EXAM FORMAT

Candidates should answer ALL questions. The exam paper will contain FOURTEEN questions: 10 short answer and 4 long answer. Answers to short answer questions are to be written on the exam paper. Answers to long answer questions are to be written in the examination answer books.

There are 60 marks in total for the paper: short answer questions are worth 2 marks each (as in the sample test) and long answer questions 10 marks each. As a guide, aim to spend 2 minutes per question mark (4 minutes per short answer, 20 minutes per long answer).

Candidates will have 2 HOURS 10 MINUTES writing time for the exam. The exam is worth 60% of the total marks for SED.

MATERIAL TO BE EXAMINED & HOW TO STUDY FOR THE EXAM

The exam will be based on all material studied in SED lectures, practicals and reading lists, covering requirements analysis, software design and testing software.

When studying for the exam, you should review the objectives and key points of each lecture. Examination and test questions are designed to test whether you have achieved the objectives and understood the key points. Also, review the core reading listed for each lecture as well as the lecture notes and browse the suggested further reading.

Attempt the sample exam questions under test conditions. Then consult your text book(s), lecture notes and other references to evaluate your answer attempts. Think about how you could improve your answer. You can also study by inventing and answering your own exam questions.

COMMENTS ON THE MARKING

Your answers should demonstrate:
breadth - knowledge of the main techniques and ideas in the question topic. Note "main" and don't try to learn verbatim all the lists in the lecture notes.
depth - show that your knowledge of techniques and ideas goes beyond their names, for example you understand the trade-offs between different methods
related topics - where relevant you should include ideas from different parts of the course such as case studies, or programming experience.
well structured answers - use your reading time (and more) to plan how you will answer questions.

Use point form and subheadings to make long answers easy to follow.

Tim French
SED Co-ordinator
May 2010
THIS PAPER CONSISTS OF 2 SECTIONS, 12 QUESTIONS
CANDIDATES SHOULD ANSWER ALL QUESTIONS
THERE ARE 60 MARKS IN TOTAL FOR THE PAPER

SECTION A
Answers for Section A are to be written in the spaces provided on the exam paper.

QUESTION 1
[2 marks]
In software models, what is an actor?

QUESTION 2
[2 marks]
Draw a UML use case diagram to represent the following WebStore system. The names of use cases and of actors are highlighted in bold.

R1. A customer may place-an-order with the retailer
R2. A customer may only place-an-order for items which are in stock
R3. Only the retailer may check-availability of stock
R4. In order to confirm-an-order with the customer, the retailer checks the stock
QUESTION 3
For the WebStore system of question 2, give an example of two design goals which conflict with one another. Explain why they conflict and suggest a way of resolving this.

QUESTION 4
What do you understand by the terms cohesion and coupling in software design?

QUESTION 5
Draw a UML class diagram to capture the following situation, “every book has chapters but only some have pictures”.
QUESTION 6  [2 marks]
Give an example of a specific problem for which a pipe and filter software architecture would be suitable.

QUESTION 7  [2 marks]
How does the Observer design pattern maintain consistency across the states of one Publisher and many Subscribers.

QUESTION 8  [2 marks]
Consider a finite state machine specification containing the transition (I, i1/o1, J) which has source state, I, destination, J and label i1/o1?
In Chow’s WP conformance test method, which of the following implementation transitions has a transfer error for transition (I, i1/o1, J)? Explain your choice briefly.

a. (I, i2/o2, J)
b. (I, i1/o2, J)
c. (I, i2/o1, K)
d. (I, i1/o1, K)
QUESTION 9
Why are code inspections so effective at finding faults in software?

QUESTION 10
Distinguish between statement, all-paths and branch coverage in structure-based, white box testing of program code.
SECTION B
Answers for Section B are to be written in the examination answer booklets.

QUESTION 11

In many software architectures, such as the 4-tier architecture shown below, the storage of persistent objects is handled by a dedicated layer. In your opinion, which design goals (performance, dependability, cost, maintenance, end user) have lead to this decision? Give reasons for your answers.

[10 marks]

---

QUESTION 12

Use the white box path testing method to generate test cases which cover all-paths in the following Java method. As well as listing test cases, show how you generated the tests.

```java
public static int entryPrice(int age, int day) {
    int price;
    if ((age>65) || (age<11))
        price=8;
    else
        price=12;
    if ((day==2) || (day==4))
        price=price-(price/4);
    return price;
}
```

[10 marks]
QUESTION 13

The Department of Computer Science and Software Engineering plans to commission an online examination system, OnLine-Exams, to enable students to sit end of semester examinations (like this one) in the Department's computer laboratories. You can assume the same style of examination papers as currently used. An external secure server will be used to provide a bank of questions for examinations. The OnLine-Exams system is required to deliver questions to authorised students at their laboratory terminals, and, at the end of the allotted time for the examination, to store the students' answers securely for marking.

A significant issue for the OnLine-Exams system is that large quantities of data (e.g. student answers) must be transferred between the students' computer and the secure answer server, in particular at the end of the examination. Discuss two different ways of solving this problem. Use rationale concepts (issue, proposals, criteria, arguments, and resolution) to present your answer.

QUESTION 14

Sketch a software architecture for the OnLine-Exams system described in Question 13 above. Use a diagram. Give a short description of the purpose of each component in the system architecture and describe the communication which occurs between components. Explain briefly how and why you chose this decomposition.

END OF EXAMINATION PAPER