

Class :II MCA

Semester : 3rd

MC1703 Software Engineering

Number of Questions : Short Answers :150

Essay Type : 25

UNIT 1

1) Define Software Engineering.

Software Engineering :

- The Application of systematic, disciplined, quantifier approach
- To the development, operations, and maintenance of software

2) What is a Process Framework?

Process Framework :

- Establishes foundation for a complete software process
- By identifying a small number of framework activities that are applicable for all software projects regardless of their size and complexity

3) What are the Generic Framework Activities?

Generic Framework Activities :

- Communication
- Planning
- Modeling
- Construction
- Deployment

4) Define Stakeholder.

Stakeholder :

- Anyone who has stake in successful outcome of Project
- Business Managers, end-users, software engineer, support people

5)How the Process Model differ from one another?

- Based on flow of activities
- Interdependencies between activities
- Manner of Quality Assurance
- Manner of Project Tracking

- Team Organization and Roles
- Work Products identify an requirement identifier

6) Write out the reasons for the Failure of Water Fall Model?

Reasons For The Failure Of Water Fall Model :

- Real Project rarely follow Sequential Flow. Iterations are made in indirect manner
- Difficult for customer to state all requirements explicitly
- Customer needs more patients as working product reach only at Deployment phase

7) What are the Drawbacks of RAD Model?

Drawbacks of RAD Model :

- Require sufficient number of Human Resources to create enough number of teams
- Developers and Customers are not committed, system result in failure
- Not Properly Modularized building component may Problematic
- Not applicable when there is more possibility for Technical Risk

8) Why Formal Methods are not widely used?

- Quite Time Consuming and Expensive
- Extensive expertise is needed for developers to apply formal methods
- Difficult to use as they are technically sophisticated maintenance may become risk

9) What is Cross Cutting Concerns?

Cross Cutting Concerns :

- When concerns cut across multiple functions, features and information

10) What are the different Phases of Unified Process?

Different Phases of Unified Process :

- Inception Phase
- Elaboration Phase
- Construction Phase
- Transition Phase
- Production Phase

11) Define the terms :

- a) Agility
- b) Agile Team

a) Agility :-

- Dynamic, Content Specific, Aggressively Change Embracing and Growth Oriented

b) Agile Team :-

- Fast Team
- Able to Respond to Changes

12) Define the terms:

a) Agile Methods

b) Agile Process

a) Agile Methods :-

- Methods to overcome perceive and actual weakness in conventional software engineering
- To accommodate changes in environment, requirements and use cases

b) Agile Process :-

- Focus on Team Structures, Team Communications, Rapid Delivery of software and it de-emphasis importance of intermediate product

13) What is the Use of Process Technology Tools?

Use of Process Technology Tools :

- Help Software Organizations
 1. Analyze their current process
 2. Organize work task
 3. Control And Monitor Progress
 4. Manage Technical Quality
 - 5.

14) Define the term Scripts.

Scripts :

- Specific Process Activities and other detailed work functions that are part of team process

15) What is the Objective of the Project Planning Process?

Objective of the Project Planning Process :

- To provide framework that enables manager to make reasonable estimates of resources, cost and schedule

16) What are the Decomposition Techniques?

Decomposition Techniques :

- Software Sizing
- Problem - Based Estimation
- Process - Based Estimation
- Estimation With Use - Cases
- Reconciling Estimates

17) How do we compute the "Expected Value" for Software Size?

- Expected value for estimation variable(size), S, can be compute as Weighted Average of Optimistic(Sopt),most likely(Sm),and Pessimistic(Spess) estimates
- $S = (Sopt+4Sm+Spess)/6$

18) What is an Object Point?

Object Point :

- Count is determined by multiplying original number of object instances by weighting factor and summing to obtain total object point count

19) What is the difference between the "Known Risks" and Predictable Risks" ?

Known Risks :-

- That can be uncovered after careful evaluation of the project plan, the business, and technical environment in which the product is being developed
- Example : Unrealistic delivery rate

Predictable Risks :-

- Extrapolated from past project experience
- Example : Staff turnover

20) List out the basic principles of software project scheduling ?

