

CS 504-Software Engineering-I

Grand Quiz December 2020

By: MCS of Virtuallians

IF there is any correction needed do it accordingly

1. Whole part structure is also called _____.
 - a. Generalization
 - b. Aggregation
 - c. Specialization
 - d. Association
2. A prototype is not the real product. It is rather just a real looking _____ of what would be eventually delivered and might not do anything useful.
 - a. Mock-up page 68
 - b. Ad-hoc
 - c. Design
 - d. Structure
3. In "Railway ticket reservation system" the roles such as inquiry, reservation, ticketing and cancellation are to be performed by the user called:-
 - a. Passenger
 - b. System analyst
 - c. System developer
 - d. System designer
4. A useful technique for evaluating the overall complexity of a proposed architecture is to look at the component's _____.
 - a. number and size of components
 - b. flow dependencies and sharing dependencies
 - c. size and cost
 - d. function points
5. _____ relationship is concerned with classes not with the class instantiates.
 - a. Association
 - b. Inheritance
 - c. Aggregation
 - d. Composition
6. Software development is a step-by-step process, and in _____ phase of software development Business Objective of an organization gets cleared.
 - a. Maintenance
 - b. Development
 - c. Definition
 - d. Vision
7. The data on which the program operates is also considered as part of the _____.
 - a. Important Data
 - b. Software page 01
 - c. Logical Data
 - d. Utility Software

8. From the following which is/are not the example(s) of illegal data flow in Data Flow Diagram (DFD)
- External Agents directly communicating with each other
 - External Agent updating information in a DataStore
 - External Agent accessing information from a Data Store
 - There must be an intermediate process which should transform data received from one external entity and then send the transformed data to the other external entity
9. In sequence diagram, the objects are organized in a _____ line and the events in a _____ time line.
- Horizontal, straight
 - Vertical, straight
 - Horizontal, vertical
 - Vertical, horizontal
10. Consider the following piece of code:
- ```
public class Square extends Shape {
 // some code
}
```
- The above code is an example of:
- Part-Whole relationship
  - Generalization/Specialization
  - Data Sharing
  - Data encapsulation
11. Requirement engineering mainly deals with the \_\_\_\_\_ of the system
- Vision Phase
  - Definition Phase page 16
  - Development Phase
  - Maintenance Phase
12. A \_\_\_\_\_ relationship indicates that one entity is composed of one or more parts which are themselves instances of that or another entity.
- Inheritance
  - Whole-part
  - Generalization
  - Specialization
13. In UML Object Model Notation \_\_\_\_\_
- C++ language-specific programs are constructed
  - mathematical problems are visualised
  - relationships among classes and sub-classes are expressed page 92
  - Graphs and tables are used to explain Software Engineering principles
14. In the case of \_\_\_\_\_, intra component linkages are stronger while inter component linkages are weak.
- high cohesion
  - low coupling page 73
  - low cohesion
  - high coupling

15. Transactions are the \_\_\_\_\_ that must be remembered through time.
- a. Events page 93
  - b. Actions
  - c. Triggers
  - d. Methods
16. The modules interacting with each other through message passing have \_\_\_\_\_ between them.
- a. low cohesion
  - b. high cohesion
  - c. low coupling page 74
  - d. high coupling
17. In UML based Object Oriented model of a system, the inheritance relation between two classes is shown by a \_\_\_\_\_ sign towards the super-class side on the association line between the two.
- a. A filled diamond
  - b. An unfilled diamond
  - c. A half arrowhead
  - d. An arrowhead page 92
18. Which statement is not according to the software engineering principles? Software engineering is a(n) \_\_\_\_\_.
- a. Balancing act
  - b. Disciplined approach
  - c. Unsystematic approach page 5
  - d. Quantifiable approach
19. A \_\_\_\_\_ is not the real product but just a real looking mock-up of what would be eventually delivered.
- a. Software
  - b. Program
  - c. Prototype (page 68)
  - d. Test Case
20. The best way to conduct a requirements validation review is to
- a. examine the system model for errors
  - b. have the customer look over the requirements
  - c. send them to the design team and see if they have any concerns
  - d. use a checklist of questions to examine each requirement
21. More powerful hardware resulted into the development of \_\_\_\_\_ powerful and \_\_\_\_\_ software.
- a. less, complex
  - b. more, complex page 4
  - c. more, simple
  - d. less, simple

