

第1章

1. Which question no longer concerns the modern software engineer?
 - A. Why does computer hardware cost so much?
 - B. Why does software take a long time to finish?
 - C. Why does it cost so much to develop a piece of software?
 - D. Why can't software errors be removed from products prior to delivery?Answer: A
2. Software is a product and can be manufactured using the same technologies used for other engineering artifacts
 - A. True
 - B. FalseAnswer: B
3. Software deteriorates rather than wears out because
 - A. Software suffers from exposure to hostile environments
 - B. Defects are more likely to arise after software has been used often
 - C. Multiple change requests introduce errors in component interactions
 - D. Software spare parts become harder to orderAnswer: C
4. WebApps are a mixture of print publishing and software development, making their development outside the realm of software engineering practice.
 - A. True
 - B. FalseAnswer: B
5. There are no real differences between creating WebApps and MobileApps.
 - A. True
 - B. FalseAnswer: B
6. In its simplest form an external computing device may access cloud data services using a web browser.
 - A. True
 - B. FalseAnswer: A
7. Product line software development depends the reuse of existing software components to provide software engineering leverage.
 - A. True
 - B. FalseAnswer: A

第2章

1. Which of the items listed below is not one of the software engineering layers?
 - A. Process
 - B. Manufacturing
 - C. Methods
 - D. ToolsAnswer: B
2. Software engineering umbrella activities are only applied during the initial phases of software development projects.

A. True

B. False

Answer: B

3. Which of these are the 5 generic software engineering framework activities?

A. communication, planning, modeling, construction, deployment

B. communication, risk management, measurement, production, reviewing

C. analysis, designing, programming, debugging, maintenance

D. analysis, planning, designing, programming, testing

Answer: A

4. Planning ahead for software reuse reduces the cost and increases the value of the systems into which they are incorporated.

A. True

B. False

Answer: A

5. The essence of software engineering practice might be described as understand the problem, plan a solution, carry out the plan, and examine the result for accuracy.

A. True

B. False

Answer: A

6. In agile process models the only deliverable work product is the working program.

A. True

B. False

Answer: B

7. A most software development projects are initiated to try to meet some business need.

A. True

B. False

Answer: A

8. In general software only succeeds if its behavior is consistent with the objectives of its designers.

A. True

B. False

Answer: B

第3章

1. Which of the following are recognized process flow types?

A. Concurrent process flow

B. Iterative process flow

C. Linear process flow

D. Spiral process flow

E. both a and c

Answer: E

2. The communication activity is best handled for small projects using six distinct actions (inception, elicitation, elaboration, negotiation, specification, validation).

A. True

B. False

Answer: B

3. A good software development team always uses the same task set for every project to insure high quality work products

A. True

B. False

Answer: B

4. Software processes can be constructed out of pre-existing software patterns to best meet the needs of a software project.

A. True

B. False

Answer: A

5. Which of these are standards for assessing software processes?

A. SEI

B. SPICE

C. ISO 9000

D. ISO 9001

E. both b and d

Answer: E

第4章

1. The waterfall model of software development is

A. A reasonable approach when requirements are well defined.

B. A good approach when a working program is required quickly.

C. The best approach to use for projects with large development teams.

D. An old fashioned model that is rarely used any more.

Answer: A

2. The incremental model of software development is

A. A reasonable approach when requirements are well defined.

B. A good approach when a working core product is required quickly.

C. The best approach to use for projects with large development teams.

D. A revolutionary model that is not used for commercial products.

Answer: B

3. Evolutionary software process models

A. Are iterative in nature.

B. Can easily accommodate product requirements changes.

C. Do not generally produce throwaway systems.

D. All of the above.

Answer: D

4. The prototyping model of software development is

A. A reasonable approach when requirements are well defined.

B. A useful approach when a customer cannot define requirements clearly.

C. The best approach to use for projects with large development teams.

D. A risky model that rarely produces a meaningful product.

Answer: B

5. The spiral model of software development

A. Ends with the delivery of the software product.

B. Is more chaotic than the incremental model.

C. Includes project risks evaluation during each iteration.

D. All of the above.

Answer: C

6. The concurrent development model is

A. Another name for concurrent engineering.

B. Defines events that trigger engineering activity state transitions.

C. Only used for development of parallel or distributed systems.

D. Used whenever a large number of change requests are anticipated.

E. Both a and b

Answer: E

7. The component-based development model is

A. Only appropriate for computer hardware design.

B. Not able to support the development of reusable components.

C. Dependent on object technologies for support.

D. Not cost effective by known quantifiable software metrics.

Answer: C

8. The formal methods model of software development makes use of mathematical methods to

A. Define the specification for computer-based systems.

B. Develop defect free computer-based systems.

C. Verify the correctness of computer-based systems.

D. All of the above.

Answer: D

9. Which of these is not one of the phase names defined by the Unified Process model for software development?

A. Inception phase

B. Elaboration phase

C. Construction phase

D. Validation phase

Answer: D

10. Which of these is not a characteristic of Personal Software Process?

A. Emphasizes personal measurement of work product.

B. Practitioner requires careful supervision by the project manager.

C. Individual practitioner is responsible for estimating and scheduling.

D. Practitioner is empowered to control quality of software work products.

Answer: B

11. Which of these are objectives of Team Software Process?

A. Accelerate software process improvement

B. Allow better time management by highly trained professionals

C. Build self-directed software teams

D. Show managers how to reduce costs and sustain quality

E. Both b and c

Answer: E

12. Process technology tools allow software organizations to compress schedules by skipping unimportant activities.

A. True

B. False

Answer: B

13. It is generally accepted that one cannot have weak software processes and create high quality end products.

A. True

B. False

Answer: A

第8章

1. Requirements engineering is a generic process that does not vary from one software project to another.

A. True

B. False

Answer: A

2. During project inception the intent of the of the tasks are to determine

A. basic problem understanding

B. nature of the solution needed

C. people who want a solution

D. none of the above

E. a, b, c

Answer: E

3. Three things that make requirements elicitation difficult are problems of

A. budgeting

B. scope

C. understanding

D. volatility

E. b, c, d

Answer: E

4. A stakeholder is anyone who will purchase the completed software system under development.

A. True

B. False

Answer: B

5. It is relatively common for different customers to propose conflicting requirements, each arguing that his or her version is the right one.

