

**THE BCS PROFESSIONAL EXAMINATION
Diploma**

October 2004

EXAMINERS' REPORT

Object Oriented Programming (Version 2)

General

The statistics suggest a 62% pass rate. Disappointingly, this is less than the April 2004 figure that was around 70%. A number of candidates are to be commended, achieving scores of 90%. However, an uncomfortably large number of candidates appeared to be badly prepared for the examination. Many achieved very low marks as a result.

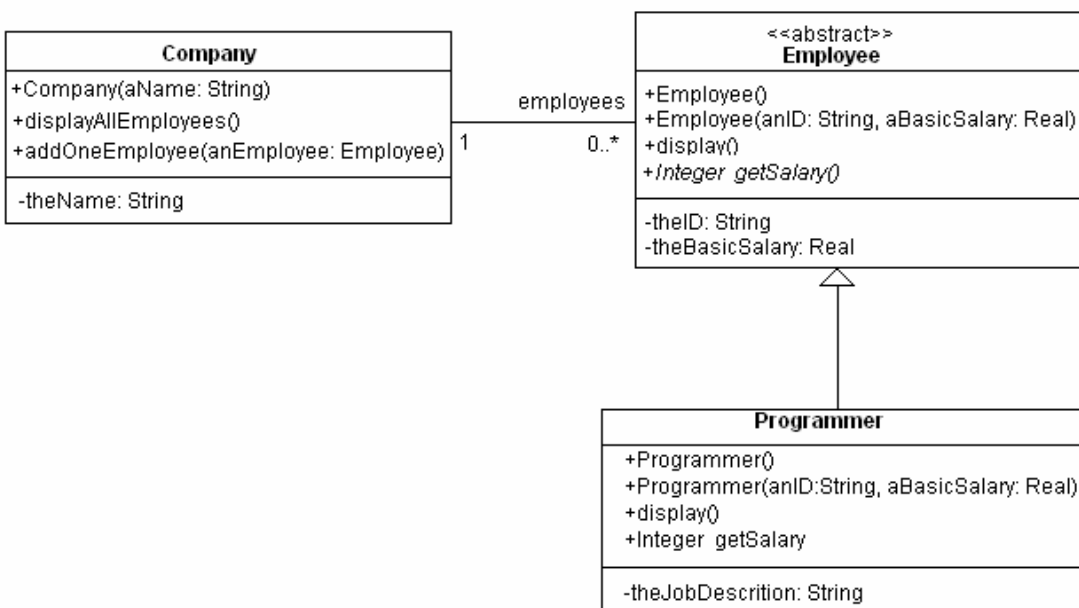
With one exception, most questions were attempted by roughly the same number of candidates. This suggests that these questions were reasonably balanced. The least popular question was question 3. As we move toward pattern-led OO development perhaps centres need to ensure this is adequately covered. This is too important an area to ignore.

Too many candidates still fail to complete the "Questions answered" box on the front of their examination scripts. Could centres and examination invigilators please remind candidates that it is their own best interest that this information is completed.

In similar vein, too many candidates fail to answer the question posed. Although we try to give candidates the benefit of the doubt this is not always possible. It is also true that some candidates put in too much or too little effort for the marks allocated to question. It would be helpful if centres could ensure that candidates have been briefed in basic examination technique.

Question 1

1. Operations and attributes of a class are given the following visibility: public, protected and private. **(6 marks)**
- a) Explain the meaning of these three terms.
 - b) Describe the information documented by the following class diagram.



