BCS Higher Education Qualification

Diploma

September 2019

EXAMINERS' REPORT

Object Oriented Programming

Question number: A1
Syllabus area: **section 2 (Concepts), parts 2.1, 2.2, 2.3 and section 3 (Design), part 3.2**
Total marks allocated: 25
Examiners' Guidance Notes
Most candidates answered this question, with more than three-quarters passing. The majority of candidates could provide an outline explanation of each concept. Marks were lost for not providing a relevant example to back up the points made, which was needed for full marks. The concept that lost most marks was aggregation, which was either not answered, or the wrong answer provided, with some candidates confusing it with inheritance.

Question number: A2
Syllabus area: **section 3 (Design), part 3.4** **section 4 (Practice), part 4.4**
Total marks allocated: 25
Examiners' Guidance Notes
This was answered by 45% of the candidates, with over half passing. For part a), lower scores went to answers where the candidate mixed up the different types of diagrams, for example, describing a class diagram as a type of behavioural diagram, or a use case diagram as a type of static diagram. Some candidates described design patterns too, instead of UML diagrams. For part b), a large number of candidates described white and black box testing, which gained some credit. For a higher mark the testing needed to be described in the context of object oriented programming, not just a general discussion of testing. To gain full credit a discussion of how UML diagrams can be used for testing was required.

Question number: A3
Syllabus area: **section 3 (Design), part 3.2** **section 4 (Practice), part 4.3**
Total marks allocated: 25
Examiners' Guidance Notes
This question was answered by over 60% of the candidates with over 80% passing. Most candidates got Part a) right. For full marks, an explanation was needed for why the diagram was wrong. Part iv) was the one most people got wrong, a Course can exist by itself, whereas an Occurrence must be related to one Course. A lot of candidates did not attempt Part b) at all, whereas other candidates got full marks. Marks were generally lost for not defining noOfCourses as a class variable and not implementing the association between the two classes. Some answers missed including the constructors, or defined them incorrectly. Some candidates included the setters and getters too, plus an example of instantiating the classes, which were not required.

Question number: A4
Syllabus area: **section 2 (Concepts) part 2.1.** **section 2 (Concepts) part 2.3 and section 4 (Practice) part 4.2 and 4.3.**
Total marks allocated: 25
Examiners' Guidance Notes
This question was attempted by 55% of candidates with 60% achieving a passing grade. In part a), many candidates knew definitions of class and instance variable, but did not address the part of the question asking about member variables. It was common to conflate these concepts with the idea of local and global scope, and variables defined in methods. In part b), it was apparent that many candidates could provide code examples to illustrate the creation of abstract and concrete classes, but failed to address the part of the question that asked specifically for situations in which each might be used. It appeared that some candidates had internalised the syntax and definitions of key terms, but were not familiar with how these concepts might be deployed in a practical scenario.

Question number: B5
Syllabus area: **section 1 (Foundations) part 1.2 and syllabus section 2 (Concepts) part 2.1, 2.2 and 2.3.** **section 2 (Concepts) part 2.1.**
Total marks allocated: 25
Examiners' Guidance Notes
This question was the second most popular question on this year's paper, having been attempted by 84% of candidates with 81% achieving a passing grade. In part a), many candidates made a reasonable attempt at describing abstraction and data hiding. In part b), most candidates understood that methods are often made public and variable often made private, but some were unable to suggest when we might make methods private and variables public. Likewise, there was some confusion over the role of the protected visibility settings, and it was sometimes confused with private. Overall, the answers to this question were satisfactory, reflected in the high average mark.

Question number: B6
Syllabus area: **section 2 (Concepts) part 2.3.** **section 2 (Concepts) part 2.2 and section 4 (Design) parts 4.2 and 4.3.**
Total marks allocated: 25
Examiners' Guidance Notes
This question was attempted by 53% of candidates with 67% of those achieving a passing grade. In part a), rather than explain the role of constructors and destructors, some candidates simply regurgitated the syntax used or pointed out that constructors have the same name as the class and no return type, and can be overloaded. This was not a satisfactory answer, and full marks were only scored by those that described what kinds of task the constructor might perform. Fewer candidates were familiar with the concept of the destructor, potentially reflecting a lack of exposure to programming languages in which these exist (such as C++). Where a response was given, most candidates suggested that constructors could be used to give initial values to variables, and that destructors could be used to de-allocate memory. Very few candidates suggested other possible actions that these methods might perform. In part b), some candidates confused the concepts of multi-level and multiple inheritance. A reasonably high number confused the concepts of hierarchical and multi-level. Some neglected to give code examples, despite these being explicitly requested, thereby losing a proportion of the available marks. Similarly, the question asked for practical situations in which each might be used, but some candidates only provided memorised syntax but were unable to suggest when it might be appropriate to use that syntax (i.e., variety of inheritance).