

BCS THE CHARTERED INSTITUTE FOR IT

BCS HIGHER EDUCATION QUALIFICATIONS
BCS Level 5 Diploma in IT

OBJECT ORIENTED PROGRAMMING

Tuesday 10th November 2020 - Afternoon

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

Time: TWO hours

Answer any Section A questions you attempt in Answer Book A
Answer any Section B questions you attempt in Answer Book B

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

Calculators are NOT allowed in this examination.

Section A
Answer Section A questions in Answer Book A

A1.

Choose **FIVE** features of the object oriented paradigm. Describe what they are, why they are important and give an example of **EACH** feature, either with code or with a diagram.

(25 marks)

A2.

a) In the context of object oriented development, what is meant by the term *design pattern*? Explain, from a programmer's point of view, the motivation for using them.

(10 marks)

b) Choose **THREE** of the following design patterns and give a detailed description of each, stating the problem they address and the basis of the solution they offer.

- i) Adapter.
- ii) Decorator.
- iii) Iterator.
- iv) Observer.
- v) Singleton.

(15 marks)

A3.

A library wishes to update its computer system and needs to store the following details:

- Borrower: idNo, name, address, date of birth, status (adult or child), no of books allowed, no of books borrowed.
- Librarian: idNo, name, address, date of birth, job title, salary, email address.
- Book: bookNo, title, main author, category, total number of copies in the library.
- Loan: date borrowed, date due, date returned, fine.

An adult borrower can borrow up to 8 books at one time, whilst a child can borrow up to 5 books. The *no of books allowed* attribute records this amount, whilst *no of books borrowed* records how many books they have currently taken out. A borrower cannot borrow any more books when they reach their limit.

A fee is charged if they return the book later than the date due. The system needs to be able to register new borrowers and to add new books to the catalogue.

a) Draw a class diagram to represent this information.

(16 marks)

b) Draw **TWO** object interaction diagrams, **ONE** to represent a valid instantiation of at least two of the above classes and **ONE** to show an invalid instantiation. Explain why the invalid one is INCORRECT.

(4 marks)

c) Write OCL code which represents a check that prevents a borrower exceeding their borrowing limit.

(5 marks)

Section B
Answer Section B questions in Answer Book B

B4.

- a) Describe a real-world practical situation in which it would be appropriate to use ad-hoc polymorphism.

(10 marks)

- b) Describe a real-world practical situation that lends itself to the use of *hierarchical inheritance*. Provide a class diagram and stub code to show how the specific classes in your scenario will be interconnected.

(15 marks)

B5.

- a) Some object oriented programming languages support *method overriding*. Describe a practical situation in which method overriding might be useful.

(10 marks)

- b) Discriminate between the inter-class relationships *aggregation* and *specialisation* and give code examples to show how they may be implemented in practice.

(15 marks)

B6.

Provide object-oriented code that implements a *stack* class that is suitable for storing integer numbers. It will have a finite capacity of 10 items. The stack is a LIFO (last in, first out) data structure and will implement the following **FIVE** operations: *push* (insert an item into the stack), *pop* (retrieve the last-added item from the stack), *isEmpty* (test if the stack is empty or not), *isFull* (test if the stack is full or not), *getNumItemsHeld* (report how many items are currently in the stack). None of these operations should display to the screen or take input from the console – all interaction with objects of the class should be via messages (method arguments and return values). Also provide a test method outside that class that instantiates and tests the stack class.

FIVE marks are available for an appropriate stack class structure. Up to **THREE** marks will be awarded for each operation (*push*, *pop*, *isEmpty*, *isFull*, *getNumItemsHeld*). The remaining **FIVE** marks are available for implementing appropriate test code.

(25 marks)

End of Examination