22. \_\_\_\_\_ pointed out the elegant conceptual integrity exhibited by layered organization of software systems, with the resulting gains in development and maintenance ease.
- David Parnas
  - Shaw and Garlan
  - Barry Boehm
  - Edsger Dijkstra page 115
23. The architecture components for product engineering are
- data, hardware, software, people
  - data, documentation, hardware, software
  - data, hardware, software, procedures
  - documentation, hardware, people, procedures
24. In Object oriented design, combining the services offered by an object with the attributes they work on, results in:
- Lower coupling and stronger cohesion
  - Lower cohesion and stronger coupling
  - Increased likelihood of reuse
  - Decreases the modularity of the system
- a only
  - band c
  - a and c
  - b and d
25. Architectural design process involves performing a number of activities which includes system structuring, control modeling and \_\_\_\_\_.
- System Analysis
  - Modular Decomposition page 120
  - Testing
  - Graphical User Interface
26. In use case diagram, the scope of the system is defined by:
- Actor
  - Entity
  - System Boundary
  - "Extends" relationship
27. Specialization means
- Calling the same method with object of child object
  - Hiding the data
  - Creating new subclasses from an existing class page 34 , 86
  - Abstraction
28. Which of the following sentence is true regarding user interface design?
- GUI interfaces are good for all tasks which a user needs to perform at an interface.
  - The higher the response time, the better is the interface
  - The simpler the interface, the efficient is the system
  - Command-line interfaces are faster for some tasks which the user needs to perform

29. In the architecture trade-off analysis method the architectural style should be described using the
- data Flow view and process view
  - data Flow view and module view
  - module view and process view
  - data Flow view, module view and process view page 136
30. In Data Flow Diagram, the entity or system, outside the boundary of this system is called
- Process
  - Data Flow
  - External Agent
  - Data Store
31. A cohesive Class is one which emphasizes on \_\_\_\_\_ unit of functionality
- Single
  - Multiple page 76
  - Static
  - Both Single and Multiple
32. The goal of \_\_\_\_\_ is to translate the customer's desire for a set of defined capabilities into a working product.
- Electrical Engineering
  - Product Engineering
  - Hardware Engineering
  - Mechanical Engineering
33. In Object Oriented Design, \_\_\_\_\_ layer contains the details that enable each object to communicate with its collaborators
- Subsystem
  - Responsibility
  - Message page 89
  - Object
34. A DFD is normally leveled (adding more levels of abstraction) as
- it is a good idea in design
  - it is recommended by many experts
  - it is easy to do it
  - it is easier to read and understand a number of smaller DFDs than one large DFD
35. As per Peter Coad's methodology, which of the following may NOT be a perfect candidate for being an object?
- Zone
  - Recipient
  - Garage
  - Password page 93 94

36. Which of the given component of software engineering framework provides different techniques that can be used to perform a particular task?
- Processes
  - Tools
  - Methods page 12
  - Quality Focus
37. To determine the architectural style or combination of styles that best fits the Proposed system requirements engineering is used to uncover
- algorithmic complexity
  - characteristics and constraints page 126
  - control and data
  - design patterns
38. In \_\_\_\_\_ relationship, a class shares the structure and behavior defined in another class.
- Aggregation
  - Composition
  - Inheritance page 86
  - Uses
39. Prototyping is used when there is regarding requirements.
- Uncertainty (page 68)
  - Confirmation
  - Conflict
  - Consensus
40. The process of utilizing our knowledge of computer science in effective production of software systems is called
- Chemical Engineering
  - Electrical Engineering
  - Computer Engineering
  - Software Engineering page 2
41. In Abbot's Textual Analysis technique, different parts of speech are identified within the text of the specification and these parts are modeled using different \_\_\_\_
- Event
  - Process
  - Operations
  - Components page 90
42. To help separate an object's external behavior from its implementation, the technique used is called \_\_\_\_\_
- Generalization
  - Association
  - Composition
  - Abstraction page 86

43. A Process in Data Flow Diagram (DFD) represents
- Flow of data
  - Transformation of data (page 49)
  - Storage of data
  - An external agent
44. In data flow diagram (DFD). Create, Update, Delete and Read operations are normally called:
- CRUD operations page 53
  - DURC operations
  - RUDC operations
  - CDUR operation
45. Collaboration diagram can show \_\_\_\_\_
- Binary messages
  - Asynchronous messages
  - Synchronous messages
- a only
  - b only
  - c only
  - both b and c page 111
46. Data cannot flow from one external entity to other external entity because
- It will get corrupted
  - It is not allowed in DFD page 59
  - An external entity has no mechanism to read or write
  - Both are outside the context of the system
47. In \_\_\_\_\_ Phase of software development, Requirement Engineer focuses on realizing the Business Object of an under developed product.
- Maintenance
  - Development page 16
  - Definition
  - Vision
48. Which of the following is not part of software architecture?
- Databases
  - data design
  - program structure
  - algorithm details
49. Selecting objects in a domain) include:
- Actors. Participants and Places
  - Only Participants
  - Only Actors
  - Only Actors and Places
50. Defining the services of an object means:
- What it does? Page 96
  - What it knows?
  - Who knows it?
  - Whom it knows?

