Answer ALL of Part I and TWO questions from Part II

You are advised to spend not more than 1 hour on Part I.

Part I

1. Below is the draft specification for a proposed software system:

   "A system is required to maintain an inventory of the contents of a warehouse. Items are delivered for storage at any time during the day and must be allocated space. An identification label must be attached to each item before storage and some items need to be stored in a refrigerated unit. An item can be stored for any period of time but some items have an expiry date by which they must be removed from the warehouse.

   When items are removed they need to be labelled, packaged for delivery and put on the correct delivery truck. The truck driver should be given a list of delivery addresses for the items.

   The system should be able to generate reports showing the current contents of the warehouse and the last day's deliveries and collections."

   a) Taking an object-oriented point of view, draw up a list of potential classes, attributes and external entities using the specification above as a guide.

   [10 marks]

   b) Starting with the classes you have identified in part a), generate a class diagram for the warehouse software system. Make sure each class is labelled with any key attributes or operations.

   [20 marks]

   c) Create object interaction diagrams for the following scenarios: delivery and storage of an item, collection and delivery of an item, generation of a report on the warehouse contents. Make sure these diagrams are consistent with the class diagram.

   [20 marks]

   [Total 50 marks]

Part II

(Answer two of the following questions.)

2. Create a context diagram and a first-level data-flow diagram to show the process of structured development, from obtaining the initial specification through to delivering the completed program. Make sure the diagrams are properly constructed and clearly show all data flows.

[Total 25 marks]

3. a) Identify and describe the key points of potential failure in the Waterfall, Prototyping and X-Model life cycle models.

[15 marks]

b) How can each life cycle model be managed to reduce the risks?

[10 marks]

[Total 25 marks]

4. a) Outline how formal methods allow a system to be verified and validated. Give examples using Z or Object-Z notation.

[15 marks]

b) If a software system fails in some respect who should be legally responsible and how is responsibility determined?

[10 marks]

[Total 25 marks]

END OF PAPER