



INSTITUTE OF AERONAUTICAL ENGINEERING (Autonomous)

Dundigal, Hyderabad - 500 043

COMPUTER SCIENCE AND ENGINEERING(DATA SCIENCE)

QUESTION BANK

Course Title	SOFTWARE ENGINEERING				
Course Code	ACDC04				
Program	B.Tech				
Semester	V	CSE(DS)			
Course Type	Core				
Regulation	IARE - UG20				
Course Structure	Theory			Practical	
	Lecture	Tutorials	Credits	Laboratory	Credits
	3	-	3	-	-
Course Coordinator	Ms. Y Sujana, Assistant Professor				

COURSE OBJECTIVES:

The students will try to learn:

I	The elicited requirements for a software development life cycles.
II	The design considerations for enterprise integration and deployment.
III	Quality assurance techniques and testing methodologies.
IV	The plan for a software project that includes the size , effort, schedule, resource allocation, configuration control, and project risk.

COURSE OUTCOMES:

After successful completion of the course, students should be able to:

CO 1	Illustrate process models, approaches and techniques for managing a software development process.	Understand
CO 2	Summarize the importance of project planning activities that accurately help in selection and initiation of individual projects and portfolios of projects in the enterprise.	Understand
CO 3	Explain software design model and behavior of a software system.	Understand
CO 4	Develop the approaches for implementaion,verification and validation including static analysis and reviews.	Apply
CO 5	Demonstrate the concept of risk management through risk identification, risk measurement and mitigation.	Understand

CO 6	Make use of earned value analysis and project metric for scheduling and improving the quality of software.	Apply
------	---	-------

QUESTION BANK:

MODULE I				
SOFTWARE PROCESS AND PROJECT MANAGEMENT				
PART A-PROBLEM SOLVING AND CRITICAL THINKING QUESTIONS				
Q.No	QUESTION	Taxonomy	How does this subsume the level	CO's
1	A set of actions for the communication activity based on the requests of client.Develop.	Apply	—	CO 1
2	Elucidate which software model is suggested if the problem stated by the client have uncertainties which lead to loss if it not planned and solved.?	Understand	The learner to explain the incremental process model.	CO 1
3	The systems developed as prototypes should not normally be used as production systems.Explain	Understand	—	CO 1
4	Which software process model is good for risk management? Explain the model. Describe how the model is used to layout the objectives, risks and plans for quality improvement.	Understand	The learner to know the concept of Iterative and Incremental Process model	CO 1
5	What is the importance of process models in Software development?List out the parameters to choose appropriate process model for a software project?	Understand	—	CO 1
6	The tools used to monitor the requirements specification in the development of software.Describe it.	understand	—	CO 1

7	Compare and contrast the estimation metrics used in LOC based analysis and FP based analysis techniques with an example.	Understand	The learner to know the concept of Estimation Metrics: LOC vs. FP	CO 1
8	The prototyping method of software development. Under what circumstances prototyping method is appropriate over other methods. Discuss it.	Understand	The learner to explain the Prototyping Method of Software Development:	CO 1
9	Consider 7 functions with their estimated lines of code. Average productivity based on historical data is 620 LOC/pm and labour rate is Rs. 8000 per mnth. Find the total estimates project cost and effort? F1 – 2340 , F2 – 5380, F3 – 6800 , F4 – 3350 , F5 – 4950 , F6 – 2140 , F7 – 8400	Analyze	—	CO 1
10	”Empirical software engineering made a difference in software development” , Justify the statement.	Evaluate	—	CO 1

PART B-LONG ANSWER QUESTIONS

1	Explain the evolving role of software?	Understand	—	CO 1
2	Define software and explain the various characteristics of software.	Understand	—	CO 1
3	Discuss on various types of software myths and the true aspects of these myths.	Understand	—	CO 1
4	Explain the layered technology of software engineering.	Understand	The learner to define software Engineering and know the concept of the layered technology of software engineering.	CO 1
5	Explain waterfall model with the help of a diagram. Give reasons for its failure.	Understand	The learner to know the concept of waterfall model.	CO 1