Basic Principles Of Software Project Scheduling :-

- Compartmentalization
- Interdependency
- Time Allocation
- Effort Validation
- Defined Responsibilities
- Defined Outcomes
- Defined Milestones

UNIT 2

21) What are the Classifications of System Engineering?

Classifications of System Engineering :

- Business Process Engineering[BPE]
- Product Engineering

22) List out the Elements in Computer-Based System?

Elements in Computer-Based System :

- Software

- Hardware
- People
- Database
- Documentation
- Procedures

23) What are the Factors to be considered in the System Model Construction?

- Assumption
- Simplification
- Limitation
- Constraints
- Preferences

24) What does a System Engineering Model accomplish?

- Define Processes that serve needs of view
- Represent behavior of process and assumption
- Explicitly define Exogenous and Endogenous Input
- Represent all Linkages that enable engineer to better understand view

25) What Architectures are defined and developed as part of BPE?

- Data Architecture
- Applications Architecture
- Technology Architecture

26) What is meant by Cardinality and Modality ?

Cardinality :-

- The number of occurrence of one object related to the number of occurrence of another object
- One to One [1 :1]
- One to Many [1 : N]
- Many to Many [M : N]

Modality :-

- Whether or not a particular Data Object must participate in the relationship

27) What are the Objectives of Requirement Analysis ?

Objectives of Requirement Analysis :

- Describe what customer requires
- Establish a basis for creation of software design
- Define a set of requirements that can be validated once the software design is built

28) What are the two additional feature of Hayley Pirbhai Model?

- User Interface Processing
- Maintenance and Self test Processing

29) Define System Context Diagram[SCD]?

System Context Diagram[SCD] :

- Establish information boundary between System being implemented and Environment which system operate
- Defines all external producers, external consumers and entities that communicate through User Interface

30) Define System Flow Diagram[SFD]?

System Flow Diagram[SFD] :

- Indicates Information flow across SCD region
- Used to guide system engineer in developing system

31) What are the Requirements Engineering Process Functions?

- Inception
- Elicitation
- Elaboration
- Negotiation
- Specification
- Validation
- Management

32) What are the Difficulties in Elicitation?

Difficulties in Elicitation :

- Problem Of Scope
- Problem Of Understanding
- Problem Of Volatility

33) List out the Types of Traceability Table?

Types of Traceability Table :

- Features Traceability Table
- Source Traceability Table
- Dependency Traceability Table
- Subsystem Traceability Table
- Interface Traceability Table

34) Define Quality Function Deployment[QFD]?

Quality Function Deployment[QFD] :

- Technique translates needs of customer into technical requirements

- “Concentrates on maximizing customer satisfaction from the software engineering process”

35) What are the Benefits of Analysis Pattern?

Benefits of Analysis Pattern :

- Seedup development of Analysis model
- Transformation of Analysis into Design model

36) What is System Modeling?

System Modeling :-

- Important Element in System Engineering Process
- Define Process in each view to be constructed
- Represent Behavior of the Process
- Explicitly define exogenous and endogenous inputs

37) Define CRC Modeling ?

CRC Modeling :-

- Class Responsibility Collaborator Modeling
- Collection of Standard Index Card .Divided into 3 sections
 1. Name of class at Top
 2. List of class Responsibilities at Left
 3. Collaborators at Right
- Classes that Cover the Information to complete its responsibilities

38) List out the Factors of Data Modeling?

Factors of Data Modeling :

- Data Objects
- Data Attributes
- Relationship
- Cardinality and Modality

39) Define Swim Lane Diagram?

Swim Lane Diagram :

- Variation of activity diagram
- Allows Modular to represent floe of activities
- Actor responsible for activity

40) What are the Selection Characteristic for Classes?

Selection Characteristic for Classes :

- Retained Information
- Needed Services
- Multiple Attribute
- Common Attribute
- Common operations
- Essential Requirements

41) Define Steps in Behavioral Model.

Steps in Behavioral Model :

- Evaluate all Use Cases
- Identify Events
- Create Sequence for each use Cases
- Build a State Diagram
- Review Model for Accuracy and Consistency

UNIT 3

41) Define the terms in Software Designing :

(a) Abstraction

(b) Modularity

(a) Abstraction :

1. Highest Level : Solution is stated in broad term using language of problem environment
2. Lower Level : More detailed description of solution is provided

(b) Modularity :

- Software is divided into separately named and addressable components, called Modules that are integrated to satisfy problem requirements

42) How the Architecture Design can be represented?

