

B4.3-R3: SOFTWARE TESTING AND QUALITY MANAGEMENT

NOTE:

1. Answer question 1 and any FOUR from questions 2 to 7.
2. Parts of the same question should be answered together and in the same sequence.

Time: 3 Hours

Total Marks: 100

1. State whether the following statements are **TRUE** or **FALSE**. In each case, justify your answer using one or two sentences.
 - a) If a class is thoroughly tested, then for any class derived from this class, the inherited methods need not be tested.
 - b) The big-bang approach is preferred for integration testing of small programs having very few modules.
 - c) Software Testing detects failures whereas Source Code inspection identifies errors.
 - d) Cyclomatic complexity correlates well with the testing difficulty of a program unit.
 - e) Test coverage analyzers are essentially static analyzers.
 - f) System testing of an object-oriented implementation of a system is considerably simpler than that of a procedural implementation of the same system.
 - g) Modern quality assurance paradigms focus on realizing high quality products through robust product testing.

(7x4)

2.
 - a) What do you understand by “code review”? What are the different types of code reviews that are normally conducted? Explain when and how code review meetings are to be conducted.
 - b) Why code review is considered to be a more efficient way to remove errors from code compared to testing?
 - c) What do you understand by path testing? For a program containing N binary branches how many test cases are necessary for path coverage? For a program containing N number of M-ary branches, how many test cases are necessary for path coverage?

(6+6+6)

3.
 - a) What do you understand by performance testing? What are the different types of performance testing?
 - b) Among the different development phases of life cycle, testing typically requires the maximum effort. Identify the main reasons behind the large effort necessary for this phase.
 - c) Design the black-box test suite for a program that accepts two strings and checks if the first string is a substring of the second string and displays the number of times the first string occurs in the second string.

(6+6+6)

4.
 - a) What are alpha, beta, and acceptance testing? What are the primary differences among these different types of testing of a software product? Explain your answer with respect to who carries out the test, when is the test carried out, and the objective of the test.
 - b) In a software development organization, identify the persons responsible for carrying out the quality assurance activities. Explain the principal tasks they perform to meet this responsibility.

- c) Suppose a developed software has successfully passed all the three levels of testing, i.e. unit testing, integration testing, and system testing. Can we claim that the software is defect free? Justify your answer.

(6+6+6)

5.

- a) List two metrics that can be determined from an analysis of a program's source code and which would correlate well with the reliability of the delivered software.
- b) What do you understand by repeatable software development? Organizations assessed at which level SEI CMM maturity achieve repeatable software development?
- c) Can reliability of a software product be determined by estimating the number of latent defects in the software? If your answer is "yes", explain how reliability can be determined from an estimation of the number of latent defects in a software product. If your answer is "no", explain why can't reliability of a software product be determined from an estimate of the number of latent defects.

(6+6+6)

6.

- a) What is the difference between process metrics and product metrics? Give two examples of each.
- b) Why effective testing real-time and embedded systems are considered more difficult than testing traditional systems? Explain a satisfactory scheme to test real-time and embedded systems.
- c) What is coding standard? Identify the problems that might occur if the engineers of an organization do not adhere to any coding standard?

(6+6+6)

7.

- a) Distinguish between the static and dynamic analysis of a program. Explain at least one metric that a static analysis tool reports and at least one metric that a dynamic analysis tool reports. How are these metrics useful?
- b) What do you understand by software configuration management? Why is software configuration management necessary for realizing high quality products?
- c) What is meant by structural complexity of a program? Define a metric for measuring the structural complexity of a program. How is structural complexity of a program different from its computational complexity?

(6+6+6)