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

Total mark out of 100.

Question 1. *The Software Development Process.*

Describe the main activities of the software design process and the outputs of these activities. Using a diagram, show possible relationships between the outputs of these activities.

Question 2. *Software Design.*

(a) Define the terms cohesion, coupling and adaptability. Explain why maximizing cohesion and minimizing coupling leads to more maintainable systems.

(b) 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.

Using a function-oriented approach, derive a high-level design for the system described above. Make reasonable assumption and clearly state them.

Question 3. *Object-oriented Design.*

A software system is to be developed for a *residential home security* system. The system uses several types of sensors that are installed inside a customer's home. They include: door and window open/closed sensors, smoke sensors, water sensors and motion detectors. The sensors are wireless and connect to a base station, also installed inside the home. The base station connects over the Internet to a central monitoring station. The base station allows the arming and disarming of the system and can be used to turn on or off any of the sensors individually. If the system is armed and any of the sensors is triggered, an alarm is sounded and the central monitoring station is notified.

Using an *object-oriented* approach, derive a design for the system above, describing possible objects that you may use for the design. Make reasonable assumption and clearly state them.

Question 4. *Software testing.*

(a) Discuss the differences between functional and structural testing and suggest how they may be used together in the defect testing process.

(b) Explain why testing can only detect the presence of errors but not their absence.

(c) Design a testing strategy for the residential home security system described in **Question 3** above.

Question 5. *Real-Time Software Systems.*

(a) Define real-time software systems.

(b) What is the difference between "soft" real-time systems and "hard" real-time systems?

(c) Draw a state machine model of the control software for the following system.

> A telephone answering machine that records incoming messages and displays the number of accepted messages on a LED display. The system should allow the telephone owner to dial in, type a sequence of numbers (identified as tones) and have the recorded messages replayed over the phone.

Question 6. *Rapid Software Development.*

(a) Explain why the rapid delivery and development of new systems is often more important to businesses than the detailed functionality of these systems.

(b) A charity has asked you to prototype a system that keeps track of all donations that have been received. This system has to maintain the names and addresses of donors, their particular interests, the amount donated, and when the donation was made. If the donation is over a certain amount, the donor may attach conditions to the donation (e.g., the money must be spent on a particular project), and the system must keep track of these and how the donation was spent.

Discuss how you would prototype this system.

Question 7. *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 8. *Client-Server Architectures.*

Your customer wants to develop a system for stock information where dealers can access information about companies and can evaluate various investment scenarios using a simulation system. Each dealer uses this simulation in a different way, according to his or her experience and the type of stocks in question. Suggest a client-server architecture for this system that shows where functionality is located. Justify the client-server system model you have chosen.