A. True

B. False

Answer: A

6. Which of the following is not one of the context-free questions that would be used during project inception?

A. What will be the economic benefit from a good solution?

B. Who is behind the request for work?

C. Who will pay for the work?

D. Who will use the solution?

Answer: C

7. Non-functional requirements can be safely ignored in modern software development projects.

A. True

B. False

Answer: B

8. In collaborative requirements gathering the facilitator

A. arranges the meeting place

B. can not be a customer

C. controls the meeting

D. must be an outsider

Answer: C

9. Which of the following is not one of the requirement classifications used in Quality Function Deployment (QFD)?

A. exciting

B. expected

C. mandatory

D. normal

Answer: C

10. The work products produced during requirement elicitation will vary depending on the
- A. size of the budget.
 - B. size of the product being built.
 - C. software process being used.
 - D. stakeholders needs.
 - E. both a and b

Answer: E

11. User stories are complete descriptions the user needs and include the non-functional requirements for a software increment.
- A. True
 - B. False

Answer: A

12. Developers and customers create use-cases to help the software team understand how different classes of end-users will use functions.
- A. True
 - B. False

Answer: A

13. Use-case actors are always people, never system devices.
- A. True
 - B. False

Answer: B

14. The result of the requirements engineering task is an analysis model that defines which of the following problem domain(s)?
- A. information
 - B. functional
 - C. behavioral
 - D. all of the above

Answer: D

15. Analysis patterns facilitate the transformation of the analysis model into a design model by suggesting reliable solutions to common problems.
- A. True
 - B. False

Answer: A

16. In agile process models requirements engineering and design activities are interleaved.
- A. True
 - B. False

Answer: A

17. In win-win negotiation, the customer's needs are met even though the developer's need may not be.
- A. True
 - B. False

Answer: B

18. In requirements validation the requirements model is reviewed to ensure its technical feasibility.
- A. True
 - B. False

Answer: B

19. The most common reason for software project failure is lack of functionality.
- A. True
 - B. False

Answer: B

第9章

1. Which of these is not an element of a requirements model?

- A. Behavioral elements
- B. Class-based elements
- C. Data elements
- D. Scenario-based elements

Answer: C

2. Which of the following is not an objective for building a requirements 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

Answer: C

3. Object-oriented domain analysis is concerned with the identification and specification of reusable capabilities within an application domain.

- A. True
- B. False

Answer: A

4. In structured analysis models focus on the structure of the classes defined for a system along with their interactions.

- A. True
- B. False

Answer: B

5. Creation and refinement of use cases is an important part of scenario-based modeling.

- A. True
- B. False

Answer: A

6. It is important to consider alternative actor interactions when creating a preliminary use case.

- A. True
- B. False

Answer: B

7. Brainstorming is one technique that may be used to derive a complete set of use case exceptions.

- A. True
- B. False

Answer: A

8. In many cases there is no need to create a graphical representation of a usage scenario.

- A. True
- B. False

Answer: A

9. UML activity diagrams are useful in representing which analysis model elements?

- A. Behavioral elements
- B. Class-based elements
- C. Flow-based elements
- D. Scenario-based elements

Answer: D

10. UML swimlane diagrams allow you to represent the flow of activities by showing the actors having responsibility for creating each data element.

- A. True

B. False
Answer: B

第10章

1. Which of the following should be considered as candidate objects in a problem space?

- A. events
- B. people
- C. structures
- D. all of the above

Answer: D

2. In the grammatical parse of a processing narrative the nouns become object candidates in the analysis model.

- A. True
- B. False

Answer: A

3. Attributes are chosen for an object by examining the problem statement and identifying the entities that appear to be related.

- A. True
- B. False

Answer: B

4. 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

Answer: D

5. Collaborators in CRC modeling are those classes needed to fulfill a responsibility on another card.

- A. True
- B. False

Answer: A

6. Which of the following items does not appear on a CRC card?

- A. class collaborators
- B. class name
- C. class reliability
- D. class responsibilities

Answer: C

7. Class responsibilities are defined by

- A. its attributes only
- B. its collaborators
- C. its operations only
- D. both its attributes and operations

Answer: D

8. A stereotype is the basis for class reuse in UML modeling.

- A. True
- B. False

Answer: B

9. An analysis package involves the categorization of analysis model elements into useful groupings.

- A. True

B. False
Answer: A

第11章

1. The behavior modeling is only used in the analysis of real-time systems.

- A. True
- B. False

Answer: B

2. For purposes of behavior modeling an event occurs whenever

- A. a state and process exchange information.
- B. the system and actor exchange information.
- C. two actors exchange information.
- D. two objects exchange information.

Answer: B

3. For purposes of behavior modeling a state is any

- A. consumer or producer of data.
- B. data object hierarchy.
- C. observable mode of behavior.
- D. well defined process.

Answer: C

4. The state transition diagram

- A. depicts relationships between data objects
- B. depicts functions that transform the data flow
- C. indicates how data are transformed by the system
- D. indicates system reactions to external events

Answer: D

5. The UML sequence diagram shows the order in which system events are processed.

- A. True
- B. False

Answer: B

6. Analysis patterns are discovered, they are not explicitly created.

- A. True
- B. False

Answer: A

7. It is not possible to justify the time required for mobile app requirements analysis.

- A. True
- B. False

Answer: B

8. Which is not one of the analysis activities that is used to create a complete analysis model?

- A. Configuration analysis
- B. Content analysis
- C. Functional analysis
- D. Market analysis

Answer: D

9. Content objects are extracted from use cases by examining the scenario description for direct or indirect content references.

