

TRUE/FALSE

1. Computer programming is part of the software engineering process, which usually involves newer object-oriented design, combined with more traditional methods of algorithm development.

ANS: T PTS: 1 REF: 32

2. Top-down development starts at the top with one concept or big idea, which is then broken down into several parts.

ANS: T PTS: 1 REF: 32

3. Modular development makes a large project more manageable.

ANS: T PTS: 1 REF: 34

4. Modular development is slow for large projects.

ANS: F PTS: 1 REF: 34

5. Modular development does not necessarily lead to a higher-quality product.

ANS: F PTS: 1 REF: 34

6. Modular development makes it easier to find and correct errors in computer software.

ANS: T PTS: 1 REF: 34

7. Modular development decreases the reusability of your solutions.

ANS: F PTS: 1 REF: 34

8. Over time, modular development allows you to build libraries of software modules for different tasks.

ANS: T PTS: 1 REF: 34

9. All computer programs, from the lowest levels of an operating system that directly control hardware to the most complex user applications, are made up of layers of short programming modules that are constantly reused in different situations.

ANS: T PTS: 1 REF: 34

10. Often, the hardest part of correcting an error in computer software is finding out exactly what is causing the error.

ANS: T PTS: 1 REF: 34

11. Objects created for one project cannot be reused by other projects.

ANS: F PTS: 1 REF: 35

12. Software design begins by writing clear specifications for what the software is supposed to do.

ANS: T PTS: 1 REF: 36

13. Top-down development and modular design are used from the beginning of the design process and help designers break a complex problem into parts that can be more easily understood.

ANS: T PTS: 1 REF: 36

14. Software testers must develop a testing plan that examines new software under all possible circumstances.

ANS: T PTS: 1 REF: 38

15. Once a method has been integration tested and debugged, it can be unit tested.

ANS: F PTS: 1 REF: 38

16. Alice objects have three kinds of methods.

ANS: F PTS: 1 REF: 39

17. Primitive methods are encapsulated methods.

ANS: T PTS: 1 REF: 39

18. In Alice, user-defined methods are encapsulated.

ANS: F PTS: 1 REF: 39

19. You can use primitive methods and user-defined methods to write new methods of your own.

ANS: T PTS: 1 REF: 39

20. Primitive methods may be used to build more complex methods, or they may be run directly.

ANS: T PTS: 1 REF: 41

21. You can run a primitive method directly by right-clicking an object or an object tile in the Object tree, and then selecting the method you want to run from the list that appears.

ANS: T PTS: 1 REF: 41

22. The first step in creating a new world is to set the scene by opening a template, adding necessary objects, and positioning things.

ANS: T PTS: 1 REF: 46

23. Alice's programming language has two primitive instructions to make an object jump up and down.

ANS: F PTS: 1 REF: 49

24. It is much easier to debug software that is written as small modules, each performing an individual task.

ANS: T PTS: 1 REF: 54

25. Once a method stops running, their values are gone unless they were saved elsewhere.

ANS: T PTS: 1 REF: 54

MULTIPLE CHOICE

1. ____ show how separate units are organized to form a single complex entity.

- a. Data charts
- b. Organizational charts
- c. UML diagrams
- d. Flowcharts

ANS: B PTS: 1 REF: 34

2. The process of top-down design leads to ____.

- a. reusable code
- b. tests for correctness
- c. encapsulation
- d. modular development

ANS: D PTS: 1 REF: 34

3. ____ makes programming easier because you need to develop the solution to a problem only once; then you can use that code whenever you need it.

- a. Encapsulation
- b. Concurrency
- c. Reusable code
- d. Bottom-down design

ANS: C PTS: 1 REF: 34

4. Object-oriented programming, by its very nature, encourages the development of ____.

- a. subprograms
- b. assignment statements
- c. reusable code
- d. functions

ANS: C PTS: 1 REF: 35

5. Computer programming and software engineering generally follow a four-step process known as the ____, in which developers design, code, test, and debug software.

