Section 4: Computing project

The project is a substantial piece of work requiring analysis and design over an extended period of time, which is organised, evaluated and presented in a report.

Candidates choose, in conjunction with their teacher, a well-defined user-driven problem which enables them to demonstrate their skills in analysis, design and software development, including programming, testing, installation, documentation and evaluation. Problems should be selected that allow candidates to demonstrate their programming skills.

Projects should be chosen to demonstrate the integrative aspects of the work and should avoid needless repetition of the demonstration of a given skill. Each candidate must submit a report on their piece of work, supported by evidence of software development including programming and testing.

The teacher marks the projects using the marking criteria in the *Guidance on Marking the Computing Project* section of this syllabus, then moderation takes place following Cambridge procedures.

4.1 Report (3 marks)
A report presenting the Coursework as specified in 4.2 to 4.6.
Content 4.1.1 Organise the report into sections as given in the syllabus 4.1.2 Word process the report 4.1.3 Documentation of each stage of the development
Learning outcomes *Candidates should be able to:* (a) organise the report (b) use word processing features where appropriate including checking of spelling and grammar (c) include the evidence specified in 4.2 to 4.6

4.2 Definition, investigation and analysis (11 marks)

Explanation of the problem to be solved, the user's requirements and how they were obtained.
There should be a clear statement of requirements, agreed with the prospective client.

Content

4.2.1 Define a problem
4.2.2 Investigate the current system
4.2.3 Record findings
4.2.4 Analyse findings
4.2.5 Identify problems/inefficiencies with current system
4.2.6 Specify requirements: user, hardware, software

Learning outcomes

Candidates should be able to:

(a) define the nature of the problem to be solved

(b) use appropriate methods to investigate the problem and to gather information; these may include questionnaires, observation, meetings and document collection, but must include an interview with the client

(c) record information/data and gather sample documents currently used

(d) identify the current processes and current data structures

(e) analyse the data and processes: candidates will be expected to use appropriate techniques such as structure diagrams/dataflow diagrams/system flowcharts to illustrate their analysis

(f) specify inefficiencies and problems apparent from the information collection

(g) derive the client's and information requirements of the system

(h) specify the required hardware and give reasons for their choice

(i) specify the required software and give reasons for their choice

(j) develop and document a clear requirement specification

4.3 Design (12 marks)

Detailed system design including data structures, input-output format and processes involved, and testing required. There should be a clear design specification.

Content

4.3.1 Overview including an agreed set of objectives
4.3.2 Output design
4.3.3 Input design
4.3.4 Data structures/model
4.3.5 Process model
4.3.6 Test plan

Learning outcomes

Candidates should be able to:

(a) agree a set of objectives with the client

(b) design and document report layouts, screen displays and/or other forms of output, drawing up detailed models of the proposed interface

(c) design and document data capture forms and/or screen layouts

(d) design and document, using appropriate techniques (for example, normalisation/E-R models), the data structures necessary to solve the inefficiencies/problems indicated in the requirements specification

(e) design and document an algorithm/pseudocode/top-down diagram or other form of process model which is/are necessary for the solution of the problem

(f) design and document a test plan that includes test data and expected outcomes

4.4 Software development, programming, testing and installation (18 marks)

A software solution that includes some programming code using a stand-alone programming language or program embedded within application software (e.g. VBA used as the front end of a database solution written by the candidate). A comprehensive test plan is developed from the design, which should show that the system works to the satisfaction of the client by providing comprehensive functional testing, both alpha and beta, of the solution. The test plan should be clearly cross-referenced to the agreed set of objectives to provide evidence that the system has been tested during development and by the client.

Content

4.4.1 Software development
4.4.2 Programming
4.4.3 Testing a software solution
4.4.4 Planning for installation and use
4.4.5 Client and user testing

Learning outcomes

Candidates should be able to:

(a) implement the proposed process model using a programming language and possibly the facilities of a software package

(b) develop the data structures of the design using the appropriate features of a software package and programming language

(c) develop inputs/outputs appropriate to the design of the solution

(d) illustrate how the software solution evolves

(e) test the software solution

(f) produce detailed output from the testing, cross-referencing to the test plan

(g) test the software solution with the client and user, providing documented evidence that the solution works, and devise a strategy for its installation

4.5 Documentation (10 marks)

The **System Maintenance Documentation** should include an explanation of the structure of the solution. All the necessary information about the system that would allow someone else to maintain and develop it should be included, for example, backup procedures/cycles, annotated code/modules, data structures used, and must include an element of adaptive maintenance in order to provide some future-proofing of the solution.