- A. True
- B. False

Answer: A

10. What are the elements of a WebApp interaction model?
- A. activity diagrams, sequence diagrams, state diagrams, interface prototype
 - B. activity diagrams, collaboration diagrams, sequence diagrams, state diagrams
 - C. use-cases, sequence diagrams, state diagrams, interface prototype
 - D. use-cases, sequence diagrams, state diagrams, sequence diagrams
- Answer: C
11. UML activity diagrams can be used to represent the user observable functionality delivered by the WebApp as well as the operations contained in each analysis class.
- A. True
 - B. False
- Answer: A
12. Configuration analysis focuses on the architecture of the user's web browsing environment.
- A. True
 - B. False
- Answer: B

第12章

1. Which of the following are areas of concern in the design model?
- A. architecture
 - B. data
 - C. interfaces
 - D. project scope
 - E. a, b, c
- Answer: E
2. The importance of software design can be summarized in a single word
- A. accuracy
 - B. complexity
 - C. efficiency
 - D. quality
- Answer: D
3. Which of these are characteristics of a good design?
- A. exhibits strong coupling between its modules
 - B. implements all requirements in the analysis model
 - C. includes test cases for all components
 - D. provides a complete picture of the software
 - E. b and d
- Answer: E
4. Which of the following is not a characteristic common to all design methods?
- A. configuration management
 - B. functional component representation
 - C. quality assessment guidelines
 - D. refinement heuristics
- Answer: A
5. What types of abstraction are used in software design?
- A. control
 - B. data
 - C. environmental
 - D. procedural
 - E. a, b, d
- Answer: E

6. Which of the following can be used to represent the architectural design of a piece of software?

- A. Dynamic models
- B. Functional models
- C. Structural models
- D. All of the above

Answer: D

7. Design patterns are not applicable to the design of object-oriented software?

- A. True
- B. False

Answer: B

8. Since modularity is an important design goal it is not possible to have too many modules in a proposed design.

- A. True
- B. False

Answer: B

9. Information hiding makes program maintenance easier by hiding data and procedure from unaffected parts of the program.

- A. True
- B. False

Answer: A

10. 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.

Answer: B

11. 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.

Answer: D

12. When using structured design methodologies the process of stepwise refinement is unnecessary.

- A. True
- B. False

Answer: B

13. Software designs are refactored to allow the creation of software that is easier to integrate, easier to test, and easier to maintain.

- A. True
- B. False

Answer: A

14. Which of the following is not one of the five design class types

- A. Business domain classes
- B. Entity classes
- C. Process classes
- D. User interface classes

Answer: B

15. 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

Answer: C

16. Which design is equivalent to the floor plan of a house?

- A. Architectural design
- B. Component-level design
- C. Data design
- D. Interface design

Answer: A

17. Which design model is equivalent to the detailed drawings of the access points and external utilities for a house?

- A. Architectural design
- B. Component-level design
- C. Data design
- D. Interface design

Answer: D

18. Which design model is equivalent to a set of detailed drawings for each room in a house?

- A. Architectural design
- B. Component-level design
- C. Data design
- D. Interface design

Answer: B

19. The deployment design elements specify the build order for the software components.

- A. True
- B. False

Answer: B

第13章

1. The best representation of system architecture is an operational software prototype.

- A. True
- B. False

Answer: B

2. The architectural representations can be an enabler for communication among project stakeholders.

- A. True
- B. False

Answer: A

3. An architectural description is often documented using an architecture template.

- A. True
- B. False

Answer: B

4. An architectural decision is often documented using an architecture decision description template.

- A. True
- B. False

Answer: A

5. An architectural genre will often dictate the architectural approach that may be used for the structure to be built.

- A. True

B. False

Answer: A

6. An architectural style encompasses which of the following elements?

A. constraints

B. set of components

C. semantic models

D. syntactic models

E. a, b, c

Answer: E

7. To determine the architectural style or combination of styles that best fits the proposed system, requirements engineering is used to uncover

A. algorithmic complexity

B. characteristics and constraints

C. control and data

D. design patterns

Answer: B

8. Before an architectural pattern can be chosen for use in a specific system it must have a code implementation to facilitate its reuse.

A. True

B. False

Answer: B

9. The criteria used to assess the quality of an architectural design should be based on system

A. accessibility

B. control

C. data

D. implementation

E. b and c

Answer: E

10. Software architectural considerations often interact with each other and moderate each other.

A. True

B. False

Answer: A

11. Developer notes are not a reliable means of documenting architectural decisions

A. True

B. False

Answer: B

12. During process of modeling the system in context, systems that interact with the target system are represented as

A. Peer-level systems

B. Subordinate systems

C. Superordinate systems

D. Working systems

E. a, b, c

Answer: E

13. Once selected, archetypes always need to be refined further as architectural design proceeds.

A. True

B. False

Answer: A

14. Which of the following is not an example of infrastructure components that may need to be integrated into the software architecture?

- A. Communications components
- B. Database components
- C. Interface components
- D. Memory management components

Answer: C

15. In the architecture trade-off analysis method the architectural style should be described using the

- A. data flow view
- B. module view
- C. process view
- D. user view
- E. a, b, c

Answer: E

16. A useful technique for evaluating the overall complexity of a proposed architecture is to look at the component

- A. cohesion
- B. flow dependencies
- C. sharing dependencies
- D. size
- E. b and c

Answer: E

17. Software architects need to create consensus among software team members and other stakeholders.

- A. True
- B. False

Answer: A

18. Pattern-based architectural reviews can be useful for project with short build cycles and volatile requirements.

- A. True
- B. False

Answer: A

19. Static architectural conformance checking assesses whether or not the source code matches the user visible requirements.

- A. True
- B. False

Answer: B

20. Architectural design has no role in agile software process models.

- A. True
- B. False

Answer: B

第14章

1. In the most general sense a component is a modular building block for computer software.

- A. True
- B. False

Answer: A

2. In the context of object-oriented software engineering a component contains

- A. attributes and operations
- B. instances of each class
- C. roles for each actor (device or user)

D. set of collaborating classes

Answer: D

3. In traditional software engineering modules must serve in which of the following roles?

A. Control component

B. Infrastructure component

C. Problem domain component

D. All of the above

Answer: D

4. Software engineers always need to create components from scratch in order to meet customer expectations fully.

A. True

B. False

Answer: B

5. Which of the following is not one of the four principles used to guide component-level design?

A. Dependency Inversion Principle

B. Interface Segregation Principle

C. Open-Closed Principle

D. Parsimonious Complexity Principle

Answer: D

6. The use of stereotypes can help identify the nature of components at the detailed design level.

A. True

B. False

Answer: A

7. Classes and components that exhibit functional, layer, or communicational cohesion are relatively easy to implement, test, and maintain.