51. In sequence diagram, the solid lines depict:
- Objects (or classes)
  - Messages being sent from one object to the other page 107
  - Life-time of an object
  - state of the object
52. Sequence diagrams:
- Provide the static behavior
  - Provide Data Flow
  - Provide a time-based view 106
  - Provide parallel data flow
53. \_\_\_\_\_ requirements cause frequent modifications in user interface
- Functional
  - Non-functional
  - Unstable page 62
  - User
- 54., security and maintainability are the types of \_\_\_\_\_ requirements.
- Non-functional page 120
  - Domain
  - Functional
  - Business
55. A change becomes \_\_\_\_\_ because of close presence of data and functions
- Accessible
  - Global
  - Private
  - Localized page 81
56. System \_\_\_\_\_ are built to allow the system engineer to evaluate the system components in relationship to one another.
- Requirements
  - Documents
  - Models page 42
  - Test cases
57. In \_\_\_\_\_ phase of software development, requirement analyst focuses on possible design of the proposed solution.
- Maintenance
  - Development (page 16)
  - Definition
  - Vision
58. The focus of sequence diagrams is:
- On objects/classes and messages exchanged among them (page 106)
  - On static Model of system
  - On object constraints
  - On the flow of Control

59. In Data Flow Diagram (DFD), data flow can:
- Only originate from an external entity
  - Only terminate in an external entity
  - Originate and terminate in an external entity
  - Either originate or terminate in an external entity but not both
60. In \_\_\_\_\_ the analyst determines all the sources of requirements and where do these requirements consume
- Data Flow Analysis
  - Source and Sink Analysis (page 40)
  - Down Parsing
  - Up Parsing
61. In a top-down system analysis, a/an \_\_\_\_\_ required to develop high level view of the system at first.
- Analyst page 54
  - Designer
  - Tester
  - Developer
62. The Use case diagram shows that which \_\_\_\_\_ interact with each use case.
- Use case
  - Actor page 32
  - Component
  - Relation
63. The context diagram is used as the top level abstraction in a \_\_\_\_\_ developed according to principles of structured analysis.
- Dataflow diagram (page 31)
  - Activity Diagram
  - State Transition Diagram
  - Use Case Diagram
64. Different messages in sequence diagrams includes:
- Simple
  - Asynchronous
  - Notify
  - Both Simple and Asynchronous page 108
65. A software requirement document describes all the \_\_\_\_\_ provided by the system along with the constraints under which it must operate.
- Conditions
  - Services page 23
  - Events
  - Processes
66. In case of a \_\_\_\_\_ message, the called routine that handles the message is completed before the caller resumes execution.
- Synchronous
  - Asynchronous
  - Bidirectional
  - a only page 108
  - b only
  - c only
  - both b and c

67. In object oriented design, \_\_\_\_\_ layer contains the data structures and algorithmic design for all attributes and operations for each object.
- Subsystem
  - Class and Object
  - Message
  - Responsibility** page 89
68. Asynchronous messages:
- are implemented as operation call
  - These block caller before response
  - occurs in multi-threaded applications** (page 109)
  - are shown by dotted line
69. In UML based object oriented model of a system, a composition relation between two objects is shown by a \_\_\_\_\_ sign on the whole side of a relation line.
- An unfilled diamond
  - A filled diamond**
  - A half diamond
  - A dot
70. In UML based Object Oriented Model of a system, the diamond sign is used to depict \_\_\_\_\_ relations between two objects/classes.
- Aggregation and Association
  - Inheritance and Association
  - Composition and Aggregation**
  - Composition, Aggregation and Association
71. A “register” in “Point of sale system” is an example of:
- Actor
  - Participant
  - Tangible thing** page 100
  - Transaction
72. A use case represents
- A class, its attributes and operation
  - An operation’s interfaces and signature
  - The role a user plays when interacting with the system** page 32
  - The system’s functionality for a particular purpose
73. In the case of \_\_\_\_\_ approach, data is decomposed according to functionality requirements.
- Object-oriented
  - Action-oriented** page 80
  - Event-oriented
  - Process-oriented
74. To construct a system model the engineer should consider one of the following restraining factors?
- Assumptions and constraints**
  - Budget and expenses
  - Data object and operation
  - Schedule and milestone

