



Australian
National
University

2015 Semester 1 - Final Examination – Paper B

Software Construction

(COMP2100/COMP2500/COMP6442)

Writing Period: 3 hour duration

Study Period: 15 minutes duration

Permitted Materials: One A4 page with notes on both sides.

*Note also the standard lab tools are available including: Java, eclipse, gedit, git, umbrello, dia, gcc, man, the calculator on the computer, ...
The Java api is available at <file:///usr/share/doc/openjdk-7-jre-headless/api/index.html>*

NO calculator permitted (physical electronic device).

Please Read The Following Instructions Carefully.

This exam will be marked out of 100 and consists of 4 questions. Questions are of unequal value. The value of each question is shown in square brackets. Questions that are partitioned into parts show the number of marks given to each part within square brackets.

Students should attempt all questions. Answers must be saved into the question's directory (Q1, Q2, Q3, Q4) using the file(s) described in the question statement. Marks may be lost for giving information that is irrelevant.

Network traffic may be monitored for inappropriate communications between students, or attempts to gain access to the Internet.

The marking scheme will put a high value on clarity so, as a general guide, it is better to give fewer answers in a clear manner than to outline a greater number of less clear answers.

Question 1 [40 marks]

Highest marks are gained by providing clear, concise, and short answers. Save your answers in the text file 'Q1answers.txt' in the directory Q1 (note this file is already created with places to add your answers, so just open up 'Q1answers.txt' and save your answers directly into it). This file can be edited using 'gedit'. Please make certain that this file is saved both as you progress through the exam and before the exam ends.

- i. [4 marks] Explain what a design pattern is. Why are design patterns useful?
- ii. [4 marks] Why are revision control systems, such as git, useful for developing software? What is the difference between a client-server revision control systems, such as svn, and distributed systems such as git or mercurial? Suppose you have a git repository set up in a local directory with a gitlab server set up as the remote fetch/push repository. What commands would you use to propagate change in a single file to the gitlab server?
- iii. [4 marks] Explain the difference between coupling and cohesion in terms of software design. Why is it advantageous to have low coupling and high cohesion in a design?
- iv. [4 marks] Explain how the "make" command determines which commands to execute within the "makefile".
- v. [4 marks] Explain what the "event dispatch thread" does in the Swing GUI framework.
- vi. [4 marks] Describe the JSON format. Illustrate your description with a JSON document that could be used for storing a person's name and shoe size.
- vii. [4 marks] Explain the Facade design pattern. When would using the Facade design pattern be useful (illustrate this with an example)?
- viii. [4 marks] What are the addition requirements for a binary tree to be a binary search tree? What is the maximum height of a binary tree that has 15 nodes? What is the minimum height of a binary tree with 15 nodes?
- ix. [4 marks] What are the advantages of using refactoring in eclipse in order to rename a method compared to that of using a simple text editor? Suppose you wish to add an external JAR to an eclipse project you are working on. How would you do this in eclipse?
- x. [4 marks] How do you set a variable in bash? Give an example of setting a variable. Describe what \${...}, \$(...), and \$[...] do in bash. (where ... is some other text)

Question 2 [30 marks]

The aim of this question is to complete a mini software development project. This involves developing a simple GUI application along with producing some documentation, testing your implementation, and demonstrating your use of some software engineering tools. The code and all other associated documentation you create must be included in the Q2 directory.

Suppose you are running a jumping castle and for safety reasons at most 10 people can be on the jumping castle at any one time.

People are allowed to go on for as long as they wish. In order to make certain there are not too many people on the castle you station an employee at the entrance/exit. This employee will count people as they enter and count them as they exit. If the jumping castle is full then a person wishing to enter will need to wait until someone first exits. Develop a GUI application



fir0002, wikipedia, GFDL 1.2

that will help the employee keep track of the number of people currently on the jumping castle. Your application must display the current number of people on the jumping castle, have an “Enter” button that increments the people count, an “Exit” button that decrements the people count, and the people count should not be permitted to go beyond 10 (also there shouldn't be a negative number of people on the castle!).

In addition to implementing an application that fulfils the requirements[10 marks] you should:

- set up and use git within the Q2 directory for version control (do at least 3 commits) [2 marks],
- complete a UML diagram (export this as a png in a file called UML.png in the Q2 directory) [4 marks], and
- undertake JUnit testing (at least 2 unit tests and 1 integration test) [4 marks].

You will also be evaluated on the overall quality of your submission [10 marks]. This includes aspects such as: design, testing coverage, code formatting, variable/field/method/class naming, documentation clarity and conciseness, code commenting, UML diagram formatting, git commit comments, etc ...

To get you started you have been provided with the “HelloWorld” gui application in the Q2 directory.