The **User Guide** should include step-by-step instructions for operating all aspects of the system, including a means of dealing with any errors that may occur. It should also include appropriate "Help" and messages within the software solution, and be present in the form of a hypertext document.

Content

4.5.1 System Maintenance Documentation
4.5.2 User Guide

Learning outcomes

Candidates should be able to:

(a) develop system maintenance documentation

(b) develop a detailed user guide

4.6 Evaluation (6 marks)
Discussion of the degree of success in meeting the original objectives as specified in the requirements specification, ease of the use of the package, acceptability to the client (including where possible a letter of acceptance from the client and reference to client and user testing results).
Content 4.6.1 Evaluate results against the agreed set of objectives 4.6.2 Evaluate the results of client and user testing
Learning outcomes *Candidates should be able to:* (a) evaluate the final system against the criteria described in the agreed set of objectives (b) evaluate the client's and user's responses to testing the system

7. Coursework

Section 4: Computing project (60 marks)

This unit assesses candidates' ability to develop a computer-based solution to a real life problem requiring the skills of analysis, design, programming, development, testing, implementation and evaluation.

Candidates should formulate the task in negotiation with their teacher. If Centres are uncertain about the appropriateness of a problem they should seek advice from Cambridge.

Assessment and moderation
All coursework is marked by the teacher and internally standardised by the Centre. Coursework is then submitted to Cambridge by the specified date.

The internally moderated marks for all candidates must be recorded on the Coursework Assessment Summary Form. This form, and the instructions for completing it, may be downloaded from **www.cie.org.uk/samples**. The database will ask you for the syllabus code (i.e. 9691) and your Centre number, after which it will take you to the correct form. Follow the instructions when completing the form.

The purpose of moderation is to ensure that the standard for the award of marks in coursework is the same for each Centre, and that each teacher has applied the same standards appropriately across the range of candidates within the Centre.

Minimum coursework requirements
If a candidate submits no work for a coursework unit, then the candidate should be indicated as being absent from that unit on the coursework mark sheets submitted to Cambridge.

If a candidate completes any work for the coursework unit, then the work should be assessed according to the criteria and marking instructions, and the appropriate mark awarded (which may be zero).

Authentication
As with all coursework, **the teacher must be able to verify that the work submitted for assessment is the candidate's own work. Sufficient work must be carried out under direct supervision to allow the teacher to authenticate the coursework marks with confidence.**

Cambridge is happy to rely on the professionalism of teachers to ensure fairness with this work.

Differentiation
In the question papers, differentiation is achieved by setting questions which are designed to assess candidates at their appropriate levels of ability, and which are intended to allow all candidates to demonstrate what they know, understand and can do.

In coursework, candidates should choose their project problem so that the work enables them to display positive achievement and to demonstrate their full range of abilities.

Recording candidates' marks
Candidates' marks must be recorded on the Individual Candidate Record Card produced by Cambridge. These forms, and the instructions for completing them, may be downloaded from **www.cie.org.uk/samples**. The database will ask you for the syllabus code (i.e. 9691) and your Centre number, after which it will take you to the correct forms. Follow the instructions when completing each form.

Please copy the Marking Grid at the back of this syllabus document and submit with the Computing Project. The Marking Grid for the Computing Project should be attached to each candidate's submission.

8. Appendix

8.1 Guidance on selecting the Computing Project

The selection of the problem for which a computerised system is to be designed and implemented is extremely important. It should be chosen by the candidate in consultation with the teacher, and should always involve a client, who requires the solution to the problem, and user(s), the person who is going to use the computerised system. The client and the user may be the same person, e.g. if a sole-trader's business requires a computerised system.

It is important to stress that the candidate should endeavour to produce a system which will solve a given problem sensibly, within the constraints of resources available to the candidate.

Since the computing project seeks to assess the systems analysis section of the specification in a practical manner, candidates should not produce a system from their own limited knowledge of the requirements of the system. The client has to be someone who is willing to be involved in the project:

- in the analysis of the problem, where the client's requirements are obtained; this may take the form of recorded interviews with the candidate
- at the software development, testing and implementation stages, where the client and/or user is involved in 'prototyping'
- at the evaluation stage, where the client is involved in checking that the system is completed as specified and, leading on from this, is then willing to write a letter of acceptance of the system, including any criticisms of it.

