

1. Which of the following is not one of the broad categories used to classify operations?

- a. Computation
- b. data manipulation
- c. event monitors

d. transformers

2. Objects inherit a class

a. Operations

- b. Name
- c. Relationships
- d. Instance

3. Data object description incorporates data and

- a. Its report
- b. occurrence

c. attributes

- d. instances

4. The maximum number of objects that can participate in a relationship is called

- a. Directionality
- b. Composition
- c. Multiplicity

d. Cardinality

5. Which of the following is not an objective for building an analysis model?

- a. define set of software requirements that can be validated
- b. describe customer requirements

c. develop an abbreviated solution for the problem

- d. establish basis for software design

6. Class based modeling defines

a. Objects

- b. Clarity
- c. Completeness
- d. Consistency

7. Name the diagram which represents the flow of activities described by the use cases and at the same time the actors who are involved in UML is

- a. Use case diagram
- b. Activity diagram
- c. Swim lane diagram**
- d. Data flow diagram

8. Term used to indicate whether or not a particular data object must participate in a relationship is

a. Modality

- b. Multiplicity
- c. Cardinality
- d. Directionality

9. Class responsibilities are defined by

- a. its attributes only
- b. its collaborators
- c. its operations only

d. both its attributes and operations

10. Classes communicate with one another via

- a. Identifier
- b. Messages**
- c. Methods
- d. Attributes

11. Which of the following are areas of concern in the design model?

- a. Architecture
- b. Data structures
- c. Project scope

d. Architecture and Data structures

12. Which design model elements are used to depict a model of information represented from the user's view?

- a. Architectural design elements
- b. Component-level design elements

c. Data design elements

- d. Interface design elements

13. Which of the following is not a goal of design engineering? To produce a model or representation that exhibits

- a. Firmness
- b. Commodity

c. Principles

- d. Delight

14. Cohesion is a qualitative indication of the degree to which a module

- a. can be written more compactly
- b. focuses on just one thing**

- c. is able to complete its function in a timely manner
- d. is connected to other modules and the outside world

15. What types of abstraction are used in software design?

- a. Control
- b. Data
- c. Procedural

d. Control, data and procedural

16. Coupling is a qualitative indication of the degree to which a module

- a. can be written more compactly
- b. focuses on just one thing

- c. is able to complete its function in a timely manner
- d. is connected to other modules and the outside world**

17. Techniques that allows a software engineer to understand how a work process is completed when several people (and roles) are included

a. **Work flow analysis**

b. UI analysis

c. Data flow analysis

d. Use case analysis

18. _____ is a measure of how well a computer system facilitates learning.

a. Completeness

b. Consistency

c. **Usability**

d. Analysis

19. Which one of these is a golden rule in interface design process?

a. Requirements gathering

b. **Reduce users memory load**

c. Interface prototype

d. Feedback from test drive

20. Interface consistency implies that

a. each application should have its own distinctive look and feel

b. input mechanisms remain the same throughout the application

c. visual information is organized according to a design standard

d. **both input mechanisms remain the same throughout the application and visual information is organized according to a design standard**

21. Mechanics for referring the refining the processing tasks that are required for software to accomplish some desired function is

a. Work flow analysis

b. **Functional decomposition**

c. Use case analysis

d. UI analysis

22. Which model depicts the profile of the end users of a computer system?

a. Design model

b. Implementation model

c. **User model**

d. User s model

23. System response time has two important characteristics. They are:

a. Interactive and knowledgeable

b. **Length and variability**

c. Correct and precise

d. Error detection and structured

24. Which one of the following is a step for user interface design modeling?

a. **Define events that will cause the state of the user interface to change.**

b. Report partitions for ease of understanding.

c. Error messages and warnings have to be presented to the user

d. Customize screen locations.

25. Which is one of four common design issues?

a. physical environment

b. navigation

c. **command labeling**

d. patterns

26. _____ from the analysis model is used to accomplish the definition of interface objects and actions applied to them.

a. ER diagrams

b. **Use cases**

c. Data structure models

d. CRC cards

27. An important step in interface design is

a. modeling

b. interface analysis

c. physical environment in which the objects should work

d. **definition of interface objects and the action applied on them**

28. Name any two design evaluation techniques.

a. application accessibility and internationalization

b. **feedback and amount of learning required by users**

c. user request help and page layout

d. system response time and user help facilities

29. The two characteristics of system response time are

a. **length and variability**

b. menu and labeling

c. analysis and patterns

d. speed and time

30. First level prototype is evaluated by

a. Developer

b. **User**

c. Designer

d. Tester

31. To collect qualitative data, _____ can be distributed to users of the prototype.

a. CASE tools

b. Reusable components of the design

c. Use cases

d. **Questionnaires**

32. The Unit test focuses on _____ and _____ within the boundaries of a component.

a. inconsistencies and omissions

b. outright errors and quality

c. **internal processing logic and data structures**

d. verification and validation of units

33. A systematic technique for constructing the software architecture while interfacing to uncover errors is called

a. Unit testing

b. **Integration testing**

- c. System testing
- d. Smoke testing

34. _____ testing is the re-execution of some subset of tests that have already been conducted to ensure that changes have not propagate d.

