

THE BCS PROFESSIONAL EXAMINATIONS
Diploma in IT

October 2006

EXAMINERS' REPORT

IT Project Management (formerly known as Project Management)

General

The number of candidates selecting this module for this sitting were similar to the number sitting in April. However the October candidates were much better prepared and the examiners were pleased to note a very significant increase in the passing rate.

An indication of the points expected by the examiners is given below, together with comments which it is hoped will assist candidates in future examinations. Any point which was valid and relevant to the question received marks. However candidates are reminded that to obtain good marks they should ensure they answer all parts of the question as set, and to carefully consider and apply their answers to any scenario which is given.

Question 1

1. BCS Holidays is an organisation that specialises in selling travel packages which consist of the flight or train to a destination and also accommodation at an hotel on arrival. This is a fiercely competitive business and good management information and strict control of costs are essential in order for the organisation to maintain its competitive position.

It has become clear that the existing computer-based booking system is no longer adequate and a decision has been made by the BCS Holidays board of directors to adopt a new more advanced booking system.

- a) Write a memorandum to the management of BCS Holidays outlining the advantages and disadvantages of acquiring an 'off-the-shelf' booking application as opposed to developing a new application in-house using its own staff. **(12 marks)**
- b) i) Explain the terms *risk exposure*, *risk reduction* and *risk mitigation/contingency*. **(5 marks)**
ii) Identify ONE risk related to the acquisition of an off-the-shelf package and ONE risk related to in-house software development. For each of the two risks, identify a risk reduction action and a risk mitigation/contingency action. **(8 marks)**

Answer Pointers

- a) 2 marks were awarded for a presentation in a satisfactory memorandum format.

The advantages of 'off-the-shelf' software included:

- Using an existing package means that there is no delay in deploying the new package
- It is possible to see the software in operation before purchase
- Avoids risks associated with software development e.g. cost overruns
- Cost is likely to be less as development costs are spread over many purchasers
- The application is likely to be reliable as it will have effectively been debugged by previous users
- The supplier can look after maintenance and upgrades

OTS disadvantages included:

- Package might not be available that meets organisation's requirements
- May be difficult to integrate standalone package with other applications belonging to the organization
- Dependency on supplier for enhancements; upgrades might not suit your needs
- No source code: so you can't enhance it yourself

- Supplier may go out of business
- The fact that competitors could acquire the same system means that there is less competitive advantage in having the system 10 marks

b)

- i) Risk exposure – an indication of the seriousness of the risk calculated by multiplying the value of the damage that a risk could incur by the probability that it will occur

Risk reduction – action taken to reduce the likelihood of the risk occurring

Risk mitigation/contingency – action taken to reduce the damage that a risk would cause when it does occur 5 marks

- ii) 1 or 2 marks were allowed for the risk, 1 mark for risk reduction and 1 mark for risk mitigation/contingency for each of the two risks

For example, inexperienced developers could cause schedule overruns on a development project; buying in experienced staff could be a risk reduction action and allowing additional time could be risk mitigation.

For OTS example, supplier goes out of business so that source code not available. Risk reduction would be to run financial checks, and risk mitigation would be to have an Escrow agreement where a copy of the code is put into the hands of a third party. 8 marks

Examiner's Comments

This was a popular question, and the marks in general tended to be good.

- a) This was usually well answered and there was evidence of good teaching and learning. However, one unproductive habit was for some candidates to make a valid point for an advantage, for example 'off-the-shelf' applications, and then repeat the same point as a disadvantage of 'in-house development'. Many candidates did not bother to use a memorandum format. In some cases to obtain full marks some explanation was needed. For example, it might be validly stated that 'off-the-shelf' solutions were likely to be more reliable, but no reason was provided as to why this should be.
- b) The definitions of the risk terms were generally rather poor and suggest that course providers need to address this topic more thoroughly. The definition of risk exposure should mention both the elements of probability and potential damage.

The examples were often generic ones, e.g. staff sickness, rather than being specifically related to off-the-shelf and in-house development. To obtain a good mark both the cause and effect involved in a risk needed to be identified, for example supplier ceasing to trade leading to users unable to get statutory changes implemented.