- a. software development cycle
- b. modular development cycle
- c. object oriented design
- d. structured design

ANS: A PTS: 1 REF: 36

6. ____ is a specialized branch of software engineering.

- a. Encapsulation
- b. Quantification
- c. Software testing
- d. Tests for correctness

ANS: C PTS: 1 REF: 37

7. ____ measure whether the program meets the original specifications.

- a. Tests for correctness
- b. User-defined tests
- c. Primitive tests
- d. Module tests

ANS: A PTS: 1 REF: 37

8. One of the unique features of Alice is its ____.
- a. use of structured programming
 - b. use of object-oriented programming
 - c. objects
 - d. drag-and-drop interface

ANS: D PTS: 1 REF: 37

9. Logic and control tiles can be found at the bottom of the ____.
- a. Tile
 - b. Editor area
 - c. File area
 - d. Details area

ANS: B PTS: 1 REF: 37

10. A(n) ____ checks to see if the method works as expected all by itself.
- a. module test
 - b. unit test
 - c. user-defined test
 - d. integration test

ANS: B PTS: 1 REF: 38

11. A(n) ____ checks to see if the method works when it is placed into a larger program in combination with other methods.

- a. module test
- b. unit test
- c. user-defined test
- d. integration test

ANS: D PTS: 1 REF: 38

12. A ____ is a short method that simulates the environment in which a newly developed method will be used.

- a. test function
- b. test module
- c. testing shell
- d. test unit

ANS: C PTS: 1 REF: 38

13. ____ are built-in, predefined methods that are part of each Alice object.

- a. Primitive methods
- b. Constructors
- c. Instance methods
- d. User-defined methods

ANS: A PTS: 1 REF: 39

14. ____ provide simple basic behaviors, such as move, turn, and roll.

- a. Primitive methods
- b. Constructors
- c. Instance methods
- d. User-defined methods

ANS: A PTS: 1 REF: 39

15. ____ are written by people who use Alice.

- a. Primitive methods
- b. Constructors
- c. Instance methods
- d. User-defined methods

ANS: D PTS: 1 REF: 39

16. The term ____ implies that the details of a method are hidden from the user; the user cannot see or edit the details of the method.

- a. personification
- b. concurrency
- c. off-camera
- d. encapsulation

ANS: D PTS: 1 REF: 39

17. The term ____ implies that an object cannot be seen on the screen with the camera in its current position.

- a. off-camera
- b. camera point at
- c. camera turn to face
- d. camera roll

ANS: A PTS: 1 REF: 43

18. ____ is the practice of writing compound names without using blank spaces, but capitalizing the first letter of each name that forms the compound name.

- a. Uppercase
- b. CamelCase
- c. Lowercase
- d. mixedCase

ANS: B PTS: 1 REF: 46

19. A(n) ____ gives us information about how the method works.

- a. method header
- b. constructor
- c. parameter
- d. instance variable

ANS: A PTS: 1 REF: 51

20. A ____ is a name for a memory location that temporarily stores a value while a method is running.

- a. constructor
- b. template
- c. module
- d. variable

ANS: D PTS: 1 REF: 54

21. ____ have data types, just like properties, and they are stored in the memory of the computer, just like properties.

- a. Shells
- b. Variables
- c. Modules
- d. Templates

ANS: B PTS: 1 REF: 54

22. ____ are associated with an object, and their values are maintained as long as the object exists.

- a. Functions
- b. Shells
- c. Properties
- d. Global variables

ANS: C PTS: 1 REF: 54

23. ____ are associated with a particular method.

- a. Variables
- b. Shells
- c. Properties
- d. Global variables

ANS: A PTS: 1 REF: 54

24. The values of ____ exist only inside the method.

- a. variables
- b. shells
- c. properties
- d. tiles

ANS: A PTS: 1 REF: 54

25. A(n) ____ is a variable whose value is passed from one method to another - just like a baton is passed from one runner to another in a relay race.

- a. object
- b. module
- c. parameter
- d. testing shell

ANS: C PTS: 1 REF: 54