75. When two components of a system are using the same global data area, they are related as
- Data coupling
  - Content coupling
  - Common coupling [google](#)
  - External coupling
76. \_\_\_\_\_ Structure represents the internal organization of the various data and control items.
- Data
  - Value
  - Information
  - Conceptual
77. The method of dividing and assigning different portions of a large system to different groups for construction is called \_\_\_\_\_
- Work Breakdown Structure [\(page 119\)](#)
  - Working Boundary Structure
  - Work Basic Structure
  - Work Breakdown System
78. "A car is made up of a body, three or four wheels, a steering mechanism, a braking mechanism and a power-engine"  
The above statement is example of :
- Whole-Part relationship [page 95](#)
  - Inheritance
  - Specialization
  - Generalization
79. In software engineering paradigm, any engineering approach must be founded on organizational commitment to \_\_\_\_\_.
- Cost
  - Scheduling
  - Quality [\(page 15\)](#)
  - Performance
80. \_\_\_\_\_ is a technique that can be used to reduce customer dissatisfaction at requirement stage.
- Analysis
  - Negotiation
  - Prototyping [\(page 71\)](#)
  - GUI
81. Which of the following activities are included in the design process of a software architecture \_\_\_\_\_ ?
- System Development and Deployment
  - Architectural Analysis and Testing
  - Requirement Specifications of the system
  - System Structuring and Modular Decomposition

82. In UML based Object Oriented model of a system, the association relation between two objects is depicted by \_\_\_\_\_.
- A straight line
  - A filled diamond sign on the whole side of the line
  - An unfilled diamond sign on the whole side of the line
  - Any arrowhead sign on one side of the line
83. \_\_\_\_\_ are kind of umbrella activities that are used to smoothly and successfully perform the construction activities
- Designee activities
  - Management activities (page 14)
  - Testing activities
  - Maintenance activities
84. Software Design discusses \_\_\_\_\_ aspect of software development
- What
  - How
  - Who
  - When
85. Include and extend relationship is used in UML notation of a/an \_\_\_\_\_.
- Activity Diagram
  - Data Flow Diagram
  - Entity Relationship Diagram
  - Use Case Model
86. External Entity may be
- source of input data only
  - source of input data or destination of results
  - destination of results only
  - repository of data
87. An object model of a system captures the \_\_\_\_\_ structure of a system.
- Static page 93
  - dynamic
  - iterative
  - Hierarchical
88. The criteria used to assess the quality of an architectural design should be based on system
- accessibility and reliability
  - data and control
  - functionality
  - implementation details
89. In case of \_\_\_\_\_ approach, decomposition of a problem revolves around data
- Object-oriented
  - Action-oriented page 80
  - Event-oriented
  - Process-oriented

90. How can we implement generalization in Object Oriented programming languages?
- Polymorphism
  - Encapsulation
  - Abstraction
  - Inheritance
91. Which of the following is NOT among one of the four layers of the Object Oriented (OO) design pyramid
- The subsystem layer
  - The class and object layer
  - The Abstract layer page 89
  - The message layer
92. \_\_\_\_\_ is a technique in which we construct a model of an entity based upon its essential characteristics and ignore the inessential details.
- Inheritance
  - Polymorphism
  - Aggregation
  - Abstraction page 79
93. The project manager would need \_\_\_\_\_ document to monitor and track the progress of the project.
- Design
  - Project
  - Requirement page 18
  - Planning
94. In the case of action-oriented approach data is decomposed according to:
- Object requirements
  - Functionality requirements
  - Corresponding domain model
  - Compatibility with object interface
95. An architectural style encompasses which of the following elements?
- Constraints, Set of Components and Semantic Models page 126
  - Set of Components and Semantic Models
  - Semantic Models and Constraints
  - Set of Components and Constraints
96. In order to determine the role and responsibilities of the identified objects, we need to consider which of the following step(s):
- Who I am?
  - What I know?
  - Who I know?
  - What I do?
- a only
  - a and b
  - b, c and d page 102
  - c and d

97. In sequence diagram, the boxes denote:

- a. Objects (or classes) page 106
- b. Messages, sent from one object to other
- c. Life-time of Objects
- d. Relationships

98. \_\_\_\_\_ is a system component that provides services to other components but would not normally be considered as a separate system.

