4.14.5 Guide to non-exam assessment documentation

1. Analysis

Marks: 9/75 (12%)

Specification 4.14.5.1 Analysis
Students are expected to:
• produce a clear statement that describes the problem area and specific problem that is being solved/investigated 1. Introduction - needs to be changed and updated
• outline how they researched the problem 4. how the problem was researched
• state for whom the problem is being solved/investigated 3. Potential user
• provide background in sufficient detail for a third party to understand the problem being solved/investigated 1. Introduction - could be exx
• produce a numbered list of measurable, "appropriate" specific objectives, covering all required functionality of the solution or areas of investigation (Appropriate means that the specific objectives are single purpose and at a level of detail that is without ambiguity.) 2. Initial objectives - might need to be changed
• report any modelling of the problem that will inform the Design stage, for example a graph/ network model of Facebook connections or an E-R model. NEED TO DO
A fully scoped analysis is one that has:
• researched the problem thoroughly
• has clearly defined the problem being solved/investigated
• omitted nothing that is relevant to subsequent stages
• statements of objectives which clearly and unambiguously identify the scope of the project
• modelled the problem for the Design stage where this is possible and necessary.

Mark scheme

Level	Mark range	Description
3	7-9	• Fully or nearly fully scoped analysis of a real problem, **presented in a way that a third party can understand**. • Requirements fully documented in **a set of measurable and appropriate specific objectives**, **covering all required functionality** of the solution or areas of investigation. • **Requirements arrived at** by considering, **through dialogue**, the needs of the intended users of the system, or recipients of the outcomes for investigative projects. • **Problem sufficiently well modelled** to be of use in subsequent stages.
2	4-6	• Well scoped analysis (but with some omissions that are not serious enough to undermine later design) of a real problem. • Most, but not all, requirements documented in a set of, in the main, measurable and appropriate specific objectives that cover most of the required functionality of a solution or areas of investigation. • Requirements arrived at, in the main, by considering, through dialogue, the needs of the intended users of the system, or recipients of the outcomes for investigative projects. • Problem sufficiently well modelled to be of use in subsequent stages.
1	1-3	• Partly scoped analysis of a problem. • Requirements partly documented in a set of specific objectives, not all of which are measurable or appropriate for developing a solution.The required functionality or areas of investigation are only partly addressed. • Some attempt to consider, through dialogue, the needs of the intended users of the system, or recipients of the outcomes for investigative projects. • Problem partly modelled and of some use in subsequent stages.

3. Design

8. Documented design

Marks: 12/75 (16%)

Specification 4.14.5.2 Design
Students are expected to articulate their design in a manner appropriate to the task and with sufficient clarity for a third party to understand how the key aspects of the solution/investigation are structured and on what the design will rely, eg use of numerical and scientific package libraries, data visualisation package library, particular relational database and/or web design framework. The emphasis is on communicating the design; therefore it is acceptable to provide a description of the design in a combination of diagrams and prose as appropriate, as well as a description of algorithms, SQL, data structures, database relations as appropriate, and using relevant technical description languages, such as pseudo-code. Where design of a user interface is relevant, screen shots of actual screens are acceptable.
Mark scheme

Level	Mark range	Description
4	10-12	Fully or nearly fully articulated **design for a real problem**, that **describes how al**l or almost all of **the key aspects of the solution/investigation are to be structured**/are structured.
3	7-9	Adequately articulated design for a real problem that describes how most of the key aspects of the solution/ investigation are to be structured/are structured.
2	4-6	Partially articulated design for a real problem that describes how some aspects of the solution/investigation are to be structured/are structured.
1	1-3	Inadequate articulation of the design of the solution so that it is difficult to obtain a picture of how the solution/ investigation is to be structured/is structured without resorting to looking directly at the programmed solution.

need to rearrange images and label them properly

5. Technical solution

Marks: 42/75 (56%)

Specification 4.14.5.3 Technical solution
Students should provide program listing(s) that demonstrate their technical skill. The program listing(s) should be appropriately annotated and self-documenting (an approach that uses meaningful identifiers, with well structured code that minimises instances where program comments are necessary).
Students should present their work in a way that will enable a third party to discern the quality and purpose of the coding. This could take the form of:
• an overview guide which amongst other things includes the names of entities such as executables, data filenames/urls, database names, pathnames so that a third party can, if they so desire, run the solution/investigation
• explanations of particularly difficult-to-understand code sections; a careful division of the presentation of the code listing into appropriately labelled sections to make navigation as easy as possible for a third party reading the code listing.
Achievement of the latter, to an extent, is linked to the skill in applying a structured approach during the course of developing the solution or carrying out the investigation.