Question 2

2. Assume that in the BCS Holidays scenario described in Question 1, a decision was made to acquire an off-the-shelf package.
- a) Describe the activities that would be required to select and acquire the software and to set up a fully operational booking system. **(15 marks)**
 - b) Explain the activities related to the project that would take place after the implementation had been completed. **(6 marks)**
 - c) Discuss the considerations that should be taken into account when setting the date for the final cutover to the new booking system. **(4 marks)**

Answer Pointers

- a) activities might have included:
- Drawing up requirements
 - Issuing invitation to tender
 - Evaluation process
 - Contract negotiation and award
 - Possible acquisition of hardware platform
 - Acceptance testing
 - Training
 - Office procedures
 - Data transfer

Up to 2 marks were allowed for each valid activity depending on the quality of the description provided **15 marks**

- b) Activities might have included:
- Lessons learnt report – about project execution issues
 - Setting up of maintenance and support
 - Post implementation review
 - Archiving/transfer of project documentation **6 marks**
- c) Consideration needed to be given to the seasonal nature of the business, for example, switching over to the new system at a quiet time of the year thus reducing the details of existing bookings to be transferred
Staff holidays would have to be taken into account as these might normally be taken in the off-season **4 marks**

Examiner's Comments

- a) This sub-section was usually quite well answered and some candidates scored the maximum marks for the section, or were very close to it. The question referred to the activities needed for the selection and acquisition of the software and setting up of the operational system: some candidates only dealt with part of this overall process. Inexplicably a few candidates gave the stages required for writing software from scratch.
- b) 'Implementation' referred to the setting up of the *operational* system, so that this question referred to those activities that are carried out after the system had become operational. Some included pre-implementation activities such as data take-on.
- c) This was generally very poorly answered with most candidates not thinking!

Question 3

3. A project has been approved to develop an accounts application for a company that has previously outsourced this function. Three departments, called A, B and C, will require access to the new system. The detailed requirements will need to be gathered from each of these departments. Although this can be done in parallel it is assumed that this will take two weeks for departments A and B, but three weeks for department C. When all the requirements have been obtained they will be consolidated into a single requirements document, which will take two weeks to complete. The consolidated requirements will be the basis for the overall software design, which will take three weeks to complete. The ordering and subsequent delivery of the hardware platform will take eight weeks and the design of the acceptance test cases will need two weeks. Building the software according to the design will require four developers to work for a duration of eight weeks. Installation of the hardware will require a week. When all these tasks have been completed then acceptance testing is scheduled to take a further three weeks.

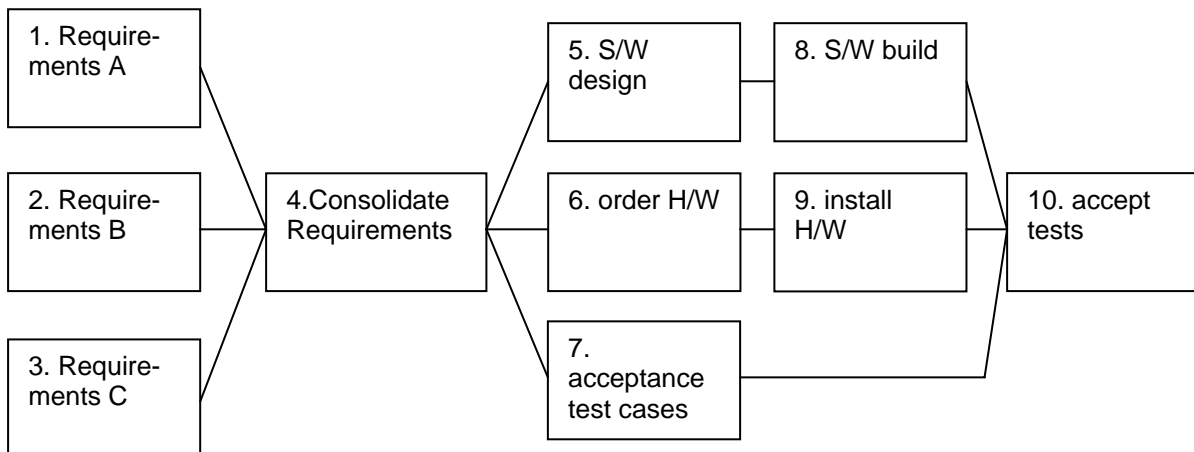
- a) Draw up an activity network for the project described above, calculating the earliest and latest start and finish dates and floats for each activity. Identify the critical path. **(15 marks)**
- b) Illustrate how the activity network can be converted into a Gantt chart and explain the advantages of this diagram over the activity network. **(10 marks)**