A. True

B. False

Answer: A

8. Software coupling is a sign of poor architectural design and can always be avoided in every system.

A. True

B. False

Answer: B

9. In component design elaboration requires which of the following elements to be describe in detail?

A. Algorithms

B. Attributes

C. Interfaces

D. Operations

E. b, c, d

Answer: E

10. In component-level design persistent data sources refer to

A. Component libraries

B. Databases

C. Files

D. All of the above

E. b and c

Answer: E

11. WebApp content design at the component level focuses on content objects and the manner in which they interact.

A. True

B. False

Answer: B

12. A WebApp functional architecture describes the key functional components and how they interact with each other.

A. True

B. False

Answer: A

13. Component-level design for mobile apps is not any different from component-based design for Web apps.

A. True

B. False

Answer: A

14. Which of these constructs is used in structured programming?

A. branching

B. condition

C. repetition

D. sequence

E. b, c, d

Answer: E

15. In component-based software engineering, the development team examines the requirements to see which are amenable to composition, rather than construction, before beginning detailed design tasks.

A. True

B. False

Answer: A

16. Which of the following is not one of the major activities of domain engineering?

A. analysis

B. construction

C. dissemination

D. validation

Answer: D

17. Which of the following factors would not be considered during component qualification?

A. application programming interface (API)

B. development and integration tools required

C. exception handling

D. testing equipment required

Answer: D

18. Which of the following is a technique used for component wrapping?

A. black-box wrapping

B. clear-box wrapping

C. gray-box wrapping

D. white-box wrapping

Answer: B

19. Which of the following is not one of the issues that form a basis for design for reuse?

A. object-oriented programming

B. program templates

C. standard data

D. standard interface protocols

Answer: A

20. In a reuse environment, library queries are often characterized using the ____ element of the 3C Model.

- A. concept
- B. content
- C. context
- D. all of the above

Answer: D

第15章

1. Which of the following interface design principles does not allow the user to remain in control of the interaction with a computer?

- A. allow interaction to be interruptible
- B. allow interaction to be undoable
- C. hide technical internals from casual users
- D. only provide one rigidly defined method for accomplishing a task

Answer: D

2. Which of the following interface design principles reduce the user's memory load?

- A. define intuitive shortcuts
- B. disclose information in a progressive fashion
- C. establish meaningful defaults
- D. provide an on-line tutorial
- E. a, b, c

Answer: E

3. The reason for reducing the user's memory load is make his or her interaction with the computer quicker to complete.

- A. True
- B. False

Answer: B

4. 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. navigational methods are context sensitive
- D. visual information is organized according to a design standard
- E. b and d

Answer: E

5. If past interactive models have created certain user expectations it is not generally good to make changes to the model.

- A. True
- B. False

Answer: A

6. 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

Answer: C

7. Which model depicts the image of a system that an end user creates in his or her head?

- A. design model
- B. user model
- C. system model
- D. system perception

Answer: D

8. Which model depicts the look and feel of the user interface along with all supporting information?

- A. implementation model
- B. user model
- C. user's model
- D. system perception

Answer: A

9. Which of these framework activities is not normally associated with the user interface design processes?

- A. cost estimation
- B. interface construction
- C. interface validation
- D. user and task analysis

Answer: A

10. Which approach(es) to user task analysis can be useful in user interface design?

- A. have users indicate their preferences on questionnaires
- B. rely on the judgement of experienced programmers
- C. study existing computer-based solutions
- D. observe users performing tasks manually
- E. c and d

Answer: E

11. Object-oriented analysis techniques can be used to identify and refine user task objects and actions without any need to refer to the user voice.

- A. True
- B. False

Answer: B

12. The computer's display capabilities are the primary determinant of the order in which user interface design activities are completed.

- A. True
- B. False

Answer: B

13. It is sometimes possible that the interface designer is constrained by environmental factors that mitigate against ease of use for many users.

- A. True
- B. False

Answer: A

14. One means of defining user interface objects and actions is to conduct a grammatical parse of the user scenario.

- A. True
- B. False

Answer: A

15. Interface design patterns typically include a complete component-level design (design classes, attributes, operations, and interfaces).

- A. True
- B. False

Answer: A

16. Several common design issues surface for almost every user interface including

- A. adaptive user profiles
- B. error handling
- C. resolution of graphics displays
- D. system response time

E. b and d

Answer: E

17. It is more important to capture the user's attention with flashy features than ergonomically sound screen layouts when building a WebApp.

A. True

B. False

Answer: B

18. Several usability measures can be collected while observing users interacting with a computer system including

A. down time for the application

B. number of user errors

C. software reliability

D. time spent looking at help materials

E. b and d

Answer: E

第17章

1. Which of the following characteristics should not be used to assess the quality of a WebApp?

A. aesthetics

B. reliability

C. maintainability

D. usability

Answer: A

2. Which of the following are design goals for every WebApp?

A. Simplicity

B. Consistency

C. Navigability

D. Visual appeal

E. All of the above.

Answer: E

3. Which of the following not part of the design pyramid for WebE design?

A. Architectural design

B. Business case design

C. Content design

D. Navigation design

Answer: B

4. With WebApps content is everything, a poorly defined user interface will be quickly overlooked by frequent users.

A. True

B. False

Answer: B

5. Which of these are WebApp interaction mechanisms?

A. Graphic icons

B. Graphic images

C. Navigation menus

D. All of the above

Answer: D

6. Screen layout design has several widely accepted standards based on human factors research.

A. True

B. False

Answer: B

7. Graphic design considers every aspect of the look and feel of a WebApp.

A. True

B. False

Answer: A

8. Content design is conducted by

A. Copywriters and graphic designer

B. Web engineers

C. both a and b

D. none of the above

Answer: C

9. Content objects have both information attributes defined during analysis and implementation specific attributes specified during design.