- a. Method
- b. Module page 121
- c. Message
- d. Relationship

99. A tangible entity in the real life is called \_\_\_\_\_.

- a. Functions
- b. Object page 85
- c. Class
- d. Sub-Class

100. Sequence of messages can be present in:

- a) Use case diagram
- b) Sequence diagram
- c) Collaboration diagram

- a. a only
- b. b only
- c. c only
- d. b and c

101. The system model template contains which of the following elements

- a. Input
- b. Output
- c. System Out
- d. Input/Output

102. Identify the TRUE statement:

- a. Normally Object Oriented design is more maintainable than functional oriented.
- b. Software with Functional oriented design does not fulfill non functional requirements.
- c. Object Oriented design cannot implement "Separation of concerns" strategy
- d. Function Oriented design does not lead to an efficient product

103. An external entity that interacts with a system is called a(n):

- a. Use case
- b. Actor
- c. Stakeholder
- d. Association

104. By levelling a DFD (adding more levels of abstraction) we mean

- a. Splitting it into different levels
- b. Make its structure uniform
- c. Expanding a process into one with more sub-processes giving more detail
- d. Summarizing a DFD to specify only the essentials

105. Software Engineering is the combination of tools, techniques and \_\_\_\_\_.
- Testing
  - Processes page 6
  - Maintenance
  - Design
106. \_\_\_\_\_ is concerned with decomposing a system into interacting sub-systems.
- System Structuring
  - Control Modeling
  - Modular Decomposition page 121
  - Work Breakdown Structure
107. There are \_\_\_\_\_ most important characteristics of an object.
- Six
  - Four
  - Two
  - Three
108. "System should maintain transaction log of every transaction"  
The above statement is an example of
- Functional requirement
  - Non-functional requirement
  - Pseudo requirement
  - Both Non-functional and Pseudo requirements
109. In object oriented approach, \_\_\_\_\_ are the people and organizations that take part in the system under consideration.
- Actors
  - Places
  - Participants
  - Tangible things
110. The modules that interact with each other through message passing have \_\_\_\_\_
- Low coupling page 73
  - High coupling
  - Low cohesion
  - High cohesion
111. Identifying Whole-part structures (Aggregations) means what are my \_\_\_\_\_
- Components page 94
  - Attributes
  - Methods
  - Messages
112. In "point of sale system" the term payment represents
- Actor
  - Participant
  - Transaction (page 99)
  - Container

113. Return values in synchronous messages are:
- Compulsory
  - May not used when response is obvious (page 109)
  - Not used at all
  - Represented by solid lines
114. GUI stand for
- Generic user interface
  - Graphic user interface
  - Generic user interaction
  - Graphical user interaction
115. Which of the following is not the object model principle?
- Abstraction
  - Encapsulation
  - Hierarchy or inheritance
  - Exposure page 86
116. The system specification describes the
- function and behavior of a computer-based system
  - implementation of each allocated system element
  - algorithmic detail and data structures
  - time required for system simulation
117. The \_\_\_\_\_ provides the software engineer with a view of the system as a whole.
- Process Model
  - Architectural Model
  - Business model
  - Requirements Model
118. Requirement engineering focuses on \_\_\_\_\_ aspect of the software development process.
- Both what and how
  - What
  - How
  - why and how
119. An arrow in Data Flow Diagram (DFD) represents
- Direction of flow of data
  - Processing of data
  - External agent
  - Internal agent
120. The \_\_\_\_\_ relationship is kind of a generalization specialization relationship.
- Bit-Byte
  - Uses
  - Binary
  - Extends

121. In the case of \_\_\_\_\_ in a system, module boundaries are not well defined.
- low cohesion
  - high coupling
  - low coupling
  - high cohesion
122. \_\_\_\_\_ Component of software engineering framework provides automated or semi-automated support in a software development.
- Processes
  - Methods
  - Quality Focus
  - Tools
123. In Data Flow Diagram (DFD), one data store cannot directly copy the data from another \_\_\_\_\_.
- Agent
  - Process
  - Data store
  - Flow
124. All the documents related to the software are also considered as part of the \_\_\_\_\_.
- Physical Document
  - Logical Document
  - Relational Database
  - Software
125. Which elements of business processing engineering are the responsibilities of the software engineer?
- business area analysis
  - business system design
  - product planning
  - information strategy planning
126. A class will be cohesive if:
- Class does not implement Complex interfaces
  - Class does not have complex methods
  - If most of the methods do not use most of the data members most of the time
  - If most of the methods use most of the data members most of the time
127. A context diagram is used
- As the first step in developing a detailed DFD of a system
  - In systems analysis of very complex systems
  - As an aid to system design
  - As an aid to programmers
128. \_\_\_\_\_ is an extremely powerful technique for dealing with complexity.
- Aggregation
  - Abstraction
  - Inheritance
  - Association