6	Explain the use of incremental process model with neat diagram.	Understand	The learner to know the concept of incremental process model.	CO 1
7	Discuss briefly about Evolutionary process models with neat diagram.	Understand	—	CO 1
8	Explain briefly about the Spiral model with neat sketch.	Understand	The learner the concept of the Spiral model	CO 1
9	Describe the use of concurrent development model (or) concurrent engineering model.	Understand	The learner to recall the the advantages of iterative development and know the difference between iterative development with Incremental delivery approach.	CO 1
10	Explain briefly about Component- Based Development model.	Understand	The learner know the Component- Based Development model.	CO 1
11	Discuss briefly about Aspect-Oriented Software Development model.	Understand	—	CO 1
12	Elaborate the importance of Estimation with examples.	Evaluate	The learner know the concept of Estimation.	CO 1
13	Explain in detail about LOC-based and FP-based estimation with an example.	Understand	The learner know the concept of LOC-based and FP-based estimation	CO 1
14	Describe COCOMO model and explain its importance.	Understand	The learner to define the COCOMO model.	CO 1
15	Explain the importance of scheduling in software development.	Understand	—	CO 1
16	Discuss the importance of earned value analysis.	Understand	The learner to know the importance of earned value analysis.	CO 1
17	Explain about reactive versus proactive risk strategies.	Understand	—	CO 1
18	Discuss in detail about different software risks.	Understand	The learner to define the software risks.	CO 1
19	Demonstrate the concept of earned value analysis.	Understand	—	CO 1
20	Write about risk management.	Understand	—	CO 1

PART-C - SHORT ANSWER QUESTIONS				
1	List out all the applications of software.	Remember	—	CO 1
2	List out the types of software myths?	Remember	—	CO 1
3	List out all the umbrella activities in process framework.	Understand	The learner to recall the concept of activities that are applicable to all software projects	CO 1
4	Discuss about software process?	Remember	—	CO 1
5	List out the different layers of software engineering.	Understand	The learner to recall the concept of different layers of software engineering	CO 1
6	Define the process pattern?	Remember	—	CO 1
7	List out the types of software process models.	Understand	The learner to know about the types of software process models.	CO 1
8	Define various steps involved in identifying a Task Set	Remember	—	CO 1
9	Give the importance of process patterns in designing a module.	Remember	—	CO 1
10	Discuss the template for process patterns.	Remember	—	CO 1
11	Explain waterfall model and who invented waterfall model	Understand	The learner first to know the concept of waterfall model.	CO 1
12	List the three types of process patterns, with suitable examples.	Remember	—	CO 1
13	List different advantages of waterfall model.	Understand	The learner first to know the advantages of waterfall model.	CO 1
14	Discuss different disadvantages of waterfall model.	Remember	—	CO 1
15	Illustrate various problems of prototyping.	Remember	—	CO 1
16	Define the use of incremental process model	Understand	The learner first to know the use of incremental process model.	CO 1

17	List out the disadvantages of spiral model	Understand	The learner first to know the disadvantages of spiral model.	CO 1
18	Discuss about component based development.	Remember	---	CO 1
19	Define how software cost is estimated.	Understand	The learner to know the concept software cost is estimated .	CO 1
20	What is the use of COCOMO model.	Understand	The learner to know the use of COCOMO model .	CO 1
21	Discuss about project scheduling?	Understand	---	CO 1
MODULE II				
REQUIREMENTS ANALYSIS AND SPECIFICATION				
PART A-PROBLEM SOLVING AND CRITICAL THINKING				
1	How feasibility studies are important in requirement engineering process.	Understand	---	CO 2
2	Why requirements validation is done in software development.	Understand	The learner to recall the requirements validation is done in software development.	CO 2
3	How does the analysis modeling help to capture unambiguous consistent requirements? Discuss several methods for requirements validation?	Understand	---	CO 2
4	The need of petri nets with an example.Explain in detail.	Understand	---	CO 2
5	Define Data dictionary. Give the importance of data dictionary with a suitable example.	Understand	The learner to explain the concept of Data dictionary	CO 2
6	Assume that you are developing a online railway reservation system. Prepare the Software Requirement Specification (SRS) document for the system	Analyze	---	CO 2