3(a) Answer Pointers

The question expected an Activity-on-Arrow (A-on-A) diagram similar to that below (preferably with dependency arrows) and with a key to the node notation being used

(7 marks for a good, clear, well labelled diagram).

Some slight variations in specific dependencies, if logical, were acceptable, as was a full, correct Activity-on-Arrow (A-on-A) diagram (although A-on-A diagrams are not specified in the syllabus).



A further 8 marks for a correct set of:

earliest and latest start and end dates, and float, for **each** activity;
together with a correctly identified critical path (named or, preferably, highlighted on the diagram);

provided all were consistent with the candidate's network diagram.

The correct values for the above diagram are:

activity	dura- tion	Earliest start	Earliest finish	latest finish	latest start	float
1. requirements A	2	0	2	3	1	1
2. requirements B	2	0	2	3	1	1
3. requirements C	3	0	3	3	0	0
4. consolidate requirements	2	3	5	5	3	0
5. software design	3	5	8	8	5	0
6. order h/w	8	5	13	15	7	2
7. acceptance test cases	2	5	7	16	14	9
8. software build	8	8	16	16	8	0
9. install H/W	1	13	14	16	15	2
10. acceptance tests	3	16	19	19	16	0

With Critical Path: 3-4-5-8-10

Examiner's Comments

Most candidates drew an A-on-N diagram similar to the above, but using with a wide variety of node templates (often with no clear key). The templates defined in the set books are preferable.

However a significant number of candidates were clearly unsure of the difference between A-on-N and A-on-A diagrams and drew some form of composite of the two (often with activities on the nodes but with durations on the arrows).

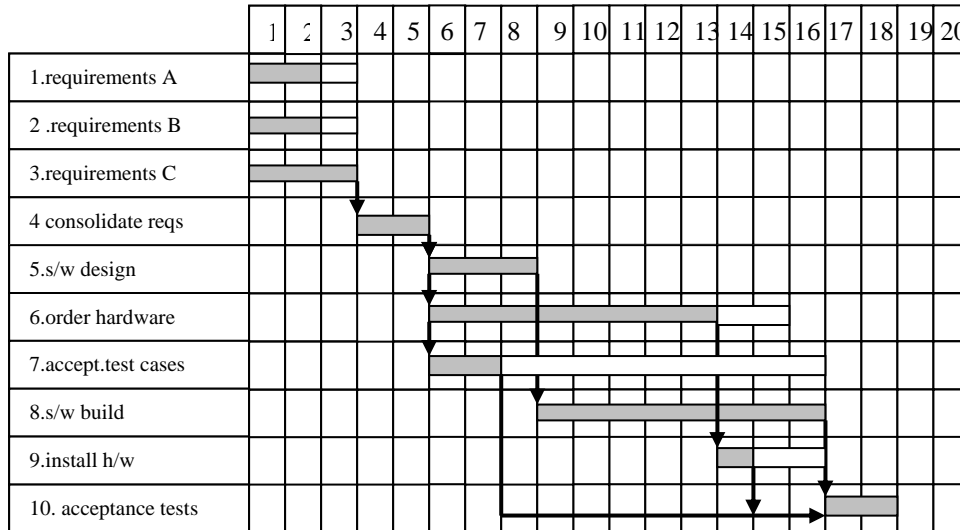
Frequently the diagram was far too cramped (candidates should use most or all of a page for such diagrams) with, as a consequence, dependencies flowing up and down (and sometimes) backwards **and** a lack of clear identification. Marks were not given where the diagram was too difficult to understand for any of the above reasons.

