THE BCS PROFESSIONAL EXAMINATIONS
Professional Graduate Diploma

April 2007

EXAMINERS' REPORT

System Design Methods

Please note that Q1c, Q2b, Q3c, Q4c and Q5b refer to a case study method briefly described in the Appendix at the end of the paper.

Question 1

1. a) Explain the following terms related to formal methods:
   (i) formal specification,
   (ii) formal verification, and
   (iii) formal development. **(6 marks)**

b) In the 1980s, many researchers predicted that by the 21st century, a large proportion of software would be developed using formal methods. There are a number of reasons why this prediction was wrong. Give at least four such reasons. **(8 marks)**

c) (i) Consider the following life cycle models: waterfall, prototyping, incremental and spiral. How would you classify the SP method process (see Appendix at end of paper)? Justify your answer. **(6 marks)**

(ii) Suppose that you want to 'incorporate' formal specification into the SP method process. Explain how you would modify this process. Justify your answer. **(5 marks)**

Answer pointers

a)

(i) Formal specification – a specification expressed in a language whose vocabulary, syntax and semantics are formally defined. Typically specification languages are based on mathematical concepts (such as set theory, logic, etc) (2 marks)

(ii) Formal verification – verification using mathematical arguments, that the code is consistent with its specification (which must be formal). (2 marks)

(iii) Formal development – a transformational development process where a formal specification is transformed through a series of more detailed representations into the code. (2 marks)

b)
Reasons:

- The use of other software engineering methods and techniques (structured, object oriented etc.) have resulted in significant improvements in software quality.
- In the 80s , software quality was the key issue. However now the critical issue for many classes of software development is time. Software must be developed quickly.
- Formal methods are not well suited to specify user interfaces which now constitute a substantial part of software systems.
- Limited scalability. Formal methods do not scale up well. They were really successful for small systems (or small parts of bigger systems).

2 marks for each reason i.e. (8 marks)

c)

(i) The SP process is based on the incremental approach with prototyping. The main iteration/loop delivers increments. The 'small' loop involves prototyping (to elicit and /or validate requirements). (6 marks)

(ii) Normally a formal specification is developed after the system requirements have been specified, but before the detailed system design. Therefore it should be part of "Design increment" stage of the SP method process. (5 marks)

Examiner's Guidance Notes

For part (a) some candidates confused the three terms.

For part (b) most candidates could identify the main reasons.

For part (c) (i) some candidates attempted to demonstrate the 'presence' of all four life cycle models in the SP method process.

For part (c) (ii) a lot of candidates suggested that formal specifications could be used during the early stages e.g. Analysis and even Feasibility Study.

Question 2

2. a) What are the main characteristics/principles of agile methods? **(5 marks)**

b) SP method (see Appendix at end of paper) specifies the development process, but it does not force a method 'user' (i.e. developer) to use a prescribed set of systems modelling techniques. Basically developers who decide to follow the SP process can use, for example, the set of structured modelling techniques, or the set of object-oriented techniques (for example, UML).

Assuming that you are required to use the SP process in your projects decide:
(i) Which structured modelling techniques you would use in different stages of the SP process, **(10 marks)**

(ii) Which UML techniques you would use in different stages of the SP process. **(10 marks)**

Briefly justify your decisions.

Answer pointers

a)

The main characteristics/principles of agile methods:

- Customer/client involvement – customers should be closely involved throughout the development process
- Incremental delivery
- People not process – team members should be left to develop and use their own ways of working without prescriptive processes
- Embrace change – expect the system requirements to change, so design the system to accommodate these changes
- Maintain simplicity – focus on simplicity of both the software being developed and the development process

1 mark for each i.e. (5 marks)

b)

(i) Possible ‘allocation’ of structured techniques (e.g. techniques used in SSADM v4) to the SP stages:
Feasibility study: Context DFD, initial ERD
Analysis: DFDs, detailed ERD, Function definitions, ELHs, ECDs/EAPs
Design increment: UPMs/EPMs
(10 marks)

(ii) Possible ‘allocation’ of UML techniques to the SP stages:
Feasibility study: Use case diagram (no details), high-level class diagram
Analysis: Use case diagram (detailed), class diagram (all attributes , operations with responsibilities defined), collaboration and sequence diagrams for key use cases, possibly state diagrams (high-level)
Design increment: Fully developed class diagram, collaboration and sequence diagrams, state diagrams, component diagrams, deployment diagrams
(10 marks)

Examiner’s Guidance Notes

For part (a) most candidates could identify the main characteristics of agile methods.

