THE BCS PROFESSIONAL EXAMINATION
Professional Graduate Diploma

April 2004

EXAMINERS' REPORT

Software Engineering

General

The scripts produced by candidates this year have improved both in the general standard of the written work, and the effort made to answer the questions set and apply knowledge in the context of the problem situation.

The examiners actively sought to set honours-level questions, and sought honours-level answers. These points are expanded upon below.

The application scenarios presented in the questions were familiar to very many candidates. However, the theoretical underpinnings that the questions attempted to draw on (analysis, design, prototyping, software modelling) varied greatly between centres. It is important that candidates study and consult the recommended text, and other relevant material.

This year there seemed to be a significant minority answering more questions than was required in the examination, and thus reduced the time for each question and limited their opportunity to obtain the full range of marks for the questions attempted.

Finally, on some scripts, student had not completed the front cover with the number of the question and part-questions attempted. This made it very difficult to find, then determine whether it was a continuation of a previous question, and assess and record the mark for that question as a whole.

Question 1

a) Give three reasons for using the black-box style of testing. (8 marks)

b) The following is a code piece from the abstract definition of a table with a maximum size, and methods for insertion and reading of a value in the table, and a function for obtaining the size of the table.

```
module  TABLE;
exports
        type Table-Type(max_size: NATURAL);
        no more than max_size entries may be stored in a table;
        procedure Insert (Table: in out Table-Type; ELEMENT: in ElementType);
        procedure Read (Table: in out Table-Type; ELEMENT: in ElementType);
        function Size(Table: in Table-Type) return NATURAL;
        provides the current size of the table
        .
        .
        .
end  TABLE
```

With reference to this abstract definition, illustrate how you would design a testing scheme based on black box testing. Include specification of the test cases you need to use. (17 marks)

Answer Pointers

a) A good answer should concentrate on higher-levels of testing, such as integration and validation testing where it is more important to identify functional errors of inter-operation than simple logic (white-box) errors. In addition, where software generators are used, there is little point in testing the generator for logic errors; so functional (or black-box) testing is appropriate.

Three descriptions, 8 marks. Two, 4 marks. One, 2 marks.

b) A good answer should recognize that the three numeric structures had legal ranges. For example,

Max-size , a positive range with value less than or equal to some number.
Table, the location of the table, a valid address-handle
Element, a range of values of ElementType

2 marks for each structure that is identified, and 1 mark for each outline test description, max. 9 marks.

In order to specify the test cases for each input domain, a black-box test should investigate each expected legal and illegal value. For example,

Max size: 3 cases. a) a negative number. b) a number greater than some maximum, say, 200. c) a valid number.

1 mark for each test case, max. 3 marks.

Table: 2 cases. a) a handle to a data area of Table-Type. b) a handle to a data area of some different type. (following rules for a set, see Pressman Ch.18).

1 mark for each test case, max. 2 marks.

Element: 3 cases. a) a value less than the range of ElementType. b) a value greater than the range of ElementType. c) a value within the range of ElementType.

1 mark for each test case, max. 3 marks.

Examiner's Guidance Notes

This was reasonably popular, perhaps because the introduction question was quite simple. However, the main aim of the question was not to elicit a description of testing, but the knowledge of how to create a test plan. Nevertheless, some answers to 'why use black-box testing' asserted that it was cheaper than 'white box' and so was to be preferred. For part (b), some candidates insisted on answering with a description of black box testing, and paid no attention to the specific detail given in the question. Several answers correctly identified that data ranges were to be designed as possible valid and invalid test cases.

Question 2

The following is an outline specification for a development project to create a web-site.

“ECICE - European Chemical Industries Centre of Excellence - is a technology transfer institute that aims to improve industrial practices in process industries. This project is about developing an information-rich web presence for ECICE. In particular, research should be made into ECICE customer behaviour, to determine why they value the services offered by ECICE. These values should be the key criteria to guide the creation of ECICE 's web image.”

a) Discuss the criteria you would use to determine the life cycle model that the development of this project should follow, and make a recommendation with reasons about selecting a suitable life cycle model. (10 marks)

b) Designers often make website functionality to meet the needs of potential customers. Identify, with reasons, three examples of functionality that would assist good communication between the ECICE website and its potential customers. (5 marks)

c) Hence or otherwise derive TWO quality criteria that should be monitored closely during the development of this project. (10 marks)

Answer Pointers

a) A good answer should draw on the detail in the question to identify suitable criteria. Such a good answer should create a balanced argument, choosing some suitable criteria and analysing the context of the question against them, then identifying issue(s) of concern, then identifying a life cycle that has an ability to cope with these issues.

