THE BCS PROFESSIONAL EXAMINATION
Professional Graduate Diploma

April 2003

EXAMINERS' REPORT

Software Engineering

General

The general readability of student scripts continues to improve. However, grammar and the volume and organisation of answers in some centres remain problematic. There were many instances in which answers to questions were buried in unnecessary and irrelevant topic details. Answers should address the question set, and be structured in a manner that gives a clear focus and a strong sense of fluency, intellectual engagement, discussion, or debate.

Unfortunately, some candidates did not appear to read in order to understand the whole question, but rather latched on to one or two keywords and produced a memory dump of everything they knew about the topic. This resulted in pages upon pages of scripts of descriptive material, with very little application to the scenario outlined.

Question 1

1. *a)* Your software manager has asked for your advice. The cost of your own in-house development averages £360 per day. Your manager is considering the purchase of a software package that is 5KLOC (five thousand lines of code) that will cost £50,000. Initial evaluation of the package indicates it will need a tailored interface to suit your company; this will involve extra code that is 1KLOC (one thousand lines of code). A complicating factor is that the supplier has offered to do the extra work for a total cost including the initial package of £65,000.

Use the parametric equations as shown in **Figure 1** below to estimate the software development cost of three alternative courses of action.

- buying the total package;
- buying the basic package for £50,000 and developing the new functionality in-house, where the in-house development style will need to be SEMI-DETACHED;
- developing the whole functionality in-house without reference to any external supplier. In this case, the development style will be ORGANIC.

(20 marks)

b) Give ONE or TWO risks that could be assigned to each course of action. In your view, does this risk appraisal significantly alter the estimates of cost? Include your reasons.

(5 marks)

Basic Parametric Model for Estimation of Effort and Duration

This basic estimator is a management model that computes effort and duration of a software development as a function of program size estimated in thousands of lines of code (KLOC). Duration is converted to cost using agreed conversion measures.

Effort = a $(\text{KLOC})^{b}$ in units of person-months,
and Duration = c $(\text{Effort})^{d}$ in terms of elapsed months.

Parameters a, b, c and d are given in the table below:

The Project	**a**	**b**	**c**	**d**
ORGANIC	2.4	1.05	2.5	0.38
SEMI-DETACHED	3.0	1.12	2.5	0.35
EMBEDDED	3.6	1.20	2.5	0.32

IF you do not have access to a calculator, use the "**Tables of Exponent Values for Basic Estimator Calculations"** *in the Appendix at the end of this question paper.* ***For fractions, interpolate between whole numbers.***

Common values for estimating time:
number of working days in a year (excl. holidays, sickness, etc.) = 220
number of working days in a month (excl. weekends, holidays etc.) = 20

Figure 1

Answer Pointers

This question attempts to test a candidate's ability to use quantitative estimating techniques.

A good answer should:
Identify cost of purchase with changes as given in question. £65,000

Calculate cost of basic purchase plus in-house changes, using formulae given. Effort of 3 person-months is computed into elapsed days charged at £360 per day for a cost of approx. £76,500 (Approximate costs are used here because not all candidates used exact arithmetic, and arithmetic as such was not the objective of this question.) (10 Marks)

Calculate the cost of completely in-house development, using formulae given. The product size could be 5KLOC or 6KLOC, either is acceptable here. The cost is calculated to be about £48,000. (10 Marks)

Give a balanced discussion of one or two risks from, for example,

- instability of user requirement if build was in-house because of 'gold-plating' - incremental adding-on of functionality, even though this appeared to be cheapest option.
- loss of developer-income if staff are taken from earning to in-house development.
- difficulty of interoperability/integration if in-house team attempt to graft an interface onto a bespoke bought-in product.
- training and acclimatisation difficulties if product is completely outsourced (though this could be mitigated by training). (5 Marks)

Examiner’s Guidance Notes

The question was quite popular; many candidates seemed to avoid it without reason because those that attempted it mostly received high grades. The method was given in the question, but even so a few candidates managed to provide their own approach as if they had not read the question. There were few marks for deviating from the method in the question. It is not difficult to apply the formula, and the test of correct application lies in using the correct parameters. Overwhelmingly, candidates did so, following the rubric of the question.

Many candidates identified mild risks that could be negotiated (contractor might deliver late – put a lateness penalty clause in the contract) or easily mitigated (contractor might go out of business and fail to maintain – use escrow arrangements).