For part (b) many candidates described modelling techniques in great detail (which was not required).
Many candidates concentrated only on a few modelling techniques such as DFDs and ER diagrams (structured) and Use Case diagrams and Class diagrams (UML) ignoring other important techniques.
Many candidates clearly indicated which techniques are suitable in different stages of the SP method. Some however did not indicate this clearly, and a few did not mention the SP stages at all!

Question 3

3. a) Compare and contrast the life cycle coverage of social-technical, engineering based, and formal methods. **(9 marks)**

b) You are in charge of five software development projects. The ‘characteristics’ of each of your projects are as follows:

- Project 1. Web-site for a local company. Relatively small system. Requirements are vague and likely to change in the near future.
- Project 2. A very large embedded system whose requirements can be easily identified and are relatively stable.
- Project 3. A ‘standard’ business application. You have developed similar systems in the past.
- Project 4. A relatively complex administrative system for one of the local hospitals. Some of the requirements seem to be pretty vague, but all the requirements are stable.
- Project 5. A small real-time control system to be used for monitoring patients in a local hospital.

Consider the following software development approaches/models:
waterfall, throw-away prototyping, evolutionary prototyping,
component-based development, formal development.

Which of the above approaches/models would you choose for each of your projects? Justify your choices. **(10 marks)**

c) Identify at least three ‘characteristics’ of applications/projects for which the SP method (see Appendix) would be a suitable choice. **(6 marks)**

Answer pointers

a) Socio-technical methods cover mainly the initial systems development life cycle activities of problem definition, feasibility study and analysis, with limited design coverage. (3 marks)

Engineering-based methods cover the majority of the systems development lifecycle from problem definition to physical systems design. However, they do not typically cover coding, physical implementation and maintenance. (3 marks)

Formal mathematical based methods tend to concentrate on the analysis and design phases of the systems development life cycle, along with some support for coding and testing. (3 marks)

b) Project 1: Evolutionary prototyping (2 marks)

Project 2: Waterfall (2 marks)

Project 3: Component-based development (2 marks)

Project 4: Throw-away prototyping (during the Requirements Analysis stage) followed by Waterfall (2 marks)

Project 5: Waterfall with some ‘critical parts to be developed using formal approach. (2 marks)

c) Characteristics:

Some requirements are vague (prototyping should be used to clarify them) (2 marks)

Requirements can be divided into different priority groups – increments (2 marks)

Business application (2 marks)

Examiner’s Guidance Notes

For part (a) some candidates were unclear as to the difference between a systems development life cycle and a methodology.

For part (b) most candidates could explain which approach to use.

For part (c) most candidates could identify the relevant characteristics

Question 4

4. a) Explain how to undertake reverse engineering. **(10 marks)**

b) Outline three ways in which IT staff could be trained in reverse engineering. **(6 marks)**

c) Consider the following re-engineering projects you are involved in:

- Project 1. Reverse engineering to 'recover' lost design documentation.
- Project 2. Re-engineering to restructure the entire system.
- Project 3. Re-engineering to restructure the entire system and to add some 'new' user requirements.
- Project 4. Re-implementation in a different programming language.

Which stages of the SP method (see Appendix) would you use in each of these projects? Justify your answers. **(9 marks)**

Answer pointers

a) The process of reverse engineering might typically involve the following:

Identifying the inputs to the system such as data entry fields on screens and possibly other input sources such as bar code readers. (2 marks)

Identifying the outputs from the system such as report screens, printed reports and possibly other forms of output such as graphics and charts. (2 marks)

Examination of the source coding for the system to determine the structure and detail of the coding. This is necessary in order to understand how the coding of the system is structured and how it works. (2 marks)

Examination of the documentation for the system. This can be beneficial for attempting to understand the design of the system. (2 marks)

Establishment and documentation of the modules within the system, the connections between modules, the main functions of individual modules and the data stores used based upon the above. (2 marks)

b) One approach would be to develop a training course for the reverse engineering process (or use a suitable commercial reverse engineering training course if one was available). (2 marks)

Use a hands-on training approach, whereby the IT staff will learn how to use the reverse engineering process while they are working on reverse engineering projects. (2 marks)

Develop a computer based training (CBT) package to train the IT staff in the reverse engineering process (or purchase a suitable commercial reverse engineering CBT package, if one was available). (2 marks)

c) Project 1. None, as the SP process does not address reverse engineering. (2 marks)

Project 2. This will involve re-designing and re-implementation, therefore Design increment and Build increment definitely will be used. (2 marks)

Project 3. This will involve re-designing and re-implementation (as Project 2) and some elements of requirements analysis, therefore Analysis possibly with Prototyping will also be used (in addition to Design and Build) (3 marks)

Project 4. Re-implementation only, therefore Build stage only. (2 marks)

Examiner's Guidance Notes

For part (a) most candidates could explain how to undertake reverse engineering.