7	Difficulty in eliciting requirements from system stakeholders. Describe it?	Understand	The learner to recall the requirements validation is done in software development.	CO 2
8	Design class hierarchy for library by using inheritance model?	Understand	—	CO 2
9	Assess on software requirement specification for banking system.	Evaluate	—	CO 2
10	Structured analysis is a development method that allows the analyst to understand the system and its activities in a logical way”. Justify.	Evaluate	The learner to explain the Structured system analysis.	CO 2

PART B-LONG ANSWER QUESTIONS				
1	What is requirement? Explain about user requirements with an example.	Understand	The learner to know the concepts of functional requirements.	CO 2
2	Explain briefly about functional requirements with an example.	Understand	—	CO 2
3	Discuss in detail about non-functional requirements with an example.	Understand	—	CO 2
4	What are system requirements? Explain in a detail.	Understand	The learner to recall the system requirements.	CO 2
5	Explain briefly about The software requirements document.	Understand	—	CO 2
6	Discuss about requirement engineering process.	Understand	The learner to recall the requirement engineering process.	CO 2
7	Discuss briefly how requirement validation is done?	Understand	The learner to define the requirement validation.	CO 2
8	Explain how requirements are managed in software project management.	Understand	The learner to recall the managed in software project management.	CO 2
9	Discuss in detail about requirement discovery with an example.	Understand	The learner to recall the requirement discovery.	CO 2
10	What is interviewing? Explain different types of interviews.	Understand	—	CO 2
11	Write short notes on requirement specification with an example.	Remember	—	CO 2
12	Define the importance of natural language specification with an example.	Understand	The learner to define natural language specification.	CO 2
13	Discuss how requirements are elucidated and validated in software project.	Understand	The learner to know about the elucidated and validated requirements in software project.	CO 2

14	Demonstrate the uses of Use cases in requirements elicitation and analysis with an example.	Remember	—	CO 2
15	What is Ethnography? Explain its importance.	Understand	The learner to define Ethnography.	CO 2
16	What do you understand about AGILE develeopment? Why it is crutial?	Understand	—	CO 2
17	Explain Software programming in terms of non techincal stakeholders.	Understand	—	CO 2
18	Can you outline what is meant by non-functional requirements?	Understand	—	CO 2
19	Can you outline what is meant by functional requirements?	Understand	—	CO 2
20	List kinds of behavioral and object models?	Understand	The learner to explain the Capability Maturity Model.	CO 2
PART C-SHORT ANSWER QUESTIONS				
1	Discuss different types of system requirements?	Understand	The learner to know the concepts of functional requirements and nonfunctional requirements	CO 2
2	What are functional requirements?	Understand	The learner to know the concept of functional requirements.	CO 2
3	Explain nonfunctional requirements?	Understand	The learner to know the concept of nonfunctional requirements.	CO 2
4	Discuss domain requirements?	Remember	—	CO 2
5	List different kinds of nonfunctional requirements?	Remember	—	CO 2
6	Define functional requirement with an example?	Understand	The learner to find functional requirement and then discusses with an example.	CO 2
7	Discuss user requirements in detail?	Understand	The learner to explain the user requirements.	CO 2

