Software Workshop – Exercises

13 October 2015

Submissions must be made using Canvas, in the following format.

SUBMISSIONS NOT COMPLYING WITH THESE GUIDELINES WILL HAVE 2 MARKS DEDUCTED.

Uploaded file must be: studentid.zip
in the zip format. Rar or tar.gz will not be accepted.

Archive must contain: Module.java (answer to question 1), Student.java (answer to question 2), TutorGroup.java (answer to question 3) and Test.java (containing your test code), MyArrayList.java and TestArrayList.java (answer to question 4).

All submissions must be made by midnight on the Sunday the exercise is due. Submissions after this time WILL NOT BE MARKED and will receive ZERO.

Your answers to question 1, 2 and 3 should be based on the classes **`Student`** and **`Module`** which we developed in the lab on Tuesday. You can find these classes on the Canvas module page.

Don't forget that every time you write a method you should test it in Test.java. You will lose marks if you don't do this.

Question 1 [3 marks]

Write a method **`equals`** for the **`Module`** class that tests if two **`Module`** objects have the same module name. It should start like this

```
public boolean equals(Module module)
```

Write code in Test.java to check this is working.

Question 2 [3 marks]

Write a method for the **`Student`** class that checks if a student is on a particular module. It should start like this:

```
public boolean onModule(Module module)
```

Write code in Test.java to check this is working.

Question 3 [9 marks]

(a) Write a new class called **`TutorGroup`** to represent tutor groups. It should have two fields: **`tutor`** (the name of the tutor as a **`String`**) and **`students`** (an **`ArrayList`** of **`Student`**). Write a constructor method that initialises the tutor name and creates the **`ArrayList`**. **[3 marks]**

(b) Write "get" and "set" methods for the **`tutor`**. Write a **`toString`** method to display information about the tutor group. **[3 marks]**

(c) Write a method **`addStudent`** to add a new student to the tutor group. Write a method **`getStudents`** which returns the **`ArrayList`** of students in the group. **[3 marks]**

For all methods you write, write code in Test.java to check they are working.

Question 4 [9 marks]

For this question, we will pretend that the ArrayList class does not exist. The object of the exercise is to see how it might be implemented.

(a) Write a class **MyArrayList** that has as attributes an array of strings, the size of the array, and the size recording the number of stored strings. Write the constructor to initialise these for an initially empty collection – use 10 for the array size. **[3 marks]**

(b) Write an **add** method that takes a **String** as an argument**.** If the array is not yet full you add this at the next available position. If the array is full, then:

1. create a new array, 10 bigger than the array holding the strings
2. copy the strings from the current array into the new array
3. make the new array be the array for the object
4. add the new string at the next available position

[3 marks]

(c) Write a **get** method that takes an **int** as an argument**.** If this is non-negative, and less than the number of stored strings, it returns the string stored at that index. Otherwise, print a message saying the index is out of bounds and return **null**.

[3 marks]

To test your work, write a class **TestArrayList** that contains a main method. You should create a **MyArrayList** object, and add 50 different strings to it. Then try getting a few strings out, both in range and out of range, to test your methods.