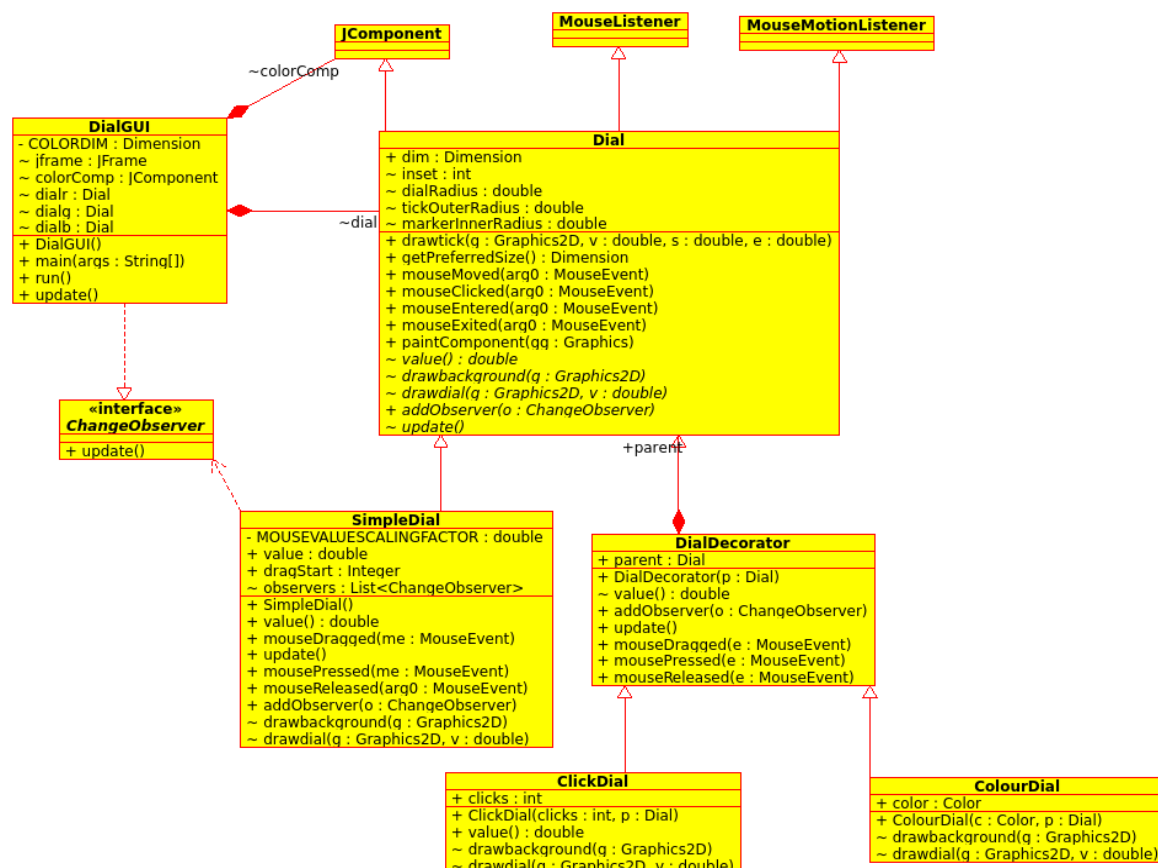
Question 3 [20 marks]

In the Q3 directory is a program that enables you to change the colour of a rectangle by “twisting” 3 dials. The dial value is changed by dragging the mouse left or right across the dial's icon. There is a dial which controls red, a dial which controls green, and one for controlling blue. These set the RGB values of the rectangle's colour. The implementation of this application is made up of 4 classes: DialGUI (with the main GUI code), Dial (the code for a simple dial), ColourDial (the code for a dial that can be coloured), and ClickDial (the code for a dial which “clicks” into an equally spaced fixed positions within the range).

Although the application works fine the design could be improved. There is considerable duplication in code. Using the decorator design pattern redesign and modify this implementation to address this issue. Remove unused classes from the directory. Change the GUI so all dials have a colour and they are all “click” dials. So within DialGUI to construct the dials you must use the code:

```
dialr = new ColourDial(Color.red, new ClickDial(10,new SimpleDial()));
dialg = new ColourDial(Color.green, new ClickDial(10,new SimpleDial()));
dialb = new ColourDial(Color.blue, new ClickDial(10, new SimpleDial()));
```

Hints: To answer this question one mainly needs to create some new classes and shift code around. To start off make “Dial” an abstract class then shift code out of this class into sub classes as needed. A possible UML diagram for a solution is given below:



Question 4 [10 marks]

The Q4 directory contains code for an implementation of a table. To test out this implementation “Demo.java” contains code that creates a table, adds elements to the table, and then looks up elements of the table. The code for adding elements has been completed, however, the code that looks up mappings in the table is yet to be implemented. The approach used to implement this table has been obscured by removing key comments and renaming key classes and methods.

Your task is to:

- determine the approach that is used for storing the table data (add your answer within a comment in the “lookup” method in Magneto.java), and
- implement the “lookup” method in Magneto.java.

You must only modify code in the “lookup” method in Magneto.java.
