

BCS THE CHARTERED INSTITUTE FOR IT

BCS HIGHER EDUCATION QUALIFICATIONS BCS Level 5 Diploma in IT

OBJECT ORIENTED PROGRAMMING

Tuesday 3rd May 2011 - 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. a) For each of the programming paradigms described below provide a phrase which describes that paradigm and brief notes which explain how programmers develop code in a language that supports the paradigm.
- i) A programming style in which solutions take the form of a set of interrelated objects;
 - ii) A programming style which emphasises the steps taken to reach a solution;
 - iii) A programming style based on a limited set of control structures.
- (15 marks)**
- b) Compare and contrast the three programming approaches outlined above setting out the advantages and disadvantages of each approach.
- (10 marks)**
- A2. a) Give a short phrase which describes each of the following object oriented concepts. For each phrase provide an example of code which implements the concept.
- i) The prototype of a method, which enables the programmer to identify the purpose of a method (via its name), the data it requires to operate (via its argument list), and any end result that is produced (its return type);
 - ii) The use of several functions with the same name which differ from each other in terms of the type of the input and the type of the output of the function;
 - iii) A mechanism which allows a subclass to provide its own implementation of a method already provided by one of its superclasses.
- (12 marks)**
- b) Programmers who use object oriented programming languages frequently design programs in which objects that are instantiated from different classes respond to identical messages. Explain why this is an important feature of object-oriented programming and give an example of how it can be achieved in code.
- (13 marks)**

Turn over]

A3. a) Give the object oriented terminology for each of the following object oriented features and supply an example of code that illustrates the feature:

- i) A blueprint for an object which defines all the data items contained in the object and the operations that are permitted for the data;
- ii) A representation of something within the domain that the program models that contains values of data and which implements operations on that data;
- iii) An operation which will manipulate the data contained in an object;
- iv) A variable which holds data that describes an individual object;
- v) A variable which holds data that is relevant to all the objects created from the same template.

(15 marks)

b) Programmers who use objectoriented languages frequently reuse code which has been developed by themselves or other programmers. Describe the primary technique which makes this possible and give a code example of it in use.

(10 marks)

Section B

Answer Section B questions in Answer Book B

- B4. a) Describe three distinct practical examples of polymorphism, providing illustrative code fragments that show how each is used. (15 marks)
- b) Distinguish between ad-hoc and parametric polymorphism. (5 marks)
- c) Describe any common dynamic data structure you are familiar with in object oriented terms. (5 marks)
- B5. a) Describe three practical techniques that enable classes to be interconnected and provide code fragments that show how each is used. (15 marks)
- b) Write code to implement the class diagram shown below. Supply a *main()* function that instantiates an object of the class and makes appropriate use of each of the functions.

bankAccountClass
- balance:double
+ bankAccountClass(balance: double)
+ getBalance(): double
+ deposit(amount: double)
+ withdraw(amount: double)
+ isOverdrawn(): boolean

(10 marks)

- B6. a) Distinguish between *accessor* and *mutator* functions, providing an example class definition containing one example of each. (7 marks)
- b) Briefly explain why a programmer may choose to designate class members as either *private*, *public* or *protected* within a single class. (6 marks)
- c) Determine class member visibility for each variable in the class hierarchy provided in **Appendix A**. Present your answer (in your answer booklet) in the form of a table like the one shown below, using the following symbols: private (-), protected (#), public (+), not visible (blank).

		CLASS			
		C1	C2	C3	C4
VARIABLE	a				
	b				
	c				
	d				
	e				
	f				
	g				
	h				
	i				
	j				
	k				
	l				

(12 marks)

Appendix A: Code Fragment for Question B3

```
class C1
{
    private:
        a;
    protected:
        b;
    public:
        c;
};

class C2: protected C1
{
    private:
        d;
    protected:
        e;
    public:
        f;
};

class C3: public C2
{
    private:
        g;
    public:
        h;
    protected:
        i;
};

class C4: private C3
{
    private:
        j;
    public:
        k;
    protected:
        l;
};
```

**** END OF EXAM ****