Programming with C++ 1st edition Mcmullen Test Bank

SKU: 9780357421901-TEST-BANK

Name:_____ Class:_____ Date:_____

Module 01: Computational Thinking

1. Which term refers to a series of steps for solving a problem or carrying out a task?
   a. pattern
   b. algorithm
   c. decomposition
   d. level of abstraction

ANSWER: b

FEEDBACK:
a. Incorrect. A pattern is a similarity found in a procedure or task.
b. Correct. Within a good algorithm, the specific series of steps or rules can be followed to successfully complete a requested task.
c. Incorrect. Decomposition divides a complex problem or task into manageable units.
d. Incorrect. The term level of abstraction relates to the amount of detail that is hidden.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.1 Algorithms
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.1 - Define the term “algorithm” as a series of steps for solving a problem or carrying out a task.
KEYWORDS: Bloom’s: Remember
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

2. Programming algorithms specify the underlying logic for the statements in a computer program.
   a. True
   b. False

ANSWER: True

FEEDBACK:
Correct Programming algorithms describe the orderly steps that need to be completed by a computer program to accomplish a task. Without a solid, well-written computer algorithm to follow, the computer program may not function correctly when solving the intended task.
Incorrect Computer programmers use programming algorithms as blueprints to guide them when writing their computer programs in languages such as C++. The programming algorithm is the foundation for the computer program, which will eventually be converted to machine-readable code executed by the computer software to accomplish a desired task. The ability of a computer program to successfully complete a task is often directly related to the soundness of the corresponding programming algorithm.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.1 Algorithms
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.2 - State that algorithms are the underlying logic for computer

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

programs.
KEYWORDS: Bloom's: Remember
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

3. Which statement best describes a computer program?
   a. It is a set of instructions that performs a specific task when executed by a digital device.
   b. It is a set of instructions executed in an application that specify the attributes for a classification of objects.
   c. It is a series of patterns that serve as the blueprints for a programming algorithm.
   d. It is a set of steps that specifies the underlying logic and structure for a classification pattern executed in an application.

ANSWER: a
FEEDBACK:
   a. Correct. A programming algorithm is a set of steps that specifies the underlying logic and structure for the statements in a computer program. The computer program is then translated into a machine-readable language the computer device can understand and executed to successfully complete a specific task.
   b. Incorrect. Instructions are executed by a digital device, not an application.
   c. Incorrect. Programming algorithms serve as the blueprints for computer programs.
   d. Incorrect. A classification pattern is a concept used in program design, not a series of steps that can be executed in an application.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.1 Algorithms
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.3 - Define the term "computer program."
KEYWORDS: Bloom's: Remember
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

4. Which example leads to an algorithm design used for an everyday technology application?
   a. The steps for changing a tire
   b. A recipe for baking brownies
   c. Making a frozen pizza in a toaster oven
   d. The payment process at an online store

ANSWER: d
FEEDBACK:
   a. Incorrect. Changing a tire is not a technology application.
   b. Incorrect. Baking brownies is not a technology application.
   c. Incorrect. Making a frozen pizza in a toaster oven is not a technology application.
   d. Correct. The payment process at an online store requires network and software technology that are used to access the online store and to purchase the desired product. An algorithm for the technological purchase will document how payment and shipping information are provided and stored, as well as how the

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

technological application ultimately completes the sale online.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.1 Algorithms
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.4 - Provide examples of algorithms used in everyday technology applications.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

5. There can be more than one programming algorithm for solving a problem or performing a task.
   a. True
   b. False

ANSWER: True
FEEDBACK:
Correct Multiple and different algorithms may exist to solve a problem or perform a task. A key difference between the algorithms to consider is efficiency, as some algorithms are more efficient than others. Efficiency may be may also be measured in different ways from execution speed to memory utilization.
Incorrect The same task can be completed in different ways using different steps to reach the same result. The steps taken to complete the same task can differ widely in specific action, length, or efficiency. However, the result of each algorithm solves the problem.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.1 Algorithms
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.5 - Confirm that there can be more than one algorithm for a task or problem and that some algorithms may be more efficient than others.
KEYWORDS: Bloom's: Remember
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