In this way, candidates can be encouraged to look beyond school or college life into the businesses and companies in the community of the surrounding area. The emphasis is on analysing an existing system, and producing a computer-based solution to fit the needs of a client.

At the end of the project, candidates should submit a concisely written and well laid out report, which should be word processed.

The solution must be implemented using a programming language and any of the following that are appropriate: pre-written modules or toolkits, applications software and programmable packages. Very brief descriptions of any programming languages or software packages used, together with reasons for their selection, should be included in the report.

For the programming the candidate should:

- annotate listings
- explain each section of the program with appropriate algorithm descriptions, which should be language independent
- define variables by name, type and function where appropriate
- define clearly and identify the purpose of subroutines and procedures.

Where part of the solution has been produced with a software package that has not involved programming, the candidate should:

- explain each section of the solution with appropriate algorithm descriptions
- define the purpose and inter-relationship of modules within the system
- clearly annotate the results produced.

The projects should be documented in a report that contains the title, a contents list, and is set out in the sub-sections identified in the 'Guidance on Marking the Computing Project'.

Appropriate evidence of development, testing and demonstration of a working system, such as screen dumps or photographs of screen layouts and printouts, paper based user documentation and a letter from the client to say that the system has been developed satisfactorily, must be included in the report.

Candidates should not submit magnetic, optical or solid state media as supporting evidence.

The computing project must involve programming and may involve the tailoring of generic software packages and may also involve the choosing and installing of hardware. It is not intended that any method of solution is better than another merely that the solution must be one that suits the problem that is being solved.

8.2 Guidance on marking the Computing Project

Computing Projects are assessed as follows:

Quality of report	3 marks
Definition, investigation and analysis	11 marks
Design	12 marks
Software development, programming, testing and installation	18 marks
Documentation	10 marks
Evaluation	6 marks
Total	**60 marks**

(a) Quality of report [Total 3 marks]

A candidate should produce a well ordered report that covers all the information from the sections set out below.

Evidence for most sections is included; there may be errors of spelling, punctuation and grammar.	1 mark
Evidence for all sections is included, the report is well ordered and there are few errors of spelling, punctuation and grammar.	2 marks
The report is complete, well organised with good use of illustrations, and there may be a few minor errors of spelling, punctuation and grammar.	3 marks

(b) Definition, investigation and analysis [Total: 11 marks]

(i) Definition – nature of the problem [3 marks]

A candidate should not expect the examiners to be familiar with the theory and practice in the area of the chosen system. There should be a brief description of the organisation (for example, firm or business) involved and the current methods used in the chosen areas that may form the basis of the project. A clear statement of the origins and form of data should be given. At this stage the exact scope of the project may not be known and it may lead to the arranging of an interview with the client.

Description of the organisation.	1 mark
Description of the organisation and the methods currently used in the area of the chosen project.	2 marks
Full description of the organisation and methods currently in use in the area of the chosen project, with a description of the origin of the data to be used and some indication of the form that data takes.	3 marks

(ii) Investigation and analysis [8 marks]

This section is the 'systems analysis'. The candidate should describe how the client requirements were ascertained (possibly by long discussions with the users: question and answer sessions should be recorded and outcomes agreed). A clear requirements specification should be defined. Alternative outline solutions should be discussed and evaluated against one another.

Some elements have been discussed but little or no client involvement.	1–2 marks
Some evidence that an attempt has been made to interview the client and some recording of it has been made. An attempt has been made to develop a requirement specification based on the information collected.	3–4 marks
Good client involvement and recording of the interview(s). Most of the necessary items have been covered including a detailed discussion of alternative approaches. A requirements specification based on the information collected is present but with some omissions.	5–6 marks
Excellent client and user involvement with detailed recording of the client's requirements. Alternative approaches have been discussed in depth. The report demonstrates a thorough analysis of the system to be computerised. A detailed requirements specification based on the information collected has been produced.	7–8 marks

(c) Design [Total: 12 marks]

(i) Nature of the solution [8 marks]

A detailed systems design (including diagrams as appropriate) should be produced and agreed with the client. Proposed record, file and data structures should be described and design limitations should be included. Design of data capture forms, input formats (with examples of screen layouts) and output formats should be included here where relevant. Process designs and a test plan for the system should also be included. The test plan should contain test data and the expected results for that data. An agreed set of objectives should also be included. These items are the design specifications, which should be agreed with the client.