It is important to stress that a good answer draws on reasons from the scenario to create a justification. There is no ‘perfect’ answer. There are several, well-reasoned, answers

A typical answer might consider the scenario given, and identify criteria that requirements are not stable (the job to do is vague); but the tools available are productive. The vague work consists of the look-and-feel of the website, and the range of services it offers. An issue that follows from these (or similar) deductions is that such HCIs are hard to get right. The issue for management is getting user involvement in approval of the chosen web designs. The question becomes “which lifecycle model is good for getting the users involved?” A series of exploratory design-build-test episodes, such as evolutionary prototyping, is an appropriate conclusion.

Max. 4 marks per criterion with rationale drawn from scenario. Max 2 marks per criterion with rationale but ignoring the scenario. Max 1 mark per criterion without any rationale.

Max. 4 marks for life cycle model that is clearly linked to identified criteria. Max 1 mark for any recommendation without clear linkage to any criteria in the scenario.

b) A good answer should again reflect on the detail of the question, and justify the planned functions of the website as logical consequences of detail in the question.

For example, ECICE claims to be a centre of excellence, to persuade businesses that it has better ways for them to operate. The website should therefore be supported by information and advice, such as downloads of strategy/research papers (showing clear industry relevance), profiles of consultants used (showing clear industry relevance), and other value-tagged documents.

Max. 5 marks for three examples of functionality well reasoned and in context.

c) A good answer should take functions already identified as important, and place these desirable functions at the heart of the quality system. Criteria, well argued and clearly linked to the question detail, were very acceptable as answers.

For example, functions arguably important from the model answers already given indicate that look-and-feel, and substance of downloads, are vital to get right.

Max. 5 marks for each criterion, well reasoned and placed firmly in context. Max. 1 mark per criterion for no reasons or linkage with scenario and previous analysis.

Examiner's Guidance Notes

This question came from a real scenario, and sought answers with insight as well as constructive analysis and design. This was a very popular question, again because it seemed more intuitive than difficult. However, very few answers tackled the question with any analysis; too many answers gave an answer without any supporting argument derived from the detail in the question. Even so, many candidates identified the importance of surveying users to gain identification of requirements.

Many candidates extended the requirements gathering phase of development into the life of the website, identifying suitable functionality as gathering preferences from customers. This development-specific activity should not be confused with operational functionality of the website.

Many candidates were not aware that a quality plan is created around targets that the development should achieve. Too many candidates described quality assurance and quality control processes without making a link to any ultimate objective of such activities.

Question 3

a) Compare and contrast the methods of software project estimating known as size-related estimates and function-related estimates. (6 marks)

b) In your view, what key practice should underpin any use of estimating systems for the planning of software development projects. (4 marks)

c) A software company has asked you to create a process improvement programme for them.

i) State THREE categories of process classification you would use. (5 marks)

ii) State THREE types of process metrics you would use. (5 marks)

iii) Discuss TWO critical success factors that will determine the outcome of this programme, with your reasons. (5 marks)

Answer Pointers

a) A good answer should describe Size-related measures as most commonly measures of Lines of Delivered Source Code (LOC). Other acceptable measures are pages of documentation, thickness of documentation, or number of object-code instructions.

Function-related measures relate to overall functionality of delivered software. Function points are the best known of this type.

The answer should go on to 'compare and contrast' these two systems. For example, estimating a size measure at the start of a project is usually by comparison with other, similar completed projects. The difficulty is finding something already completed to act as a guide.

Estimating this measure at the start of a project may be by counting features in the requirements definition, or waiting until a high-level design exists then counting the number of design elements that need to be produced as code.

The difficulty is knowing the rate of development by your own people with new tools, or the rate of development by new people with old tools, or similar combinations of uncertainty.

Max. 6 marks for comparing and contrasting the two systems. Max 2 marks of only one system is described.

b)
A good answer should recognise that the key practice to underpin any estimating system is measurement of own organisation performance in order to calibrate rates of development. All other estimates are guesses, to a greater or lesser extent.

4 marks for identifying measurement as key.

c)
i. A good answer should derive three types of process classification. For example, one way of tackling this is to choose any three of classic quality-cycle processes of plan-do-check-act:
Planning - estimating activity durations.
Doing – software lifecycle development activities (requirements capture, design, code)
Checking – test case design, test case generation, testing.
Acting – process- and people-intensive activities such as reviews and inspections.

Appropriate alternative schemes were accepted if the process classes were clearly ‘different’.

Max. 5 marks for three clear types, 2 for two and 1 for one.

ii. Following on from the classifications of process, a good answer should identify units of measure for these processes.

For example; durations and time measures through the life cycle; effort as in person-days for production of deliverables through the life cycle; and frequency counts of events such as defects discovered or change requests made.
Other metric systems were acceptable, where linked with reasons to process classes already described.

5 marks for three, 3 marks for two, 1 mark for one.

iii.. A good answer should identify tests for real improvement, such as

- increased profitability (or reduction of development costs) because of increased efficiency, and
- reduced number of customer complaints or contract deviations or quality exceptions because of fewer defects.

Other, well-founded criteria were acceptable.

5 marks for two with reasons, 2 marks for one with reasons, 1 mark for an answer without clear rationale.

