THIS PAPER IS NOT TO BE REMOVED FROM THE EXAMINATION HALLS

UNIVERSITY OF LONDON **CO3314 ZA**

BSc Examination

COMPUTING AND INFORMATION SYSTEMS AND CREATIVE COMPUTING

Software Engineering Management

Thursday 9 May 2013 : 2.30 – 4.45 pm

Duration: 2 hours 15 minutes

There are five questions in this paper. Candidates should answer **THREE** questions. All questions carry equal marks and full marks can be obtained for complete answers to **THREE** questions.

Only your first three answers, in the order that they appear in your answer book, will be marked.

Questions involving a description or explanation should, wherever possible, be accompanied by an appropriate example.

A hand held calculator may be used when answering questions on this paper but it must not be pre-programmed or able to display graphics, text or algebraic equations. The make and type of machine must be stated clearly on the front cover of the answer book.

NOTE:

AN OUTLINE SPECIFICATION FOR A HYPOTHETICAL SYSTEM IS USED IN ALL OF THE QUESTIONS IN THIS PAPER AND IS APPENDED TO THIS PAPER AS PAGE 4.

READ IT CAREFULLY BEFORE ATTEMPTING TO ANSWER ANY QUESTIONS.

Question 1

Measurement, Cost estimation

Read the task specification of the software for monitoring the operations of an onboard train system, given in the appendix, and answer all parts of this question.

a. Which are the types of measurement used in the software engineering process. **6 marks**

b. (i) Estimate approximately the total effort in thousands lines of code (KLOC) required to implement each function of the software onboard train system. **15 marks**

(ii) Explain briefly when a software project will be considered well managed. **4 marks**

Question 2

Software life-cycle

Read the task specification of the software for monitoring the operations of an onboard train system, given in the appendix, and answer all parts of this question.

a. Develop a requirements specification for the onboard train software, and focus especially on the following:

(i) Identify the functions that have to show data on the overview status display. **12 marks**

(ii) Identify the functions that have to show data on the detailed status monitor. **6 marks**

b. Give the main stages of the ideal software life-cycle. **7 marks**

Question 3

Dependability measurement and Reliability Modelling

Read the task specification of the software for monitoring the operations of an onboard train system, given in the appendix, and answer all parts of this question.

a. Reason briefly about the reliability of the onboard train software and Find some unexpected events that may occur and prevent delivery of required options. **12 marks**

b. Explain how we compute the expected number of faults *m* discovered by time *u* according to the Goel-Okumoto NHPP model, assuming that the mean number of faults is *n*, and *z* is the rate of fault activation. **6 marks**

c. What could be the problems behind incorrect assumptions made in reliability models, such as the lack of independence of failures for example. **7 marks**

Question 4

Software inspections

Read the task specification of the software for monitoring the operations of an onboard train system, given in the appendix, and answer all parts of this question.

a. Briefly describe the procedure for inspections of software design.
9 marks

b. Define some measures that can be used for inspection of the functions of the onboard train software system according to the methodology of Fagan.
16 marks

Question 5

Safety-critical software

Read the task specification of the software for monitoring the operations of an onboard train system, given in the appendix, and answer all parts of this question.

a. Identify which are the functions in the onboard train software that have a high probability of residual fault and may lead to critical failures during operation. **16 marks**

b. Discuss briefly how fault removal can be performed. **9 marks**

APPENDIX: TASK SPECIFICATION

The task is to design software for monitoring the operations of an onboard train system. This software program should collect data for the functioning of the onboard subsystems, and show these data on the display. The subsystems data should include: the locomotive status, the energy subsystem status and the train schedule. The only person who is allowed to use the system is the train driver. He is allowed to overview the status of the subsystems on the display and request reports, all of his allowances should be implemented in the software as options. The display should be divided into two windows: the first window is called overview status display, and the second window is called detailed status monitor.

The first option is to show on the overview status display data such as: 1) the locomotive data, including the current speed, the maximum allowed speed and the throttle settings; 2) the engine energy subsystem data, including the alternator volts, the oil temperature, the water pressure and the drawbar force; and 3) the schedule data, including the start station, the destination station, and the arrival time. The software should have a common database where all these data are stored, so that the control module can take them at each time instant and visualize them on the display in order to allow the train driver to monitor the train movement.

The second option is to allow the driver to request a report for the current status of a particular subsystem, in which case the software should provide the corresponding data. This is the control module which is responsible for taking the data from the database and visualizing them on the detailed status monitor. This detailed status monitor should prompt the driver which system to inspect.

The third option given to the driver is to request a report for all subsystems in case of a failure. Servicing a failure by the software should include providing access to each data item in each subsystem upon receiving a request from the driver. This requires the system to save into an external file a copy of all current data in case of a failure, while keeping the movement without stopping the train. Each subsystem should turn on a red light button on the overview display to prompt the driver to look at the detailed status monitor.

The fourth option offered to the driver is to print a trip report after arrival at the destination station. The software should then compute the trip duration, the amount of fuel used, the average speed, the number of failures. There should be also provided some data for the functioning of the subsystems, like the maximum engine temperature, the average alternator volts etc. The software should allow the driver to print the report and save it the database. The database should only be accessible by the driver with a password.

END OF PAPER