Some vague discussion of what the system will do with a brief diagrammatic representation of the new system.	1–2 marks
The major objectives of the new system have been adequately summarised, *but omissions have been made.* There is a brief outline of a design specification, including mock-ups of inputs and outputs, process model described (including a diagram: structure diagram, dataflow diagram or system flowchart). However, there is a lack of completeness with omissions from the process model, inputs and outputs. Data structures have been identified but there may be inadequate detail. The test plan may be incomplete.	3–4 marks
A clear set of objectives has been defined and a full design specification is included, but there may be some errors or logical inconsistencies, for example validation specified may be inadequate or field lengths incorrect. There is clear evidence that a response to the design has been obtained from the client, and any comments have been acted upon.	5–6 marks
A clear set of agreed objectives with a detailed and complete design specification, which is logically correct. There are also detailed written descriptions of any processes/modules and a clear, complete definition of any data structures. The specification is sufficient for someone to pick up, develop and test an end result using the software and hardware specified in the requirements specification.	7–8 marks

(ii) Intended benefits [2 marks]

There should be some discussion of the relative merits of the intended system and of the previous mode of operation. This may include any degree of generality beyond the original scope of the system.

One valid benefit of the new system has been identified and explained.	1 mark
The benefits of the new system have been comprehensively described.	2 marks

(iii) Limits of the scope of the solution [2 marks]

This may include volume (sizing limitations) and limitations of the facilities used. For full marks there must be some estimate of the size of the files required for the implemented system.

A discussion of what the system limitations are.	1 mark
A detailed description of the system limitations has been given, including the estimate of the size of the files required for the implemented system.	2 marks

(d) Software development, programming, testing and installation [Total: 18 marks]

(i) Development [4 marks]

A technical description of how the solution relates to the design specification produced and agreed with the user should be included.

Program listings or evidence of tailoring of a software package is provided in the form of printouts. The developed solution does not fulfil the design specification. A teacher may award 1 mark if they have been shown the system working satisfactorily and there is no hard evidence in the project report.	1 mark
Program listings or evidence of tailored software packages are provided in the form of printouts. Data structures are illustrated as part of the listings where appropriate, detailing their purpose. There is some annotation evident to illustrate how the package was tailored for a particular purpose or to indicate the purpose of sections of code in a program listing. The developed solution has logical flaws and does not fulfil the design specification.	2–3 marks
Program listings or evidence of tailored software packages are provided in the form of printouts. Data structures are illustrated as part of the listings where appropriate, detailing their purpose. There is a full set of printouts showing input and output as well as data structures. The developed solution does fulfil the design specification.	4 marks

(ii) Programming [5 marks]

There should be clearly set out program listings that demonstrate the technical competence of the candidate. Candidates should make good use of the facilities of a procedural programming language as part of their solution.

A program listing showing code written by the candidate is included.	1–2 marks
Some technical competence in programming shown by a program listing that makes use of meaningful identifier names, indentation and formatting to show the control structures used. The code should be annotated with some comments so that the logic of the solution can be followed.	3–4 marks
Good technical competence in programming shown by a self-documented program listing that makes good use of meaningful identifier names, indentation and formatting to show the control structures used. The code should be annotated with comments so that the logic of the solution can be easily followed.	5 marks

(iii) Testing [5 marks]

An attempt should be made to show that all parts of the system have been tested, including those sections dealing with unexpected or invalid data as well as borderline cases. Showing that many other cases of test data are likely to work – by including the outputs that they produce – is another important feature. Evidence of testing is essential. Comments by teachers and others are of value, but the test plan must be supported by evidence in the report of a properly designed testing process. The examiner must be left in no doubt the system actually works to the satisfaction of the client. This evidence may be in the form of hardcopy output and screen dumps.