A. True

B. False

Answer: A

10. Content objects are not normally chunked into Web pages until the implementation activities begin.

A. True

B. False

Answer: B

11. Content architecture and WebApp architecture are pretty much the same thing for many WebApps?

A. True

B. False

Answer: B

12. Which of the following is not one of the content architectural structures used by web engineers?

A. linear

B. grid

C. hierarchical

D. parallel

Answer: D

13. MVC is a three layer architecture that contains a

A. machine, view, content objects

B. model, view, and content objects

C. model, view, and controller

D. machine, view, controller

Answer: C

14. Web navigational design involves creating a semantic navigational unit for each goal associated with each defined user role.

A. True

B. False

Answer: A

15. To allow the user to feel in control of a WebApp, it is a good idea to mix both horizontal and vertical navigation mechanisms on the same page.

A. True

B. False

Answer: B

16. Component level design for WebApps is very similar to component level design for other software delivery environments.

A. True

B. False

Answer: A

17. Which of these is not one of the design activities associated with object-oriented hypermedia design?

A. abstract interface design

B. conceptual design

C. content design

D. navigational design

Answer: C

18. UML does not have any representation schemas that are useful in building WebApp design models.

A. True

B. False

Answer: B

第20章

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

A. True

B. False

Answer: A

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

A. True

B. False

Answer: A

3. 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

Answer: B

4. Review metrics can be used to assess the efficacy of each review activity.

A. True

B. False

Answer: A

5. Defect density can be estimated for any software engineering work product.

A. True

B. False

Answer: A

6. Agile software developers are aware that software reviews always take time without saving any.

A. True

B. False

Answer: B

7. The level of review formality is determined by which of the following?

A. amount of preparation

B. reviewer follow-up

C. size of project budget

D. structure of review

E. a, b, d

Answer: E

8. An informal review may consist of which of the following?

- A. casual meeting
- B. desk check
- C. inspection
- D. pair programming
- E. a and b

Answer: E

9. 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

Answer: D

10. 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. reject the product due to severe errors
- E. a and d

Answer: E

11. 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. none of the above

Answer: C

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

- A. True
- B. False

Answer: A

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

- A. True
- B. False

Answer: B

第22章

1. In software quality assurance work there is no difference between software verification and software validation.

- A. True
- B. False

Answer: B

2. The best reason for using Independent software test teams is that

- A. software developers do not need to do any testing
- B. strangers will test the software mercilessly
- C. testers do not get involved with the project until testing begins
- D. the conflicts of interest between developers and testers is reduced

Answer: D

3. What is the normal order of activities in which traditional software testing is organized?
- A. integration testing, system testing, unit testing, validation testing.
 - B. unit testing, validation testing, system testing, integration testing
 - C. unit testing, integration testing, validation testing, system testing
 - D. validation testing, system testing, integration testing, unit testing

Answer: C

4. By collecting software metrics and making use of existing software reliability models it is possible to develop meaningful guidelines for determining when software testing is done.

- A. True
- B. False

Answer: A

5. Which of the following strategic issues needs to be addressed in a successful software testing process?

- A. conduct formal technical reviews prior to testing
- B. specify requirements in a quantifiable manner
- C. use independent test teams
- D. wait till code is written prior to writing the test plan
- E. a and b

Answer: E

6. Which of the following need to be assessed during unit testing?

- A. algorithmic performance
- B. code stability
- C. error handling
- D. execution paths
- E. c and d

Answer: E

7. Units and stubs are not needed for unit testing because the modules are tested independently of one another.

- A. True
- B. False

Answer: B

8. Top-down integration testing has as its major advantage(s) that

- A. low level modules never need testing
- B. major decision points are tested early
- C. no drivers need to be written
- D. no stubs need to be written
- E. b and c

Answer: E

9. Bottom-up integration testing has as its major advantage(s) that

- A. major decision points are tested early
- B. no drivers need to be written
- C. no stubs need to be written
- D. regression testing is not required

Answer: C

10. Regression testing should be a normal part of integration testing because as a new module is added to the system new

- A. control logic is invoked
- B. data flow paths are established
- C. drivers require testing
- D. all of the above

Answer: E

11. Smoke testing might best be described as
- A. bulletproofing shrink-wrapped software
 - B. rolling integration testing
 - C. testing that hides implementation errors
 - D. unit testing for small programs
- Answer: B
12. When testing object-oriented software it is important to test each class operation separately as part of the unit testing process.
- A. True
 - B. False
- Answer: B
13. The OO testing integration strategy involves testing
- A. groups of classes that collaborate or communicate in some way
 - B. single operations as they are added to the evolving class implementation
 - C. operator programs derived from use-case scenarios
 - D. none of the above
- Answer: A
14. Since many WebApps evolve continuously, the testing process must be ongoing as well.
- A. True
 - B. False
- Answer: A
15. Testing MobileApps is not different than testing WebApps.
- A. True
 - B. False
- Answer: B
16. The focus of validation testing is to uncover places that s user will be able to observe failure of the software to conform to its requirements.
- A. True
 - B. False
- Answer: A
17. Software validation is achieved through a series of tests performed by the user once the software is deployed in his or her work environment.
- A. True
 - B. False
- Answer: B
18. Configuration reviews are not needed if regression testing has been rigorously applied during software integration.
- A. True
 - B. False
- Answer: B
19. Acceptance tests are normally conducted by the
- A. developer
 - B. end users
 - C. test team
 - D. systems engineers
- Answer: B
20. Recovery testing is a system test that forces the software to fail in a variety of ways and verifies that software is able to continue execution without interruption.
- A. True
 - B. False
- Answer: B

21. Security testing attempts to verify that protection mechanisms built into a system protect it from improper penetration.
- A. True
 - B. False
- Answer: A
22. Stress testing examines the pressures placed on the user during system use in extreme environments.
- A. True
 - B. False
- Answer: B
23. Performance testing is only important for real-time or embedded systems.
- A. True
 - B. False
- Answer: B
24. Debugging is not testing, but always occurs as a consequence of testing.
- A. True
 - B. False
- Answer: A
25. Which of the following is an approach to debugging?
- A. backtracking
 - B. brute force
 - C. cause elimination
 - D. code restructuring
 - E. a, b, c
- Answer: E

