B3.3-R4: SOFTWARE ENGINEERING & CASE TOOLS

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.

a) What do you understand by software crisis? What are the possible remedies for it?

b) What are the major advantages of first constructing a working prototype before developing the actual software? Is there any disadvantage of constructing a prototype?

c) What are the important differences between a model-oriented specification method and a property-oriented specification method? Name at least one representative popular property-oriented specification technique, and one representative model-oriented specification technique.

d) What do you understand by a layered software design? What are the advantages of a layered design? Explain your answer by using a suitable example.

e) What is User Interface Design? Explain in brief.

f) What is a baseline in the context of software configuration management? After a revision of any configuration items, explain how a baseline can be updated to form a new baseline?

g) Define two metrics based on which the user interface of an application software can be evaluated.

(7x4)

2.

a) Draw a class diagram using the UML syntax to represent the fact that a travel agency owns fleet of 100 vehicles consisting of vehicles of the types Tata Indica, Maruti van, and Mahindra Xylo. The regular customers of the travel agency can rent any vehicle they want. The details of the customers such as the name, address, and phone number are maintained by the agency.

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.

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.

(6+6+6)

3.

a) Identify the criteria that you would use to decide which one of two otherwise correct alternate function-oriented design solutions to a problem is superior.

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 the principles of abstraction and decomposition? Why these two principles are considered important in software engineering?

(6+6+6)

4.

a) What are the reasons to opt for reverse engineering?

b) What is Change Control Process? Explain.

c) What do you understand by ”build-and-fix” style of software development? Diagrammatically depict the typical activities in this style of development and their ordering. Identify at least four major problems that would arise, if a large professional software development project is undertaken using a ”build-and-fix” style of software development.

(6+6+6)

5.

a) Define three metrics based on which it would be possible to determine the goodness of an object-oriented software design.

b) Explain why reuse is more difficult in software development compared to hardware development. How can reusability of a piece of a code be enhanced?

c) What do you understand by repeatable software development? Organizations assessed at which level of SEI CMM maturity achieve repeatable software development?

(6+6+6)

6.

a) What is the difference between the functional and the non-functional requirements of a system? Identify at least two functional requirements of a typical bank ATM (Automated Teller Machine) system. Also identify one non-functional requirement for an ATM system.

b) Explain the cleanroom approach to software development. What are its advantages over conventional testing?

c) What do you mean by an abstract class? Abstract classes cannot have instances. What is then the use of defining abstract classes?

(6+6+6)

7.

a) What are the different types of maintenance that a software product might need? If a software product costed Rs. 10,000,000/- for development, compute the annual maintenance cost given that every year approximately 5% of the code needs modification. Identify the factors which render the maintenance cost estimation inaccurate?

b) Through a simple plot explain how the reliability of a software product changes over its lifetime. Draw the reliability change for a hardware product over its life time and explain why the two plots look so different.

c) What do you mean by modular design? What are the different factors that affect the modularity of a design? How can you assess the modularity of a design? What are the advantages of a modular design?

(6+6+6)