**Suggestion**

**Software Engineering (CS602)**

1. Why SRS is called the black-box specification of a system?
2. Briefly explain the characteristics of a good SRS.
3. Analyze different phases of SDLC with diagram.
4. Explain the Spiral model with a neat diagram.
5. What is FP? Determine the method to compute FP in detail?
6. What are the advantages of function point over LOC?
7. Explain different symbols used in DFD?
8. Distinguish between functional and nonfunctional requirements
9. Differentiate between function-oriented and object-oriented design.
10. Discuss the advantages of prototype model over waterfall model.
11. Distinguish between logical DFD and Physical DFD
12. Explain the importance of feasibility study for a project?
13. What are the different types of coupling and cohesion?
14. How are the concepts of cohesion and coupling useful in arriving at good software design?
15. What are the top-down and bottom-up designs?
16. What is testing? Explain the types and levels of testing.
17. Consider a software project using semi-detached mode with 300KLOC. Find out effort estimation, development time and person estimation.
18. Explain acceptance testing.
19. What are the various categories of risks? Give an overview about Risk Management
20. What is the effect of risk management activity over the total cost of project?
21. What is SQA?
22. Assume a software project with 5 activities T1 to T5. Duration of 5 activities in weeks are 3, 2,3,5,2 respectively. T2 and T4can start when T1 is complete. T3 can start when T2 is complete.T5 can start when both T3 and T4 are complete.
23. Draw activity network for the project. When is the latest start date of the activity T3? What is the float of the activity T4? Which activities are on the critical path?
24. Explain when and why are PERT charts and Gantt charts used in case of project management.
25. Compare between verification and validation.
26. Explain the nomenclature of black box testing and white box testing.
27. Draw the control flow graph for the following function. From that determine its cyclomatic complexity and the independent paths.

void max(int a, int b, int c) {

if (a > b && a > c) {

printf("a is greatest");

} else if (b > c) {

printf("b is greatest");

} else {

printf("c is greatest");

}

return;

}

1. Draw the control flow graph for the below code. From that determine its cyclomatic complexity and the independent paths.

int compute\_gcd(int x, int y)

{

While(x=!y) { if(x>y) then x=y; else y=y-x;}

return x;

}

1. A 400KLOC embedded system is to be developed and you have a choice of hiring from two developers. One developer has very highly capable application experience (0.82) with very low programming language experience (1.14) and other has very low application experience (0.29) and high programming language experience (0.95). Who will the choice with respect to Total Effort, Development Time?
2. What is partitioning?
3. Which lifecycle models would you follow for developing following applications? Justify your selection. 1) a game 2) a text editor c) Banking application d)Online ticket booking application
4. Illustrate the project planning activities with a diagram.
5. Discuss the role of stub and driver.
6. “Software does not wear out or tear out.” Explain.
7. “Non-functional requirements are also essential for customer satisfaction”-Justify this statement with suitable justification.
8. Demonstrate the software engineering layers with a neat diagram.

37. Write short notes

1. COCOMO Model
2. Roles of a system analyst
3. Acceptance testing
4. PERT chart and Gantt chart
5. Integration testing
6. Boundary value analysis
7. All UML Diagrams in detail
8. Structure chart
9. reliability metrics
10. Feasibility analysis
11. RAD Model
12. V-model