第23章

1. With thorough testing it is possible to remove all defects from a program prior to delivery to the customer.
- A. True
 - B. False
- Answer: B
2. Which of the following are characteristics of testable software?
- A. observability
 - B. simplicity
 - C. stability
 - D. all of the above
- Answer: D
3. The testing technique that requires devising test cases to demonstrate that each program function is operational is called
- A. black-box testing
 - B. glass-box testing
 - C. grey-box testing
 - D. white-box testing
- Answer: A
4. The testing technique that requires devising test cases to exercise the internal logic of a software module is called
- A. behavioral testing
 - B. black-box testing
 - C. grey-box testing

D. white-box testing

Answer: D

5. What types of errors are missed by black-box testing and can be uncovered by white-box testing?

A. behavioral errors

B. logic errors

C. performance errors

D. typographical errors

E. b and d

Answer: E

6. Program flow graphs are identical to program flowcharts.

A. True

B. False

Answer: B

7. The cyclomatic complexity metric provides the designer with information regarding the number of

A. cycles in the program

B. errors in the program

C. independent logic paths in the program

D. statements in the program

Answer: C

8. The cyclomatic complexity of a program can be computed directly from a PDL representation of an algorithm without drawing a program flow graph.

A. True

B. False

Answer: A

9. Condition testing is a control structure testing technique where the criteria used to design test cases is that they

A. rely on basis path testing

B. exercise the logical conditions in a program module

C. select test paths based on the locations and uses of variables

D. focus on testing the validity of loop constructs

Answer: B

10. Data flow testing is a control structure testing technique where the criteria used to design test cases is that they

A. rely on basis path testing

B. exercise the logical conditions in a program module

C. select test paths based on the locations and uses of variables

D. focus on testing the validity of loop constructs

Answer: C

11. Loop testing is a control structure testing technique where the criteria used to design test cases is that they

A. rely basis path testing

B. exercise the logical conditions in a program module

C. select test paths based on the locations and uses of variables

D. focus on testing the validity of loop constructs

Answer: D

12. Black-box testing attempts to find errors in which of the following categories

A. incorrect or missing functions

B. interface errors

C. performance errors

D. none of the above

E. a, b, c

Answer: E

13. Graph-based testing methods can only be used for object-oriented systems

A. True

B. False

Answer: B

14. Equivalence testing divides the input domain into classes of data from which test cases can be derived to reduce the total number of test cases that must be developed.

A. True

B. False

Answer: A

15. Boundary value analysis can only be used to do white-box testing.

A. True

B. False

Answer: B

16. Orthogonal array testing enables the test designer to maximize the coverage of the test cases devised for relatively small input domains.

A. True

B. False

Answer: A

17. Test derived from behavioral class models should be based on the

A. data flow diagram

B. object-relation diagram

C. state transition diagram

D. use-case diagram

Answer: C

18. Documentation does not need to be tested.

A. True

B. False

Answer: B

19. Real-time applications add a new and potentially difficult element to the testing mix

A. performance

B. reliability

C. security

D. time

Answer: D

第24章

1. It is not possible to test object-oriented software without including error discovery techniques applied to the system OOA and OOD models.

A. True

B. False

Answer: A

2. The correctness of the OOA and OOD model is accomplished using formal technical reviews by the software quality assurance team.

A. True

B. False

Answer: B

3. The consistency of object-oriented models may be judged by reviewing the CRC card model.

A. True

B. False

Answer: A

4. Test case design for OO software is driven by the algorithmic detail of the individual operations.

A. True

B. False

Answer: A

5. Integration testing of object-oriented software can be accomplished by which of the following testing strategies?

A. Cluster testing

B. Glass-box testing

C. Thread-based testing

D. Use-based testing

E. a, c, d

Answer: E

6. Validation of object-oriented software focuses on user visible actions and outputs from the system.

A. True

B. False

Answer: A

7. Encapsulation of attributes and operations inside objects makes it easy to obtain object state information during testing.

A. True

B. False

Answer: B

8. Use-cases can provide useful input into the design of black-box and state-based tests of OO software.

A. True

B. False

Answer: A

9. Fault-based testing is best reserved for

A. conventional software testing

B. operations and classes that are critical or suspect

C. use-case validation

D. white-box testing of operator algorithms

Answer: B

10. Scenario-based testing

A. concentrates on actor and software interaction

B. misses errors in specifications

C. misses errors in subsystem interactions

D. both a and b

Answer: A

11. Random order tests are conducted to exercise different class instance life histories.

A. True

B. False

Answer: A

12. Which of these techniques is not useful for partition testing at the class level

A. attribute-based partitioning

B. category-based partitioning

C. equivalence class partitioning

D. state-based partitioning

Answer: C

13. Multiple class testing is too complex to be tested using random test cases.
A. True
B. False
Answer: B
14. The state model can be used to derive test cases based on the dynamic behavior of an object-oriented system.
A. True
B. False
Answer: A

第25章

1. Which of the following is not one of the dimensions of quality used to assess a WebApp?
A. Content
B. Maintainability
C. Navigability
D. Usability
Answer: B
2. WebApps require special testing methodologies because WebApp errors have several unique characteristics.
A. True
B. False
Answer: A
3. Since WebnApps evolve continuously, the testing process is an on-going activity, conducted by the Web support staff using regression tests.
A. True
B. False
Answer: A
4. Test planning is not used in WebApp testing.
A. True
B. False
Answer: B
5. As the WebApp architecture is constructed which types of testing are used as integration tests?
A. Component testing
B. Content testing
C. Navigation testing
D. Usability testing
E. both a and c
Answer: E
6. Which of the following is not one of the objectives of WebApp content testing?
A. Find organizational or structure errors
B. Identify linking errors
C. Uncover semantic errors
D. Uncover syntactic errors
Answer: B
7. Database testing is very rarely a part of WebApp content testing.
A. True
B. False
Answer: B

8. The overall strategy for interface testing is to uncover errors

- A. in navigation semantics
- B. in overall usability
- C. related to specific interface mechanisms
- D. both a and c

Answer: D

9. Which of the following is not a WebApp interface mechanism?

- A. Browser
- B. Cookies
- C. Forms
- D. Links

Answer: C

10. When testing WebApp interface semantics, each use-case is used as input for the design of a testing sequence.

- A. True
- B. False

Answer: A

11. Usability tests should be designed and executed by intended users for a given WebApp.

- A. True
- B. False

Answer: B

12. WebApp compatibility testing is conducted to be sure that the user model for usage scenario matched the user category assigned to a given user.

