THE BRITISH COMPUTER SOCIETY

THE BCS PROFESSIONAL EXAMINATIONS
BCS Level 6 Professional Graduate Diploma in IT

SYSTEMS DESIGN METHODS

25th April 2007, 10.00 a.m.-1.00 p.m.
Answer THREE questions out of FIVE. All questions carry equal marks.
Time: THREE hours.

*The marks given in brackets are **indicative** of the weight given to each part of the question.*

Calculators are NOT allowed in this examination.

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

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)**

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.

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 SPmethod would be a suitable choice. (See **Appendix** at end of paper). **(6 marks)**

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 would you use in each of these projects? (See **Appendix** at end of paper). Justify your answers. **(9 marks)**

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)**

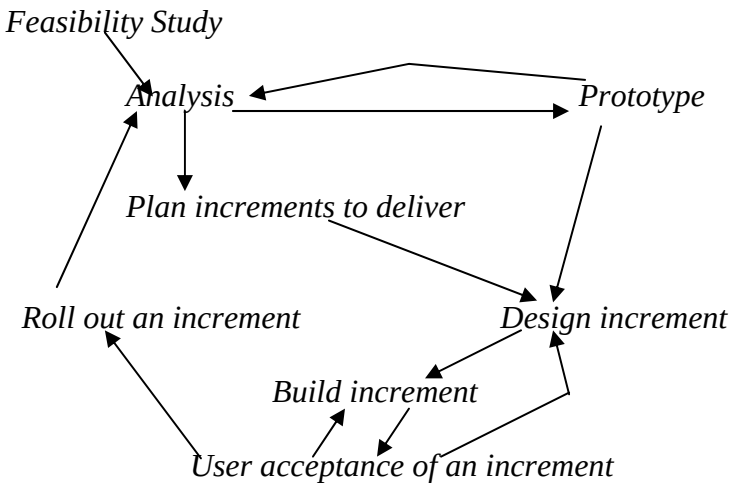
b) Explain the difference between validation and verification (V&V) in software projects. Suggest various V&V activities/techniques suitable for different stages of the SP process. (See **Appendix** at end of paper).

Your answer should include a brief justification of your 'allocation' of V&V activities/techniques to the SP stages. **(13 marks)**

APPENDIX (for use in Questions 1c, 2b, 3c, 4c and 5b)

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.