6. Why are computer scientists interested in algorithm efficiency?
   a. Because efficient algorithms always produce more secure computer programs.
   b. Because efficient algorithms always produce a programming blueprint that can easily be applied to any programming language.
   c. Because efficient algorithms always produce a computer program that uses the least amount of code.
   d. Because efficient algorithms tend to produce computer programs that in turn operate efficiently, quickly, and reliably.

ANSWER: d
FEEDBACK:
a. Incorrect. The security of a computer program is not defined by or directly related to the efficiency of the algorithm.

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

b. Incorrect. In general, an algorithm should be applicable to various programming languages. However, some programming languages may be able to utilize the algorithm easier and more efficiently than another.

c. Incorrect. Efficient algorithms tend to produce computer programs that operates efficiently, quickly, and reliably. The number of line of code needed to meet these criteria is not always considered.

d. Correct. Computer scientists are interested in designing what they call “good” algorithms that lead to efficiency, speed, and reliability in programs. Good algorithms have characteristics related to five areas. First, input values apply to a set of specified inputs. Second, the algorithm produces one or more outputs. Third, the algorithm terminates after a finite number of steps. Fourth, each step of the algorithm is clear and unambiguous. Fifth, the algorithm successfully produces the correct output.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.1 Algorithms
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.6 - Explain why computer scientists are interested in algorithm efficiency.
KEYWORDS: Bloom’s: Analyze
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

7. What characteristic best applies to an effective algorithm?

a. The algorithm applies to one specific input.

b. The algorithm will terminate only when the user wants it to terminate.

c. The algorithm produces at most one output.

d. The algorithm produces one or more outputs.

ANSWER: d
FEEDBACK:

a. Incorrect. An effective algorithm design should apply to a set of specific inputs.

b. Incorrect. An effective algorithm design should ensure a resulting program terminates after a finite amount of steps.

c. Incorrect. An effective algorithm design should produce one or more outputs.

d. Correct. Computer scientists like algorithms designed with five main qualities that produce a computer program that will operate efficiently, quickly, and reliably. These five qualities focus on characteristics for inputs, outputs, finiteness, precision, and effectiveness.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.1 Algorithms
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.7 - List the characteristics of an effective algorithm.
KEYWORDS: Bloom’s: Remember
DATE CREATED: 1/11/2021 7:32 AM

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

DATE MODIFIED: 1/11/2021 7:32 AM

8. A programmer usually has multiple alternative choices for writing an algorithm for an everyday technology task.
   a. True
   b. False

ANSWER: True

FEEDBACK:
Correct An alternate algorithm for an everyday technology task requires the use of one of the three conventional ways a programmer can come up with an algorithm: using a standard algorithm, performing a task manually and documenting the steps, and, finally, applying computational thinking techniques. By using one of these techniques, the programmer will most likely create a successful algorithm for an everyday technology task.

Incorrect Before coding, programmers consider various algorithms that might apply to a problem. First, the programmer can use a standard algorithm. Second, he or she can perform the task manually, stepping through the process, recording those steps, and then analyzing their effectiveness. Third, a programmer can apply computational thinking techniques. For an everyday technology task, one or more of these algorithm choices may be available and useful.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.1 Algorithms
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.8 - Write a list of steps (an algorithm) for accomplishing a simple, everyday technology application.
KEYWORDS: Bloom's: Apply
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

9. One way a programmer can come up with an algorithm for accomplishing a simple, everyday technology application by using a standard algorithm.
   a. True
   b. False

ANSWER: True

FEEDBACK:
Correct A programmer can come up with an algorithm in three ways: using a standard algorithm, performing a task manually and documenting the steps, and, finally, applying computational thinking techniques. For a simple, everyday technology application any of these three techniques might work.

