

Chapter 2: Principles of Programming and Software Engineering

Multiple Choice Questions:

- 1) A(n) _____ is a step-by-step specification of a method to solve a problem within a finite amount of time.
- a) prototype
 - b) module
 - c) solution
 - d) algorithm

Answer: d.

- 2) The first phase of the life cycle of software is the _____ phase.
- a) design
 - b) risk analysis
 - c) specification
 - d) coding

Answer: c.

- 3) Which of the following is an example of a syntax error?
- a) a program encounters an instruction to divide by zero
 - b) an array subscript in a program goes out of range
 - c) the beginning of a while loop is written as “whille” instead of “while”
 - d) an algorithm that calculates the monthly payment of a loan displays incorrect results

Answer: c.

- 4) Self-contained units of code are called _____.
- a) modules
 - b) invariants
 - c) algorithms
 - d) exceptions

Answer: a.

- 5) The data flow among the modules of a program is specified during the _____ phase of the software life cycle.
- a) specification
 - b) design
 - c) coding
 - d) verification

Answer: b.

- 6) The communication mechanisms among modules are called _____.
- a) algorithms
 - b) solutions
 - c) prototypes
 - d) interfaces

Answer: d.

- 7) Modularity describes a program that is organized into _____.
- a) loosely coupled and highly cohesive modules
 - b) tightly coupled and highly cohesive modules
 - c) loosely coupled modules which are not cohesive
 - d) tightly coupled modules which are not cohesive

Answer: a.

- 8) A statement about a particular condition at a certain point in an algorithm is called a(n) _____.
a) base case
b) invariant
c) assertion
d) loop invariant

Answer: c.

- 9) A(n) _____ is a condition that is always true at a particular point in an algorithm.
a) assertion
b) exception
c) constant
d) invariant

Answer: d.

- 10) The syntax errors of a program are removed during the _____ phase of the program's life cycle.
a) verification
b) coding
c) testing
d) refining
e) maintenance

Answer: b.

- 11) User-detected errors are corrected during the _____ phase of the software life cycle.
a) maintenance
b) coding
c) refining
d) testing
e) verification

Answer: a.

- 12) A(n) _____ is a collection of data and a set of operations on the data.
a) method
b) exception
c) abstract data type
d) data structure

Answer: c.

- 13) A(n) _____ is a construct that can be defined to store a collection of data.
a) data structure
b) module
c) abstract data type
d) method

Answer: a.

- 14) Which of the following is an example of a logical error?
a) an algorithm that calculates the monthly payment of a loan displays incorrect results
b) an array subscript in a program goes out of range
c) a program expects a nonnegative number but reads -23
d) the beginning of a while loop is written as "whille" instead of "while"

Answer: a.

15) A set of objects that have the same type is called a(n) _____.

- a) data structure
- b) abstract data type
- c) module
- d) class

Answer: d.

16) An instance of a class is known as a(n) _____.

- a) module
- b) object
- c) abstract data type
- d) data structure

Answer: b.

17) _____ describes the ability of a class to derive properties from a previously defined class.

- a) Encapsulation
- b) Inheritance
- c) Polymorphism
- d) Information hiding

Answer: b.

18) With _____, objects can determine appropriate operations at execution time.

- a) encapsulation
- b) inheritance
- c) polymorphism
- d) information hiding

Answer: c.

19) A top-down design identifies actions by focusing on the _____ in the problem statement.

- a) nouns
- b) verbs
- c) adjectives
- d) adverbs

Answer: b.

20) Object-oriented design identifies objects by focusing on the _____ in the problem statement.

- a) nouns
- b) verbs
- c) adjectives
- d) adverbs

Answer: a.

21) A(n) _____ is a mechanism for error handling.

- a) prototype
- b) class
- c) exception
- d) abstract data type

Answer: c.

22) A module reacts to an exception that another module throws by _____ the exception.

- a) guarding
- b) throwing
- c) accepting
- d) catching

Answer: d.

23) Which of the following is NOT used to find logical errors?

- a) breakpoints
- b) watches
- c) System.out.println statements
- d) exceptions

Answer: d.

24) The specifications of a module describe its _____.

- a) public view
- b) private view
- c) methods
- d) implementation

Answer: a.

25) During the design phase of the software life cycle, the program is divided into _____.

- a) invariants
- b) loops
- c) modules
- d) prototypes

Answer: c.

26) During the _____ phase of the software life cycle, formal methods may be used to prove the correctness of an algorithm.

- a) specification
- b) verification
- c) testing
- d) design

Answer: b.

27) Which of the following is an example of an exception?

- a) a statement does not contain a semicolon at the end
- b) an array subscript in a program goes out of range
- c) an algorithm that converts a temperature in Celsius to Fahrenheit displays the wrong results
- d) a program expects a positive number but reads -23

Answer: b.

28) A prototype program is created during the _____ phase of the software life cycle.

- a) design
- b) specification
- c) coding
- d) testing

Answer: b.

29) Which of the following is NOT a part of the maintenance phase of the software life cycle?