A collection of hardcopy test run outputs with no test plan, or a test plan with no hardcopy evidence may also be present. A teacher may award 1 mark if they have been shown the system working satisfactorily and there is no hard evidence in the project report.	1 mark
There is little evidence of testing with a badly developed test plan with clear omissions. There is no description of the relationship between the structure of the development work and the testing in evidence.	2 marks
There should be hardcopy evidence from at least eight different test runs cross-referenced to the test plan. However, not all cases have been tested.	3–4 marks
Evidence of each test run cross-referenced to the test plan is present in the report. Testing should include as many different paths through the system as is feasible, including valid, invalid and borderline cases. Marks may be lost for lack of evidence of a particular test run.	5 marks

(iv) Installation [4 marks]

It is recognised that the client may not fully install and use the system, although this is the ultimate aim. However, to score any marks in this section, there must be some evidence that the client has seen the system in operation. This can be done in a number of ways: such as by inviting the client to see the product and allowing the candidate to demonstrate the system, or by taking the system to the client involved. There should be an installation plan written, including details of system changeover, training required and details of user testing.

Details of system changeover have been documented. Some evidence of client and/or user testing is given, usually by questionnaire or written comments by fellow students or others who were not directly involved in the development of the system.	1 mark
An implementation plan with details of system changeover and training required. There is written evidence available from the client indicating that they have seen the system in operation.	2–3 marks
A clear and detailed implementation plan, including planned system changeover, training required and detailed stages of user testing. There is written evidence available from the client and/or user that they have tested the system and agree with the strategy for implementation.	4 marks

(e) Documentation [Total: 10 marks]

(i) System maintenance documentation [4 marks]

Much of the documentation will have been produced as a by-product of design and development work and also as part of writing up the report to date. The contents of the manual should, where relevant, include the following: record, file and data structures used; data dictionary; data flow (or navigation paths); annotated program listings; detailed flowcharts; details of the algorithms used and adaptive maintenance to provide for some future-proofing.

All parts of the manual should be fully annotated, since this is important for subsequent maintenance of the system. The specifications of the hardware and software on which the system can be implemented should be included.

Some items are present with some annotation attempted.	1–2 marks
One or two omissions, but the rest is present and annotation is used sensibly.	3–4 marks

(ii) User guide [6 marks]

Clear guidance, as friendly as possible, should be given to the user for all operations that they would be required to perform. These would include input format with screen displays, print options, backups (file integrity routines), security of access to data and a guide to common errors that may occur. (Note: the candidate would **not** be required to copy out large volumes of any underlying software's user guide, but to produce a non-technical and easy-to-follow guide for someone with little computer knowledge.) Some mention here of the relationship between items of software and the data they deal with may be relevant.

The user guide should be well presented with an index and, where necessary, a glossary of the terms used. Alternatively, an electronic guide could be based around hypertext links (screen dumps will be required).

An incomplete guide, perhaps with no screen displays. Some options briefly described, but difficult for the user to follow.	1–2 marks
All but one or two options fully described (for example, backup routines not mentioned). Mostly, the options are easy for the user to follow with screen displays.	3–4 marks
A full user guide with all options described well presented (possibly as booklet), with an index and a glossary. No omission of any of the options available (including backup routines, guide to common errors). Marks may be lost for inadequate descriptions of some options. For full marks, good on-screen help should exist where this is a sensible option, and be present in the form of a hypertext document.	5–6 marks

(f) Evaluation [Total: 6 marks]

(i) Discussion of the degree of success in meeting the original objectives [3 marks]

This discussion should demonstrate the candidate's ability to evaluate the effectiveness of the completed system. The agreed set of objectives should be matched to the achievements, taking into account the limitations. Client and user evaluation is also essential, and should arise from a questionnaire or, preferably, direct evaluation. For full marks it is important that the user provides sets of data as they are likely to occur in practice, and that the results arising from such data be taken into account. This data is typical data rather than test data, and it may show up faults or problems that the candidate's own test data failed to find.

Some discussion about the success, or otherwise, of the work, but with no reference to the specification set out in **(c)(i)**.	1 mark
Some discussion about a number of the objectives set out in **(c)(i)**, but some omissions or inadequate explanation of success or failure.	2 marks
A full discussion, taking each objective mentioned in **(c)(i)** and explaining the degree of success in meeting them, indicating where in the project evidence can be found to support this, or giving reasons why they were not met.	3 marks

(ii) Evaluation of the client's and user's response to the system [3 marks]

It is important that neither the client nor the user is assumed to be an expert in computer jargon, so some effort must be made to ensure that the system is user-friendly. It will be assumed that the client will have considerable knowledge of the underlying theory of the business being computerised. Clarity of menus, clear on-screen help and easy methods of inputting data are all examples of how the system can be made user-friendly. Here, marks are awarded for the degree of satisfaction that the client indicates in the acceptance procedure. Could the system or its results be used? Was the system specification achieved? Do any system faults still exist? The candidate should evaluate the client's response to the final version of the system.

