UNIVERSITY OF LONDON

BSc EXAMINATION 2019

For Internal Students of
Royal Holloway

DO NOT TURN OVER UNTIL TOLD TO BEGIN

CS2800: Software Engineering
CS2800R: Software Engineering

PAPER FOR FIRST SIT/RESIT CANDIDATES

Time Allowed: **TWO hours**

Answer ALL questions

Calculators are NOT permitted

1. For each of the following pairs of related software engineering concepts you must:

   - *Describe* carefully each of the two concepts.
     A good description could be about three lines of text. Enough to explain the concept to a new student on CS2800.
   - Show that you understand how the two concepts are *connected*.
     For example, they have the same or contrasting goals, or they may be techniques that rely on each other to work.

   (a) *Revision Control System* and *TDD*. [10 marks]

   (b) *Coupling* and *Cohesion*. [10 marks]

   (c) *Code Release* and *Candidate Release Branch*. [10 marks]

   (d) *Code Smell* and *Refactoring*. [10 marks]

```
 1 public static void CS2800_19(int cow, int horse) {
 2   if (cow % 2 == 1) {
 3     System.out.println("Hello");
 4   } else {
 5     System.out.println("Goodbye");
 6   }
 7   switch (cow + horse) {
 8    case 0:
 9     System.out.println("No quadrupeds");
10     break;
11    case 1:
12     System.out.println("A lonely quadruped");
13     break;
14    default:
15     System.out.println("Buy a field");
16     break;
17   }
18 }
```

Figure 1: Code for use in answering Question 2

2. (a) What (in the course CS2800) are the two key properties of *good* code? [2 marks]

   (b) Describe the process of writing code using TDD. [6 marks]

   (c) Describe all paths testing. [2 marks]

   (d) How many tests are required for all paths testing of the code shown above Question 2? [2 marks]

   (e) Sketch the control flowgraph for the code shown above Question 2. [6 marks]

   (f) Describe how you are able to calculate the Cyclomatic complexity of the code shown above Question 2 from its control flowgraph. [2 marks]

3. (a) In a Source Code Control System (like SVN) what is a delta of a file. [2 marks]

   (b) Why does SVN use negative deltas? [2 marks]

   (c) You are working on a project managing buildings for a lettings agency.

   All the source code is kept in an SVN archive.

   You have been asked to fix some language issues (spelling mistakes in strings and poor translations into French)

   You have also been asked to add a new input screen that collects data on flammable cladding for all the buildings we manage.

   i. With respect to SVN and working in a team, what simple steps do you need to take to begin these tasks? [4 marks]

   ii. A colleague is tasked with helping you on the new screen feature. Describe a work flow that will allow you both to work independently on the same feature.
   You should describe the set-up, the daily procedures, and how you finally incorporate your feature into the code base. [8 marks]

   iii. You find that the new feature is long lived - it will take more than a month to complete. What must you do regularly to make sure that you will have no issues when you have finished with your feature? [2 marks]

   iv. It is urgent that one of your spelling corrections be incorporated into the code base for the next release before you have finished all of the work on that task.
   Describe in detail the process of doing a cherry pick three way merge to incorporate just some of your changes back into the main code base without finishing work on the task.
   In particular you must describe exactly how a three way merge works in SVN, the problems that may occur, and how you can solve them. [4 marks]

   (d) How can SVN be used to judge the *quality of the code submitted* by a particular software engineer? [4 marks]

4. (a) What is the State design pattern? Motivate the use of a State design pattern with a real world example. [3 marks]

   (b) Explain how we might refactor a Calculator (Java) object, that can be either infix or reverse polish, to introduce the State pattern. [3 marks]

   (c) The Singleton pattern is often considered an anti-pattern.

      i. Give two clear design problems associated with using singletons and describe how the basic implementation of Singleton is modified to overcome the issue in Java. [4 marks]

      ii. Describe a creational pattern that can easily be used to replace singleton and carefully explain how it overcomes the issues that you have described. [4 marks]

END