Question 2

2. *a)* Describe the differences between Quality Assurance (QA) and Quality Control (QC) with reference to a software development project. **(6 marks)**

b) Some writers about software development methods claim that "Quality is free". Do you agree with this? Give your reasons. **(12 marks)**

c) How would you measure the effectiveness of QA and QC on a software development project? Give reasons for your answers. **(7 marks)**

Answer Pointers

This question attempts to assess a candidate’s practical understanding of application software design and implementation principles.

A good answer should:

Distinguish between Quality Assurance as a plan of quality criteria, activities, personnel and Quality Control events that will be executed to assure that the product conforms to contractual quality criteria. (6 Marks)

Make a balanced, reasoned discussion on the thesis “*quality is free”.* Such discussion should identify the costs of quality and debate whether the benefits return more than the cost, or not.

For example, consider that 30% of project development is testing, and that a software product is under maintenance for much longer than it is under development. Efforts during development to limit the cost of testing and the cost of maintenance might well be cheaper in the long run. The alternative, cut down on testing and leave maintenance to others, can create disappointed customers and loss of business, even legal disputes. It seems that a very good case can be made for the argument that quality during software development can avoid expensive problems during software operation, maintenance and eventual retirement. In that way, the quality becomes an investment that returns far more than it costs, and so appears to be *free*. (12 Marks)

Describe how effectiveness of Quality Assurance could be measured by compliance with IEE or ISO standards via the quality audit process.

Identify suitable development metrics such as error rates, amount of rework, and numbers of bugs founding order to measure the result of effective Quality Control. (7 Marks)

Examiner's Guidance Notes

a) This question was very popular. Most candidates correctly identified the 'promise' nature of QA and the 'promise delivery' nature of QC.

However, many completely missed the way that the QA promise is tailored around the needs of the product. They asserted that 'top notch quality' is always applied, regardless.

Some few candidates evinced broad enthusiasm for quality without successfully distinguishing between QA and QC. This earned very low marks.

b) Most responses to the invitation 'discuss *quality is free*' were poor. Very many denied the thesis (*quality is free*) and proceeded to describe a quality utopia of managed design process with checks at every stage without ever identifying why this panoply of control was held to be necessary.

Some candidates started by decrying the thesis and discovered as they wrote that the lifecycle framework of controlled development was explained by management attempts to prevent poor market image and product refusal by customer. However, only one or two candidates in the whole exam correctly addressed the whole lifecycle costs of software, to recognise that reduction of rework during development and maintenance after installation might save more money than error prevention activities during development. Too many candidates held a rigid, developmental mind-set that refused to recognise that a software product has a life after development is completed. Marks were awarded for answers that were supported by reasons, whether they agreed or disagreed with the thesis of the question. Very many answers were simple lists of assertions without rationale.

c) Most candidates commented that effectiveness of QA could be measured by compliance with IEE or ISO standards via the quality audit process. Fewer had such a succinct view of Quality Control.

Many candidates identified product-related metrics (Mean time between failure, Mean time to repair) as effectiveness tests, but this overlooked the context given in the question, to consider 'on a software development project'. These contribution attracted few marks.

Some candidates omitted one or other of the QA/QC pair. This lost half the marks immediately.

Question 3

3. The management team of a local college has given you a 12-month contract to develop software for a very powerful computer workstation donated by a local electronics company. A wide range of system software is available including operating systems, compilers, and program debuggers. The company has also supplied an integrated CASE tool, but no application software. In appointing you, the College Principal is particularly interested in developing a Student Records System that is portable, reusable, and scaleable, but tailored to the teaching and learning environment of the college.

Identify and discuss the software engineering design and implementation principles that, when applied, produce application software that is fit for the purpose of the college. **(25 marks)**

Answer Pointers

This question attempts to assess a candidate's practical understanding of application software design and implementation principles.

A good answer should:

Demonstrate an appreciation of design approaches that address the system requirements issues of portability, reuse, and scalability. At the design stage, design reuse can be addressed, and methodology for ensuring this is important. Thus, object-oriented analysis and design could be particularly relevant;

Recognise the importance of CASE tools and the possibility for automated consistency checking of documentation and verification of a designed component's declaration and definition are important. (7 Marks)

Show awareness of problems between design and implementation divergence. In particular, the portability and reuse quality attributes mentioned by the Principal can be affected by choices made at the design stage; portability requires an implementation language governed by international standards with compilers for a variety of platforms. Reuse requires a language system that supports abstraction, modularity, independent and separate compilation, and code repositories;

Recognise the importance of coupling and cohesion in any coding exercise. (9 Marks)

However, scalability appears to be more concerned with the system and its number of users. Thus, this entails performance and efficiency factors and may conflict with the universal qualities of reuse and portability.