- Architectural Design can be represented by one or more different models. They are,
 1. Structural Models
 2. Framework Models
 3. Dynamic Models
 4. Process Models

43) What is the Advantage of Information Hiding?

Advantage of Information Hiding :

- During testing and maintenance phase if changes require that is done in particular module without affecting other module

44) What types of Classes does the designer create?

- User interface Classes
- Business Domain Classes
- Process Classes
- Persistent Classes
- System Classes

45) What is Coupling?

Coupling :-

- Quantitative measure of degree to which classes are connected to one another
- Keep coupling as low as possible

46) What is Cohesion?

Cohesion :

- Indication of relative functional strength of a module
- Natural extension of Information Hiding
- Performs a single task, requiring little integration with other components

47) Define Refactoring.

Refactoring :

- Changing software system in the way that does not alter external behavior of code

48) What are the Five Types of Design classes?

Five Types of Design classes :

- User Interface Classes
- Business domain Classes
- Process Classes
- Persistent Classes
- System Classes

49) What are the Different types of Design Model? Explain.

Different types of Design Model :

- Process Dimension :
 - Indicate evolution of Design model as design tasks executed as part of software process
- Abstraction Dimension :
 - Represent level of detail as each element of analysis model is transformed into design equivalent

50) List out the Different elements of Design Model?

Different Elements of Design Model :

- Data Design Elements
- Architectural Design Elements
- Interface Design Elements
- Component Level Design Elements
- Deployment Level Design Elements

51) What are the Types of Interface Design Elements?

Types of Interface Design Elements :

- User Interfaces
- External Interfaces
- Internal Interfaces

52) What Types of Design Patterns are available for the software Engineer?

Types of Design Patterns :

- Architectural patterns
- Design Patterns
- Idioms

53) Define Framework.

Framework :

- Code Skeleton that can be fleshed out with specific classes or functionality
- Designed to address a specific problem at hand

54) What is the Objective of Architectural Design?

Objective of Architectural Design :

- Model overall software structure by representing component interfaces, dependencies and relationships and interactions

55) What are the important roles of Conventional component within the Software Architecture?

- Control Component : that coordinates invocation of all other problem domain
- Problem Domain Component : that implement Complete or Partial function required by customer
- Infrastructure Component : that responsible for functions that support processing required in problem domain

56) What are the Basic Design principles of Class-Based Components?

Basic Design principles of Class-Based Components :

- Open-Closed Principle[OCP]
- Liskov Substitution Principle[LSP]
- Dependency Inversion Principle[DIP]
- Interface Segregation Principle[ISP]
- Release Reuse Equivalency Principle[REP]
- Common Closure Principle[CCP]
- Common Reuse Principle[CRP]

57) What should we consider when we name components?

- Components
- Interface
- Dependencies and Inheritance

58) What are the Different Types of Cohesion?

Different Types of Cohesion :

- Functional
- Layer
- Communicational
- Sequential
- Procedural
- Temporal
- Utility

59) What are the Different Types of Coupling?

Different Types of Coupling :

- Content Coupling
- Common Coupling
- Control Coupling
- Stamp Coupling
- Data Coupling
- Routine Call Coupling
- Type Use Coupling
- Inclusion or Import Coupling
- External Coupling

60) What is Program Design Language [PDL]?

Program Design Language [PDL] :

- Also called Structured English or Pseudocode
- Pidgin Language in that it uses the vocabulary of one language and overall syntax of another

UNIT 4

61) What are the Basic Principles of Software Testing?

Basic Principles of Software Testing :

- Traceable to Customer Requirements
- Planned long before Testing begins
- Pareto Principles applied to Software testing
- Begin small and progress towards testing
- Exhaustive testing is not possible
- Conducted by independent third party

62) List out the Characteristics of Testability of Software?

Characteristics of Testability of Software :

- Operability
- Observability
- Controllability
- Decomposability
- Simplicity
- Stability

- Understandability

63) List out various Methods for finding Cyclomatic Complexity ?