- a. Regression te sting
- b. Smoke testing
- c. Integration testing
- d. Sandwich testing

35. _____ refers to the set of activities that ensure that software correctly implements a specific function.

- a. Verification
- b. Validation
- c.
- d.

36. Are we building the right product? It is

- a. Verification
- b. Validation
- c.
- d.

37. The process of uncovering errors, diagnosing them and correcting them is called

- a. testing
- b. planning
- c. debugging
- d. designing

38. Initially, tests focus on each component individually for functionality of the component and this test is called

- a. Unit Testing
- b. Integration Testing
- c. System Testing
- d. Basis path testing

39. In an object oriented testing strategy the focus of testing is

- a. a unit
- b. a class
- c. a package
- d. a component

40. _____ testing uses top-down tests for upper levels of the program structure, coupled with bottom-up tests for subordinate levels.

- a. Sandwich testing
- b. Integration testing
- c. Smoke testing
- d. Regression testing

41. Which one of these belongs to integration testing in the OO context?

- a. Class testing
- b. Thread based testing
- c. Validation testing
- d. Formal technical review

42. Software is tested to uncover _____ that were made inadvertently as it was designe d and constructed

- a. faults
- b. errors
- c. bugs
- d. mistakes

43. _____ testing is designed to test the run-time performance of software within the context of an integrated system.

- a. Integration testing
- b. Performance testing
- c. System testing
- d. Validation testing

44. Brute force, backtracking, cause elimination are strategies used in

- a. art of debugging
- b. system testing
- c. smoke testing
- d. performance testing

45. Alpha test is conducted at _____ and beta test is conducted at _____.

- a. end user s site , developer s site
- b. developer s site, end user s site
- c.
- d.

46. Recovery testing is a part of Integration testing.

- a. True
- b. False
- c.
- d.

47. Loop testing is a black box testing technique.

- a. True
- b. False
- c.
- d.

48. Validation testing begins at the culminations of _____ testing.

- a. System testing
- b. Smoke testing
- c. Integration testing
- d. OO testing

49. Software _____ is how easily a computer program can be tested.

- a. testability
- b. operability
- c. Decomposability
- d. Stability

50. During security testing the tester plays the role of the individual who desires to

- a. debug the system
- b. identify errors during integration
- c. penetrate the system
- d. check for validations

51. _____ test examines some fundamental aspect of a system with little regard for the internal logical structure of the software.

a. black box test

b. white box test

c. glass box test

d. stability test

52. _____ test is also called glass box test.

a. white box test

b. black box test

c. stability test

d. operability test

53. _____ test guarantees that all independent paths within a module have been exercised at least once.

a. black box test

b. white box test

c. glass box test

d. stability test

54. The _____ depicts logical control flow and is a notation for the representation of control flow.

a. Basis path testing

b. Flow graph notation

c. Test case notation

d. Control structure testing

55. _____ is a black box testing method that divided the input domain of a program into classes of data from which test cases can be derived.

a. Equivalence partitioning

b. Graph based testing methods

c. Boundary value analysis

d. Orthogonal array testing

56. _____ metrics for testing leads to the design of effective test cases and evaluate the efficacy of testing.

a. in-process metric

b. complexity metric

c. statement and branch coverage metric

d. defect related metric

57. One of the following is a metric used to estimate the cost or effort required to design, code and test the software.

a. Number of external inquiries

b. Lines of code

c. Function based metric

d. Defect related metric

58. _____ is a quantitative measure of the degree to which a system component or process possesses a given attribute.

a. Measurement

b. Metric

c. Software quality

d. Product quality

59. One among the following is a metric for the analysis model.

a. system size

b. component level metrics

c. in-process metric

d. complexity metric

60. _____ according to Sears is a worthwhile design metric for human / computer interfaces.