8	Explain the need for system requirement.	Remember	—	CO 2
9	Discuss about requirement validation?	Understand	The learner to know the concept of Ethnography technique.	CO 2
10	Explain about requirement engineering process	Remember	—	CO 2
11	Discuss about requirement discovery	Remember	—	CO 2
12	Discuss about Requirements classification and organization.	Understand	The learner to show the Requirements classification and organization.	CO 2
13	Compare functional and non functional requirements.	Remember	—	CO 2
14	Demonstrate the use of Ethnography technique.	Understand	The learner to explain about requirement validation.	CO 2
15	What is Scenarios? Explain.	Remember	The learner to recall the characteristics of Effective interviewers.	CO 2
16	Define the characteristics of Effective interviewers.	Understand	—	CO 2
17	Give an example for requirement validation.	Remember	The learner to define various types of validation techniques.	CO 2
18	Discuss various types of validation techniques that can be used individually or in conjunction with one another:	Understand	—	CO 2
19	Explain about Requirements management planning	Remember	—	CO 2
20	Discuss about Requirements change management.	Remember	The learner to define the requirement review.	CO 2
21	Explain requirement review?	Understand	The learner to define the systems developed as prototypes should not normally be used as production systems.	CO 2
22	Define data dictionary?	Understand	The learner to define the develop a set of actions for the communication activity.	CO 2
23	Discuss data flow model?	Understand	The learner to know the concepts of software process	CO 2

24	Explain Data dictionary in the process of requirements analysis, state machine model of a microwave oven?	Remember	The learner to know the concepts of umbrella activities in process framework.	CO 2
MODULE III				
SOFTWARE DESIGN				
PART A- PROBLEM SOLVING AND CRITICAL THINKING				
1	How do we assess the quality of a software design? Outline about software quality guidelines.	Understan	The learner to know the concepts of taxonomy of software design in the design of user interface.	CO 3
2	Design pattern that you encounter in a category of everyday things.Describe it.	Understand	The learner to know about design pattern.	CO 3
3	Data abstractions and the procedural abstractions that can be used to manipulate them explain with examples.	Understand	The learner to know about variations and effectiveness of data abstractions and the procedural abstractions	CO 3
4	Demonstrate the architecture of a house or building as a metaphor, Draw comparison with software architecture. How are the disciplines of classical architecture and software architecture similar? How do they differ?	Understand	The learner to justify that How are the disciplines of classical architecture and software architecture similar.	CO 3
5	In your own words, define coupling. Explain different categories of coupling	Understand	—	CO 3
6	List out the steps for conducting component level design	Understand	—	CO 3
7	Elucidate the functionalities of architectural design for software development. What should we consider when we name components?	Understand	The learner to know the functionalities of architectural design for software development	CO 3
8	The various steps to develop a decision table.Explain.	Understand	The learner to explain the various steps to develop a decision table.	CO 3

9	Classify elements of interface analysis and highlight the principles of task analysis and modeling	Understand	Understand	CO 3
10	Why are control components necessary in traditional software and generally not required in object-oriented software?	Understand	The learner to define the analysis and design model.	CO 3
PART B-LONG ANSWER QUESTIONS				
1	Explain briefly about the design process and also explain its characteristics.	Understand	The learner to recall the characteristics of good interface design.	CO 3
2	Discuss briefly the following fundamental concepts of software design: i) Abstraction ii) Modularity iii) Information hiding.	Understand	The learner to recall the various design concepts.	CO 3
3	Illustrate the importance of design classes. Explain different types of design classes.	Understand	—	CO 3
4	Discuss in detail about architectural design elements and interface design elements.	Understand	—	CO 3
5	Explain the importance of component level design and deployment level design elements.	Understand	The learner to know the architectural design elements and interface design elements	CO 3
6	What is software architecture? Why it is important explain with an example.	Understand	The learner to recall the component level design and deployment level design elements	CO 3
7	Explain briefly about transform mapping with an example	Understand	The learner to define the software architecture.	CO 3
8	Demonstrate the importance of Archetypes with an example.	Understand	The learner to Know the concept of transform mapping.	CO 3
9	Discuss in detail about different architectural styles.	Understand	—	CO 3