The most problematic value calculations were usually latest start time and earliest finish time, and floats were often omitted, suggesting a lack of understanding of each of their meanings. Disappointingly, the critical path was often not named at all and was very rarely highlighted on the diagram.

For completeness it helps also to state the minimum duration, although this was not requested specifically in the question and thus no marks were lost through not stating it.

Question 3 (b)
Answer Pointers

The question expected an illustration similar to that below (or an equivalent full description) – 6 marks.



with

- Vertical axis – activity identifiers
- Horizontal axis – time intervals (weeks)
- Scaled task bars – showing the earliest time when each can take place
- all dependencies and floats

Up to 4 marks (one each) for clear explanations of advantages, which could include:

- A clearer picture of when activities will actually take place, e.g. drawn to scale
- Easier to identify concurrent activities and therefore resource clashes
- Easier for resource scheduling
- Easier to schedule in non-project activities e.g. leave
- A more intuitive format for staff and others (cf holiday planners)
- Easier to highlight current status, e.g. part/full task completion to date
- Easier to show subtasks

Examiner’s Comments

Key words here were “illustrate” and, more importantly, “explain”. A diagram was expected, although a full description was acceptable, but a mere list of “advantages” (with no explanation at all) did not obtain full marks.

Most Gantt charts were well drawn, but several did not show dependencies clearly (if at all) – and thus task floats could not really be shown. For instance, some “bars” were too thick, with no gap between the rows, making it impossible to show such dependencies. Others did not really illustrate the need for a correct reliable time scale.

For clarity it helps to display the task names on the vertical axis (as above) rather than squeeze them into (or above) the task bar itself.

Gantt charts should normally display tasks top down, left to right, but some were drawn bottom up (i.e. starting from the bottom left) again making it difficult to display dependencies.

The advantages of Gantt charts were usually well understood, but were often just listed rather than being explained.

Question 4

4. For a particular project, it has been decided that the project manager should produce a report for the project board (or steering committee) at the end of each four week period.

- a) Describe the items of information which should appear in this report. **(9 marks)**
- b) Explain how the project manager would obtain the data used to produce the information contained in the report. **(9 marks)**
- c) A problem with the project is that the users keep asking for changes to the requirements. Further investigation reveals that the users find it difficult to visualise how the system will actually work until they are given something they can try out. Discuss what might be done to deal with this problem. **(7 marks)**

Part a) Answer Pointers

9 marks for a good clear **description** of 6 **different** items of information required in this type of (board level) progress report. These could include:

- Dates of reporting period
- Activities scheduled to be completed within time period which have or have not been completed
- Activities scheduled to start within the time period which have or have not been started
- Targets for next month
- Staffing – any changes e.g. leavers/starters
- Costs to date: budgeted and actual costs for completed activities, costs of uncompleted activities, and projected total costs
- Changes to scope of the project
- Risks – currently identified risks and their status
- Outlook for project

Up to 3 marks were not allocated if only a list of such information was supplied, with no description at all.

Examiner's Comments

The key points for this report are that it is:

- a) 4-weekly,
- b) intended for the project board.

Thus it needs to be at a summary level, whilst informing the board adequately and clearly of the current status of **all** aspects of the project.

Most candidates understood this, but there was a clear tendency to concentrate on actual progress issues at task level – e.g. by providing lists of tasks and their current status - without putting this detailed progress reporting into an overall context. Wider considerations, such as staffing issues, were frequently omitted.

Several candidates concentrated too much on low level detail – e.g. by including all staff time sheets in such a report. Such detail would not be appropriate for the board.