a. coupling metrics

b. class size

c. methods inheritance factor

d. layout appropriateness

61. The physical connections between elements of the OO design represent _____.

a. Cohesion

b. Completeness

c. Coupling

d. Sufficiency

62. Cohesion metrics and Coupling metrics are metrics for _____ level design.

a. Component level

b. Class oriented

c. Method inheritance factor

d. Operation oriented

63. _____ is a process by which numbers or symbols are assigned to the attributes of entities in the real world in such a way as to define them accordingly to clearly defined rules.

a. product metrics

b. assessment of quality

c. quantitative assessment of quality

d. measurement

64. _____ is conformance to explicitly stated functional and performance requirements, explicitly documented development standards and implicit characteristics that are expected of all professionally developed software.

a. Measurement

b. Metrics

c. Software quality

d. Product metrics

65. This metric is the maximum length from the node to the root of the tree.

a. Number of children

b. Depth of the inheritance tree

c. Coupling between object classes

d. Width of the tree

66. This metric indicates the number of classes (or methods) that can access another

class's attributes, a violation of encapsulation.

- a. LCOM
- b. PAP
- c. PAD**
- d. NOR

67. To ensure that methods do not generate side effects Lack of cohesion in methods (LCOM) value must be as _____ as possible.

- a. high
- b. less**
- c.
- d.

68. _____ is computed with the following formula: .

- a. Software Maturity Index(SMI)**
- b. Public Access to data members(PAD)
- c. Percent public and protected (PAP)
- d. Fan in (FIN)

69. Fanin in an OO context indicates _____

- a. encapsulation
- b. abstraction
- c. inheritance**
- d. polymorphism

70. Metrics for OO systems focus on measurement that can be applied to the _____ and _____ characteristics like encapsulation, information hiding.

- a. class and design**
- b. operators and operands
- c. coupling and cohesion
- d. human and computer interaction

71. The number of distinct operators and operands, the total number of operator and operand occurrences measures are used in _____

- a. Cyclamate complexity
- b. Halstead's theory**
- c. Product metrics
- d. Object oriented metrics

72. The controllable factors (determinants) in improving software quality and organizational performance are process, people , _____ and _____

- a. measures and metrics
- b. product and technology**
- c. skill and motivation
- d. methods and tools

73. LOC measures are programming language dependent that when productivity is considered.

- a. True**
- b. False
- c.

d.

74. Among the following which two are process metrics.

- a. lines of code and function points
- b. effort and cost**
- c. review hours and delivered source lines
- d.

75. Function Point measures are programming language dependent.

- a. True
- b. False**
- c.
- d.

76. Number of support classes is one among several OO metrics.

- a. True**
- b. False
- c.
- d.

77. _____ is a detailed sequence of steps that describes the interaction between the user and the application.

- a. Key classes
- b. Support classes
- c. Scenario scripts**
- d. Subsystem

78. A _____ is an aggregation of classes that support a function that is visible to the end-user of a system.

- a. scenario scripts
- b. key classes
- c. support classes
- d. subsystems**

79. Mention any two web engineering project metrics.

- a. scenario scripts and number of key classes
- b. number of support classes and subsystems
- c. number of static pages and number of internal page link**
- d. use cases and lines of code

80. Mention any two Indirect measures of the product .

- a. lines of code and execution speed
- b. defects reported over set period of time and cost
- c. effort and time estimate
- d. quality and efficiency**

81. _____ metrics are derived by normalizing quality and /or productivity measures by considering the size of the software that has been produced.

- a. direct
- b. indirect
- c. size oriented**
- d. process oriented

82. _____ is the degree to which the software performs its required function.

- a. Correctness
- b. Integrity
- c. Usability
- d. Maintainability

83. _____ is a measure of the filtering ability of quality assurance and control activities as they are applied throughout all process framework activities.

- a. Usability
- b. Integrity
- c. Defect removal efficiency
- d. Maintainability

84. The most common measure for correctness is

_____.

- a. KLOC
- b. Function point
- c. Security
- d. Defects per KLOC

85. COCOMO stands for

- a. Customer cost model
- b. Constructive cost model
- c. Consistent cost model
- d. Configuration cost model

86. $DRE = E_i / (E_i + E_f)$ where E_i is

- a. number of errors found during software engineering activity i
- b. number of defects found before delivery in activity i
- c. number of defects found after delivery in activity i
- d. number of errors found in the i iteration of testing.

87. MTTC stands for

- a. Maintenance to track changes
- b. Mean time to change
- c. Metrics to translate change
- d. Measure to track changes

88. _____ is the probability that an attack of a specific type will occur within a given time.

