

THE BRITISH COMPUTER SOCIETY

THE BCS PROFESSIONAL EXAMINATIONS
BCS Level 5 Diploma in IT

OBJECT ORIENTED PROGRAMMING

17th October 2007, 2.30 p.m.-4.30 p.m.

Answer FOUR questions out of SIX. All questions carry equal marks

Time: TWO hours

*The marks given in brackets are **indicative** of the weight given to each part of the question.*

Calculators are NOT allowed in this examination.

1. a) The majority of programs manipulate data structures. Explain the way in which object-oriented programming languages can be used to create data structures. **(5 marks)**
- b) During the execution of a program any data structures it uses must be assigned to some part of the computer's memory. Explain how this is achieved in an object oriented programming language with which you are familiar. **(5 marks)**
- c) Some data structures (such as counters) need to be initialised when they are created. Show how this initialisation can be achieved using an object oriented programming language. **(5 marks)**
- d) Show how the mechanism you described in part c) operates when a data structure is derived from another simpler structure via single inheritance. **(5 marks)**
- e) When a program has finished using a data structure, the memory it occupies can be returned to the pool of available memory. Describe a mechanism that achieves this without the programmer having to explicitly release memory. **(5 marks)**
2. a) Explain the effect of the following modifiers in the declaration of variables and methods:
- i) public;
 - ii) private;
 - iii) protected.
- (6 marks)**
- b) A class MyClass has only one constructor. Discuss the way the class might be used if its declaration is as follows:
- ```
class MyClass {
 private MyClass(){
 .
 .
 }
 .
 .
}
```
- (5 marks)**
- c) Justify the inclusion of a mechanism for supporting encapsulation in an object oriented programming language. **(7 marks)**
- d) Write a set of guidelines that set out good practice when writing code in a language that supports encapsulation. **(7 marks)**

**Turn over]**

3. a) Explain the meaning of the following terms within the context of an object oriented programming language:
- i) method;
  - ii) message;
  - iii) overloading;
  - iv) overriding.
- (8 marks)**
- b) The terms in part a) of this question describe features of programming that are implemented in all the major object-oriented programming languages and similar features are not found in structured programming languages. What advantages do object-oriented languages gain through the implementation of these features? **(8 marks)**
- c) Explain how overloading and overriding contribute to the implementation of polymorphism in object oriented languages. **(9 marks)**

4. *Zodak Printers are a company that specialises in digital printing over the web. New customers first have to register to use the system by supplying a username, password and their email address, a confirmation email is sent to this address and customers must respond to the email to complete their registration. Once validated new customers can then use the system.*

*All customers must login using their username and password; if these match the stored details they can then access the system.*

*The customer must create an album for each set of photographs they want to upload for printing. Each customer has a file storage limit of 200mb, if they exceed this limit, they can not load any further photographs.*

*Once the customer has loaded their photographs, they can place their order by choosing how many copies of each photograph they want and what size. Then they can proceed to check out, where they will be prompted for their delivery and credit card details. If these are validated, then their photographs are put on a queue for printing. The customer can optionally print out the details of the order.*

*At regular intervals a photo technician checks the quality of the photographs being printed and will add the appropriate toner if required.*

*Once a week, the Administrator sends out an email of special offers to customers who have not placed an order for 3 months.*

- a) Draw a use case diagram for this system. **(15 marks)**
- b) Discuss how use case diagrams and descriptions provide an overview of the user requirements of a system. Within your answer include examples from the above system. **(10 marks)**

5. A Bank wishes to keep information on its customers. The proposed Customer class has the following instance variables:

|                |         |
|----------------|---------|
| customerNo:    | String  |
| customerName:  | String  |
| date of Birth  | Date    |
| credit rating: | Integer |

Customer number and name are string data types, credit rating is a number between 0-20, where 0 represents customers not yet rated.

A class variable is also required, called noOfCustomers, which will be incremented each time a Customer instance is created.

Using an object-oriented programming language that you are familiar with, write code to perform the following. Where appropriate include suitable integrity checks:

- i) Show the declaration of the Customer class, including any *setters* and *getters* methods. **(15 marks)**
- ii) Declare two constructors as follows, both constructors should increment the class variable appropriately:
  - The first is a default constructor that has no parameters and sets the instance variables to either "not known" for the strings, 0 for the integer and 1<sup>st</sup> January 1900 for the date (assume there is a Date constructor that accepts dates in a string format).
  - The second takes 4 parameters, one for each of the instance variables. **(8 marks)**
- iii) Show how both constructors could be instantiated. **(2 marks)**

6. The Unified Modelling Language 2.0 (UML) comprises of 13 different diagrams. These can be broadly categorised as:

- i) Structure diagrams
- ii) Behaviour diagrams
- iii) Interaction diagrams

- a) For each category, give a description of one diagram that falls into the category; include a simple example of its use and explain when you would use it. **(15 marks)**
- b) Discuss how object-oriented code can be tested; within your answer explain which UML diagrams can be used to aid testing. **(10 marks)**

**\*\* END OF PAPER \*\***