Question 4

Part b) Answer Pointers

Full 9 marks for an informed clear explanation of 4 or more different sources of such report information, recognising the relative importance of, and the type(s) of information that might be obtained from, each source. Typical sources could include:

- Checkpoint meetings with staff

- Timesheets from staff, together with estimates of completion for specific tasks that each is working on
- a project accounting system (which probably gets information from timesheets), and/or the organisation's accounting system
- Requests for change – from change management system
- Risk register – could be updated at checkpoint meetings etc
- Staffing details – from HR reports; induction/termination procedures
- User feedback

Up to 5 marks were not allocated for candidates who gave a list of sources with no relevant explanation.

Examiner's Comments

Important issues here were the relative importance of each identified information source, as well as the type(s) of information that might be obtained from each.

Most candidates recognised this, though often the relative importance aspect was not then considered. There was also a tendency to concentrate too much on time sheets rather than individual discussions with staff in order to anticipate likely future progress as well as record actual current progress.

Some candidates included the tools used to present the information (e.g. project plans showing progress to date) rather the source(s) from which this information was derived. Thus a project plan (e.g. Gantt chart) is a source of information **against which** current progress can be illustrated, but it is not a "full" source in its own right.

Very few of the less important, lower priority, sources were mentioned.

Part c) Answer Pointers

There are two distinct issues here (the effect and the underlying cause):

- a) dealing with the increasing number of requests for change (3 marks);
- b) providing the eventual users with a better understanding (visualisation) of the system under development (4 marks);

Suitable methods could be:

For a): use of a **change control system**, where a change management group (which has representatives of development and client/user management) have to assess the importance of each requested change and approve, reject or suspend the change. Exception reports might have to be made to the overall project board where the overall cost/duration of the project could exceed planned tolerances

For b): use of **prototyping/screen mock-ups**, early user manuals etc so that users get an idea of the way the new application will work in practice at an early point in the project

Examiner's Comments

The context of the question was important here, as well as the need to differentiate between the two different problems that, ideally, needed to be resolved separately.

Most candidates recognised the need to use prototyping to enable users to visualise the proposed new system, but often did not then realise that this would probably require a change in the current development method, and possibly a delay in the project delivery date. Few mentioned other possible ways of resolving this particular issue.

When considering the increasing number of change requests, some candidates suggested that there should have been a more rigorous definition of requirements, and therefore none (or few) of the change requests should be accepted. Others suggested repeating the requirements definition

stage. Both ignored the underlying problem and would not, on their own, have helped produce an acceptable new system.

Many candidates did not address this second issue at all, and disappointingly few mentioned the need for some form of practical change control system, with full consideration of the each of the requests being made.

Question 5

5. a) Explain the difference between *quality control* and *quality assurance*, giving TWO examples of each relating to software development projects. (10 marks)
- b) Describe how the acceptance testing phase of a project would be planned, executed and managed. (15 marks)

Part a) Answer Pointers

2 marks for each good clear definition of quality control and quality assurance, highlighting the difference:

Quality control - the focus is on checking the products created by the project/system and eliminating or reworking those that are defective

Quality assurance – the focus is on checking that the appropriate quality control processes are being carried out effectively

6 marks (nominally 1.5 per example) for 2 examples of each.

E.g.:

quality control examples: checking acceptance test cases against the requirements document to ensure that all the required functions in the new software product will be tested; acceptance testing to ensure delivered software meets its requirements;

quality assurance examples: checking acceptance test documentation to see that the execution of all test cases has been carried out and suspected errors have been recorded; checking that there is a record that all errors having been dealt with fully.

Examiner's Comments

Very few candidates distinguished clearly between the two concepts. QC is the activity of checking quality, whereas QA ensures that all such activities take place. Neither is related solely to "meeting requirements", nor is any form of testing (per se) necessarily a QC activity in itself – the quality aspect is in ensuring that such testing is comprehensive and thorough.

Several candidates did not apply their examples to the systems development environment (as required by the question).

Sometimes "quality control" was (wrongly) confused with "project control", and "quality assurance" was confused with "post-project review" (or some other post-project activity). It is essentially an "in-project" concept.

Question 5

Part b) Answer pointers

Three distinct stages of acceptance testing were identified in the question.