It is important that the client and the user become active participants in this section, and that their responses are reported and evaluated by the candidate.

Some effort has been made to make the system user-friendly, but the user still has difficulty using the system.	1 mark
The system is mostly user-friendly, but there is room for improvement (e.g. no on-screen help has been provided). The client indicates that the system could be used, but there are some faults which need to be rectified.	2 marks
A fully user-friendly system has been produced. The client indicates that the system fully meets the specification given in section **(b)**, and there are no known faults in the system.	3 marks

Paper 4: Marking Grid (attach to Project)

Computing Project marking details

		Total Project mark	**[___/60 marks]**
Candidate name:		**Candidate number:**	
Centre name:		**Centre number:**	

(a) Quality of report	**[___/3 marks]**

		Mark	Comments
1	Evidence for most sections is included; there may be errors of spelling, punctuation and grammar.		
2	Evidence for all sections is included, the report is well ordered, and there are few errors of spelling, punctuation and grammar.		
3	The report is complete, well organised with good use of illustrations, and there may be a few minor errors of spelling, punctuation and grammar.		

(b) Definition, investigation and analysis	**[___/11 marks]**

(i) Definition – nature of the problem **[__/3 marks]**

		Mark	Comments
1	Description of the organisation.		
2	Description of the organisation and the methods currently used in the area of the chosen project.		
3	Full description of the organisation and methods currently in use in the area of the chosen project, with a description of the origin of the data to be used and some indication of the form that data takes.		

(ii) Investigation and analysis **[__/8 marks]**

		Mark	Comments
1–2	Some elements have been discussed but little or no user involvement.		
3–4	Some evidence that an attempt has been made to interview the client and some recording of it has been made. An attempt has been made to develop a requirement specification based on the information collected.		
5–6	Good client involvement and recording of the interview(s). Most of the necessary items have been covered including a detailed discussion of alternative approaches. A requirements specification based on the information collected is present but with some omissions.		
7–8	Excellent client involvement with detailed recording of the client's requirements. Alternative approaches have been discussed in depth. The report demonstrates a thorough analysis of the system to be computerised. A detailed requirements specification based on the information collected has been produced.		

(c) Design	[___/12 marks]

(i) Nature of the solution **[__/8 marks]**

		Mark	Comments
1–2	Some vague discussion of what the system will do with a brief diagrammatic representation of the new system.		
3–4	The major objectives of the new system have been adequately summarised, but omissions have been made. There is a brief outline of a design specification, including mock-ups of inputs and outputs, process model described (including a diagram: structure diagram, dataflow diagram or system flowchart). However, there is a lack of completeness with omissions from the process model, inputs and outputs. Data structures have been identified but there may be inadequate detail. The test plan may be incomplete.		
5–6	A clear set of objectives have been defined and a full design specification is included but there may be some errors or logical inconsistencies, for example validation specified may be inadequate or field lengths incorrect. There is clear evidence that a response to the design has been obtained from the client, and any comments have been acted upon.		
7–8	A clear set of agreed objectives with a detailed and complete design specification, which is logically correct. There are also detailed written descriptions of any processes/modules and a clear, complete definition of any data structures. The specification is sufficient for someone to pick up, develop and test an end result using the software and hardware specified in the requirements specification.		

(ii) Intended benefits **[__/2 marks]**

		Mark	Comments
1	One valid benefit of the new system has been identified and explained.		
2	The benefits of the new system have been comprehensively described.		