10	Demonstrate how a system represent in architectural context with an example.	Understand	—	CO 3
11	Discuss briefly about the golden rules for the user interface design.	Understand	The learner to know the concepts of user interface design.	CO 3
12	Explain briefly about the importance of task analysis and modeling.	Understand	—	CO 3
13	Discuss in detail about user interface design patterns with an example.	Understand	—	CO 3
14	Explain briefly about different common design issues in user interface design.	Understand	The learner to know about details of common design issues in user interface design.	CO 3
15	List the basic design principles for designing class based components.	Understand	—	CO 3
16	What is Cohesion? Explain its importance in designing class based components.	Understand	The learner to know the importance in designing class based components.	CO 3
17	Discuss in detail about Coupling and also explain different categories of Coupling.	Understand	—	CO 3
18	Compare and contrast Coupling and Cohesion in designing class based components.	Understand	The learner to compare Coupling and Cohesion in designing class based components.	CO 3
19	Describe about graphical design notation in designing traditional components.	Understand	—	CO 3
20	Explain the necessary steps to build decision table in designing traditional components.	Understand	The learner to define the cohesion.	CO 3
PART C - SHORT ANSWER QUESTIONS				
1.	Explain why design is important in software engineering	Remember	—	CO 1
2.	Discuss analysis and design model	Understand	The learner to know about concept of software design.	CO 1

3.	Describe about software quality guidelines	Understand	The learner to know about the concept software quality.	CO 1
4.	How do we assess the quality of a software design	Remember	—	CO 1
5.	What characteristics are common to all design methods?	Understand	The learner to recall the concept of all design methods.	CO 1
6.	Write a short note on Abstraction.	Understand	The learner to define the abstraction.	CO 1
7.	Discuss the importance of design pattern	Remember	—	CO 1
8.	Explain about the modularity in design concepts.	Understand	The learner to identify the modularity in design concepts.	CO 1
9.	Discuss the concept of Information Hiding and Functional Independence.	Remember	—	CO 1
10.	Discuss the importance of refactoring in software design.	Understand	The learner to recall the importance of refactoring in software design	CO 1
11.	Write a short note on design classes in software design.	Understand	The learner to identify the design classes in software design.	CO 1
12.	Discuss about interface design elements in the design model	Remember	—	CO 1
13.	Write short notes on component level and deployment level design elements	Remember	—	CO 1
14.	Define software architecture with its importance	Understand	—	CO 1
15.	Write short notes on architectural descriptions	Remember	The learner to know the sources of representing the system in context	CO 1
16.	Explain taxonomy of architectural styles	Understand	The learner to correlate with specified domains and implemented domains.	CO 1
17.	Write a short notes on architecture patterns	Remember	—	CO 1
18.	Demonstrate about representing the system in context	Understand	The learner to define the analysis and design model.	CO 1

19.	Define the roles of archetypes in architectural design	Remember	The learner to know the concepts of activities in design process.	CO 1
20.	Write short notes on architectural mapping using data flow	Understand	The learner to recall the characteristics of good interface design.	CO 1
21.	What is user interface design	Understand	—	CO 1
22.	Demonstrate the importance of user analysis	Remember	The learner to know about concept of software design.	CO 1
23.	Discuss about Use cases and Task elaboration	Remember	The learner to know about the concept software quality.	CO 1
24.	Write short notes on Workflow analysis	Understand	—	CO 1
25.	Define various User interface design steps	Understand	The learner to recall the concept of all design methods.	CO 1
26.	Write any three user interface design issues	Remember	The learner to define the abstraction.	CO 1
27.	List out the golden rules for interface design	Understand	—	CO 1
28.	What is a component?	Understand	The learner to identify the modularity in design concepts.	CO 1
29.	What belongs to a component according to object oriented view	Remember	—	CO 1
30.	List any two basic design principles that are applicable to component-level design	Understand	The learner to recall the importance of refactoring in software design	CO 1
31.	What should we consider when we name components?	Remember	The learner to identify the design classes in software design.	CO 1
32.	Write a short notes on cohesion	Remember	—	CO 1

