UNIVERSITY OF LONDON

CM2010

BSc EXAMINATION

COMPUTER SCIENCE

Software Design and Development

Release date: Thursday 9 September 2021 at 12:00 midday British Summer Time

Submission date: Friday 10 September 2021 by 12:00 midday British Summer Time

Time allowed: 24 hours to submit

INSTRUCTIONS TO CANDIDATES:

Section A of this assessment paper consists of a set of **10** Multiple Choice Questions (MCQs) which you will take separately from this paper. You should attempt to answer **ALL** the questions in Section A. The maximum mark for Section A is **40**.

Section A will be completed online on the VLE. You may choose to access the MCQs at any time following the release of the paper, but once you have accessed the MCQs you must submit your answers before the deadline or within **4 hours** of starting whichever occurs first.

Section B of this assessment paper is an online assessment to be completed within the same 24-hour window as Section A. We anticipate that approximately **1 hour** is sufficient for you to answer Section B. Candidates must answer **TWO** out of the THREE questions in Section B. The maximum mark for Section B is **60**.

Calculators are not permitted in this examination. Credit will only be given if all workings are shown.

You should complete **Section B** of this paper and submit your answers as **one document**, if possible, in Microsoft Word or a PDF to the appropriate area on the VLE. You are permitted to upload 30 documents. However, we advise you to upload as few documents as possible. Each file uploaded must be accompanied by a coversheet containing your **candidate number**. In addition, your answers must have your candidate number written clearly at the top of the page before you upload your work. Do not write your name anywhere in your answers.

SECTION B

Candidates should answer any **TWO** questions from Section B.

Question 1

In this question, you will be considering the development of a control panel for an offshore wind turbine installation. An offshore wind turbine is an electricity generation system that floats out to sea on a platform somewhat similar to an oil rig platform. The control panel allows the user to monitor the status of the turbine and to control the flow of electricity from the platform to the grid.

(a) Draw a flowchart or other diagram showing the lifecycle of the software system. The flowchart should show each of the following items and when they occur. Also consider if certain cycles should be repeated, and indicate this on the diagram.

i. Unit testing

ii. Requirements testing

iii. Usability testing

iv. Accessibility testing

v. Security testing

[10]

(b) Write out TWO requirements for the monitoring part of the control interface using the EARS syntax. [4]

(c) Explain the difference between usability testing and accessibility testing. [2]

(d) Consider the system usability scale (SUS) and Nielsen's 10 usability heuristics.

i. State TWO similarities and TWO differences between these usability evaluation methods. [4]

ii. Users for the control interface will sometimes be wearing heavy waterproof overalls. Do you think these two methods for evaluating usability will be effective for these users? Justify your answer [4]

(e) You have been asked to develop a strategy to evaluate the security of the code base for the control interface software. Describe THREE actions you would take in carrying out this evaluation. [6]

Question 2

This question concerns the interactions between modules. You are developing some statistical functions for a new graph plotting and statistical analysis library. You are working on the functions in the following list:

i. A sorting function called sort

ii. A median function called median

iii. a mode function called mode which calculates the mode

iv. A mean function called mean

v. A standard deviation function called stddev

(a) Define module coupling and module cohesion. [2]

(b) For each function, write out a function signature showing the inputs and outputs. The programming language is not important but you must specify the inputs and outputs somehow. [5]

(c) For each function, write a list of which other functions it interacts with to compute its output from its inputs. What form of module coupling is going on here? [5]

(d) Describe an example from your own experience of programming where you had to make a decision about module **coupling**. [2]

i. State how you could have solved that problem using minimal coupling. [2]

ii. State how you could have solved that problem using stronger coupling. [2]

(e) Now you are working on the graph plotting part of the software. This consists of the following modules:

i. Rendering which takes the graph data and renders it to a bitmap image.

ii. Data processing layer which takes the raw data and processes it into a form for the graphical layer.

iii. Data acquisition layer, which reads in the raw data for the data processing layer.

Do you think these modules appear to have high cohesion? Justify your answer in terms of desirable and undesirable forms of module cohesion. [6]

(f) Describe an example from your own experience of programming where you had to make a decision about module **cohesion**. [2]

  i. State how you could have solved that problem using minimal coupling. [2]

  ii. State how you could have solved that problem using stronger coupling. [2]

Question 3

This question is concerned with version handling using git.

(a) You are introducing some colleagues to git. Describe THREE features of the git system and why they are important. Use diagrams as necessary. [9]

(b) One of your colleagues thinks it would be much easier if everybody just worked on their own copies of the codebase and you met up weekly to merge the code and resolve conflicts manually. Write an explanation of why git supports this workflow. [4]

(c) Name and state the purpose of THREE git commands. [3]

(d) Run these THREE commands on your system and provide a screenshot of your terminal and some of the output of the command. [3]

(e) Visit the github website and locate a repository with plenty of activity. Do not choose react or tensorflow - try and find something that other students might not find. Using git log and other appropriate techniques, investigate the repository in order to answer the following questions:

i. What is the address of the repository on github? [1]

ii. Describe the branching and merging activity in the repository. For example, do branches tend to get merged? [4]

iii. What kind of contribution patterns are there? Are there lots of small contributions from lots of people or just a few main developers? Make it clear which specific observations you have made. [4]

iv. If you wanted to contribute to this repository, how would you go about identifying things the developers are requesting help with? [2]

END OF PAPER

.