(iii) Limits of the scope of the solution **[___/2 marks]**

		Mark	Comments
1	A discussion of what the system limitations are.		
2	A detailed description of the system limitations has been given, including the estimate of the size of the files required for the implemented system		

(d) Software development, programming, testing and installation	[___/18 marks]

(i) Development **[__/4 marks]**

		Mark	Comments
1	Program listings or evidence of tailoring of a software package is provided in the form of printouts. The developed solution does not fulfil the design specification. A teacher may award 1 mark if they have been shown the system working satisfactorily and there is no hard evidence in the project report.		
2–3	Program listings or evidence of tailored software packages are provided in the form of printouts. Data structures are illustrated as part of the listings where appropriate, detailing their purpose. There is some annotation evident to illustrate how the package was tailored for a particular purpose or to indicate the purpose of sections of code in a program listing. The developed solution has logical flaws and does not fulfil the design specification.		
4	Program listings or evidence of tailored software packages are provided in the form of printouts. Data structures are illustrated as part of the listings where appropriate, detailing their purpose. There is a full set of printouts showing input and output as well as data structures. The developed solution does fulfil the design specification.		

(ii) Programming **[__/5 marks]**

		Mark	Comments
1–2	A program listing showing code written by the candidate is included.		
3–4	Some technical competence in programming shown by a program listing that makes use of meaningful identifier names, indentation and formatting to show the control structures used. The code should be annotated with some comments so that the logic of the solution can be followed.		
5	Good technical competence in programming shown by a self-documented program listing that makes good use of meaningful identifier names, indentation and formatting to show the control structures used. The code should be annotated with comments so that the logic of the solution can be easily followed.		

(iii) Testing **[___/5 marks]**

		Mark	Comments
1	A collection of hardcopy test run outputs with no test plan, or a test plan with no hardcopy evidence may also be present. A teacher may award 1 mark if they have been shown the system working satisfactorily and there is no hard evidence in the project report.		
2	There is little evidence of testing with a badly developed test plan with clear omissions. There is no description of the relationship between the structure of the development work and the testing in evidence.		
3–4	There should be hardcopy evidence from at least eight different test runs cross-referenced to the test plan. However, not all cases have been tested.		
5	Evidence of each test run cross-referenced to the test plan is present in the report. Testing should include as many different paths through the system as is feasible, including valid, invalid and borderline cases. Marks may be lost for lack of evidence of a particular test run.		

(iv) Installation **[__/4 marks]**

		Mark	Comments
1	Details of system changeover have been documented. Some evidence of client and/or user testing is given, usually by questionnaire or written comments by fellow students or others who were not directly involved in the development of the system.		
2–3	An implementation plan with details of system changeover and training required. There is written evidence available from the client indicating that they have seen the system in operation.		
4	A clear and detailed implementation plan including planned system changeover, training required and detailed stages of user testing. There is written evidence available from the client and/or user that they have tested the system and agree with the strategy for implementation.		

(e) Documentation	[__/10 marks]

(i) Systems maintenance documentation **[__/4 marks]**

		Mark	Comments
1–2	Some items are present with some annotation attempted.		
3–4	One or two omissions, but the rest is present and annotation is used sensibly.		

(ii) User Guide **[___/6 marks]**

		Mark	Comments
1–2	An incomplete guide, perhaps with no screen displays. Some options briefly described but difficult for the user to follow.		
3–4	All but one or two options fully described (for example, backup routines not mentioned). Mostly, the options are easy for the user to follow with screen displays.		
5–6	A full user guide with all options described well presented (possibly as booklet) with an index and a glossary. No omission of any of the options available (including backup routines, guide to common errors). Marks may be lost for inadequate descriptions of some options. For full marks, good on-screen help should exist where this is a sensible option, and be present in the form of a hypertext document.		

(f) Evaluation	[__/6 marks]

(i) Discussion of the degree of success in meeting the original objectives **[__/3 marks]**

		Mark	Comments
1	Some discussion about the success, or otherwise, of the work, but with no reference to the specification set out in (c)(i).		
2	Some discussion about a number of the objectives set out in (c)(i), but some omissions or inadequate explanation of success or failure.		
3	A full discussion, taking each objective mentioned in (c)(i) and explaining the degree of success in meeting them, indicating where in the project evidence can be found to support this, or giving reasons why they were not met.		

(ii) Evaluation of the client's and user's response to the system **[__/3 marks]**

		Mark	Comments
1	Some effort has been made to make the system user-friendly, but the user still has difficulty using the system.		
2	The system is mostly user-friendly, but there is room for improvement (e.g., no on-screen help has been provided). The client indicates that the system could be used, but there are some faults which need to be rectified.		
3	A fully user-friendly system has been produced. The user indicates that the system fully meets the specification given in section (b), and there are no known faults in the system.		