- Number of Regions
- Cyclomatic Complexity $V(G)$, for Flow Graph

$$V(G) = E - N + 2$$
- Cyclomatic Complexity $V(G)$

$$V(G) = P + 1$$

64) Define Smoke Testing ?

Smoke Testing :

- Integration testing
- Commonly used when software products are being developed

65) What are the Attributes of Good Test?

Attributes of Good Test :

- High probability of finding errors
- Not Redundant
- "Best of Breed"
- Neither too Simple nor too complex

65) Define White Box Testing.

White Box Testing :

- Also called Glass Box Testing
- Test case design uses Control Structure of Procedural Design to derive test cases

66) Define Basic Path Testing.

Basic Path Testing :

- White Box Testing
- Enable test case designer to derive a logical complexity measure of a procedural design
- Use this measure as a Guide for defining a basis set of execution paths

67) Define the terms :

a) Graph Matrices

b) Connection Matrices

Graph Matrices :-

- To develop software tool the data structure used is Graph Matrix
- Square Matrix
- Size equals number of nodes on the Flow graph

Connection Matrices :-

- If Link Weight =1 => Connection Exists
- If Link Weight =0 => Connection Does not Exist

68) What is Behavioral Testing?

Behavioral Testing :

- Also Known as Black Box Testing
- Focuses on Functional Requirement of software
- Enables Software engineer to derive set of input condition that fully exercise all functional requirements of a software

69) What are the Benefits of conducting Smoke Testing?

Benefits of conducting Smoke Testing :

- Integration Risk is Minimized
- Quality of end-product is improved
- Error diagnosis and Correction are simplified
- Progress is easy to assess

70) What errors are commonly found during Unit Testing?

- Misunderstood or incorrect arithmetic precedence
- Mixed Mode Operations
- Incorrect Initializations
- Precision Accuracy
- Incorrect Symbolic representation of expression

71) What problems may be encountered when Top-Down Integration is chosen?

- Delay are test until stubs replace with actual modules
- Develop stubs that perform limited functions that simulate the actual module
- Integrate the software from the bottom of the hierarchy upward

72) What are the Steps in Bottom-Up Integration?

Steps in Bottom-Up Integration :

- Low level components are combined into clusters perform specific software sub function
- Driver is written to coordinate test case input and output
- Cluster is tested
- Drivers are removed and clusters are combined moving inward in program structure

73) What is Regression Testing?

Regression Testing :

- Re-execution of some subset of tests that have already been conducted
- To ensure changes have not propagated unintended side effects

74) What are the Characteristics of "Critical Module"?

Characteristics of "Critical Module" :

- Addresses several software requirements
- Has High Level Of Control
- Complex or error prone
- Has Definite Performance Requirements

75) What are the Properties of Connection Matrices?

Properties of Connection Matrices :

- Probability that link will execute
- Processing time expended during traversal of link
- Memory required during traversal of link
- Resource required during traversal of link

76) What is Flow Graph Notation?

Flow Graph Notation :-

- Simple notation for representing Control Flow
- Draw only when Logical Structure of component is complex

77) Define Cyclomatic Complexity?

Cyclomatic Complexity :-

- Software Metric
- Quantitative measure of Logical Complexity
- Number of Independent Paths in the basis set of program

78) What is Equivalence Partition?

Equivalence Partitions :-

- Derives a input domain of a program into classes of data from which test cases are derived
- Set Of Objects have link by relationships as Symmetric, Transitive and Reflexive an equivalence class is present

79) List out the possible errors of Black Box Testing?

Errors of Black Box Testing :

- Incorrect or Missing Functions
- Interface Errors
- Errors in Data Structures or external databases
- Behavioral or Performance errors
- Initialization or Termination errors

80) Define Data Objects.

Data Objects :

- Represent Composite Information
- External entity, thin, occurrence or event, role, organizational unit, place or structure
- Encapsulates Data only

UNIT 5

81) What are the Components of the Cost of Quality?

Components of the Cost of Quality :

- Quality Costs
- Prevention Costs
- Appraisal Costs

82) What is Software Quality Control?

Software Quality Control :

- Involves series of inspections, reviews and tests
- Used throughout software process to ensure each work product meets requirements placed upon it

83) What is Software Quality Assurance?

