A10.2-R4: SOFTWARE TESTING AND QUALITY MANAGEMENT

NOTE:

___

IMPORTANT INSTRUCTIONS:

1.   There are **TWO PARTS** in this Module/Paper.   **PART ONE** contains **FOUR** questions and **PART TWO** contains **FIVE** questions.

2.   **PART ONE** is to be answered in the **OMR ANSWER SHEET** only, supplied with the question paper, as per the instructions contained therein.   **PART ONE** is **NOT** to be answered in the answer book.

3.   Maximum time allotted for **PART ONE** is **ONE HOUR**. Answer book for **PART TWO** will be supplied at the table when the answer sheet for **PART ONE** is returned. However, candidates, who complete **PART ONE** earlier than one hour, can collect the answer book for **PART TWO** immediately after handing over the answer sheet for **PART ONE**.

___

TOTAL TIME: 3 HOURS   **TOTAL MARKS: 100**
                                                                            (PART ONE – 40; PART TWO – 60)

PART ONE
(Answer all the questions)

1.   **Each question below gives a multiple choice of answers. Choose the most appropriate one and enter in the "OMR" answer sheet supplied with the question paper, following instructions therein.   (1x10)**

1.1   Integration testing mainly targets to detect which one of the following types of errors?
A)   Design errors
B)   Algorithm errors
C)   Requirements errors
D)   Interfacing errors

1.2   Data flow testing is a control structure testing technique where the criteria used to design test cases is that they
A)   rely on basis path testing
B)   exercise the logical conditions in a program module
C)   select test paths based on the locations and uses of variables
D)   focus on testing the validity of loop constructs

1.3   During unit testing of a software, why is it important to test the boundary values?
A)   It reduces test costs as boundary values are easily computed by hand.
B)   Debugging is easier when testing boundary values.
C)   The correct execution of a function on all boundary values proves that the function is correct.
D)   Programming the boundary conditions is usually error-prone in practice.

1.4   Which one of the following attributes of a program can be inferred from its Cyclomatic complexity?
A)   Computational complexity
B)   Lines of code (LoC)
C)   Executable code size
D)   Understandability

1.5   Which one of the following is true of a pure top-down integration testing approach?
A)   Requires only stubs to be developed for testing
B)   Requires only drivers to be developed for testing
C)   Requires both stubs and drivers to be developed for testing
D)   Requires neither stubs nor drivers to be developed for testing

1.6   Which one of the following types of program models is normally used to design an integration test plan?
A)   CFG (Control Flow Graph)
B)   DFD (Data Flow Graph)
C)   Structure chart
D)   State chart

1.7   Which one of the following is the basic focus of modern quality paradigms?
A)   Process assurance
B)   Product assurance
C)   Thorough testing
D)   Thorough testing and rejection of bad products

1.8   Which one of the following would be true of a development organization assessed at the SEI CMM *repeatable* level of software development?
A)   The software developed by the organization is guaranteed to be of high quality.
B)   Success in the development of a software can be repeated by the organization in similar software development projects.
C)   Success in development of a software can be repeated by the organization in all software development projects that the organization might undertake.
D)   When the same development team is chosen to develop another software, they can repeat their success.

1.9   Which one of the following is not a software verification technique?
A)   Review
B)   Simulation
C)   Unit testing
D)   Stress testing

1.10   In which one of the following SDLC models, testing activities are spread over the entire development life cycle?
A)   Iterative waterfall model
B)   V model
C)   Prototyping model
D)   Classical waterfall model

2. **Each statement below is either TRUE or FALSE. Choose the most appropriate one and ENTER in the "OMR" answer sheet supplied with the question paper, following instructions therein.** **(1x10)**

2.1   Adherence to coding standards is normally checked during the system testing stage.

2.2   Introduction of additional sequence type of statements in a program can not cause the Cyclomatic complexity of the program to increase.

2.3   If a base class is thoroughly tested, then the inherited methods in the derived class need not be tested.

2.4   Code walkthrough for a module is normally carried out after unit testing of the module is complete.

2.5   As testing continues, the rate of growth of reliability slows down representing a diminishing return of reliability growth with testing effort.

2.6   If an organization assessed at SEI CMM level 2, then it must be having a documented process and must be following it rigorously.

2.7   While verification is concerned with phase containment of errors, the aim of validation is that the final product be error free.

2.8   When any of the objects under configuration control is changed and the configuration is updated, a new baseline gets formed.