(14 marks)

- c) What constructs of an object oriented programming language would you expect to use in order to implement the class diagram shown in b)? (5 marks)

Answer Pointers

Question		Mark
1	This question examines Part 3 (Design) and Part 4 (Practice) of the syllabus	
(a)	public: visible to any object protected: visible to receiving object and subclass objects private: visible to the receiving object	2 2 2
(b)	<p>Descriptions based on the following:</p> <p>A Company object is associated with zero or more Employees. Each Employee is associated with one Company. A Programmer is special kind of Employee.</p> <p>A Company object is instantiated by a constructor that takes a single String parameter. It may be asked to display all Employees or to add an Employee to itself. It has theName as a private attribute. It holds its Employees in a collection named employees.</p> <p>An Employee object cannot be instantiated as the Employee class is abstract. It acts as the superclass for Programmers. It has a default and a parameterised constructor. In the former case no actual parameter is supplied. In the latter case a String and a Real actual parameter is supplied . It also has a concrete display operation and an abstract getSalary operation.</p> <p>The Programmer class specialises the Employee class. A Programmer object can be instantiated using a default or a parameterised constructor. In the former case no actual parameter is supplied. In the latter case a String and a Real actual parameter is supplied. The Programmer class inherits the display operation from Employee and redefines it. It also inherits the abstract operation getSalary and provides an implementation for it. It has visibility of the protected attribute theBasicSalary.</p>	3 2 4 5
(c)	Class declaration (including method and fields), visibility modifiers, class extension (inheritance), abstract class/method declaration, method redefinition, collections	5

Examiners' Comments

This question was a popular choice. The average mark was ~45%. Two failings were identified and centres preparing students for the examination need to consider these.

- OK
- There was a tendency among many candidates to simply catalogue the features of the classes rather than explore the implication of the architecture. For example, many candidates failed to record the outcome of having class Employee as abstract. Similarly, they did not identify that operation getSalary as deferred and hence required a definition in the subclass Programmer. Candidates should look more deeply into these designs and make important observations.
- This was quite poorly answered with too many candidates offering large tracts of program code without explanation. Further, the question was deeper than coding seeking to identify some of the key features that are found in OO programming languages.

Question 2

2. a) A class is required to hold the basic details of bank accounts. The proposed Account class has the following instance variables:

number: Integer
 name: String
 balance: Integer

A class variable called numberOfAccounts is also required. It is incremented each time an Account instance is created.

Using an object oriented programming language with which you are familiar, write code to:

- i) Show the declaration of the fields for the Account class. **(4 marks)**
- ii) Declare TWO constructors. The first should be a default constructor with no parameters that sets instance variables to either “not known” for the Strings or 0 for Integers. The second constructor should take three parameters, one for each of the instance variables. Both constructors should increment the class variable accordingly. **(7 marks)**
- iii) Show how both constructors given in ii) could be used to instantiate an object of the class Account. **(2 marks)**
- iv) Show the declaration of a getter method and setter method for one of the instance variables of the class Account. **(2 marks)**
- b) Giving your reasons, state which development style is most appropriate for the construction of a software system when using object technology. **(10 marks)**

Examiners' Comments

This was intended to be a straightforward question that should have been answerable by most candidates. The average mark was ~60%.

- a) On the whole there were no major problems with this question. However, it was surprising how many candidates could not instantiate an object of the class Account.
- b) It was pleasing to note that virtually all candidates recognised that an iterative incremental style of software development is the most appropriate for OO systems development. With the popularity of agile methods this kind of approach is becoming increasingly important.

Answer Pointers

Question		Mark
2	This question examines Part 2 (Concepts) and Part 4(Practice) of the syllabus	
(a)	(Java) (i) private int number; private String name; private int balance; private static int numberOfAccounts = 0;	4
	(ii) public Account() { number = 0; name = “not known”; balance = 0;	7

	<pre> numberOfAccounts++; } or public Account(int aNumber, String aName; int aBalance) { number = aNumber; name = aName; balance = aBalance; numberOfAccounts++; } </pre>	
	<pre> (iii) Account acc1 = new Account(); Account acc2 = new Account(123, "James Smith", 100); </pre>	2
	<pre> (iv) public int getNumber() { return number; } or public String getName() { return name; } or public int getBalance() { return balance; } </pre>	2
(b)	<p>Iterative and incremental. A discussion/justification should typically include the following:</p> <ul style="list-style-type: none"> in keeping with the class as being highly cohesive and loosely coupled, a version of the system can be “up-and-running” early, reduces risk of failure, avoids a “big bang” approach, suites small to medium teams of developers, possible to develop one class at a time, possible to develop one (or a selection of) method(s) at a time, testing can also be iterative and incremental, possible to develop one use case at a time 	10

