UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

INFR11038 SOFTWARE ARCHITECTURE, PROCESS AND MANAGEMENT (LEVEL 11)

Wednesday 7th May 2014

14:30 to 16:30

INSTRUCTIONS TO CANDIDATES

Answer any TWO questions.

All questions carry equal weight.

CALCULATORS MAY NOT BE USED IN THIS EXAMINATION

Year 4 Courses

Convener: I. Stark
External Examiners: A. Cohn, T. Field

THIS EXAMINATION WILL BE MARKED ANONYMOUSLY

1. (a) Explain the related software metrics of *Coupling* and *Cohesion.* [*4 marks*]

   (b) Your manager has asked for all components to be highly-cohesive. Suggest a way in which you could easily, but artificially, make all components highly-cohesive and why you should not do this. What would be the sign that you had done this? [*3 marks*]

   (c) Your manager now wishes to measure the coupling in your system.

      i. You and your co-worker both use automated tools to measure this but report somewhat different results. Suggest why this might be. [*2 marks*]

      ii. As the project proceeds, why might you keep both measurements? [*2 marks*]

   (d) Choose an architectural pattern, from the course or your own reading. Describe the pattern and explain how it would help with cohesion and coupling. [*4 marks*]

   (e) On the course the concept of a “patterns compiler” for *design patterns* was discussed. It was reasoned that such a compiler does not exist because particular design patterns often exist because expressing the solution in a way that is usefully parameterised is a challenge. Instead, a colleague has suggested attempting to build a pattern *recogniser*. Such a tool would *recognise* the application of a given design pattern via the structure of the code, rather than any naming conventions. Without referring to the feasibility of this project, explain what you think of this idea. [*4 marks*]

   (f) It has been suggested that *“that which does not get measured, does not get done.”* State and explain whether you agree with this. [*6 marks*]

2. (a) Your software development company has renewed support for a legacy application and has therefore agreed to re-commence its maintenance. The legacy application has not been developed in several years but now requires updating. You are going to need to refactor the code before you can begin to make the required changes. Explain what refactoring is, making sure you say what the purpose is. [*3 marks*]

(b) One of your co-workers suggests throwing out the legacy application and starting again from scratch. Explain two reasons why you are nervous to do this. [*2 marks*]

(c) Your manager wishes to know how long it will take to improve the legacy source code.

i. Estimates are always difficult to get correct, however, state and explain two reasons why you are reluctant to give an estimate *specifically* in this case. [*4 marks*]

ii. A colleague suggests that a better question is to make a deadline and ask how much can be done in this time frame. State and explain your opinion of this suggestion. [*2 marks*]

(d) Explain an item of deferred work that you would *not* regard as "Technical Debt" and why. [*2 marks*]

(e) During a meeting for a project you are working on, a colleague makes the following statement *"Design patterns are too often useless since they solve problems one would not have without the rigid object-oriented framework and with immutable state anyway."* How would you respond? You may address all or some of your response to the individual, to the team or to the team leader. [*4 marks*]

(f) Model based estimation techniques such as COCOMO use data from previous projects in order to infer the relationships between properties of the project and the property that we wish to estimate, such as the eventual size of the project or effort required to complete the project. Discuss whether you consider this a worthwhile endeavour. [*8 marks*]

3. (a) Explain why adding more people to a late software development project may make it later. [*2 marks*]

(b) One of the Joel Test questions asks “Do you fix bugs before writing new code?” Do you agree that this is essential? It may help your answer to clarify what you believe this sentence means. [*4 marks*]

(c) You are on a project and find that there are still many issues left in the bug tracker. A colleague suggests “Sacrificing One Person” each week to be responsible for the elimination of issues. Each week the sacrificed person should be a different team member. In order to ensure that the sacrificed team member performs over their week, the number of bugs in the bug-tracker is measured. Explain a risk involved here specific to the method of measurement. [*4 marks*]

(d) Explain your thoughts on the proposition that “*Design Patterns* are best used as a good way for novice programmers to learn from experts.” [*3 marks*]

(e) Define the distinction between *agile* and *heavy-weight* methodologies and discuss agile methodologies in relation to the rise in cloud-computing, focussing on advantages and disadvantages of adopting an agile methodology, specifically for cloud-based applications. [*6 marks*]

(f) Describe a project most suited to a heavy-weight methodology. [*6 marks*]