Examiner’s Guidance Notes
This question was again set to seek answers that exhibited analysis and design rather than rote learning. Most candidates successfully distinguished between size-related estimates and function-related estimates.

The area of the question was process improvement, and many candidates confused ‘process’ with ‘development process’, and therefore identified steps in a life cycle model as classifications of process to improve. In reality, all stages of any lifecycle model are one class of process, development processes.

Although many candidates drew a link between process improvement and the Capability Maturity Model, very few indeed realised that the precursor to ‘process focus’ is measurement, or the simple fact that before anything can be improved there must exist the capability of doing it twice the same way.

‘Critical success factors’ were consistently underplayed. Very few candidates described the cost-savings and money profit that can follow from ‘getting it right’. Other candidates described lower-level ‘outcomes’ such as reduced error rates, or avoidance of rework.

Question 4

As chief software architect, you have been commissioned to develop an integrated web-enabled bill payment system for a private telecommunications company. The timetable for implementation is short and some concerns were raised by your client about the lack of benefits in deriving a software model before any system development can take place.

Write a detailed report that aims to assure the client of the benefits of constructing a prototype. The report should also provide details of the approach, methods and tools you intend to adopt in the construction of the first bill payment prototype. (25 marks)

Answer Pointers

This question assesses the candidate’s knowledge of the benefits of prototyping, and general awareness of the range of methods and tools to be used in the construction of the prototype.

A good answer should:

a) - be written and structured in the form of a report with appropriate titles and section headings.
- explain the purpose of a prototype and justify its use in this context such as the long-term cost of incorrect or incomplete user requirements.

(8 Marks)

b) Identify and define the possible prototyping approach available and select and justify the approach that will be adopted for this project. For example, approaches include throwaway and evolutionary prototyping. Further, the selection and justification of an approach will depend on whether additional preliminary work is required (unlikely to please the client), and the basic system requirements are stable and well understood.

(9 Marks)

c) Identify the current methods and tools available and discuss how they might be deployed in the project. Methods and tools include fourth generation languages for report generation, screen designs, reusable software components, and formal specification and prototyping environments.

(8 Marks)

Examiner’s Guidance Notes

This was quite a popular question with some very good answers that focused on the specific user requirements. It is particularly important to structure the answer in the form of a report (headings, section title, and numbering), attempt to address the clients concerns, show some understanding of bill payment, and demonstrate your knowledge of prototyping methods and how the method selected might be applied to the application scenario outlined.

The relatively weak answers either focussed on web technology alone, or process models, without making any particular reference to the clients concern or the application requirements. These answers were often very descriptive in nature.

Question 5

A local bank has given you the requirements specification for a system to manage customer accounts. In particular, the specification identifies the data, the processes, and the user interface expected. The data include name, address, current balance, and the transaction type, amount, and date. Likewise, the processes include the creation, update, and deletion of customer accounts, and deposit and withdrawal transactions keyed in at a terminal.

Based on the information given, use an appropriate design method to define and develop design outlines for the following deliverables:

i) Data design
ii) architectural design
iii) interface design
iv) component-level design. (25 marks)

Answer Pointers

This question assesses the candidate's understanding of aspects of design through the selection and application of a design method to the problem outline given.

A good answer should:

- present a clear definition of the overall design philosophy being used, be it bottom-up or top-down, or process-driven or data-driven.
  (3 marks)
- Provide a definition and description of:
  Data design: transform information about system requirements into suitable data structures
  (3 marks)
  Architectural design: represents the interrelationships between sub-systems elicited from the system specification;
  (3 marks)
  Interface design: represents the communication pattern between the systems, sub-systems and their users;
  (3 marks)
  Component-level design: transform sub-system elements into procedural descriptions.
  (3 marks)

  In terms of developing design outlines for the account management system, three or more of the following diagrams as appropriate might be used to represent the various design deliverables:

  Class Diagrams; ERD diagrams; Data Flow diagrams; State Transition diagrams; Entity Life Histories; Sequence Diagrams; Structure Charts, Process Specs, etc. The design deliverables are:

  Identify and use appropriate notation for data representation for the entities Customer Account, and Transaction, and the attributes of each (name, address, balance, transaction amount etc)
  (4 marks)

  Identify and represent the processes and their relationship in time, message/data exchanged, and hierarchy.

(6 marks)

Examiner's Guidance Notes

This question wasn't the most popular, but produced a wide range of differing answers. Some answers emphasised the design of input and output screens, others developed process designs to varying levels of detail, using the notations of methodologies such as OOD, SSADM, UML, and YSM. Those candidates that developed extensive screen designs did not appear to understand or recognise the relationship between interface design as an aspect of modular design and the other deliverables listed in the question.

The work submitted was very good in some cases, but candidates should ensure that they distribute the allocated time equally across the four aspect of design highlighted in the question. Thus, many marks were lost by only addressing one or two deliverables.