Incorrect Before coding, programmers consider various algorithms that might apply to a problem. First, the programmer can use a standard algorithm. Second, he or she can perform the task manually, stepping through the process, recording those steps, and then analyzing their effectiveness. Third, a programmer can apply computational thinking techniques. For a simple, everyday technology application, a standard algorithm technique might exist.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.1 Algorithms

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.9 - Write an alternate algorithm for the everyday technology task.
KEYWORDS: Bloom's: Apply
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

10. What concept refers to a set of techniques designed to formulate problems and their solutions so that a programmer can select efficient algorithms?

a. algorithm termination closure
b. standard algorithm processing
c. computational thinking
d. hierarchical modeling

ANSWER: c
FEEDBACK:

a. Incorrect. The concept of algorithm termination closure does not exist.
b. Incorrect. The concept of standard algorithm processing does not exist.
c. Correct. Computational thinking is a set of techniques designed to formulate problems and their solutions. A programmer can use computational thinking techniques such as decomposition, pattern identification, and abstraction to analyze an algorithm's design efficiency. With this analysis, a programmer may then use the most efficient algorithm to serve as a blueprint for an efficient computer program
d. Incorrect. The concept of hierarchical modeling does not exist.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.1 Algorithms
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.1.10 - Select the more efficient of the two algorithms you have written.
KEYWORDS: Bloom's: Apply
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

11. What term refers to the process of dividing an extensive app into smaller parts?

a. abstraction
b. decomposition
c. computational thinking
d. pattern identification

ANSWER: b
FEEDBACK:

a. Incorrect. Abstraction refers to hiding data.
b. Correct. It is usually easier to deal with several smaller components instead of one large, complex application. Decomposition is the process that helps create

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

the smaller, more manageable pieces of the app.

c. Incorrect. Computational thinking is a set of techniques designed to formulate problems and their solutions.

d. Incorrect. Pattern identification refers to the process of finding similarities in procedures and tasks.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.2 Decomposition
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.2.1 - Define the term "decomposition" as a technique for dividing a complex problem or solution into smaller parts.
KEYWORDS: Bloom's: Remember
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

12. Why is decomposition an important tool for computer scientists?

a. It creates algorithm designs with a flat layout, leading to programs that are more efficient.

b. It creates algorithm designs with a high level of abstraction, leading to programs that are more efficient.

c. It creates algorithm designs with a low level of abstraction, leading to programs that are more efficient.

d. It creates algorithm designs with more manageable pieces, leading to programs that are more efficient and maintainable.

ANSWER: d
FEEDBACK:

a. Incorrect. Algorithm designs are not characterized by layout.

b. Incorrect. Algorithm designs are not characterized by abstraction.

c. Incorrect. Algorithm designs are not characterized by abstraction.

d. Correct. The decomposition process creates smaller, more manageable modules. These in turn are easier to work with when compared to the original large, complex application. Efficiencies are gained by working with smaller, manageable pieces.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.2 Decomposition
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.2.2 - Explain why decomposition is an import tool for computer scientists.
KEYWORDS: Bloom's: Analyze
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

13. A programmer uses decomposition when devising an algorithm for a complex problem or task.

a. True

b. False

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

ANSWER: True

FEEDBACK:

Correct Decomposition is a process that creates a set of smaller tasks from a larger application. An algorithm is a series of steps for solving a problem or carrying out a task. An algorithm is created for each of the tasks that results from the decomposition process.