MODULE IV

TESTING AND IMPLEMENTATION

PART A-PROBLEM SOLVING AND CRITICAL THINKING

Q.No	QUESTION	Taxonomy	How does this subsume the level	CO's
------	----------	----------	---------------------------------	------

1.	Who should perform the validation test— the software developer or the software user? Justify your answer.	Understand	Learner to recall the software failures and faults.	CO 4
2.	Develop a complete test strategy for the SafeHome system.	Apply	Learner to know the concepts of complete test strategy.	CO 4
3.	Using your own words, describe the difference between verification and validation.	Understand	—.	CO 4
4.	Is unit testing possible or even desirable in all circumstances? Provide examples to justify your answer.	Analyze	—	CO 4
5	In your own words define system testing? Explain briefly about system testing.	Understand	The learner to define the system testing.	CO 4
6.	Can you talk about a time when you didn't see eye-to-eye with a team member? How did you resolve the problem?	Understand	The learner to define the system testing.	CO 4
7.	what are the critical principles that you follow as a software developer	Understand	—	CO 4
8.	Have you ever had to adapt to a new situation as a software engineer?	Understand	The learner to recall the importance of validation testing.	CO 4
9.	Has there been a circumstance in which you missed a deadline? What strategy would you use to complete the task now?	Understand	—	CO 4
10.	What would be your definition of debugging in software development?	Understand	The learner to know the concepts of software implementation techniques.	CO 4
PART B- LONG ANSWER QUESTIONS				
1.	Explain about the importance of test strategies for conventional software.	Understand	Learner to know the concepts of integration testing process and system testing process.	CO 4

2.	What is testing? Discuss in detail about Black-Box testing.	Understand	The learner to identify the importance of test strategies for conventional software.	CO 4
3.	Compare and contrast black box testing and White Box testing.	Understand	The learner to identify the black box testing and White Box testing.	CO 4
4.	Discuss in detail about basis path testing and explain its importance.	Understand	—	CO 4
5.	Explain briefly about control structure testing.	Understand	The learner to know the concepts of control structure testing.	CO 4
6.	What is system testing? Explain briefly about system testing.	Understand	The learner to define the system testing.	CO 4
7.	Discuss in detail about the art of debugging.	Understand	—	CO 4
8.	Demonstrate the importance of validation testing?	Understand	The learner to recall the importance of validation testing.	CO 4
9.	Write short notes on regression and white box testing.	Understand	—	CO 4
10.	Explain about various software implementation techniques.	Understand	The learner to know the concepts of software implementation techniques.	CO 4
11.	How is Pseudo Code different from structured English?	Understand	—	CO 4
12.	What is regression testing?	Understand	The Learner to know about the regression testing.	CO 4
13.	What is the difference between STLC and SDLC	Understand	The Learner to know about the unit testing.	CO 4
14.	Describe Equivalence partitioning testing?	Understand	—	CO 4
15.	What is the difference between static and dynamic testing?	Understand	The learner to know about the alpha and beta testing.	CO 4
16.	Explain the concepts of Datadriven and retsting.	Understand	The learner to know about the performance testing	CO 4
17.	Compare test scenarios, test cases and test script.	Understand	—	CO 4

