**ASSIGNMENT NO.01(Unit-01, CO-01)**

**Session 2023-24**

**SUB: Software Engineering (KCS-601)**

1. How is software different from a program? How it is different from hardware?
2. List the impact of software engineering on developing software.
3. What are the four important attributes that all software products should have?
4. Discuss the characteristics of the software.
5. What are the different software components?
6. What are the driving factors that determine the cost of any software?
7. What are the responsibilities and challenges for software engineers?
8. Describe the software crisis.
9. Why is it difficult to build correct software?
10. What are the desired characteristics of a software process model?
11. Discuss different phases of the classical waterfall model.
12. Discuss the advantages of the waterfall model of SDLC. What are the limitations of waterfall model?
13. What is iterative life cycle model? Explain its working.
14. What is evolutionary life cycle model? Explain its working.
15. Explain what is a software prototype?
16. What is capability maturity model (CMM)? Explain its structure.
17. Do you agree with the following statement? “The emphasis of exploratory programming is on error correction while the software engineering practices emphasize error prevention”. Give reason to justify your answer.
18. What factors have contributed to the making of the present software crisis? What are the possible solutions to the present software crisis?
19. What are the desired characteristics of a software process model?
20. What are the important activities that are carried out during the feasibility study phase?
21. What is a prototype? Under what circumstances is it beneficial to construct a prototype? Does the construction of a prototype always increase the overall cost of software development?
22. Explain why the spiral life cycle model is considered to be a Meta model.
23. Illustrate the working of spiral model. Also write down the advantages of using spiral model.

**ASSIGNMENT NO.02(Unit-02, CO-02)**

**Session 2023-24**

**SUB: Software Engineering (KCS-601)**

1. What are the objectives of requirement analysis?
2. Define functional and non-functional requirements.
3. What is feasibility study? Discuss different types of feasibility studies.
4. What do you mean by a requirement in software engineering? What are the different classes of requirements in software engineering?
5. What do you mean by good SRS? What are the characteristics of good SRS?
6. Some researchers feel that “feasibility analysis should be a parallel activity with requirement engineering”. Comment.
7. What is the purpose of systematic modeling for the purpose of analysis?
8. Explain the representative s of DFD and ERD. What is the difference between the term ‘entity’ in DFD and ERD?
9. What are the different object-oriented models used as modern modeling techniques? Explain their focuses of abstraction.
10. Why is the SRS document also known as the black-box specification of a system?
11. What is it important for a software development organization to obtain ISO 9001 certification?
12. What is repeatable software development? Organizations assessed at which level of SEI CMM maturity achieve repeatable software development?
13. What are different component of SRS document? Also list the benefits of SRS document.
14. What do you mean by software quality? Why does one need to measure the quality of the software? What are the common quality metrics for software?
15. Write notes on software quality assurance (SQA).
16. Describe the difference between verification and validation and explain why validation is particularly difficult process?
17. Describe three principle activities involved in software quality management.
18. How do you define reliability? Discuss various models for reliability allocation.
19. Discuss reliability assessment models.
20. What do you mean by software reliability and software availability? Also discuss how they are measured?

**ASSIGNMENT NO.03(Unit-03, CO-03)**

**Session 2023-24**

**SUB: Software Engineering (KCS-601)**

1. What characteristics of software are important for measurement? Why does the industry need separate metrics for software products and processes?
2. Describe the various strategies of design. Which design strategies is most popular and practical?
3. Explain how size oriented metrics differ from function oriented metrics. Discuss the pros and cons of each.
4. Explain the following: Coupling, cohesion, Structure chart, and Pseudo-code.
5. Define the following with example:
6. Function oriented design.
7. Top-down and bottom-up design
8. Object Oriented design
9. What is cohesion and coupling in the context of software design? How would you improve a software design that displays very low cohesion and high coupling?
10. Explain with examples, top-down and bottom-up approach in software design.
11. Why is it considered a dominating factor in the success of a project? What are the main attributes of a software project to be planned during project planning?
12. What are the estimations of development efforts and duration of the project based upon them? Explain how intermediate COCOMO provides better estimation as compared to basic COCOMO.
13. Illustrate Halestead’s software science of software measurement and metrics.
14. How can be calculate the cyclomatic complexity in basic path testing? Assume a procedure and calculate the cyclomatic complexity for that procedure.
15. What are the top-down and bottom-up design methodologies?
16. What are their areas of applications? Give at least two examples of software development problems where these methodologies can be followed for design.
17. Using a schematic diagram show the order in which the following are estimated in the COCOMO estimation technique: cost, effort, duration, size. Suppose you are developing a software product in the organic mode. You have estimated the size of the product to be about 100,000 lines of code. Compute the nominal effort and the development time.