Incorrect The concepts of an algorithm and decomposition are different. An algorithm is a series of steps for solving a problem or carrying out a task. Decomposition, on the other hand, is the process of breaking a larger, complex app into smaller tasks. A programmer can create smaller algorithms for each of smaller modules identified in the decomposition process.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.2 Decomposition
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.2.3 - Differentiate the concepts of algorithms and decomposition.
KEYWORDS: Bloom's: Analyze
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

14. Which phrase describes an example of structural decomposition?
    a. using algorithmic pattern recognition to identify possible variables
    b. dividing a mobile banking app into hierarchical structural units
    c. breaking down a two-factor authentication module into smaller actions, processes, or steps
    d. looking for two-factor authentication logical and physical objects that a computer program will manipulate

ANSWER: b

FEEDBACK:

a. Incorrect. This response is not part of any type of decomposition.
b. Correct. Yes—for structural decomposition, we must consider dividing a mobile banking app into hierarchical structural units.
c. Incorrect. This response refers to functional decomposition.
d. Incorrect. This response refers to object-oriented decomposition.

POINTS: 1
DIFFICULTY: Difficult
REFERENCES: 1.2 Decomposition
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.2.4 - Identify examples of structural decomposition.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

15. Which phase describes an example of functional decomposition?
    a. using algorithmic pattern recognition to identify possible variables
    b. dividing a mobile banking app into structural units

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

c. breaking down a two-factor authentication module into smaller actions, processes, or steps
d. looking for two-factor authentication logical and physical objects that a computer program will manipulate

ANSWER: c
FEEDBACK:
a. Incorrect. This response is not part of any type of decomposition.
b. Incorrect. This response refers to structural decomposition.
c. Correct. Yes—for functional decomposition, a programmer must consider the smaller actions, processes, or steps that will be necessary.
d. Incorrect. This response refers to object-oriented decomposition.
POINTS: 1
DIFFICULTY: Difficult
REFERENCES: 1.2 Decomposition
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.2.5 - Identify examples of functional decomposition.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

16. Which phrase describes an example of object-oriented decomposition?
a. using algorithmic pattern recognition to identify possible variables
b. dividing a mobile banking app into structural units
c. breaking down a two-factor authentication module into smaller actions, processes, or steps
d. looking for two-factor authentication logical and physical objects that a computer program will manipulate

ANSWER: d
FEEDBACK:
a. Incorrect. This response is not part of any type of decomposition.
b. Incorrect. This response refers to structural decomposition.
c. Incorrect. This response refers to functional decomposition.
d. Correct. Yes—for object-oriented decomposition, a programmer must consider the logical and physical objects that a computer program will manipulate in an application or module.
POINTS: 1
DIFFICULTY: Difficult
REFERENCES: 1.2 Decomposition
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.2.6 - Identify examples of object-oriented decomposition.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

17. Suppose a programmer is functionally decomposing a two-factor authentication process. Which statement illustrates an example of a module that requires further decomposition?
a. Prompt for user ID.

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

b. Prompt for password.
c. Generate a one-time PIN.
d. Validate user ID and password match.

ANSWER: d

FEEDBACK:
a. Incorrect. A user ID prompt is a one-time act and needs no further decomposition.
b. Incorrect. A prompt for a password is a one-time act and needs no further decomposition.
c. Incorrect. The generation of a one-time PIN is a one-time act and needs no further decomposition.
d. Correct. With decomposition, the idea is to break down an extensive application or module by dividing the application or module into smaller parts. To validate a user ID and password match, there must be additional decomposition steps to look up the stored user ID and password and then compare the entered data with the stored data.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.2 Decomposition
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.2.7 - Provide examples of decomposition in technology applications.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

18. What statement describes how dependencies and cohesion relate to decomposition?
a. An effective breakdown relates only to maximized dependencies among the various parts.
b. An effective breakdown relates only to minimized cohesion among the various parts.
c. An effective breakdown maximizes dependencies and minimizes cohesion among the various parts.
d. An effective breakdown minimizes dependencies and maximizes cohesion among the various parts.

