

**Department of Applied Mathematics
Delhi Technological University**

ASSIGNMENT

**Course: MC-310 Software Engineering
(2019-2020 Even Semester)**

1. Explain how the universal use of the Web has changed software systems.
2. Suggest why it is important to make a distinction between developing the user requirements and developing system requirements in the requirements engineering process.
3. Explain why incremental development is the most effective approach for developing business software systems. Why is this model less appropriate for real-time systems engineering?
4. Suggest how an engineer responsible for drawing up a system requirements specification might keep track of the relationships between functional and non-functional requirements.
5. Explain all the levels of COCOMO model. Assume that the size of an organic software product has been estimated to be 32,000 lines of code. Determine the effort required to develop the software product and the nominal development time.
6. Explain why testing can only detect the presence of errors, not their absence.
7. Assume that the initial failure intensity is 20 failures/CPU hr. The failure intensity decay parameter is 0.02/failure. We have experienced 100 failures up to this time.
 - (i) Determine the current failure intensity.
 - (ii) Find the decrement of failure intensity per failure.
 - (iii) Calculate the failures experienced and failure intensity after 20 and 100 CPU hrs. of execution.
 - (iv) Compute additional failures and additional execution time required to reach the failure intensity objective of 2 failures/CPU hr.

Use Logarithmic Poisson execution time model of software reliability for the calculations.

8. Explain why a software system that is used in a real-world environment must change or become progressively less useful.
9. Explain why the environment in which a computer-based system is installed may have unanticipated effects on the system that lead to system failure. Illustrate

your answer with an example.

10. Explain why there is a need for risk assessment to be a continuing process from the early stages of requirements engineering through to the operational use of a system.
11. Explain the steps involved in project planning. Discuss the various factors that affect a project plan.
12. Explain the SEI Capability Maturity Model (CMM).
13. “The best programmers do not always make the best software managers.” Explain.
14. Explain why the intangibility of software systems poses special problems for software project management.
15. What is risk? Identify six possible risks that could arise in software projects. Discuss how you would manage those risks at different phases.
16. What is software quality? Discuss software quality attributes.
17. Explain why the process of project planning is iterative and why a plan must be continually reviewed during a software project.
18. Some very large software projects involve writing millions of lines of code. Explain why the effort estimation models, such as COCOMO, might not work well when applied to very large systems.
19. What are the objectives of testing? Explain why testing can only detect the presence of errors, not their absence.
20. What are the different levels of testing and the goals of the different levels? For each level, specify the most suitable testing approach.
21. What is software reliability? Discuss the following models of software reliability
 - (a) Basic Execution Time model
 - (b) Jelinski-Moranda model
22. List some problems that will come up if the methods used for developing small software are used for developing large software systems.
23. If the primary goal is to make software maintainable, list some of the things you will do and some of the things you will not do during coding and testing.
24. Suggest five possible problems that could arise if a company does not develop effective configuration management policies and processes.
25. Discuss reverse engineering and re-engineering.

END