- a. Integrity
- b. Security
- c. Threat
- d. Attack

89. SEI stands for

- a. Software Enterprise Inc
- b. Software Engineering Institute
- c. Software Enterprise Integration
- d. Software Engineering Integration

90. A quality objective for a software team is to achieve DRE that approaches ____.

- a. 0.01
- b. 0.1
- c. 1
- d. 0.001

91. Staff turnover, poor communication with the customer are risks which are extrapolated from past experience and are called _____ risks

- a. technical risks
- b. known risks
- c. predictable risks
- d. unpredictable risks

92. Software risk always involves two characteristics

- a. fire fighting and crisis management
- b. known and unknown risks
- c. uncertainty and loss
- d. staffing and budget

93. Software risk impact assessment should focus on consequences affecting

- a. planning, resources, cost, schedule
- b. marketability, cost, personnel
- c. business, technology, process
- d. performance, support, cost, schedule

94. Reactive risk strategies are also called fire fighting mode.

- a. True
- b. False
- c.
- d.

95. The reason for refining risks is to break them into smaller units having different consequences.

- a. True
- b. False
- c.
- d.

96. Building an excellent product or system that no one really wants is

- a. technical risk
- b. business risk
- c. known risk
- d. project risk

97. The risk which gives the degree of uncertainty that the project schedule will be maintained and that the product will be delivered in time is _____.

- a. known risk
- b. unknown risk
- c. schedule risk
- d. technical risk

98. _____ risks identify potential design, implementation, interface, verification and maintenance problems.

- a. technical risk
- b. business risk
- c. known risk
- d. project risk

99. Three major categories of risks are

a. business risks, personnel risks, budget risks

b. project risks, technical risks, business risks

c. planning risks, technical risks, personnel risks

d. management risks, technical risks, design risks

100. Risk projection attempts to rate each risk in two ways

a. likelihood and size

b. likelihood and probability

c. likelihood and impact

d. likelihood and mitigation

101. Software _____ and _____

_____ analysis are software

quality assurance activities that focus on the

identification and assessment of

potential hazards that may affect software

negatively and cause an entire system

to fail.

a. RM and MI

b. Risk management and risk

c. Safety and hazard

d. Cost and risk mitigation

102. The overall risk exposure RE is determined by the product of _____

and _____.

a. probability of occurrence for a risk and cost to the project should the risk occur

b. probability of risk not occurring and cost of the project should the risk occur

c. probability of risk not occurring and cost of project

d. probability of identification of risk and cost of risk impact

103. A risk referent level is a risk component value (performance, cost, support, schedule) or combination of values that cause a project to be terminated.

a. True

b. False

c.

d.

104. Risk information sheets (RIS) are never an acceptable substitute for a full risk mitigation, monitoring, and management (RMMM) plan.

a. True

b. False

c.

d.

105. Risk management and contingency planning assumes that mitigation efforts have failed and that risk has become a reality.

a. True

b. False

c.

d.

106. The risk projection and analysis techniques described in the chapter are applied iteratively as the software project proceeds.

a. True

b. False

c.

d.

107. Risk monitoring involves watching the risk indicators defined for the project and not determining the effectiveness of the risk mitigation steps themselves.

a. True

b. False

c.

d.

108. An effective risk management plan will need to address which of the following issues?

a. risk avoidance

b. risk monitoring

c. contingency planning

d. risk avoidance ,risk monitoring and contingency planning

109. Hazard analysis focuses on the identification and assessment of pote ntial hazards that can cause

a. project termination

b. schedule slippage

c. external problems

d. entire system to fail

110. Which factors affect the probable consequences likely if a risk does occur?

a. risk cost

b. risk timing

c. risk resources

d. risk predictability

111. The goal of quality assurance is to provide management with the data needed to determine which software engineers are producing the most defects.

a. True

b. False

c.

d.

112. Quality management is an umbrella activity that is applied throughout the software process.

a. True

b. False

c.

d.

113. There is no need to assess customer satisfaction when trying to determine the quality of a piece of software.

- a. True
- b. False**
- c.
- d.

114. People who perform software quality assurance must look at the software from the customer's perspective.

- a. True**
- b. False
- c.
- d.

115. Quality costs may be divided into costs associated with

- a. prevention, appraisal, and failure**
- b. people, process, and product
- c. customers, developers, and maintenance
- d. all of the above

116. Variation control in the context of software engineering involves controlling variation in the _____

- a. process applied
- b. resources expended
- c. product quality attributes
- d. process applied , resources expended and product quality attributes**

117. The key concept of quality control is that all work products

- a. are delivered on time and under budget
- b. have complete documentation
- c. have measurable specifications for process outputs**
- d. are thoroughly tested before delivery to the customer

118. Defect amplification models can be used to illustrate the costs associated with using software from its initial deployment to its retirement.