Question 3

3. a) Describe what is meant by the term *Design Pattern*. (2 marks)
- b) Give an example of ONE specific design pattern. Your answer should include a justification for its existence, and a description of it in use supported by appropriate UML diagrams. (15 marks)
- c) Using an object oriented programming language with which you are familiar, give an outline implementation of the design pattern in b). Your answer should focus on those elements that pertain to the design pattern. (8 marks)

Examiners' Comments

Clearly this was perceived as a difficult question. However many of those candidates who attempted it scored high marks with an average of ~52%. At the other extreme some candidates did rather poorly. However, they also tended to do poorly in other questions as well.

- a) This was a relatively easy question to answer and virtually all candidates were given full marks. However, several candidates gave far too much information for just the 2 marks allocated!

- b) No real problems with most answers but some candidates developed illegal UML diagrams. Typically they were composed of English text rather than properly specified operations and attributes.
- c) No real problems.

Answer Pointers

Question		Mark
3	This question examines Part 3 (Design) of the syllabus	
(a)	The solution to a problem in a context. or A commonly accepted approach to solving a particular kind of problem	2
(b)	Typically one of the more popular Design Patterns is expected e.g. Adapter, Decorator, Iterator, Observer or Singleton. Justification Description A UML class diagram and interaction diagram is expected to support a description of it in use.	5 6 4
(c)	The code should clearly show how the Design Pattern is implemented.	8

Question 4

4. In the following class diagram a University is shown as an aggregation of many Students. The diagram reveals that objects of the Student class can be sent the messages getName and getAge.
- a) Construct a collaboration diagram, showing the University and three students, to illustrate how the University would produce a list of the names of those students that are over the age of 21. Give a detailed explanation for the construction of this diagram. **(9 marks)**
 - b) Show a sequence diagram for the same scenario as described in part a) of this question. Give a detailed explanation for the construction of this diagram. **(9 marks)**
 - c) Describe what is shown by these two types of diagram. Identify the individual strengths of collaboration and sequence diagrams. **(7 marks)**

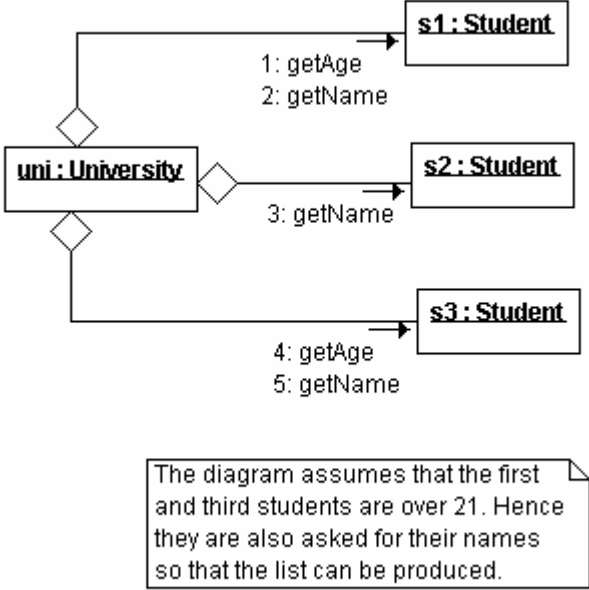
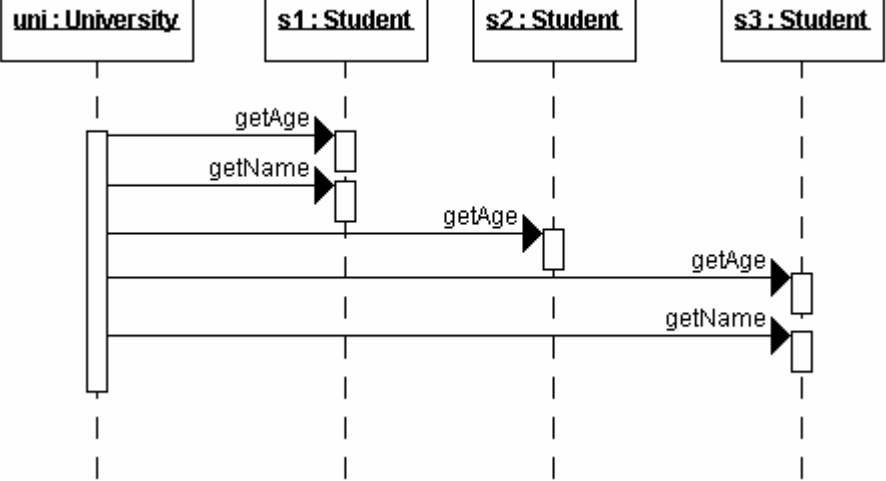