- A. True
- B. False

Answer: B

13. Which test case design technique(s) are appropriate for WebApp component-level testing?

- A. Boundary value analysis
- B. Equivalence partitioning
- C. Path testing
- D. All of the above

Answer: D

14. The purpose of WebApp navigation syntactic testing is to ensure the correct appearance of each navigation mechanism.

- A. True
- B. False

Answer: B

15. Both Web engineers and non-technical users conduct navigation semantics testing for WebApps.

- A. True
- B. False

Answer: A

16. Which of following is not one of the elements that need to be considered when constructing WebApp server-side configuration tests?

- A. Browser compatibility
- B. Database software integration
- C. Operating system compatibility
- D. System security measures

Answer: A

17. To design client-side configuration tests each user category is assessed to reduce the number of configuration variables to a manageable number.

- A. True

B. False

Answer: A

18. Which of the following is not a testable WebApp security element?

A. Authentication

B. Encryption

C. Firewalls

D. Penetration

Answer: D

19. WebApp performance tests are designed to

A. asses WebApp usability

B. evaluate page loading times

C. simulate real-world loading situations

D. test network connectivity

Answer: C

20. Load testing involves determining the input of which 3 variables?

A. N, T, D

B. N, T, P

C. T, D, P

D. N, D, P

Answer: A

21. WebApp stress testing is a continuation load testing.

A. True

B. False

Answer: A

第29章

1. Which of these are valid software configuration items?

A. case tools

B. documentation

C. executable programs

D. test data

E. All of the above.

Answer: E

2. Which of the following is not considered one of the four important elements that should exist when a configuration management system is developed?

A. component elements

B. human elements

C. process elements

D. validation elements

Answer: D

3. Once a software engineering work product becomes a baseline it cannot be changed again.

A. True

B. False

Answer: B

4. Which configuration objects would not typically be found in the project database?

A. design specification

B. marketing data

C. organizational structure description

D. test plans

E. b and c

Answer: E

5. Modern software engineering practices usually attempt to maintain SCI's in a project database or repository.

A. True

B. False

Answer: A

6. A data repository meta model is used to determine how

A. information is stored in the repository

B. well data integrity can be maintained

C. easily the existing model can be extended

D. all of the above

Answer: D

7. Many data repository requirements are the same as those for a typical database application.

A. True

B. False

Answer: A

8. The ability to track relationships and changes to configuration objects is one of the most important features of the SCM repository.

A. True

B. False

Answer: A

9. Which of the following tasks is not part of software configuration management?

A. change control

B. reporting

C. statistical quality control

D. version control

Answer: C

10. A basic configuration object is a __ created by a software engineer during some phase of the software development process.

A. program data structure

B. hardware driver

C. unit of information

D. all of the above

Answer: C

11. Version control systems establish a change set as part of their primary functionality.

A. True

B. False

Answer: B

12. Change control is not necessary if a development group is making use of an automated project database tool.

A. True

B. False

Answer: B

13. When software configuration management is a formal activity the software configuration audit is conducted by the

A. development team

B. quality assurance group

C. senior managers

D. testing specialists

Answer: B

14. The primary purpose of configuration status reporting is to
- A. allow revision of project schedule and cost estimates by project managers
 - B. evaluate the performance of software developers and organizations
 - C. make sure that change information is communicated to all affected parties
 - D. none of the above
- Answer: C
15. Configuration issues that need to be considered when developing Web and Mobile Apps include:
- A. content
 - B. cost
 - C. people
 - D. politics
 - E. a, b, c
- Answer: E
16. Web and Mobile App configuration objects can be managed in much the same way as conventional software configuration objects except for:
- A. content items
 - B. functional items
 - C. graphic items
 - D. user items
- Answer: A
17. Content management establishes a process by which Web content is rendered on the user's display screen.
- A. True
 - B. False
- Answer: B
18. Change management for Web and Mobile Apps is best handled in agile manner.
- A. True
 - B. False
- Answer: A
19. One reason that version control is difficult for WebApps is that in an uncontrolled environment, you can have multiple authors making changes to the same files from multiple locations without any realizing it.
- A. True
 - B. False
- Answer: A
20. Requiring developers to check Web configuration items in and out and sending affected stakeholders e-mail messages automatically are good ways to deal with configuration auditing and reporting for WebApps.
- A. True
 - B. False
- Answer: A

第31章

1. Effective software project management focuses on
- A. people, performance, payoff, product
 - B. people, product, performance, process
 - C. people, product, process, project
 - D. people, process, payoff, product
- Answer: C

2. Organizations that achieve high levels of maturity in people management have a higher likelihood of implementing effective software engineering processes.

A. True

B. False

Answer: A

3. The first step in project planning is to

A. determine the budget.

B. select a team organizational model.

C. determine the project constraints.

D. establish the objectives and scope.

Answer: D

4. Process framework activities are populated with

A. milestones

B. work products

C. QA points

D. all of the above

Answer: D

5. Project management is less important for modern software development since most projects are successful and completed on time.