- a) correcting user-detected errors
- b) adding more features to the software
- c) proving the correctness of algorithms
- d) modifying existing features of the software to better suit the users

Answer: c.

30) Which of the following is an example of an error in input data?

- a) a program encounters an instruction to divide by zero
- b) an array subscript in a program goes out of range
- c) an algorithm that converts a temperature in Celsius to Fahrenheit displays the wrong results
- d) a program expects a positive number but reads -23

Answer: d.

True/False Questions:

- 1) A programmer can decrease the time needed to develop a program by starting to write the code of the program before creating a solution design.

Answer: False.

- 2) During the specification phase of the life cycle of software, the programmer should specify what enhancements to the program are likely in the future.

Answer: True.

- 3) The specifications of a module indicate how that module should be implemented.

Answer: False.

- 4) Logical errors of a program are removed during the coding phase of the software life cycle.

Answer: False.

- 5) Coding is a relatively minor phase in the software life cycle.

Answer: True.

- 6) The cost of software includes the consequences of the software not behaving correctly.

Answer: True.

- 7) If two programs perform the same task, then the faster program is always better.

Answer: False.

- 8) Abstraction separates the purpose of a module from its implementation.

Answer: True.

- 9) A module indicates that an error has occurred by catching an exception.

Answer: False.

- 10) Syntax errors are corrected during the maintenance phase of the software life cycle.

Answer: False.

Short Answer Questions:

- 1) What are some of the questions that must be answered during the specification phase of the software life cycle?

Answer: What is the input data? What data is valid and what data is invalid? Who will use the software, and what user interface should be used? What error detection and error messages are desirable? What assumptions are possible? Are there special cases? What is the form of the output? What documentation is necessary? What enhancements to the program are likely in the future?

- 2) What are the nine phases of the software life cycle?

Answer: Specification, design, risk analysis, verification, coding, testing, refining, production, and maintenance.

- 3) List the information provided in a method's precondition.

Answer: The information provided in a method's precondition includes the method's formal parameters, any class named constants that the method uses, and any assumptions that the method makes.

- 4) List the information provided in a method's postcondition.

Answer: The information provided in a method's postcondition includes the method's effect on its parameters – or in the case of a valued method, the value it returns – and any other action that has occurred.

5) What is a loop invariant?

Answer: A loop invariant is a condition that is true before and after each execution of an algorithm's loop.

6) What is the benefit of procedural abstraction?

Answer: Procedural abstraction enables the use of a method, without knowing the particulars of its algorithm, as long as a statement of its purpose and a description of its parameters are present.

7) What is meant by information hiding?

Answer: Information hiding refers to hiding the details of a module from public view and ensuring that no other module can temper with these hidden details.

8) What is meant by the object-oriented approach to modularity?

Answer: The object-oriented approach to modularity produces a collection of objects that have behaviors. An object is a component of a modular solution that combines data and operations on the data.

9) What are the three principles of object-oriented programming?

Answer: The three principles of object-oriented programming are:

- Encapsulation: objects combine data and operations
- Inheritance: classes can inherit properties from other classes
- Polymorphism: objects can determine appropriate operations at execution time

10) What are some of the benefits of modularity?

Answer: Modularity makes it easier to construct large programs. Modularity also isolates errors which makes it easier to debug the program. Also, modular programs are easier to read. Modularity also isolates modifications to the code, and eliminates redundant code.

11) What is an accessor method?

Answer: An accessor method is a method that returns a data field's value.

12) What is a mutator method?

Answer: A mutator method is a method that sets a data field's value.

13) What are some ways of finding logic errors?

Answer: Logic errors can be found by using watches, breakpoints, or temporary System.out.println statements.

14) What is meant by procedural abstraction?

Answer: Procedural abstraction refers to separating the purpose of a method from its implementation.

15) What is a prototype program and what is it used for?

Answer: A prototype program is a program that simulates the behavior of portions of the desired software product. It is used to clarify the specifications of the software.

16) What is an Application Programming Interface (API) ? Why is it useful?

Answer: A collection of preexisting software that is part of the run-time library of your Java installation. It contains predefined classes and methods for commonly used data structures and operations so that you do not have to reinvent the wheel.

17) What are the four parts necessary to proving that a loop is correctly implemented?

Answer: First, the loop invariant must be true initially. Second, executing the loop from one iteration to the next must preserve the truth of the loop invariant. Third, the loop invariant must capture the correctness of the algorithm (i.e. must be materially correct). Finally, the loop must terminate.

18) In a UML diagram, what does it mean if there is an arrow pointing from class A to class B?

Answer: Class A is a subclass of class B.

19) What are some good guidelines for deciding on a name for a method of a class?

Answer: The name should sound like a verb (e.g. `findArea()` instead of just `area()`). Also the first letter should be lowercase, and all other letters should be lowercase except the initial letters of subsequent words.

20) Name two important features of fail-safe programming.

Answer: First, error checking of input. Second, checking of the program's own logic and assumptions before critical operations are performed.