Software Quality Assurance :

- Set of auditing and reporting functions
- Assess effectiveness and completeness of quality control activities

84) What are the Objective of Formal Technical Reviews?

Objective of Formal Technical Reviews :

- Uncover errors in function, logic and implementation for representation of software
- Software represented according to predefined standard
- Verify software under review meets requirements
- Achieve software developed in Uniform Manner
- Make projects more manageable

85) What Steps are required to perform Statistical SQA?

- Information about software defects is collected and categorized
- Attempt is made trace each defect
- Using Pareto principle, isolate 20%
- Once vital causes are identified, correct problems that cause defects

86) Define SQA Plan.

SQA Plan :

- Provides roadmap for instituting SQA
- Plan serves as template for SQA activities that instituted for each software project

87) What is Baseline criteria in SCM ?

- Help to control Change
- Specification or product that has been formally

- Reviewed and agreed upon serves as basis for future development
- That can be change only through formal change control procedures

88) Define Status Reporting ?

- Also called Configuration Status Reporting
- Is a SCM task that answers
 1. What Happened ?
 2. Who did it ?
 3. When did it happen ?
 4. What else will be affected ?

89) What is the Origin of changes that are requested for software?

Origin Of Change :-

- New Business or Market Condition
- New Customer Needs
- Reorganization or business growth/downsizing
- Budgetary or Scheduling constraints

90) List out the Elements of SCM?

Elements of SCM :-

- Component Elements
- Process Elements
- Construction Elements
- Human Elements

91) What are the Features supported by SCM?

Features supported by SCM :

- Versioning
- Dependency tracking and change Management
- Requirements tracking
- Configuration Management
- Audit trails

92) What are the Objectives of SCM Process?

Objectives of SCM Process :

- Identify all items, collectively define software configuration
- Manage changes to one or more these items
- Facilitate construction of different version of an application
- Ensure that the software quality is maintained

93) What are the issues to be considered for developing tactics for WebApp Configuration Management?

- Context
- People
- Scalability

94) Define CASE Tools.

CASE Tools :

- Computer Aided Software Engineering
- It is a System software
- Provide Automated support for software process activities
- Includes program used to support software process activities
- Such as Requirement Analysis, System Modeling, Debugging and Testing

95) How do we define Software Quality?

Software Quality :

- Conformance to explicitly stated functional and performance requirements, explicitly documented development standards
- Implicit characteristics, expected for professional developed software

96) Define the terms :

- a) Quality of Design
- b) Quality of Conformance

Quality of Design :

- Characteristics, designer specify fro an item

Quality of Conformance :

- Degree to which design specifications are followed during manufacturing

97) What are the Type of CASE Tools?

Types of CASE Tools :-

- Upper CASE Tools
- Lower CASE Tools

98) Define Software Reliability?

Software Reliability :

- Probability of failure-free operation of computer program in a specified environment for a specified time

99) How the Registration process of ISO 9000 certification is done?

- Registration process of ISO 9000 certification has the following stages
 1. application
 2. Pre-assessment

3. Document Review and Adequacy of audit
4. Compliance Audit
5. Registration
6. Continued Surveillance

100) What are the Factors of Software Quality?

Factors of Software Quality :

- Portability
- Usability
- Reusability
- Correctness
- Maintainability

Short Answers

1. Define Software Engineering

The establishment and use of sound engineering principles in order to obtain economically software that is reliable and works efficiently on real machines.

2. Differentiate Software engineering methods, tools and procedures.

Methods: Broad array of tasks like project planning, cost estimation etc..

Tools: Automated or semi automated support for methods.

Procedures : Holds the methods and tools together. It enables the timely development of computer software.

3. Write the disadvantages of classic life cycle model.

Disadvantages of classic life cycle model :

- (i) Real projects rarely follow sequential flow. Iteration always occurs and creates problem.
- (ii) Difficult for the customer to state all requirements
- (iii) Working version of the program is not available. So the customer must have patience.

4. What do you mean by task set in spiral Model?

Each of the regions in the spiral model is populated by a set of work tasks called a task set that are adopted to the characteristics of the project to be undertaken.

5. What is the main objective of Win-Win Spiral Model?

The customer and the developer enter into the process of negotiation where the customer may be asked to balance functionality ,performance and other product against cost and time to market.

