Examiner Report	
Qualification Name	Higher Education Qualification
Qualification Level	Diploma
Date/ Series	April 2024
Module	Software Engineering
Question no.	**comments**
A1	This question is in 3 main parts a) This part had two subsections. Many answers to this part were very poor. Few candidates gave the three reasons for common software failure and many failed to give an example. Those minority of candidates who did give a fuller account gave reasonable examples related to the use of ambiguous language possible double meaning of user requirements etc'. The few candidates that went on to cover the next subsection had difficulty in explaining how an engineered approach could reduce the reasons for failure that they explained in the examples from the first subsection. Subsequently very low or no marks were available. b) This question was concerned with the software crisis being linked to lack of agreed best practice and required an example to illustrate the answer. Many candidates were able to recognise the key role of technological change and the inability of software standards to keep pace. However very few candidates went on to give pertinent examples. Those candidates who did achieved full marks. c) This question required candidates to give four examples of how requirements engineering applies rigour to capture and documentation. Very few candidates were able to go beyond an acknowledgement that requirements engineering is based on management, and provides a framework together with documentation to systematically process the requirements. Very few candidates gave the four examples required and those who did attempt an example were unable to give four separate examples but tended to repeat a single aspect such as documentation.

Question no.	comments
A2	This question is in five main parts a) This question required examples of engineered and non-engineered approaches to software design. The specificity of the question to the software design phase was lost on almost all candidates. Those candidates who did attempt an answer often found a mark or two by incidentally mentioning differences in traceability to requirements and having testable designs. b) This part of the question on examples of measurement of system quality was well answered by many candidates who were able to include pertinent items such safety, reliability and testability However many candidates failed to offer relevant examples. c) This part of the question was concerned with the key elements of software costs and how cost-driven methods can compromise software quality. The question required two examples . Many candidates were able to give a reasonable key element of cost and around half of all candidates were able to give at least one example. For some candidates the question was either avoided or the answer failed to address the effect on quality. d) This part of the question required two examples of how software engineering principles might help the effectiveness of a development team. Many candidates failed to fully read the question and did not attempt to provide examples. Those candidates that gave reasonable answers addressed items such as pair programming, collaboration and team structure. e) This part of the question required the identification of two process models. Many candidates were able to provide two models and were able to discuss something about the underlying philosophy, however very few candidates could extend this into a broader description of the subsequent tools and methods.
Question no.	**comments**
A3	This question was the least popular choice for candidates. The question is in three parts. a) This part of the question required an explanation of the meaning of separation of concerns and asked for an example. Many candidates were able to identify the partitioning of software into distinct sections and were able to identify why it is used. However, most candidates did not give an appropriate example subsequently losing the majority of marks available for the question. An example such as the layers of a protocol.

	b) This part required an explanation of ways in which encapsulation can be achieved in OO design. Very few candidates were able to describe the use of access modifiers or to any of the accessors and mutators. A few candidates were able to provide a suitable illustrative example. c) This part offered a diagram showing the MVC design pattern and required candidates to base a revised diagram based on the offered application set in the question The majority of candidates found difficulty in correctly realising an example. Most were unable to correctly provide a correct logic to the diagrams they did attempt. The UML notation was generally not correct. Only a very small minority of candidates gained reasonable marks for this part of the question.

Question no.	**comments**
B4	This question is in 4 main parts a) This part was concerned with discussing evaluation of a software tool for editing and writing software. Candidates were required to discuss five key features. Many candidates were able to mention benefits such as syntax highlighting, autosensing and an ability to embed other tools. For a considerable number of candidates, the question proved difficult and many simply listed one or two editing tool names. b) This part required candidates to refer to the tool mentioned in their answer to part a. The question required the candidate to identify appropriate metrics with which to make a meaningful comparison to any other editing tool which benefits code productivity. Many candidates chose to omit this part completely. Other candidates interpreted the meaning of metrics as purely numeric, subsequently describing CoCoMo models, function points and other quantitative models which bore no relation to the tools from part a. Those candidates that mentioned accuracy of debugging, formatting quality, or accuracy of predictive text amongst other similar metrics tended to gain high marks. c) This part was concerned with a comparison of multi-tool versus IDE development environments. The question required two points in favour and two points against for each tool. Some candidates were able to answer this part, but a large majority restricted the answer to one tool only. For a significant number of candidates, the question proved difficult, and discussions were around irrelevant matters and did not address the question. d) This part required candidates to define a repository schema then discuss issues in using a standardised repository with a bespoke software development company. Very few candidates attempted this part of the question. The answers that were produced gave a reasonable definition, but many failed to note

	that standardised repositories are better suited to product development companies not bespoke companies as the requirements for each project will change. Answers also tended to neglect the possibilities of different naming conventions between projects and different documentation requirements amongst other similar issues in using a standardised product in a non-standardised development environment.

Question no.	comments
B5	This question is in 3 main parts a) This part required a discussion of various costing techniques that the candidate is familiar with and state the most appropriate for the given brief. Most candidates were able to gain reasonable marks with a discussion of techniques. In recommending the appropriate technique many candidates ignored the scenario given in the question preamble and did not mention techniques such as analogy or expert judgement. b) This part was concerned with customer complaints of failure in a software mobile app supplied by the company and asked for four software engineering practices that would reduce product faults. Many candidates were able to answer with SE practices such as Traceability of requirements, testability of design etc’. Many candidates simply mentioned a practice but did not further elaborate on how the practice would improve the failure rate. c) This part was in two sub sections. For i) Candidates were required to describe appropriate test techniques for the type of software. Many candidates found this part difficult with very few mentioning the integrated nature of the testing from unit through to acceptance testing. Consequently, few attempts gained high marks For subsection ii) This part required an example of how a test for inclusion in the tender would be documented as a sample of quality documentation. Many candidates chose to ignore this subsection. The majority of attempts that were made failed to achieve a pass in this subsection. Many did not include a sample that details what was being tested or expected v actual result.

Question no.	Comments
B6	This question is in 3 main parts a) The question asked for four quality techniques that can be applied to a software project to **measure** and **monitor**

	progress during **implementation and testing** Very few candidates answered this part with any degree of success. Many failed to note that they were measures for progress and monitoring at the later stages of the process. Many answers simply listed quality assurance methods or testing. Subsequently gaining no marks. Those answers which did acknowledge the context could have maximised marks by covering Lines of Code, Function points, complexity amongst other relevant measures. b) This part required the defining of four types of software maintenance. Many candidates mentioned at least three and gained some marks. The question further extended into requiring examples of software engineering practice that would support the type of maintenance. Very few candidates were able to provide examples. An approach that mentioned amongst other things, traceability from requirements to code would help corrective maintenance. Code structure and naming helping with adaptive maintenance. These and similar considerations for the other maintenance types would have gained maximum marks. c) This question required an answer that described the three phases of risk management. Many candidates were unable to correctly identify the correct phases. The question extended to requiring candidate to show where in the SDLC stages any particular type of of the phases can be applied. Many candidates ignored this extended part of the question. Some candidates were able to define an appropriate test at a given stage and state and explain why it is appropriate and state actual/expected results together with suggested remedial action for the phase