4.14.3.3.1 Completeness of solution (15 marks)

Level	Mark range	Description
3	11-15	**A system that meets almost all of the requirements of a solution/an investigation** (**ignoring any requirements that go beyond the demands of A-level**).
2	6-10	A system that achieves many of the requirements but not all. The marks at the top end of the band are for systems that include some of the most important requirements.
1	1-5	A system that tackles some aspects of the problem or investigation.

4.14.3.3.2 Techniques used (27 marks)

Level	Mark range	Description	Additional Information
3	19-27	The techniques used are appropriate and demonstrate **a level of technical skill equivalent to those listed in Group A in Table 1**. **Program(s) demonstrat**e(s) that **the skill** required for this level has been applied sufficiently to demonstrate proficiency.	Above average performance: **Group A equivalent algorithms and model programmed** more than **well to excellent**; all or almost **all excellent coding style characteristics**; more than to **highly effective solution.** Average performance: Group A equivalent algorithms and/or model programmed well; majority of excellent coding style characteristics; an effective solution. Below average performance: Group A equivalent algorithms and/or model programmed just adequately to fully adequate; some excellent coding style characteristics; less than effective to fairly effective solution.
2	10-18	The techniques used are appropriate and demonstrate a level of technical skill equivalent to those listed in Group B in Table 1. Program(s) demonstrate(s) that the skill required for this level has been applied sufficiently to demonstrate proficiency.	Above average performance: Group B equivalent algorithms and model programmed more than well to excellent; majority of excellent coding style characteristics; more than to highly effective solution. Average performance: Group B equivalent algorithms and/or model programmed well; some excellent coding style characteristics; effective solution. Below average performance: Group B equivalent algorithms and/or model programmed just adequately to fully adequate; all or almost all relevant good coding style characteristics but possibly one example at most of excellent characteristics; less than effective to fairly effective solution.
1	1-9	The techniques used demonstrate a level of technical skill equivalent to those listed in Group C in Table 1. Program(s) demonstrate(s) that the skill required for this level has been applied sufficiently to demonstrate proficiency	Above average performance: Group C equivalent model and algorithms programmed more than well to excellent; almost all relevant good coding style characteristics; more than to highly effective simple solution. Average performance: Group C equivalent model and algorithms programmed well; some relevant good coding style characteristics; effective simple solution. Below average performance: Group C equivalent algorithms and/or model programmed in a severely limited to limited way; basic coding style characteristics; trivial to lacking in effectiveness simple solution.

4.14.3.4.1 Table 1: Example technical skills

Group	Model (including data model & data structure)	Algorithms
A	**Complex data model in database (eg several interlinked tables)** **Hash tables, lists, stacks, queues, graphs, trees or structures of equivalent standard** **Files(s) organised for direct access** **Complex scientific/mathematical/robotics/ control/business model** **Complex user-defined use of object-orientated programming (OOP) model, eg classes, inheritance, composition, polymorphism, interfaces** **Complex client-server model**	• **Cross-table parameterised SQL** • **Aggregate SQL functions** • **User/CASE-generated DDL script** • **Graph/Tree Traversal** • **List operations** • **Linked list maintenance** • **Stack/Queue Operations** • **Hashing** • **Advanced matrix operations** • **Recursive algorithms** • **Complex user-defined algorithms (eg optimisation, minimisation, scheduling, pattern matching) or equivalent difficulty** • **Mergesort or similarly efficient sort** • **Dynamic generation of objects based on complex user-defined use of OOP model** • **Server-side scripting using request and response objects and server-side extensions for a complex client-server model** • **Calling parameterised Web service APIs and parsing JSON/XML to service a complex client-server model**
B	Simple data model in database (eg two or three interlinked tables) Multi-dimensional arrays Dictionaries Records	• Single table or non-parameterised SQL • Bubble sort • Binary search Writing and reading from files • Simple user defined algorithms (eg a range of mathematical/statistical calculations)