For part (b) most candidates could describe different appropriate training approaches.

For part (c) some candidates appeared to find difficulty applying the available techniques to the projects described.

Question 5

5. a) You are an IT consultant who has been hired to advise upon the development of a very large scale IT system for a government department. Outline and critically evaluate three quality assurance approaches that could be used to improve the quality of the design process for such a large systems development project. **(12 marks)**

b) Explain the difference between validation and verification (V&V) in software projects. Suggest various V&V activities/techniques suitable for different stages of the SP process (see Appendix). Your answer should include a brief justification of your 'allocation' of V&V activities/techniques to the SP stages. **(13 marks)**

Answer pointers

a) Reviews, which involve supervisors or managers formally assessing the quality of design artefacts, produced in order to identify errors in the designs. However, staff may feel intimidated by this since the results of reviews might be used for deciding pay rises and promotions. Staff may therefore be reluctant to admit to errors and make corrections. (4 marks)

Walkthroughs involve a group of peers informally examining design documents in order to identify and discuss possible improvements. Hopefully this should be an egoless environment since the staff involved will be at the same level within the organisation, and staff may be more willing to admit to errors and make corrections. (4 marks)

Quality circles involve a group of peers discussing mechanisms for improving the quality of designs, for example developing more rigorous design standards or adjusting the design techniques used. This can be used for on-going improvement of the systems design process, rather than being a mechanism for identifying improvements to individual designs. (4 marks)

b) 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. (2 marks)

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

Feasibility Study: Review of Feasibility document (1 mark)

Analysis: Review of Requirements document (1 mark)

Prototype: Acceptance testing (2 marks)

Design increment: Walkthrough or/and Review (2 marks)

Build: Various tests including regression testing, possibly code inspections (2 marks)

User acceptance: Acceptance testing (1 mark)

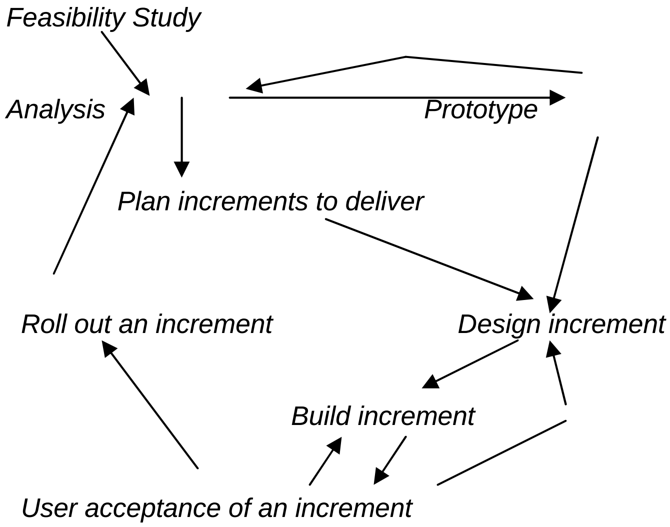
Examiner's Guidance Notes

For part (a) most candidates discussed appropriate quality assurance approaches.

For part (b) most candidates could explain the difference between validation and verification, but found some difficulty in selecting techniques for the stages of the given method.

APPENDIX

Consider the following software development method. The method is called SP and its process and stages are detailed below.
The method includes the following stages: Feasibility Study, Analysis, Prototype, Plan Increments to Deliver, Design Increment, Build Increment, User Acceptance of an Increment, Roll out of an Increment.
The development process is as follows (see Figure below).



The stages are briefly described below:

Feasibility Study. Scope the development in terms of proposed solutions and produce both a business case and first-cut project plan. Find out who/what the system will interact with.
Analysis. Explore the requirements for the software to such a degree that the scope is confirmed and that sufficient information is provided to allow an incremental delivery plan to be produced. A closed iterative loop with the prototype stage exists as a means of eliciting requirements.
Prototype. Elicit requirements through the construction of prototypes (screens). Some prototypes may evolve to a working increment.
Plan Increment. Develop a plan for producing user services in terms of a set of incremental deliveries.
Design increment. Develop increment design specification.
Build increment. Implement increment – this involves coding, assembling and testing the software comprising a particular increment. Regression testing of previous increments must also be taken into account.
User acceptance of an increment. *Ensure acceptance of an increment before it is 'rolled out' i.e. installed in the live environment.*
Roll out. *Install an increment in the live environment.*