UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

Informatics 2C – Software Engineering

Degree Examination

Date: ?? December 2010

Time: ??? - ??? (1 hour)

INSTRUCTIONS TO CANDIDATES

1. Check that the question paper contains pages 1–6. If it does not, inform the invigilator.
2. You should attempt as many questions as possible.
3. The questions in this paper may vary widely in difficulty. During your first pass through the paper, you are advised not to dwell on questions to which the answer is not readily apparent.
4. Write your answer to each question in the box or table provided. If you wish to write more in answer to a question, continue on the blank page opposite, indicating that you have done so.
5. Please write *legible* and *concise* answers.
6. The marks allocated to each part of a question are indicated in the margin. There are 100 marks in total.

Question 1

NB all marks in this paper are allocated in even numbers, to make the total a convenient 100 without making the paper too long!

1. In a use case diagram, what is an actor? [4 marks]

   a kind of user of the system (2 marks), or a kind of other external entity such as another system (2 marks)

2. And what is a use case? [4 marks]

   a task (or coherent work unit) that the system can carry out (4 marks)

3. Consider an application that allows a user to edit a Java program, compile and run it. It also provides check in and check out access to an existing version control system via the internet. Draw a use case diagram for this system. [12 marks]

   4 marks for actors (User, VCS), 4 for use cases (Edit, Compile, Run, Checkin, Checkout), 4 for associations (from User to all use cases, from VCS to Checkin and Checkout)

Question 2

1. Given an example of a version control system that uses a Copy-Modify-Merge model. [4 marks]

   CVS, or SVN

2. Given an example of a version control system that uses a Lock-Modify-Unlock model. [4 marks]

   RCS

3. List three advantages that distributed version control systems have, compared to the systems mentioned above. [12 marks]

   Any 3 (4 marks for each) of:

   (a) reduces dependence on single physical node

   (b) allows people to work (including check in, with log comments etc.) while disconnected

   (c) much faster VC operations

   (d) much better support for branching

   (e) makes it easier to republish versions of software

Question 3

Consider the following code fragment.

```
public Object visit(ParameterDeclarationAS host, Object data) {
  FunctionParameter functionParameter = q.
  createFunctionParameter();
  this.putAst(host, functionParameter);
  functionParameter.SetName(host.getName());
  return functionParameter;
}
```

1. Annotate the code fragment to show three ways in which the readability of the code could be improved. [6 marks]

   Any 3 of: q is an unhelpful name (2 marks); bad indentation on the next line (2 marks); bad capitalisation of SetName (2 marks); parameter `data` is never used (2 marks).

2. Either below or on the blank page facing this one, draw a sequence diagram demonstrating what happens when an actor sends message `visit(h,d)` to an object `o`, invoking the code shown above. Show as much detail as you can. [14 marks]

   They need lifelines for: the actor; the object o; q; functionParameter:FunctionParameter (which could, but need not, be shown displaced down the page at the call to createFunctionParameter); host:ParameterDeclarationAS. 2 marks for any of these, 4 if all are correct. They need messages visit;createFunctionParameter;putAst (should be shown, since we asked for detail);getName;setName, with correct parameters. 2/4/6 marks for getting any, at least 3, all of these correct, rsp. Remaining 4 marks for any two of: return arrows, activation bars, correct notation for use of return value of getName.

Question 4

1. Explain, as generally as you can, when a module A is said to *depend* on a module B. [4 marks]

When a change to B may necessitate a change to A (2 for the general idea, 2 for getting it the right way round!))

2. Give two specific examples of ways in which a Java class A may depend on a Java class B. [4 marks]

E.g., when A extends B; when an object of class A must send a message to an object of class B. 2 marks for each.

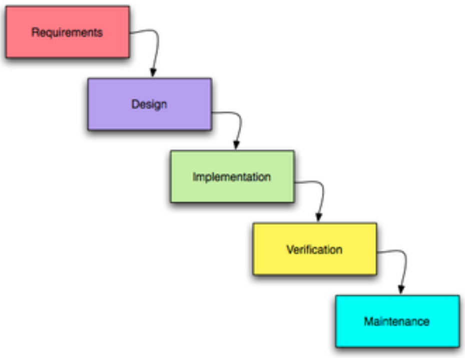
3. Draw a fragment of UML to show that class A depends on class B, without showing what kind of dependency it is. Show also that A has a private attribute a of class String, and a public operation toString that takes no arguments and returns a String. [12 marks]

2 marks for both class rectangles with names A and B. 4 marks for dependency arrow from A to B. 2 marks for attribute, 2 marks for operation, 2 for showing compartments.

Question 5

1. Draw a diagram to illustrate the waterfall model of software development. [8 marks]

   2 marks for general shape, 2 for arrows, 4 for correct activities. (This image, from Wikipedia, is what's in the slides, but I'd accept e.g. Deployment instead of Maintenance.)

2. What is wrong with the waterfall model? (You are not expected to discuss alternative models.) [4 marks]

   It's unrealistic (2 marks). In practice it is impossible, e.g., to complete design before doing any implementation. The model gives no guidance about how to handle the inevitable iteration (2 marks).

CONTINUED OVERLEAF

3. You are the project manager with overall responsibility for the development of the software in a new autonomous car (that is, a car which drives itself, given only the coordinates of the destination). *Briefly* describe the considerations that will guide your choice of development methodology, in the light of the characteristics of this project. You are not expected to choose a named methodology, or to give an exhaustive description. [8 marks]

The system is safety critical (2 marks) so more than acceptability to the customer must be shown: regulation will require a written safety case and the methodology must permit one to be developed (2 marks). The software will have to interact with hardware, not only with people (2 marks) which will limit requirements flexibility and probably make frequent releases impractical (2 marks). (Other answers possible, give credit for anything reasonable.)