2.9   Branch coverage is a stronger test coverage criterion compared to path coverage-based testing.

2.10 Once the McCabe's Cyclomatic complexity of a large program has been determined, the set of linearly independent paths of the program are easily identified.

3. **Match words and phrases in column X with the closest related meaning/ word(s)/phrase(s) in column Y. Enter your selection in the "OMR" answer sheet supplied with the question paper, following instructions therein.** **(1x10)**

	X		Y
3.1	Software debugging	**A.**	Condition coverage testing
3.2	Black-box testing approach	**B.**	Repeatable process
3.3	White-box testing approach	**C.**	Carried out after a debugging session is complete.
3.4	SEI-CMM Level 3	**D.**	Mutation testing
3.5	Regression testing	**E.**	A technique used for test coverage measurement.
3.6	Fault-based testing	**F.**	Beta testing
3.7	Performance testing	**G.**	Can detect unreachable code
3.8	Static analysis	**H.**	Technique used to ensure that the development steps have been carried out correctly.
3.9	Dynamic analysis	**I.**	Testing of a category of non-functional requirements
3.10	A type of system testing	**J.**	Cause-effect graphing
		K.	Quality assurance process
		L.	Boundary value analysis is helpful
		M.	Steps to detect and correct errors

4. **Each statement below has a blank space to fit one of the word(s) or phrase(s) in the list below. Enter your choice in the "OMR" answer sheet supplied with the question paper, following instructions therein.** **(1x10)**

A.	Scenario coverage	**B.**	Control Flow Graph	**C.**	Grey box
D.	Branch coverage	**E.**	Testability	**F.**	Test script
G.	Module interfacing	**H.**	SRS document	**I.**	Security testing
J.	Orthogonal array testing	**K.**	Design document	**L.**	Dataflow diagram
M.	Equivalence testing				

4.1   A _____ is an encoding of a test case as a short program.

4.2   _____ of a requirement denotes the extent to which it is possible to determine whether an implementation of the requirement conforms to it in terms of both functionality and performance.

4.3   _____ is a stronger testing criterion than statement coverage criterion.

4.4   A _____ of a program describes the sequence in which the different instructions of the program get executed.

4.5   _____ targets to test the pair-wise combinations of all the selected variables.

4.6   The aim of integration testing is to determine _____ errors.

4.7   A test suite designed based on the design models of an object-oriented program is called _____ test suite.

4.8   System test suite is designed based on the _____.

4.9   _____ is a performance testing technique.

4.10   A popular use case-based testing technique is _____ testing.

PART TWO
(Answer **ANY FOUR** questions)

5.
a) Discuss the relative merits of adopting ISO 9001 certification or the SEI CMM-based quality practices by a software organization.
b) What do you understand by defect prevention? Explain use of periodic reviews and pareto analysis as defect prevention techniques.
c) What do you understand by Key Process Area (KPA), in the context of SEI CMM? Would an organization encounter any problems, if it tries to implement higher level SEI CMM KPAs before achieving the lower level KPAs? Justify your answer.

(5+5+5)

6.
a) What are alpha, beta, and acceptance testing? How do these different types of testing of software differ from one another? Explain your answer with respect to who carries out the test, when is the test carried out, and the objective of the test.
b) What is a test plan? Discuss a test plan template and briefly explain each item in your test plan template.
c) Briefly describe the V model. Outline its strengths and weaknesses.

(5+5+5)

7.
a) What do you understand by the term phase containment of errors? Why phase containment of errors is is important? How can phase containment of errors be achieved?
b) What is the difference between code inspection and code walkthrough? Give an example of at least two types of errors detected during code inspection and at least two types of errors detected during code walkthrough?
c) In a software development organization whose responsibility is it to ensure that the products are of high quality? Explain the principal tasks they perform to meet this responsibility.

(5+5+5)

8.
a) What do you understand by a symbolic debugger? What are the other popular techniques for debugging? Briefly outline how debugging is performed by using a symbolic debugger.
b) What do you mean by black-box testing? Design black-box test suites for a function that checks whether a character string (of up to 25 characters in length) is a palindrome.

(7+8)

9.
a) What do you understand by the term integration testing? Which types of defects are uncovered during integration testing? What are the different types of integration testing methods that can be used to carry out integration testing of a large software product? Compare the merits and demerits of these different integration testing strategies.
b) When in the life-cycle of a software product, are the non-functional requirements tested? How are the different non-functional requirements tested? Explain your answer with respect to various categories of non-functional requirements.
c) What do you understand by test coverage analysis? Why is it necessary to perform test coverage analysis? Define at least two test coverage metrics.

(5+5+5)