasgow

Tuesday 25 April 2017
2:00 – 3:30pm
(Duration: 1 hour 30 minutes)

DEGREES OF MSci, MEng, BEng, BSc, MA and MA (Social Sciences)

Java and Object Oriented Software Engineering 2

(Answer all questions)

This examination paper is worth a total of 60 marks.

The use of calculators is not permitted in this examination.

INSTRUCTIONS TO INVIGILATORS:

Please collect all exam question papers and exam answer scripts and retain for school to collect. Candidates must not remove exam question papers.

1. This question concerns the Java programming language. (15 marks total)

(a) Identify **five stylistic problems** with the following Java class. [5]

```
public class My_Class {
       java.util.ArrayList values = new java.util.ArrayList();

boolean doSomething(int value) {
  if (values.contains(value) == true) {
    return false;
  } else {
    values.add(value);
    return true;
  }
}
}
```

(b) Describe how you would modify this Java class to make it **immutable**. You may illustrate your answer with fragments of Java source code, but this is not essential. [5]

```
public class Class2 {
  private String[] values;

  public Class2 (String[] values) {
    this.values = values;
  }

  public String[] getValues() {
    return values;
  }

  public void setValues (String[] values) {
    this.values = values;
  }
}
```

(c) Explain, with a diagram, the role of the **UI delegate** in the Java Swing GUI library. [3]

(d) Explain why it is necessary to override the `hashCode()` method whenever you override the `equals()` method in a class. [2]

2. This question concerns Java class design. (15 marks total)

First, read the following description of an online catalogue system:

> The catalogue contains a range of sets of a building toy – let's call it "Brikz". Each building set record includes the following information: a unique set number (which is defined to be a number between 1 and 80 000), a name, a theme (e.g., "Ninjas", "Rescue Vehicles", "Knights"), the number of pieces included in the set, and a retail price.
>
> The catalogue stores the complete list of sets, and allows users to query and filter the sets in various ways: for example, they can get a list of all sets from a given theme, all sets that have at least a minimum number of pieces, or all sets in a given price range. Each of these queries returns a list of the sets from the catalogue that meet the criterion.

(a) Write a class definition for `BrikzSet`, incorporating all of the attributes mentioned above. Be sure to use appropriate data types and access modifiers. You should also define a public constructor for `BrikzSet`, which should initialise all instance fields. The constructor should also check that the parameters are sensible, and should throw an `IllegalArgumentException` if it is given bad input. You do not need to define any instance methods. [7]

(b) Define the `Catalogue` class and a write a `getSetsByTheme()` method that takes a theme name and returns all sets that match that theme. You can assume that the `BrikzSet` class has any necessary get/set methods. [6]

(c) The catalogue is to be modified so that the `getSetsByTheme` method should return a list of sets that is **sorted by price**. Describe how you would modify the classes defined above to deal with this situation. You may illustrate your answer with fragments of Java source code, but this is not essential. [2]

3. This question concerns software testing (15 marks total)

(a) Explain the concept of test criteria subsumption. [3]

(b) Assume the program function `ABS` below that returns the absolute value of the integer passed to the function as a parameter

```
public class Code1Testing
{
/*
 * INPUT: An integer.
 * OUTPUT: The absolute value if the input integer.
 */
1 int ABS(int x)
2 {
3  if(x < 0)
4    x = -x;
5  return x;
6 }

}
```

(i) Draw the control flow diagram for `ABS`. [3]

(ii) Why is the complete path testing of `ABS` theoretically possible but not practical? [3]

(c) Assume the predication a && b && c consisting of 3 conditions. Enumerate the test cases needed to satisfy the following:

(i) Multiple condition coverage. [3]

(ii) Modified condition/decision coverage. [3]

4. This question concerns software architectures (15 marks total)

(a) Why is the format of the URI of a RESTful API is very important? [2]

(b) Classify the HTTP common Methods based on their Idempotence and non-Idempotence. [5]

(c) Describe the different levels of Richardson Maturity Model used during RESTful API design. [8]