6. Which of the software engineering paradigms would be most effective? Why?

Incremental / Spiral model will be most effective.

Reasons:

- (i) It combines linear sequential model with iterative nature of prototyping
- (ii) Focuses on delivery of product at each increment
- (iii) Can be planned to manage technical risks.

7. Who is called as the Stakeholder?

Stakeholder is anyone in the organization who has a direct business interest in the system or product to be built.

8. Write the objective of project planning ?

It is to provide a framework that enables the manager to make reasonable estimates of resources, cost and schedule.

9. What is Boot Strapping?

A sequence of instructions whose execution causes additional instructions to be loaded and executed until the complete program is in storage.

10. Write a short note on 4GT.

Fourth Generation Technique. 4GT encompasses a broad array of software tools. Each tool enables the software developer to specify some characteristics of software at a higher level.

11. What is FP ? How it is used for project estimation ?

Function Point. It is used as the estimation variable to size the each element of the software. It requires considerably less detailed. Estimated indirectly by estimating the number of inputs, outputs, data files, external interfaces.

12. What is LOC ? How it is used for project estimation?

LOC : Lines of Code. It is used as estimation variable to size each element of the software. It requires considerable level of detail..

13. Write the formula to calculate the effort in persons-months used in Dynamic multi variable Model?

Software Equation : $E = [LOC * B^{0.333}/P]^3 * (1/t^4)$ Where E is effort in person-months, t is project duration, B is special skills factor, P is productivity parameter.

14. What is called object points?

It is an indirect software measure that is computed using counts of the number of screens, reports and components.

15. What are the four different Degrees of Rigor ?

Four different degrees of Rigor are

- Casual
- Structured
- Strict
- Quick reaction

16. Write about Democratic Teams in software development. (Egoless Team)

It is egoless team. All team members participate in all decisions. Group leadership rotates from member to member based on tasks to be performed.

17. What are the two project scheduling methods ?

- PERT- Program Evaluation and Review Techniques
- CPM- Critical Path Method

18. What is called support risk?

The degree of uncertainty that the resultant software will be easy to correct , adapt and enhance.

19. What is RMMM?

Risk Mitigation, Monitoring and Management Plan. It is also called Risk Aversion.

20. What are four impacts of the project risk?

Catastrophic, Critical, Marginal, Negligible.

21. List the tools or methods available for rapid prototyping.

Rapid prototyping (Speed)

- (i) 4GT
- (ii) Reusable software components
- (iii) Formal specification and prototyping environments.

22. What is the need for modularity ?

Need for modularity: Easier to solve a complex problem. Can achieve reusability. Best effort and complexity reduces.

23. What are the five criteria that are used in modularity?

- Modular Decomposability
- Modular composability
- Modular understandability
- Modular continuity

Modular protection

24. What is Software Architecture?

The overall structure of the software and the ways in which that software provides conceptual integrity for the system.

25. What are the models are used for Architectural design?

Structural models
Framework models
Dynamic models
Process models
Functional models

26. What is cohesion?

It is a measure of the relative functional strength of a module. (Binding)

27. What is Coupling?

Measure of the relative interdependence among modules.

(Measure of interconnection among modules in a software structure.)

28. List the coupling factors.

Interface complexity between modules
Reference to the module
Data pass across the interface.

29. Define Stamp coupling.

When a portion of the data structure is passed via the module interface , then it called stamp coupling.

30. Define common coupling.

When a number of modules reference a global data area, then the coupling is called common coupling.

31. Define temporal cohesion.

When a module contains tasks that are related by the fact that all must be executed with the same span of time, then it termed as temporal cohesion.

32. Write a short note on structure charts.

These are used in architectural design to document hierarchical structure, parameters and interconnections in a system. No Decision box . The chart can be augmented with module by module specifications of I/P and O/P parameters as well as I/P and O/P attributes.

33. What do you mean by factoring?

It is also called vertical partitioning. It follows Top-Down strategy. We can say that there are some top level modules and low level modules.

Top level modules ---- Control functions ,actual processing work

Low level modules ----Workers. Performing all input computation and output tasks.

34. What is Aesthetics?

Aesthetics : It is a science of art and beauty. These are fundamental to software design, whether in art or technology.

