

NATIONAL EXAMS - MAY 2019

17-Comp-A6, Software Engineering

3 Hours Duration

Notes:

1. If doubt exists as to the interpretation of a question, the candidate is urged to submit with the answer paper a clear statement of any assumptions made.
2. No calculator is permitted. This is a CLOSED book exam.
3. Answer any five of the eight questions.
4. Any five questions constitute a complete paper. Only the first five questions as they appear in your answer book will be marked.
5. All questions have equal weight.

Marking Scheme

1. (a) 15 marks; (b) 5 marks.
2. (a) 10 marks; (b) 10 marks.
3. 20 marks.
4. (a) 10 marks; (b) 10 marks.
5. (a) 10 marks; (b) 10 marks.
6. (a) 5 marks; (b) 5 marks; (c) 10 marks.
7. 20 marks.
8. (a) 4 marks; (b) 4 marks; (c) 4 marks; (d) 4 marks; (d) 4 marks.

Total mark out of 100.

Question 1. *The Software Development Process.*

- (a) List the stages of the software development life cycle and briefly describe each stage.
- (b) Contrast and compare these stages to the stages of purchasing and owning a piece of equipment, such as a car or a refrigerator. In particular, contrast the life cycle costs of owning the equipment with that of software. How are they similar and how are they different? Justify your answer.

Question 2. *Software Design and Object-Oriented Design.*

- (a) Discuss the differences between object-oriented and function-oriented design.
- (b) A software system is to be developed for a microprocessor-based *Insulin Delivery System (IDS)* in a hospital. The system works by using a micro-sensor embedded in the patient to measure blood parameters that are proportional the sugar level. These parameters are then sent to a pump controller. This controller computes the sugar level, judges how much insulin is required and sends signals to a miniaturized pump to deliver the insulin via a permanently attached needle.

Using an object-oriented approach, derive a design for the IDS described above. Make reasonable assumption and clearly state them.

Question 3. *Function-Oriented Design.*

Using a function-oriented approach, derive a high-level design for the system outlined below. Make reasonable assumptions about the system and state them clearly.

A software system is to be developed to hold details of newspaper and magazine deliveries in a small town. In addition to recording which households take newspapers and magazines, the system includes billing details and details of customer vacations, when newspapers are not delivered. For each delivery person, the system prints a daily list of which newspapers and magazines are to be delivered to which households. The system is also able to produce summary information showing how many copies of each newspaper were sold each day in the week. Bills for each customer can be printed at the end of the month and delivered with the first delivery each month.

Question 4. *Software Reuse and Portability.*

- (a) In an object-oriented programming language, *information-hiding* and *inheritance* can be used to adapt software components for reuse. Describe information-hiding and inheritance, and the pros and cons of using each to support code reuse.
- (b) You have been assigned the task of implementing a calendar and clock which gives time and date information. This has to operate on a range of computers from 8-bit micros to 64-bit special purpose processors. Design and implement an abstract data type for representing the calendar and clock that can be readily ported from machine to machine.

Question 5. *Software Safety.*

Consider the Insulin Delivery System (IDS) described in **Question 2** above. A low blood sugar level, even for a short term, is a serious condition that can result in brain damage and ultimately death. A high blood sugar level, for a long term, can result in eye damage, kidney damage and heart problems.

- (a) Conduct a software hazard analysis of the IDS described above. What are the hazards that can occur in the system? What is the risk associated with each hazard?
- (b) Using fault tree analysis, discover the conditions that might cause each of the hazards you identified above.

Question 6. *Distributed Software Systems.*

- (c) Explain why distributed systems are inherently more scalable than centralized systems. What are the likely limits on the scalability of distributed systems?
- (d) What is the difference between a fat-client and a thin-client approach to client-server systems development?
- (e) Using a distributed object approach, propose a distributed architecture for a national store e-commerce system, where users can check product availability and buy products online. State any assumptions you make clearly.

Question 7. Configuration Management.

Configuration management is the management of system change. When a system is maintained, the role of the configuration management team is to ensure that changes are incorporated in a controlled way.

A common problem with system building occurs when physical file names are incorporated in system code and the file structure implied in these names differs from that of the target machine. Write a set of programmer's guidelines that help avoid this and other system building problems that you can think of.

Question 8. Software Reliability.

Suggest appropriate reliability metrics for the following classes of software systems. Give reasons for your choice of metric. Suggest also approximate acceptable values of the system reliability.

- (a) A system that monitors patients in a hospital intensive care unit.
- (b) A word processor.
- (c) An automated vending machine control system.
- (d) A system to control braking in a car.
- (e) A management report generator.