129. In multiprocessing applications, different execution threads may pass information to one another by sending \_\_\_\_\_ to each other.
- Interrupt calls
  - Synchronous messages
  - Asynchronous messages
  - System calls
130. Normally a system is more easy to modify if its modules have
- High coupling and high cohesion
  - High coupling and low cohesion
  - Low coupling and high cohesion
  - Low coupling and Low cohesion
131. UML is a language for
- High-level Programming
  - Low-level Programming
  - Modeling and Design
  - Creating diagrams only
132. \_\_\_\_\_ is not the part of Peter Coad methodology
- Select actors
  - Select participants
  - Select places
  - Select actions
133. Which of the following is the external quality of software product?
- Correctness
  - Concision
  - cohesion
  - Low coupling
134. While establishing the services for an object, the goal is to keep data and action together for \_\_\_\_\_ coupling and \_\_\_\_\_ cohesion
- Lower, Higher
  - Higher, Lower
  - Lower, Lower
  - Higher, Higher
135. A necessary supplement to transform or transaction mapping needed to create a complete architectural design is
- entity relationship diagrams
  - the data dictionary
  - processing narratives for each module
  - test cases for each module
136. Construction activities are directly related to software \_\_\_\_\_.
- Management
  - Planning
  - Quality Assurance
  - Development

137. OOD results in a design that achieves a number of different levels of \_\_\_\_\_.
- a. Operation
  - b. Event page 89
  - c. Modularity
  - d. Process
138. At which stage of software development loop, results are delivered?
- a. Problem definition
  - b. Solution integration
  - c. Technical development
  - d. Status quo
139. \_\_\_\_\_ Requirements are often called product features.
- a. Functional
  - b. Non-functional
  - c. Developer
  - d. User
140. Strong cohesion implies that
- a. All parts of component have a class logical relationship with each other
  - b. All part of component do not have a close logical relationship with each other
  - c. Component is dynamic in nature
  - d. Component is static in nature
141. \_\_\_\_\_ Diagram does not capture control flow information; it just shows the flow of the data in a system.
- a. Sequence
  - b. Data Flow
  - c. Activity
  - d. Class
142. A life line represents the object's life during the interaction in a sequence diagram while its notation is depicted by \_\_\_\_\_
- a. Solid Lines
  - b. Dotted Lines
  - c. Full Arrow
  - d. Curved Lines
143. An attribute that varies over time, e.g. price of an item, should be replaced by a/an \_\_\_\_\_ with an effective date and value
- a. Additional Class
  - b. Additional Method/Function
  - c. Interface
  - d. Structure
144. Which of the given component of software engineering framework demands rational and Timely development of a software?
- a. Tools
  - b. Methods
  - c. Quality Focus
  - d. Processes

145. A context diagram
- a. describes detailed design of a system
  - b. is a DFD which gives an overview of the system
  - c. is a detailed description of a system
  - d. is not used in drawing a detailed DFD
146. Which one is not a part of Software Development phase?
- a. Construction
  - b. Scope
  - c. Project Vision
  - d. Definition
147. Software architecture must address \_\_\_\_\_ requirement of a software system
- a. Functional
  - b. Non Functional
  - c. User interface Requirements
  - d. Both Functional and Non Functional
148. If Cat is derived from Mammal Class, and Mammal is derived from Animal Class, then:
- a. Cat will inherit Animal's functions and data
  - b. Cat will not inherit Animal's functions and data
  - c. Cat will not be able to access any class
  - d. Cat is allowed to access only the Mammal's Class
149. Which of the following is not supported by a maintainable design?
- a. Change
  - b. debugging
  - c. Adding new features
  - d. Higher maintenance cost
150. The condition that must be met before the use case can be invoked, is called:
- a. Pre-Condition
  - b. Post-Condition
  - c. Pre-Assertion
  - d. Post-Assertion
- 151.