

**BCS Professional Examination 2017  
Professional Graduate Diploma**

**March 2017**

**Examiners' Report**

**System Design Methods**

**Section A**

**Basic Elements of System Design Methods**

1. a) Discuss how use case and class diagrams can be used to cross-check each other.  
(6 marks)
- b) Discuss how data flow diagrams and entity relationship diagrams can be used to cross-check each other.  
(6 marks)
- c) The ABC method (see Appendix at end of paper) specifies the development process but it does not prescribe any set of systems modeling techniques. Assuming that you are required to use the Unified Modeling Language (UML) techniques, decide which techniques you would use in different stages of the method. Briefly justify your decisions.  
(13 marks)

**Answer pointers**

a)

The following (or suitable alternatives) would be expected:

The methods in the class diagram should correspond with the use cases in the use case diagram. There should not be a method that does not correspond to at least one use case, and there should not be a use case that does not correspond to at least one method.

(6 marks)

b)

The following (or suitable alternatives) would be expected:

The data stores in the data flow diagram should correspond with the entities in the entity relationship diagram. There should not be an entity that does correspond with a data store. Each data store should correspond with one or more entities.

(6 marks)

c)

The following 'allocation' of UML techniques (or appropriate alternatives) would be expected:

Feasibility: Use case diagram (with major use cases), Class diagram (with main classes)

(2)

Requirements analysis: Use case diagram (detailed), Class diagram (mainly with analysis classes), Sequence diagrams (for all or more complex use cases), possibly State machines/diagrams (for more complex classes)

(3)

Design: Class diagram, Sequence diagrams, State machines/diagrams – all these with extra details introduced by design.

(3)

Implementation: primarily Component and Deployment diagrams.

(2)

Activity diagrams could be used in different stages e.g. in 'early' stages to model business processes, and later to specify logic of more complex use cases and class operations.

(3)

(13 marks)

### **Examiner's Guidance Notes**

Part a). Some candidates provided reasonable and passable answers. Many answers were unrelated e.g. some candidates discussed class diagrams and use cases only instead of concentrating on cross-checking.

Part b). The evidence shows that this part (like part a)) caused many problems. Many answers were unrelated e.g. some candidates discussed data flow diagrams and entity relationship diagrams only instead of concentrating on cross-checking.

Part c). Most candidates answered part c) reasonably well i.e. they discussed an 'allocation' of UML techniques to different stages of the method. However, some candidates discussed structured techniques (DFDs, ERDs, etc.) instead.

### **Construction of a Method**

2. a) Explain with the aid of diagrams the following software life cycle models: Waterfall, Incremental, Spiral. Explain also how the Spiral model incorporates both the Waterfall and the Incremental approaches. **(15 marks)**

b) Consider the following software life cycle models: Waterfall, Incremental development, Evolutionary prototyping. For each of these life cycle models discuss briefly whether the ABC method process (see Appendix at end of paper) is based on it. Justify your answers. **(10 marks)**

### **Answer pointers**

a)

Waterfall – diagram and brief explanation are required (3)

Incremental – diagram and brief explanation are required (4)

Spiral - diagram and brief explanation are required (4)

Waterfall – follow the spiral once (2)

Incremental – track the spiral a number of times, once for each increment (2)

**(15 marks)**

b)

Waterfall: ABC method is definitely not based on this process. From the diagram it is clear that this is a highly iterative process which delivers software in 'parts'. The waterfall approach is not really iterative and delivers software 'in one go'.

**(3)**

Incremental: ABC method is clearly based on this approach. In the incremental approach software is delivered in increments/parts and this involves a number of iterations.

**(3)**

Evolutionary prototyping: it involves building prototypes which 'evolve' to the final system. It is therefore an iterative process. However, the diagram suggests that prototyping focuses on Requirements analysis (and capture) only. In addition, the system is delivered in parts instead of 'evolving'.

Conclusion – ABC method is not based on this approach.

**(4)**

### **Examiner's Guidance Notes**

Part a). Most candidates answered the first section of part a) reasonably well i.e. they explained all three software life cycle models. Some answers however were too detailed and partly irrelevant e.g. detailed discussion of fact gathering techniques, detailed discussion of testing techniques, etc.

There is evidence that the second section caused candidates some problems. Only some candidates properly explained how the Spiral model incorporates both the Waterfall and the Incremental approaches.

Part b). Many candidates answered this part reasonably well. In particular, they noticed that the ABC method is not based on the waterfall approach, but it is based on the incremental approach. There is evidence that evolutionary prototyping caused some problems.

### **Selecting a Method**

3. a) There are a wide variety of systems design techniques available for developing IT systems. Discuss why certain design techniques are more relevant to certain types of IT systems than other design techniques. For example, compare the suitability of different design techniques for developing safety critical systems and business information systems. **(10 marks)**
- b) The ABC method (see Appendix at end of paper) is suitable for applications which have certain characteristics. Compare the suitability of this method for developing safety critical systems and business information systems. **(6 marks)**
- c) Assume that you are a project manager of three projects with the following characteristics:
- Project 1. A medium size business application whose requirements can be easily prioritized and grouped.
  - Project 2. A 'standard' business application. You have developed similar systems in the past and expect to develop more in the future.
  - Project 3. A relatively small web-site for a local business. Requirements are vague and are likely to change in the future.

