

98-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 calculators permitted. This is a closed book exam.
3. Answer any five of the nine 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.

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. *Object-Oriented Software Design.*

Identify possible objects in the following system and develop an object-oriented design for it. Make and state reasonable assumptions about the system when deriving the design.

A group diary and time management system is intended to support the timetabling of meetings and appointments across a group of co-workers. When an appointment is to be made that involves a number of people, the system finds a common slot in each of their diaries and arranges the appointment for that time. If no common slots are available, it interacts with the user to rearrange his or her personal diary to make room for the appointment.

Question 3. *Formal Methods.*

- (a) Explain what is meant by “formal methods” in Software Engineering.
- (b) What are the advantages of formal methods to software engineers?
- (c) What is the difference between algebraic and model-based approaches to formal specification?
- (d) An abstract data type representing a stack has the following operations associated with it:

New:	Bring a stack into existence
Push:	Add an element to the top of the stack
Top:	Evaluate the element at the top of the stack
Retract:	Remove the top element from the stack and return the modified stack
Empty:	True if there are no elements on the stack

Define this abstract data type using an algebraic specification.

Question 4. *Real-Time Systems.*

- (a) Define real-time software systems.
- (b) Using examples, explain why real-time systems usually have to be implemented using concurrent processes.
- (c) Design a process architecture for an environmental monitoring system that collects data from a set of air quality sensors situated around a city. There are 5,000 sensors organized into 100 neighborhoods. Each sensor must be interrogated four times per second. When more than 30% of the sensors in a particular neighborhood indicate that the air quality is below an acceptable level, local warning lights are activated. All sensors return the readings to a central computer, which generates reports every 15 minutes on the air quality in the city.

Question 5. *Software Testing.*

- (a) Explain why testing can only detect the presence of errors, not their absence.
- (b) Some believe that developers should not be involved in testing their own code but that all testing should be the responsibility of a separate team. Give arguments for and against testing by the developers themselves.
- (c) Derive a set of tests for a method called 'cutWhiteSpace' in a 'Paragraph' object that, within the paragraph, replaces sequences of blank characters with a single blank character.

Question 6. *Software Project Management.*

- (a) Identify the main stages of risk management in software engineering projects.
- (b) Explain why the best programmers do not always make the best software managers.
- (c) You are asked by your manager to deliver software to a schedule that you know can only be met by asking your project team to work unpaid overtime. All team members have young children. Discuss whether you should accept this demand from your manager or whether you should persuade your team to give their time to the organization rather than to their families. What factors might be significant in your decision?

Question 7. Critical Systems.

- (a) What is the fundamental difference between hardware and software failures? Given this difference, explain why hardware reliability metrics are often inappropriate for measuring software reliability.
- (b) A safety-critical software system for treating cancer patients has two principal components:
- A radiation therapy machine that delivers controlled doses of radiation to tumor sites. This machine is controlled by an embedded software system.
 - A treatment database that includes details of the treatment given to each patient. Treatment requirements are entered in this database, and are automatically downloaded to the radiation therapy machine.

Identify three hazards that may arise in this system. For each hazard, suggest a defensive requirement that will reduce the probability that these hazards will result in an incident. Explain why your suggested defence is likely to reduce the risk associated with the hazard.

Question 8. Software Quality.

- (a) What is software quality management?
- (b) List the three main activities of software management for large systems.
- (c) Explain why it is difficult to validate the relationship between internal product attributes, such as program size or number of procedure parameters, and external attributes, such as maintainability.

Question 9. Distributed Software Systems.

- (a) Explain why deploying software as a service can reduce the IT support costs for a company. What additional costs might arise if this deployment model is used?
- (b) A company wishes to move from using desktop applications to accessing the same functionality remotely as services. Identify three risks that might arise and suggest how these risks may be reduced.