The answer should also address the verification and validation of the software using automated tools, potential users, peers (structured walkthroughs), professional testers, and formal and informal methods. (9 Marks)

Examiner's Guidance Notes

This was a popular question with some very good answers that focussed on specific user requirements. However, there were also poor answers provided in which one or two keywords have been identified such as life cycle, and a memory dump produced of everything known about life cycle methods and stages. This resulted in pages upon pages of scripts of descriptive material, with very little application technique or business focus.

Question 4

4. A specialist bookshop, wishing to enter the online services market, would like to develop an online ordering system within the next six months.

a) Identify and compare TWO viable but distinct process models that might be used for this particular project. **(10 marks)**

b) Select ONE of the process models identified in part *a)* and provide a complete description of its application in the development of the bookshop project. **(15 marks)**

Answer Pointers

This question assesses the candidate's awareness of the appropriateness of certain process models for applications and projects with specific requirements.

A good answer should:

a) identify two process models such as the linear sequential and "Prototyping" models and provide a brief comparison of their approach to software development. For example, the answer could emphasise such things as the latter being more user-centred and interactive than the former. (10 Marks)

b) identify the specific project environment described, the time-constraints and user requirements as key factors determining model choice before proceeding to present a critical appraisal of the various stages of the process model selected.

The time-constraint and the importance of the user-interface make end-user involvement a critical success factor. This would very much favour the prototyping approach. Thus, prototype the system requirements with user, build model, ask user to test-drive model, and revise requirements accordingly.

However, one must ensure that each revision results in either progress towards the final system or the definitive functional specification. In case of the latter, where the candidate presents a linear sequential model may be justified through the extensive use of assembling standard components during the coding phase.

In either case, sufficient time must remain to test the system performance online or beta-testing the interface with bookshop customers. (15 Marks)

Examiner's Guidance Notes

This question was answered much better by candidates compared to questions 3 and 5 with some effort made to apply concepts to the business scenario outlined. The first part of the question was answered well and candidates from some centres seem to have opted for pages and pages of stages of life cycle descriptions to the detriment of the second section that required more analytical thinking.

Question 5

5. Consider the case of a small company wishing to develop an in-house payroll system for its 100 regular contracted staff who are all paid on a daily basis. The fundamental elements of the application software will involve both data structures and algorithms. However, the application can only be designed using a data-driven or process-oriented approach.

Evaluate the merit of each approach to design, and illustrate your answer with examples of how each approach would be applied to produce a design solution for the payroll system. **(25 marks)**

Answer Pointers

This question attempts to assess the candidate's knowledge and ability to apply data-driven and process-oriented techniques to the possible solution of a business problem.

A good answer should:

a) present a clear definition of the two approaches such as data-driven design emphasise the identification and modelling of the data to be used. The processes required will be determined by the data.

Process-driven design emphasise the behavioural aspect of the required system. The data required to "fuel" that behaviour then becomes secondary.

Both approaches should deliver the product required by the user. However, data-driven approaches might be considered to be easier to maintain because data is generally stable, and any such change would be visible to the whole application. (10 Marks)

b) Give examples, using appropriate diagramming notation of applying each approach to the design of the Payroll system. Thus, a data-driven approach starts from data on staff. File and record structures would be developed, identifying individual fields and attributes.

The data, structured in a manner that use notation for repetition, sequence, and selection can then be used to provide a program skeletal. Thus, the payroll file could be an iteration of employee records, where each employee record is a sequence of attributes.

Procedural design for the payroll system involves functional decomposition – identify the functions and their relationship at an appropriate level of granularity.

Thus, the top-level function could be “Generate Payroll”, based on sub-functions “Get Employee Details”, “Compute Daily Pay”, and “Output Payslip”. (15 Marks)

Examiner’s Guidance Notes

This question was the least popular amongst candidates, as it required both knowledge of, and the ability to apply, the contrasting approaches of process-driven and data-driven methods for application development. Again, some centre candidates, with what I consider to be very "shallow" knowledge of technical development concepts, construed the question as yet another opportunity to provide pages of descriptions on the stages of the project life cycle.

Other candidates produced an object-oriented solution in terms of diagrams and notation and this was marked according to its payroll application and representational accuracy. In the case of the latter, with the growing use of OO methods as the focal point for software engineering activities, more traditional process and data-driven methods may become increasingly redundant.