	Text files File(s) organised for sequential access Simple scientific/mathematical /robotics/ control/business model Simple OOP model Simple client-server model	● Generation of objects based on simple OOP model ● Server-side scripting using request and response objects and server-side extensions for a simple client-server model ● Calling Web service APIs and parsing JSON/XML to service a simple client-server model
C	Single-dimensional arrays Appropriate choice of simple data types Single table database	Linear search Simple mathematical calculations (eg average) Non-SQL table access

4.14.3.4.2 Table 2: Coding styles

Style	Characteristic
Excellent	● **Modules (subroutines) with appropriate interfaces.** ● **Loosely coupled modules (subroutines) – module code interacts with other parts of the program through its interface only.** ● **Cohesive modules (subroutines) – module code does just one thing.** ● **Modules(collections of subroutines) – subroutines with common purpose grouped.** ● **Defensive programming.** ● **Good exception handling.**
Good	● Well-designed user interface ● Modularisation of code ● Good use of local variables ● Minimal use of global variables ● Managed casting of types ● Use of constants Appropriate indentation ● Self-documenting code ● Consistent style throughout ● File paths parameterised
Basic	● Meaningful identifier names ● Annotation used effectively where required

6. Testing

could be done as a video?

Marks: 8/75 (11%)

Specification 4.14.5.4 Testing
Students must provide and present in a structured way for example in tabular form, clear evidence of testing. This should take the form of carefully selected and representative samples, which demonstrate the robustness of the complete, or nearly complete, solution/thoroughness of investigation and which demonstrate that the requirements of the solution/investigation have been achieved. The emphasis should be on producing a representative sample in a balanced way and not on recording every possible test and test outcome. Students should explain the tests carried out alongside the evidence for them. This could take the form of:

- an introduction and overview
- the test performed
- its purpose if not self-evident
- the test data
- the expected test outcome
- the actual outcome with a sample of the evidence, for example screen shots of before and after the test, etc, sampled in order to limit volume.

Level	Mark range	Description
4	7-8	**Clear evidence, in the form of carefully selected representative samples, that thorough testing has been carried out.** This **demonstrates** the robustness of the complete or nearly complete solution/thoroughness of investigation and **that the requirements of the solution/ investigation have been achieved**.
3	5-6	Extensive testing has been carried out, but the evidence presented in the form of representative samples does not make clear that all of the core requirements of the solution/ investigation have been achieved. This may be due to some key aspects not being tested or because the evidence is not always presented clearly
2	3-4	Evidence in the form of representative samples of moderately extensive testing, but falling short of demonstrating that the requirements of the solution/ investigation have been achieved and the solution is robust/ investigation

		thorough. The evidence presented is explained.
1	1-2	A small number of tests have been carried out, which demonstrate that some parts of the solution work/some outcomes of the investigation are achieved. The evidence presented may not be entirely clear.

8. Evaluation

Marks: 4/75 (5%)

Specification 4.14.5.5 Evaluation
Students should consider and assess how well the outcome meets its requirements. Students should obtain independent feedback on how well the outcome meets its requirements and discuss this feedback. Some of this feedback could be generated during prototyping. If so, this feedback, and how/why it was taken account must be presented and referenced so it can be found easily. Students should also consider and discuss how the outcome could be improved more realistically if the problem/investigation were to be revisited.

Level	Mark	Description
4	4	● **Full consideration given to how well the outcome meets all of its requirements.** ● **How the outcome could be improved** if the problem was revisited is discussed and given detailed consideration. ● **Independent feedback obtained** of a useful and realistic nature, evaluated and discussed in a meaningful way.
3	3	● Full or nearly full consideration given to how well the outcome meets all of its requirements. ● How the outcome could be improved if the problem was revisited is discussed but consideration given is limited. ● Independent feedback obtained of a useful and realistic nature but is not evaluated and discussed in a meaningful way, if at all.
2	2	● The outcome is discussed but not all aspects are fully addressed either by omission or because some of the requirements have not been met and those requirements not met have been ignored in the evaluation. ● No independent feedback obtained or if obtained is not sufficiently useful or realistic to be evaluated in a meaningful way even if attempted.
1	1	● Some of the outcomes are assessed but only in a superficial way. ● No independent feedback obtained or if obtained is so basic as to be not worthy of evaluation.