Simplicity, Elegance(refinement), clarity of purpose.

35. What do you mean by common coupling?

Common coupling : When a number of modules reference a global data area , then the coupling is called common coupling.

36. Write about Real Time Systems.

It provides specified amount of computation with in fixed time intervals. RTS sense and control external devices, respond to external events and share processing time between tasks.

37. Define Distributed system .

It consists of a collection of nearly autonomous processors that communicate to achieve a coherent computing system.

38. Compare Data Flow Oriented Design with data structure oriented design

Data flow oriented design : Used to represent a system or software at any level of abstraction.

Data Structure oriented design : It is used for representing information hierarchy using the three constructs for sequence, selection and repetition.

39. Define Architectural Design and Data Design.

Architectural Design : To develop a modular program structure and represent the relationships between modules.

Data Design : To select the logical representations of data objects , data storage and the concepts of information hiding and data abstraction.

40. What are the contents of HIPO diagrams?

Visual table of contents, set of overview diagrams, set of detail diagrams.

41. What are the aspects of software reuse.

Software development with reuse

Software development for reuse

Generator based reuse

Application system reuse

42. Define Configuration Status Reporting .

What happened ? Who did it?

When it happened? What else will be affected?

It is also called status accounting.

43. What is the need for baseline?

Need for Baseline :

(i) Basis for further development

(ii) Uses formal change control procedure for change

(iii) Helps to control change

44. Define SCM.

It is an umbrella activity that is applied throughout software process. It has a set of tracking and control activities that begin when a software engineering project begins and terminates only when the software project is taken out of operation.

45. List the SCM Activities.

(i) Identify a change

(ii) Control change

(iii) Ensure that change is being properly implemented

(iv) Report changes to others who may have an interest

46. What is meant by software reusability?

A software component should be designed and implemented so that it can be reused in many different programs.

47. What is CASE ?

CASE : Computer Aided Software Engineering

CASE provides the engineer with the ability to automate manual activities and to improve engineering insight.

48. Write the distinction between SCM and software support.

SCM : It has a set of tracking and control activities that begin when a software engineering project begins and terminates only when the software project is taken out of operation.

Software support : It has a set of software engineering activities that occur after software has been delivered to the customer and put into operation.

49. What is the difference between basic objects and aggregate objects used in software configuration.
 Basic Objects : It represents unit of text. E.g Section of requirement specification, Source listing for a component
 Aggregate objects: Collection of basic objects. And other aggregate objects. E.g Full design specification
50. What is configuration Audit?
 Has the change specified in ECO been made?
 Formal technical review been conducted?
 Software Engineering procedures for noting the change, recording it, reporting it been followed?
 SCI is updated?

Essay Type Questions(in Brief)

51. Explain Linear Sequential Model and prototyping model in detail
 Linear Sequential Model :
 Explanation, Diagram , Advantages, Disadvantages
 Prototyping model:
 Explanation, Diagram , Advantages, Disadvantages
52. Explain Spiral model and win-win spiral model in detail. .
 Spiral Model :
 Six Task Regions : Customer Communication
 Planning
 Risk Analysis
 Engineering
 Construction and Release
 Customer Evaluation
 Diagram , Details of four circles
 Win-Win spiral model:
 The customer and the developer enter into the process of negotiation, where the customer may be asked to balance functionality, performance, and other product against cost and time to market.

Activities, diagram ,explanation

53. Explain incremental model in detail
 Explanation of increments in the stages of
 Analysis, Design, Code, Test.
54. Discuss about fourth generation techniques.
 4GT :
 It encompasses a broad array of software tools. Each tool enables the software developer to specify some characteristics of software at a higher level.
 Explanations of : 4GT Tools
 4GT Paradigm
 Current state of 4GT approaches

55. Explain the Activities of Project Planning
 Software scope with an example (Conveyor Line Sorting System)
 Resources
 Hardware/ Software Tools
56. Explain the cost estimation procedure using COCOMO Model.
 It is algorithmic cost model. (One of the Empirical estimation model)
 COCOMO Model: 10 steps
 3 different sizing options
 Explanation
57. Explain the following:
- (i) Delphi Cost Estimation
 - (ii) Putnam Estimation model
 - (iii) Decomposition approach

Ans :