**ASSIGNMENT NO.04(Unit-04, CO-04)**

**Session 2023-24**

**SUB: Software Engineering (KCS-601)**

1. What do you mean by test case? Describe top-down and bottom-up testing strategies along with their advantages and disadvantages. Illustrate Regression testing also.
2. List the various objectives of testing.
3. Discuss the alpha and beta testing of products along with their advantages and disadvantages.
4. Write short notes on :
5. Unit Testing
6. Integration testing
7. Acceptance testing
8. What are drivers and stubs modules in context of integration and unit testing of a software product? Why are stubs and drivers modules required?
9. Explain white-box testing in detail.
10. Explain regression and acceptance testing in detail.
11. Discuss the differences between black-box and structural testing and suggest how they can be used together in the defect testing process?
12. What is the difference between coding standards and coding guidelines? Why these are considered important in software development organization?
13. Write down any five important coding standards and guidelines that you would recommend.
14. What is code walk through? What are important rules for code review and code walk through? Also explain its benefits of review and code walk through.
15. What is the need for testing in software engineering? When should testing start in SDLC? Distinguish between black-box and white-box testing.
16. What is regression testing? How are the best cases selected from original test suite for regression testing?
17. When is the compliance to coding standards checked during the development process? List two coding standards one each for (i) enhancing the readability of the code and (ii) reuse of the code. Differentiate between alpha testing and beta testing of products.
18. What are the top-down and bottom-up testing methodologies? What are their areas of applications? Give at least two examples of software development problems where these methodologies can be followed for testing.

**ASSIGNMENT NO.05(Unit-05, CO-05)**

**Session 2023-24**

**SUB: Software Engineering (KCS-601)**

1. What is McCabe’s Cyclomatic complexity? Why is it important to compute Cyclomatic complexity of a procedure? Compute the Cyclomatic complexity of a module with 3 ‘if-else’ blocks, 2 ‘do-while’ loops, 2 ‘for’ loops, and total (including these mentioned blocks) 125 statements.
2. What do you mean by software maintenance? What is adaptive maintenance of the software? Why does the environment of the software change?
3. What do you understand by the terms: a CASE tool and a CASE environment?
4. Discuss the role of the data dictionary in a CASE environment.
5. What features are supported by a good prototyping CASE tool?
6. During code generation what supports might be expected from CASE tools?
7. What are the software project management supports that might be expected from CASE tools?
8. What are the software reverse engineering supports that might be available from CASE tools?
9. What are the different types of maintenance that a software product might need? Why is such maintenance required? If a software product costs Rs.1,000,000 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.
10. Suppose you have estimated the nominal development time of a moderate-sized software product to be 5 months. You have also estimated that it will cost Rs.50,000 to develop the software product. Now, customer wants to accelerate the delivery time by 10%. How much additional cost would you charge the customer for this accelerated delivery? Irrespective of whether you take less time or more time to develop the product, you are essentially developing the same product. Why then does the effort depend on the duration over which you develop the product?
11. Describe COCOMO model. Suppose a project is to be developed in organic mode and you have estimated the size of product to be about 100,000 LOC. Compute the nominal effort and development time.
12. What are different categories of software development projects according to the COCOMO estimation model? Give examples of software product development projects belonging to each of these categories.
13. What is the importance of software maintenance?
14. Why the cost of maintenance is so high? Explain
15. What is Boehm’s cost estimation model for software maintenance? Explain.
16. Discuss the causes of software maintenance problems.
17. What are various types of software maintenance? Differentiate between perfective and corrective maintenance.
18. How do you estimate software maintenance costs? Explain.
19. What are the ways for reducing the need of software maintenance? Illustrate.
20. What is reverse engineering? Discuss levels of reverse engineering.
21. What is re-engineering? Differentiate between re-engineering and new development.
22. What are appropriate reverse engineering tools? Discuss it.
23. What are the benefits of software re-engineering?