18.	list the parameters which can be useful to know how the quality of test execution.	Understand	The learner to identify coding practices.	CO 4
19.	Explain the concept of test deliverables.	Understand	The learner to recall the characteristics of testability.	CO 4
20.	What is the procedure of conducting risk analysis. Explain in detail about it.	Understand	The learner to know the testing in software development.	CO 4
PART C- SHORT ANSWER QUESTIONS				
1.	What are the characteristics of testability?	Understand	The Learner to know about the integration testing.	CO 4
2.	Define various test characteristics.	Remember	—	CO 4
3.	Write short notes on internal and external views of testing.	Understand	The Learner to know about the internal and external views of testing.	CO 4
4.	What is white box testing?	Understand	The Learner to know about the white box testing.	CO 4
5.	Discuss the importance of graph matrices in basis path testing.	Remember	—	CO 4
6.	Explain different steps that can be applied to derive the test cases.	Understand	The learner to recall the different steps that can be applied to derive the test cases.	CO 4
7.	What is loop testing? Write a short notes on loop testing.	Understand	The Learner to know about the loop testing.	CO 4
8.	Define condition testing and data flow testing.	Remember	—	CO 4
9.	What are the differences between verification and validation?	Understand	The learner to know the concept of verification and validation.	CO 4
10.	Demonstrate about boundary value analysis in black box testing.	Remember	—	CO 4
11.	Discuss in detail about graph-based testing methods.	Remember	—	CO 4
12.	What is regression testing?	Understand	The Learner to know about the regression testing.	CO 4

13.	Write short notes on unit testing and explain its environment.	Understand	The Learner to know about the unit testing.	CO 4
14.	What is the use of integration testing? Explain its types.	Remember	---	CO 4
15.	Discuss about alpha and beta testing.	Understand	The learner to know about the alpha and beta testing.	CO 4
16.	Define stress and performance testing.	Understand	The learner to know about the performance testing	CO 4
17.	Explain about debugging process.	Remember	---	CO 4
18.	Define various debugging strategies.	Remember	The learner to identify coding practices.	CO 4
19.	Define the importance of refactoring technique in software implementation.	Understand	The learner to recall the characteristics of testability.	CO 4
20.	Write short notes on coding practices.	Understand	The learner to know the testing in software development.	CO 4

MODULE V

PROJECT MANAGEMENT

PART A-PROBLEM SOLVING AND CRITICAL THINKING

Q.No	QUESTION	Taxonomy	How does this subsume the level	CO's
1.	Describe five software application areas in which software safety and hazard analysis would be a major concern.	Understand	The learner to know the concepts of software application.	CO 5
2.	Can you outline some useful software metrics?	Understand	---	CO 5
3.	Are you aware of any software engineering project management tools?	Understand	The learner to know the importance of project scheduling.	CO 5
4.	Can you describe a time when you had to provide an estimate for a client.	Understand	---	CO 5
5.	Can you talk about a time when you had to plan for a software engineering project?	Understand	The learner to Estimate the Software Development Time for a Client Project	CO 5

6.	What do you understand about “Agile” development? Why is it crucial?	Understand	The learner to explain the Agile” development	CO 5
7.	Suppose you have a budgeted cost of a project as Rs. 9,00,000. The project is to be completed in 9 months. After a month you have completed 10 percent of project at a total expense of Rs. 1,00,000. The planned completion should have been 15 percent. You need to determine whether the project is on-time and on budget? Use Earned value analysis approach and interpret.	Evaluate	---	CO 5
8.	How is Pseudo Code different from structured English?	Understand	---	CO 5
9.	Can you describe functional programming?	Understand	---	CO 5
10.	Describe in detail COCOMO model for software cost estimation. Use it to estimate the effort required to build software for a simple ATM that produce 12 screens, 10 reports and has 80 software components. Assume average complexity and average developer maturity. Use application composition model with object points.	Evaluate	---	CO 5
PART B- LONG ANSWER QUESTIONS				
1.	Explain in detail about Reactive versus Proactive Risk Strategies	Understand	The learner to know about the concepts of Reactive versus Proactive Risk Strategies.	CO 5
2.	Write briefly about Risk mitigation, monitoring, and management.	Understand	---	CO 5