Consider also the following software development approaches/models: Waterfall, Incremental, Evolutionary prototyping, Throw-away prototyping and Component-based development.

Which of the above approaches/models would you choose for each of your projects? Your choices should be properly justified. **(9 marks)**

### **Answer pointers**

a)

Safety critical systems require a high degree of precision in terms of design, in order to attempt to ensure that the system operates in a specified manner. For such systems formal mathematically based systems design approaches may be more appropriate as such design methods can define requirements in an unambiguous manner. **(5 marks)**

Business information systems require detailed design in terms of data and their relationships, and the processing of such data. Systems design approaches that model such requirements for example class diagrams in UML or entity relationship diagrams in structured approaches, as well as use case diagrams from UML or data flow diagrams from structured approaches would be appropriate for the design of such systems. **(5 marks)**

**(10 marks)**

b)

Characteristics of suitable applications:

- Some requirements are vague (prototyping should be used to clarify them)
- Requirements can be divided into different priority groups – increments/parts
- Business application

Conclusion – ABC method is more suitable for business information systems.

(6 marks)

c)

Project 1: Incremental (requirements are grouped, so it is easy to identify increments)

Project 2: Component-based development (components of previous systems can be used, and new components can be produced for future projects)

Project 3: Evolutionary prototyping (requirements are vague and unstable, so there is a need for rapid prototyping leading to the final product)

(3 marks for each correct answer)

(9 marks)

### **Examiner's Guidance Notes**

Part a). The evidence shows that this part caused candidates some problems. A lot of candidates discussed methods and life cycle models instead of design techniques. For example, they discussed Agile methods, the waterfall model, etc.

Part b). Many candidates answered this part reasonably well. However, many answers were unclear and without proper conclusions.

Part c). Most candidates answered this part reasonably well.

## **Section B**

### **Introducing a Method**

4. a) Discuss why both developers and users should be trained in the use of a new systems design method. **(10 marks)**

b) Assume that your software organization has been using a traditional structured method based on the Waterfall approach for many years. Now they want to introduce the new method ABC (see Appendix at end of paper) that incorporates UML. Discuss how introducing the ABC method and UML may impact:

- (i) the method users (i.e. developers),
- (ii) the system users.

**(15 marks)**

### **Answer pointers**

4. a)

Systems developers should be trained in the use of a new systems design method, otherwise they might not use the design techniques within the systems design method accurately, or consistently, and the developers might take longer than necessary to use such techniques. **(5 marks)**

User should be trained in the use of a new systems design method, otherwise it might be difficult for them to check designs, or to provide appropriate feedback to the developers. **(5 marks)**

4. b)

i) Developers should be trained in using an incremental approach and prototyping (as ABC is based on both). **(3 marks)**

As they are not familiar with UML, they should also be given proper training in using UML techniques.

**(3 marks)**

(ii) Users may be required to check and sign off various models and documents specifying user requirements (e.g. use case diagrams, class diagrams). Therefore, these new models/diagrams should be explained to them (a basic training should be provided). (3 marks)

ABC method is partly based on prototyping which requires a substantial participation from users. (3 marks)

ABC method delivers a system in increments/parts. Each increment must be accepted by users. This requires 'more frequent' participation of users. (3 marks)

### **Examiner's Guidance Notes**

Most candidates answered part a) reasonably well and discussed why training would be required. However, some candidates discussed potential training approaches rather than why the training would be required.

Most candidates answered part b) reasonably well and appropriately discussed the likely impact on method users and system users.

### **Evaluation and tuning of a method**

5. a) Explain the difference between validation and verification (V&V) in software projects. Suggest various V&V activities and techniques suitable for different stages of the ABC method process (see Appendix at end of paper). Your answer should include a brief justification of your 'allocation' of V&V activities/techniques to the ABC stages. (12 marks)

b) Discuss what software tools could support the quality control of design diagrams, program specifications and programs during a software development project. (13 marks)

### **Answer pointers**

5. a)

Validation: A set of activities that ensure that the phase product which has been delivered is traceable to customer/user requirements i.e. it satisfies customer/user requirements.

Verification: A set of activities that ensure that a product emerging from any phase of the development process meets its specification. (2 marks)

Requirements analysis: Mainly 'static' techniques can be used such as reviews, inspections, etc. It is also possible to use dynamic validation techniques such as prototyping. (2 marks)

Design: mainly static techniques such as reviews. It is also possible to use dynamic techniques (and tools) such as design prototypes, design simulators, etc. (3 marks)

Implementation and Testing 1: primarily 'dynamic' techniques such as testing. Possible code inspection and walkthroughs. (3 marks)

Integration and Testing 2:

Testing 1 primarily involves module testing techniques.

Testing 2 involves integration tests and acceptance testing. (2 marks)

5. b)

Diagramming tools could be used to create and update relevant design diagrams to a professional standard. (4 marks)

CASE tools could assist in creating and updating design diagrams and program specification and could analyse code for complexity and modularity to a professional standard.

(5 marks)

Debugging tools could assist in developing program code to a professional standard. Some programming tools could assist in structuring and formatting program code to a professional standard.

(4 marks)

**Examiner's Guidance Notes**

Most candidates answered part a) reasonably well and allocated appropriate verification and validation activities / techniques to the different stages of the systems design method.

Most candidates answered part b) reasonably well and appropriately discussed how the stated software tools could support quality control. However, some candidates discussed different software tools from those stated in the question.