B3.3-R3: SOFTWARE ENGINEERING AND 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) Do you agree with the following statement: “The focus of exploratory programming is error correction while the software engineering principles emphasize error prevention?” Give the reasoning behind your answer.

b) What is the difference between functional and structural testing? Can one be used in place of another?

c) What do you understand by format requirements specification? What are its advantages over informal or semiformal specification?

d) What is the difference between an error and a failure? Give one example of each.

e) What is a design pattern? What are the advantages of using design patterns? Name and briefly explain at least one design pattern.

f) What do you understand by component-based software engineering? What are its advantages over traditional software engineering?

g) Identify two important problems that you would face if you are developing a large software product and you are not using any configuration management tools.

(7x4)

2.

a) Explain different types of requirements problems that should be identified and resolved during the requirements analysis activity. Illustrate by a suitable example.

b) Identify the CASE support that can be availed of during a large maintenance effort concerning a large legacy software.

c) What do you understand by the term integration testing? Which types of defects are uncovered during integration testing? Describe 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.

(6+6+6)

3.

a) What do you understand by data abstraction? How does data abstraction help in reducing the coupling in a design solution?

b) When during the development process is the compliance with coding standards checked? List two coding standards each for (i) enhancing readability of the code, (ii) reuse of the code.

c) What do you mean by the term software `reengineering` and `reverse engineering`? Why are these required?

(6+6+6)

4.

a) What is the difference between a coding standard and a coding guideline? Why are these considered important in software development?

b) What do you understand by “build-and-fix” model of software development? Give the shortcomings of this approach.

c) What do you understand by association relation among classes? Give an example of binary and a ternary association relation between two classes.

(6+6+6)

5.

a) What criteria would you use to distinguish a good software design from a bad design? Discuss some metrics that can be used to judge the goodness of a 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) List four metrics that can be determined from an analysis of a program’s source code and would correlate well with the reliability of the delivered software.

(6+6+6)

6.

a) Explain why the effort, time and cost required to develop a program using the build and fix style increase exponentially with the size of the program? How do software engineering principles help tackle this rapid rise in development time and cost?

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

c) What is a real-time system? Why can’t traditional design techniques be satisfactorily used for designing real-time systems? Explain the extension to the traditional design technique necessary for designing real-time systems.

(6+6+6)

7.

a) Do you agree with the following assertion? “An object-oriented software can be adequately tested by testing the methods of all the classes in the system and then performing integration and system testing.” If your answer is “yes”, explain how the methods can be tested; if your answer is “no”, explain how an object-oriented program can be adequately tested.

b) 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.

c) Why is it important to properly document a software product? What are the different types of documents that need to be developed?

(6+6+6)