Examiners' Comments

Again, a popular question with an average mark of ~50%. Parts 4(a) and 4(b) suffered from little or no explanations when they were explicitly requested in the question. Those candidates that did offer some explanation were generally shallow. In many answers candidates failed to identify that Students would only be asked for their name if their age exceeded 21. Hence the diagrams need

to emphasise that `getAge` was requested first, and that `getName` was only requested where the criteria was met.

Answer Pointers

Question		Mark
4	This question examines Part 3 (Design) of the syllabus	
(a)	 <pre> classDiagram class uni : University class s1 : Student class s2 : Student class s3 : Student uni "1" *-- "1" s1 uni "1" *-- "1" s2 uni "1" *-- "1" s3 uni --> s1 : 1: getAge uni --> s1 : 2: getName uni --> s2 : 3: getName uni --> s3 : 4: getAge uni --> s3 : 5: getName </pre> <p>The diagram assumes that the first and third students are over 21. Hence they are also asked for their names so that the list can be produced.</p>	9
(b)	 <pre> sequenceDiagram participant uni as uni: University participant s1 as s1: Student participant s2 as s2: Student participant s3 as s3: Student uni->>s1: getAge uni->>s1: getName uni->>s2: getAge uni->>s3: getAge uni->>s3: getName </pre>	9
(c)	<p>Collaboration and sequence diagrams are examples of interaction diagrams. Their role is to reveal the interaction (i.e. message passing) between objects. In the sequence diagram the order of the messages is determined by the timeline. In the collaboration diagram the ordering is shown by numbering of the messages. This ordering is probably better shown by the sequence diagram. However, the collaboration diagram also reveals the architecture of the objects which is otherwise implicit in the sequence diagram.</p>	7

Question 5

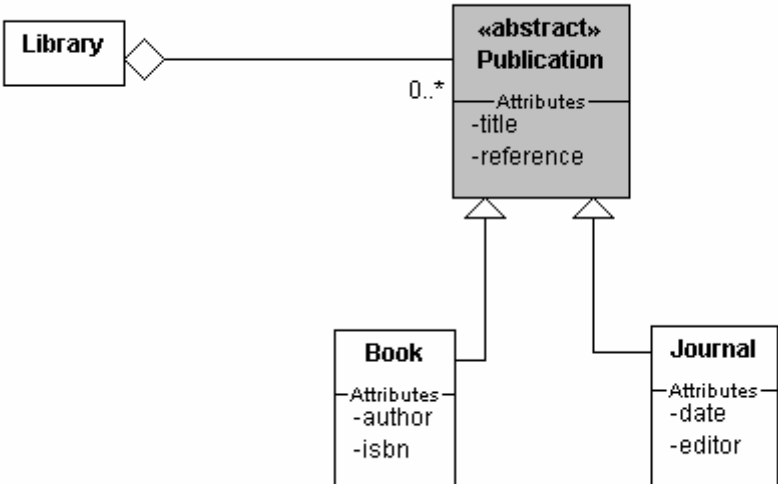
5. A lending library holds a large number of publications that may be books or journals. Both books and journals have a title and a unique library reference number. Each book also has an author and an international standard book number (ISBN). Journals have a date of publication and the name of the editor.
- a) Construct a class diagram for this scenario giving a detailed explanation for the resulting architecture. For each class in the diagram show the attributes that have been assigned to each. (8 marks)
 - b) Identify the operations required by each class for the library to produce a list of publications that are currently on loan to the library's borrowers. (8 marks)
 - c) In part a) of this question you are required to display the details of each publication. Identify within the class diagram how such a display operation might be implemented. Your answer should reveal what object oriented features are being exploited. (9 marks)

Examiners' Comments

Although this was a popular question many candidates did not score particularly well. The average mark was 40%.