- (i) Delphi cost estimation
 Procedures to calculate
 - (ii) Putnam estimation model (Dynamic multi variable model)
 Explanation of the software equation
 - (iii) Decomposition approach
 Write an algorithm
58. Explain the organizational structure of the software development.
 Explanations of
 Project structure
 Programming team structure
 Management by objectives.
59. Explain the process of ‘ Risk Analysis and Management.’
 Risk Identification
 Risk Estimation
 Risk Assessment
 Risk Management and Monitoring
 Risk Refinement
60. Explain the following (i) Software requirement specification.
 (ii) Specification Review

Ans :

- (i) Software Requirement Specification :
 Information Description
 Functional Description
 Behavioral Description
 Validation criteria
 Bibliography and appendix
 Preliminary user’s manual
 - (ii) Specification Review : Explanation
61. Explain the types of coupling and cohesion.
 Coupling : Measure of the relative interdependence among modules.
 Types: Data coupling , Stamp coupling, control coupling, External coupling,
 Common coupling, Content coupling
 Cohesion : It is a measure of the relative functional strength of a module.
 Types: Coincidentally cohesive, Logically cohesive, Temporal cohesion,
 procedural cohesion, communicational cohesion, High cohesion, sequential cohesion.

62. Explain the various software design concepts

Explanations of Abstraction, Refinement, Modularity, Software Architecture , Control hierarchy, Structural partitioning, Data structure , Software procedure , Information hiding, Verification, Aesthetics.

63. Explain Software Design Documentation in detail.

Design Documentation :

(Explanation of the following items and sub items)

Scope

Reference Documents

Design Description

Modules

File Structure and global data

Requirements Cross Reference

Test provisions

Packaging

Special Notes

Appendices

64. Discuss the design procedure for Real time and distributed system software.

Real Time and distributed system design :

Real Time systems : It must provide specified amounts of computation within fixed time intervals. (Explanation)

Distributed system : It consists of a collection of nearly autonomous processors that communicate to achieve a coherent computing system.

(Explanation)

65. Explain Jackson system development with an example.

Steps are : Entity Action step

Entity Structure step

Initial modeling step

Function step

System Timing step

Implementation step

Example : University with two campuses.

66. Explain Software Design Notations

Explanations of

Data Flow diagram , Structure charts, HIPO diagrams, procedure template, pseudocode, structured flow chart, Structured English, Decision tables.

67. Explain Data Flow Oriented design in detail.

The objective of this design is to provide a systematic approach for the derivation of program structure.

Design and information flow

Design process considerations

(Atleast one of the following with an example)

Transform flow and analysis

Transaction flow and analysis

68. Explain programming standards in detail

Explanation of all standards.

69. What is software reuse? Explain the various aspects of software reuse.

A software component should be designed and implemented so that it can be reused in many different programs.

Explanation of Aspects :

Software development with reuse
Software development for reuse
Generator based reuse
Application system reuse

70. Describe the various software configuration management tasks in detail.

Brief explanations of

SCM Definition
Activities
Process
Baselines
Software Configuration Items
Identification of objects
Version control
Change control
Configuration Audit
Status reporting

71. Write notes on Version Control and Change control

Version control : Description

Representations : (Evolution graph, Object Pool)

Change control : Description

Process of change control

72. What are CASE tools and their usage in Software Engineering ? Discuss each tool in brief.

Business process Engineering tools
Process modeling and management tools
Project planning tools
Risk Analysis tools
Project management tools
Requirements tracing tools
Documentation tools
System software tools
Quality Assurance tools
Database management tools
Software configuration management tools
Analysis and design tools
PRO/SIM tools
Interface design and development tools
Prototyping tools
Programming tools
Web development tools
Integration and testing tools
Static Analysis tools
Dynamic analysis tools
Test management tools
Client/Server testing tools
Re-Engineering tools

73. Explain Integrated CASE Environment in detail.

Explanations of

Integrated CASE Environment
Benefits

Integration Architecture

74. Explain CASE repository in detail

Definition

Functions

Features and content

DBMS features.

Special features of CASE

Repository features.

75. Explain Building blocks for CASE

CASE Tools

Integrated framework

Portability services

Operating system

Hardware platform

Environment Architecture

