
SECTION B

- Answer **ALL** questions in the answer booklet provided.

Short Answer Questions **[15 + 15 = 30 marks]**

a) Consider the following problem statement:

You have been hired by a major online retailer to develop a new software application to receive and manage home delivery of publications (books and magazines). The customers of the retailer can register their details using the web site including name, address, and telephone numbers. They can then browse the publications available for sale, including individual titles (books) and subscription items (number of issues over a defined period), adding those items to their shopping trolley. Once the order is complete, the retailer staff will package any individual books and current edition of a magazine and deliver them to the customer. The total cost of goods purchased, plus a supplemental charge of $5 per book or $10 per subscription, is charged to the customer's account. A record of any items that are not available are kept in the database as backorders and are automatically added to the next order from the customer.

Your task is to prepare a UML class diagram illustrating the concepts in the problem statement above. Your class diagram must demonstrate the following elements:

- Attributes, properties, and methods;
- Collections;
- Abstract classes;
- Inheritance;
- Associations (any type); and
- Cardinality.

b) For this task you are required to write C# code satisfying the following requirements:

- An abstract class named Building which has:
  - Two string attributes for the address and phone number;
  - Read-only properties encapsulating the above attributes;
  - A method ChangePhone which allows only derived classes to provide a new value for the phone number attribute;
  - A custom constructor which takes address and phone number parameters and initialises the attributes defined above; and
  - A ToString() method which returns a string containing the above data in the format:
    address: *phone number*
- An class named RentedBuilding which inherits from the Building class and has:
  - A decimal attribute for the rental per week;
  - A read-only property encapsulating the above attribute;
  - A custom constructor which takes address, phone number, and rental per week, passes the address and phone number to the base class constructor and initialises the rental per week attribute defined above;
  - A ToString() method which returns a string containing all of the stored data (with rental in currency format) in the format:
    address: *phone number* （*rental per week*）

[Section B Total: 30 marks]

[Grand Total: 120 marks]