Up to 5 marks for a description of each.

The following actions might be included within each stage:

Planning:

- Produce a detailed requirements document to include, among other things, the main transactions in the new application;

- Then produce an overall testing plan. The type of testing to be carried out would depend on each of the requirements, e.g. if reliability and performance is important, then volume tests would be carried out.
- Prepare test cases; listing all test input data and the full expected results.

Execution

- When software development has been completed, a test version would be released for testing;
- Test cases would be executed and the actual outcomes would be compared with the expected. Where they differ formal error reports would be completed.

Management

- Ensure that each error report is reviewed jointly by the developer and user management and an adjudication made to confirm whether an actual error has been found which needs correction
- Ensure that all agreed corrections are made and new version(s) of the software released.
- Then check that not just the incorrect component but the whole system is re-tested – i.e. regression testing

Examiner's Comments

The emphasis here should be on:

- a) recognising each of the 3 stages (as stated in the question) distinctly
- b) identifying the "What?" and the "How?" of dealing with each of them.

Many candidates did not break down their answer in this way and often then concentrated on just one (often the actual testing) stage, instead of all three, without the required depth or structure. In several instances, the answer comprised general comments on the nature of acceptance testing, without relating them to any of these three stages.

Several candidates seemed to confuse "acceptance testing" with "implementation" and then considered the different methods of implementing a new or replacement application system.

Question 6

6. a) Using the example of the BCS Holidays scenario given in Question 1 and assuming that a decision is made to develop the new software from the beginning, using in-house staff, explain which activities the users could be involved with during the execution of the new project. **(13 marks)**
- b) Developer X is a software developer who is employed permanently by BCS Holidays and is paid £24,000 per year. Developer Y is an independent contractor who is usually paid £800 a week. The project requires the use of a software developer for 14 weeks. Discuss the advantages and disadvantages of using contract staff as opposed to permanent staff in general and also in the specific cases of developer X and developer Y. **(12 marks)**

Answer Pointers

- a) The activities with which users could be involved included:

Requirements gathering – users need to be interviewed and would need to make existing documentation available

Design – users could be involved in evaluating prototypes of the user interfaces and in generally providing feedback on the different options for the structure of the application

Acceptance test cases – would have a key role in producing these

Acceptance testing and checking – being involved in the data input would give users some early experience of using the new application

Data takeon – users would be motivated to ensure that this is done accurately as they would have to live with the consequences

Training – obviously users would be the recipients of training. Some key users could be trained who could then train the others

Hardware/software installation – would need to be consulted about which users would need which types of equipment and which types of access to the system. 13 marks

b) use of in-house staff vs. external contractors

The advantages of using in-house staff included:

- They are likely to have knowledge of existing applications and processes within the organisation;
- They are already on the payroll – so no direct additional costs;
- Expertise in maintenance of the new system is kept with the organisation;
- Security/confidentiality concerns should be less
- Less expensive daily rates

The advantages of using contractors included:

- They are not employed when not needed – this could reduce costs
- They can bring in outside expertise
- They might be eager to please because of possibility of new contracts
- Work normally carried out by permanent staff would be uninterrupted.

8 marks

In the specific scenario:

If, say, X gets four weeks holiday a year then the number of days worked a year would be 48×5 i.e. 240 days and the daily rate for X would be $24000/240 = £100$ a day, or £500 a week. It is likely that there would be other costs relating to social security, pensions, holidays etc that permanent staff would incur. The contractor, Y, on the other hand is being paid £800 a week or $14 \times 800 = £11,200$ for the project. There would be no further cost once the project has been completed.

4 marks

Examiner's Comments

This question may have suffered from being the last on the paper, many candidates gave very little and incomplete answers.

- a) Generally this sub-section was reasonably well answered, but a common error was to provide a general description of the software development cycle and not mention where the users' involvement might be.
- b) Usually poor use was made of the examples of X and Y. Sometimes statements were made with no reasoning provided to support them e.g. 'in-house developers are slower'. Clearly the examiners would need to know why the candidate felt in house developers were slower.