- a. True
- b. False**
- c.
- d.

119. The purpose of software reviews is to uncover errors in work products so they can be removed before moving on to the next phase of development

- a. True**
- b. False
- c.
- d.

120. In general the earlier a software error is discovered and corrected the less costly to the overall project budget

- a. True**
- b. False
- c.
- d.

121. SQA is a planned and systematic pattern of actions that are required to ensure high quality in software.

- a. True**
- b. False
- c.
- d.

122. Which one of the following is not an SQA activity?

- a. Audits
- b. Recording of noncompliance
- c. Reviews SE activities for compliance with defined software process
- d. White box Testing**

123. Which of these activities is not one of the activities recommended to be performed by an independent SQA group?

- a. prepare SQA plan for the project
- b. review software engineering activities to verify process compliance
- c. report any evidence of noncompliance to senior management
- d. serve as the sole test team for any software produced**

124. Statistical quality assurance involves

- a. using sampling in place of exhaustive testing of software
- b. surveying customers to find out their opinions about product quality
- c. tracing each defect to its underlying cause, isolating the "vital few" causes, and moving to correct them**
- d. tracing each defect to its underlying causes and using the Pareto principle to correct each problem found

125. Which of the following are objectives for formal technical reviews?

- a. allow senior staff members to correct errors
- b. assess programmer productivity
- c. determining who introduced an error into a program
- d. uncover errors in software work products**

126. The purpose of software reviews is to uncover errors in work products so they can be removed before moving on to the next phase of development

- a. True**
- b. False
- c.
- d.

127. Defect amplification models can be used to illustrate the costs associated

with using software from its initial deployment to its retirement.

- a. True
- b. False**
- c.
- d.

128. At the end of a formal technical review all attendees can decide to

- a. accept the work product without modification**
- b. modify the work product and continue the review
- c. reject the product due to stylistic discrepancies
- d.

129. Attempts to apply mathematical proofs to demonstrate that a program conforms to its specifications are doomed to failure.

- a. True
- b. False**
- c.
- d.

130. Sample driven reviews only make sense for very small software development projects.

- a. True
- b. False**
- c.
- d.

131. In any type of technical review, the focus of the review is on the product and not the producer.

- a. True**
- b. False
- c.
- d.

132. A review summary report answers which three questions?

- a. terminate project, replace producer, request a time extension
- b. what defects were found, what caused defects, who was responsible
- c. what was reviewed, who reviewed it, what were the findings**
- d.

133. In general the earlier a software error is discovered and corrected the less costly to the overall project budget.

- a. True**
- b. False
- c.
- d.

134. A Defect amplification model is used to illustrate the generation and detection of errors during the preliminary ____, ____, and ____ steps of a software engineering process.

- a. design, detail design, coding**
- b. communication, planning and coding
- c. communication, elaboration and implementation
- d. inception, elaboration and implementation

135. Six Sigma methodology defines three core steps.

- a. analyze, improve, control
- b. analyze, design, verify
- c. define, measure, analyze**
- d. define, measure, control

136. The audience to a formal review presentation consist of

- a. Only technical staff
- b. only customers
- c. analyst, developers and technical staff.
- d. customers, management and technical staff**

137. Software reliability problems can almost always be traced to

- a. errors in accuracy
- b. errors in design**
- c. errors in operation
- d.

138. Poka-yoke devices are mechanisms that lead to the

- a. creation of quality processes with minimal resources
- b. determining causes of software defects
- c. prevention of potential quality problems**
- d.

139. The ISO quality assurance standard that applies to software engineering is

- a. ISO 9000:2004
- b. ISO 9001:2000**
- c. ISO 9002:2001
- d. ISO 9003:2004

140. Which of the following is not a section in the standard for SQA plans recommended by IEEE?

- a. budget**
- b. documentation
- c. reviews and audits
- d. tests

141. A special set of ISO guidelines have been developed to help interpret the standard for use in software process and that is

- a. ISO 9000-1
- b. ISO 9000-2
- c. ISO 9000-3**
- d. ISO 9000-4

142. Software safety is a quality assurance activity that focuses on hazards that

- a. affect the reliability of a software component
- b. may cause an entire system to fail**
- c. may result from user input errors
- d. prevent profitable marketing of the final prod