3.	What is scheduling? Explain the importance of scheduling in the project management.	Understand	—	CO 5
4.	Discuss the importance of project scheduling.	Understand	The learner to know the importance of project scheduling.	CO 5
5.	Write about how to define a task set for the software project.	Understand	—	CO 5
6.	How the risk identification in done in risk management.	Understand	—	CO 5
7.	Compare the pros and cons of COCOMO and COCOMO II Models?	Understand	The learner to know the concepts of COCOMO and COCOMO II Models.	CO 5
8.	Explain in detail about software measurement and discuss various metrics.	Understand	—	CO 5
9.	Explain the process of integrating metrics within the software process.	Understand	—	CO 5
10.	Discuss in detail about various metrics for small organizations.	Understand	The learner to recall the various metrics for small organizations.	CO 5
11.	List the steps involved in activity planing and explain them.	Understand	—	CO 5
12.	Explain the process of Delphi method and write its advantages and disadvantages.	Understand	—	CO 5
13.	Explain the process of functional point analysis.	Understand	The learner to know about the software project scheduling.	CO 5
14.	Explain functional point analysis with sample cases for component for different complexity.	Understand	—	CO 5
15.	Give detail explanation about Scheduling and Tracking	Understand	—	CO 5
16.	Outline the key differences between COCOMO and COCOMO II Models.	Understand	The learner to recall the various metrics for small organizations.	CO 5

	Explain in detail about Reactive versus Proactive Risk Strategies	Understand	The learner to know about the concepts of Reactive versus Proactive Risk Strategies.	CO 5
17.	Describe the differences between —known risks‖ and —predictable risks.‖	Understand	The learner to identify the known risks and predictable risks.	CO 5
18.	Describe steps involved in project scheduling process, project timeline chart and task network.	Understand	The learner to recall the earned value analysis	CO 5
19.	Write about software maintenance, PERT - CPM for scheduling , RMMP	Understand	—	CO 5
20.	Brief about calculating Earned value measures.	Understand	The learner to know the concepts of metrics for software quality.	CO 5
PART C- SHORT ANSWER QUESTIONS				
1.	Write short notes on estimation.	Understand	The learner to recall the importance of cost estimation in software development.	CO 5
2.	Give an example of LOC-based cost estimation	Remember	—	CO 5
3.	Write a short notes on FP-based cost estimation	Remember	—	CO 5
4.	Discuss about outsourcing.	Remember	—	CO 5
5.	What is planning? Write about project planning process.	Understand	The learner to define planning and know about the project planning process.	CO 5
6.	Write a short note on COCOMO II model.	Remember	—	CO 5
7.	Discuss about proactive risk strategies.	Understand	The learner to know the concepts of proactive risk strategies.	CO 5
8.	Demonstrate various types of software risks.	Understand	The learner to define various types of software risks.	CO 5
9.	Write a short note on risk identification.	Remember	—	CO 5
10.	Explain how overall project risk will be assessed.	Understand	The learner to know the project risk will be assessed.	CO 5

11.	What is risk projection (or) risk estimation?	Remember	—	CO 5
12.	Elaborate the RMMM.	Remember	—	CO 5
13.	Discuss about software project scheduling.	Understand	The learner to know about the software project scheduling.	CO 5
14.	What are the basic principles of software project scheduling?	Remember	—	CO 5
15.	Define the relationship between people and effort.	Remember	—	CO 5
16.	How should effort be distributed across the software process workflow?	Understand	The learner to identify the distributed across the software process workflow	CO 5
17.	Discuss about time-line charts with an examples.	Understand	The learner to know the concepts of time-line charts.	CO 5
18.	Write a short note on earned value analysis.	Understand	The learner to recall the earned value analysis	CO 5
19.	Determine process metrics and software process improvement	Remember	—	CO 5
20.	Illustrate the various metrics for software quality.	Understand	The learner to know the concepts of metrics for software quality.	CO 5

Course Coordinator:
Ms.Y Sujana, Assistant professor

HOD CSE(DS)