- a) On the whole the class diagram was properly constructed by most candidates. However several candidates specialized the Library class into a Book or Journal. This is clearly not sensible! Far too many candidates did not read this question carefully enough and gave no explanation of the class diagram. Marks were lost as a result.
- b) Perhaps this question was not clear enough. Generally it was poorly answered by most candidates.
- c) This question sought to discover if the candidate really understood the OO paradigm by identifying the use of polymorphic substitution and redefined methods.

Answer Pointers

Question		Mark
5	This question examines Parts 2 and 3 (Concepts and Design) of the syllabus	
(a)	 <pre> classDiagram class Library class Publication { <<abstract>> -title -reference } class Book { -author -isbn } class Journal { -date -editor } Library o-- "0..*" Publication Publication < -- Book Publication < -- Journal </pre>	8

(b)	<pre> classDiagram class Publication { <<abstract>> -Operations +isOnLoan +display } class Book { -Operations +display } class Journal { -Operations +display } class Library Publication < -- Book Publication < -- Journal Library *-- "0..*" Publication </pre>	8
(c)	<p>In the solution for part (b) the display method is defined in each of the three publication classes. The display operation in the Publication class determines that the Library can send this message to all publications irrespective of whether they are Books or Journals. This operation will, of course, only provide the reference and title values. In the subclass Book, for example, the redefined version of display will call the superclass version of the operation and augment it by also displaying the author and ISBN. Similarly for the Journal class.</p>	9

Question 6

6. a) Explain what is meant by the following terms:

- i) Class
- ii) Object instance
- iii) Specialisation
- iv) Operation overloading
- v) Operation redefinition

(15 marks)

b) Give a detailed account of the substitution principle used in object oriented systems. Your answer should include an example of how this is used.

(10 marks)

Examiners' Comments

Part (a) was generally well answered. The average mark was ~40%. The problem for candidates seemed to be with part (b). Here, many students appeared to answer their 'own' question! For those who stayed with the question, there were tendencies to either dump all their knowledge in this area or resort to large tracts of coding. In contrast, the question sought an explanation of substitution and how it is deployed in OO systems.

Answer Pointers

Question		Mark
6	This question examines Part 2 (Concepts) of the syllabus	
(a)	<ul style="list-style-type: none"> (i) A class is an abstraction for some problem-domain entity. The class describes the features for it including its state and behaviour. (ii) An instance or object instance represents a particular example of some class. The instance would have state and behaviour 	15

	<p>(iii) A specialised class is a subclass of another, inheriting all the features of its superclass. The specialised class can have additional attributes and operations and can redefine the behaviour of an inherited operation.</p> <p>(iv) Two operations with the same name but different signatures are described as overloaded operations. The signature comprises the number and type of parameters. It is the compiler's responsibility to disambiguate usage.</p> <p>(v) A subclass may choose to override the implementation of an inherited operation. Typically this would involve calling the overridden operation in the superclass than augmenting it to perform the additional distinguishing features of the subclass.</p>	3 marks for each element
(b)	<p>The principle of substitution states that an object of a subclass can be used anywhere an instance of a superclass is expected. Hence in the solution to question 5(a) we see that the Library believes it has a collection of Publication objects. In truth the collection comprises Book and Journal objects. This is possible because a Book can substitute for a Publication since it is a subclass of the latter.</p> <p>This substitution is possible since a subclass inherits all the features of its superclass. Hence in solution 5(b) all Publication objects can be sent the display message. Even were it not redefined in the subclasses it would be inherited.</p>	10