A. True

B. False

Answer: B

6. Which of the following is not considered a stakeholder in the software process?

A. customers

B. end-users

C. project managers

D. sales people

Answer: D

7. The best person to hire as a project team leader is the most competent software engineering practitioner available.

A. True

B. False

Answer: B

8. The best project team organizational model to use when tackling extremely complex problems is the

A. closed paradigm

B. open paradigm

C. random paradigm

D. synchronous paradigm

Answer: B

9. Which factors should be considered in choosing the organizational structure for a software team?

A. degree of communication desired

B. predicted size of the resulting program

C. rigidity of the delivery date

D. size of the project budget

E. a, b, c

Answer: E

10. One of the best ways to avoid frustration during the software development process is to

A. give team members more control over process and technical decisions.

B. give team members less control over process and technical decisions.

C. hide bad news from the project team members until things improve.

D. reward programmers based on their productivity.

Answer: A

11. Small agile teams have no place in modern software development.

A. True

B. False

Answer: B

12. Which of these software characteristics is not a factor contributing to project coordination difficulties?

A. interoperability

B. performance

C. scale

D. uncertainty

Answer: B

13. Which of these software characteristics are used to determine the scope of a software project?

A. context, lines of code, function

B. context, function, communication requirements

C. information objectives, function, performance

D. communications requirements, performance, information objectives

Answer: C

14. The major areas of problem decomposition during the project scoping activity are the

A. customer workflow

B. functionality to be delivered

C. process used to deliver functionality

D. software process model

E. b and c

Answer: E

15. Product and process decomposition occurs simultaneously as the project plan evolves.

A. True

B. False

Answer: A

16. When can selected common process framework activities be omitted during process decomposition?

A. when the project is extremely small in size

B. any time the software is mission critical

C. rapid prototyping does not require their use

D. never the activities are invariant

Answer: D

17. How does a software project manager need to act to minimize the risk of software failure?

A. double the project team size

B. request a large budget

C. start on the right foot

D. track progress

E. c and d

Answer: E

18. The W5HH principle contains which of the following questions?

A. Why is the system being developed?

B. What will be done by whom?

C. Where are they organizationally located?

D. How much of each resource is required?

E. a, c d

Answer: E

19. Which of these are critical practices for performance-based project management?

- A. assessing product usability
- B. defect tracking against quality targets

第34章

1. Software projects are inevitably late and there is nothing that can explain why.

- A. True
- B. False

Answer: B

2. It is unethical to undertake a project that you know in advance cannot be completed by the customer's deadline, unless you inform the customer of the risk and establish a project plan that can deliver the needed system incrementally.

- A. True
- B. False

Answer: A

3. Which of the following is not one of the guiding principles of software project scheduling:

- A. compartmentalization
- B. market assessment
- C. time allocation
- D. effort validation

Answer: B

4. Doubling the size of your software project team is guaranteed to cut project completion time in half.

- A. True
- B. False

Answer: B

5. The software equation can be used to show that by extending the project deadline slightly

- A. fewer people are required
- B. you are guaranteed to meet the deadline
- C. more lines of code can be produced
- D. none of the above

Answer: A

6. The 40-20-40 rule suggests that the least of amount of development effort be spent on

- A. estimation and planning
- B. analysis and design
- C. coding
- D. testing

Answer: C

7. A task set is a collection of

- A. engineering work tasks, milestones, deliverables
- B. task assignments, cost estimates, metrics
- C. milestones, deliverables, metrics
- D. responsibilities, milestones, documents

Answer: A

8. The task (activity) network is a useful mechanism for

- A. computing the overall effort estimate
- B. detecting intertask dependencies
- C. determining the critical path
- D. specifying the task set to the customer

E. b and c

Answer: E

9. Tasks that lie on the critical path in a task network may be completed in any order as long as the project is on schedule.

A. True

B. False

Answer: B

10. Two tools for computing critical path and project completion times from activity networks are

A. CPM

B. DRE

C. FP

D. PERT

E. a and d

Answer: E

11. Timeline charts assist project managers in determining what tasks will be conducted at a given point in time.

A. True

B. False

Answer: A

12. The best indicator of progress on a software project is the completion

A. of a defined engineering activity task

B. of a successful budget review meeting on time

C. and successful review of a defined software work product

D. and successful acceptance of project prototype by the customer

Answer: C

13. Since iterative process model work best for object-oriented projects it is impossible to determine whether an increment will be completed on time or not.

A. True

B. False

Answer: B

14. WebApp projects only require the creation of a macro schedule.

A. True

B. False

Answer: B

15. The purpose of earned value analysis is to

A. determine how to compensate developers based on their productivity

B. provide a quantitative means of assessing software project progress

C. provide a qualitative means of assessing software project progress

D. set the price point for a software product based on development effort

Answer: B

16. Earned value analysis is a technique that allows managers to take corrective action before a project crisis develops.

A. True

B. False

Answer: A

第35章

1. Proactive risk management is sometimes described as fire fighting.

A. True

B. False

Answer: B

2. 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

Answer: C

3. Three 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

Answer: B

4. Generic risks require far more attention than product-specific risks.

A. True

B. False

Answer: B

5. A risk item checklist would contain known and predictable risks from which of these categories?

A. product size

B. development environment

C. staff size

D. process definition

E. all of the above

Answer: E

6. Questions that should be asked to assess the overall project risk include:

A. Have top managers formally committed to support the project?

B. Are end-users committed to the project and proposed system being built?

C. Are requirements fully understood by development team and customers?

D. Does the proposed budget have time allocated for marketing?

E. a, b, c

Answer: E

7. 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

Answer: D

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

A. likelihood and cost

B. likelihood and impact

C. likelihood and consequences

D. likelihood and exposure

Answer: C

9. Risk tables are sorted by

A. probability and cost

B. probability and impact

C. probability and consequences

D. probability and exposure

Answer: B

10. Individual team members can make their own estimate for a risk probability and then develop a consensus value.

A. True

B. False

Answer: A

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

A. risk cost

B. risk timing

C. risk scope

D. risk resources

E. b and c

Answer: E

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

A. True

B. False

Answer: B

13. Effective risk management plan needs to address which of these issues?

A. risk avoidance

B. risk monitoring

C. contingency planning

D. all of the above

Answer: D

14. 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

Answer: B

15. Hazard analysis focuses on the identification and assessment of potential hazards that can cause

A. project termination

B. schedule slippage

C. cost overruns

D. an entire system to fail

Answer: D

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

A. True

B. False

Answer: B