ANSWER: d

FEEDBACK:
a. Incorrect. An effective breakdown minimizes dependencies and maximizes cohesion among the various parts.
b. Incorrect. An effective breakdown minimizes dependencies and maximizes cohesion among the various parts.
c. Incorrect. An effective breakdown minimizes dependencies and maximizes cohesion among the various parts.
d. Correct. In practice, there may be several viable ways to apply decomposition, but an effective breakdown follows the principles of decomposition, which indicate a programmer should minimize dependencies and maximize cohesion among the various parts.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.2 Decomposition
QUESTION TYPE: Multiple Choice

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.2.8 - Explain how dependencies and cohesion relate to decomposition.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

19. Which term refers to the process of finding similarities in procedures and tasks?
    a. pattern identification
    b. decomposition
    c. computational thinking
    d. abstraction

ANSWER: a
FEEDBACK:
    a. Correct. Pattern identification is one of the three common computational thinking techniques designed to formulate problems and their solutions. The other computational thinking techniques are decomposition and abstraction to devise efficient algorithms.
    b. Incorrect. Decomposition refers to the process of dividing an extensive app into smaller pieces. ing techniques are decomposition and abstraction to devise efficient algorithms.
    c. Incorrect. Computational thinking refers to a set of techniques designed to formulate problems and their solutions.
    d. Incorrect. Abstraction refers to hiding data.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.3 Pattern Identification
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.3.1 - Define the term "pattern identification" as a technique for recognizing similarities or characteristics among the elements of a task or problem.
KEYWORDS: Bloom's: Remember
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

20. Which idea represents an example of a fill-in-the blank pattern?
    a. looking for patterns in algorithm steps that allow the algorithm to work for any input number value, leading to variable use
    b. allowing three password entry attempts before locking a user out of an online account
    c. defining a class called Restaurant with attributes of restaurant name, hours of operation, and menu
    d. using a variable to represent many different numbers

ANSWER: a
FEEDBACK:
    a. Correct. Fill-in-the blank pattern results start with pattern identification. From there, a common numeric value entry can be modified into a variable to allow many different inputs into the algorithm.
    b. Incorrect. This idea refers to a repetitive pattern.

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

c. Incorrect. This idea refers to the concepts of objects and attributes, not patterns.
d. Incorrect. This idea refers to the use of a variable, not a pattern.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.3 Pattern Identification
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.3.2 - Identify examples of fill-in-the blank patterns.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

21. Which idea represents an example of a repetitive pattern?
a. looking for patterns in a template used to specify the attributes for an abstract object
b. allowing three password entry attempts before locking a user out of an online account
c. defining a class called Restaurant with attributes of restaurant name, hours of operation, and menu
d. using a variable to represent many different numbers

ANSWER: b
FEEDBACK:
a. Incorrect. Templates identify fields that contain data and do not contain patterns.
b. Correct. Allowing three password entry attempts before locking a user out of an online account creates multiple, repetitive lines of code. This inefficient code can be consolidated using a repeat statement.
c. Incorrect. Defining a class and its attributes relates to object-oriented decomposition.
d. Incorrect. Using a variable is part of the concept of abstraction.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.3 Pattern Identification
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.3.3 - Identify examples of repetitive patterns.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

22. What phrase identifies an example of single classification of objects from a classification pattern?
a. vehicles represented as cars
b. restaurant name, hours of operation, and menu
c. a user ID, a password, and a mobile number
d. car color, make, model, and VIN number

ANSWER: a

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

FEEDBACK:

a. Correct. Classification patterns also come in handy if you want to design programs based on the interactions among a variety of objects, rather than a step-by-step algorithm. In some programming circles, templates are called classes because they specify the attributes for a classification of objects.
b. Incorrect. These represent attributes for a class, possibly referred to as "Restaurant."
c. Incorrect. These represent attributes for a class, possibly referred to as "Login Credentials."
d. Incorrect. These represent attributes for a class, possibly referred to as "Vehicle."

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.3 Pattern Identification
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.3.4 - Identify examples of classification patterns.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

23. Recognizing that each user of a social media site has a login credential set is an example of pattern identification.

a. True
b. False

ANSWER: True
FEEDBACK:
Correct Everyone who subscribes to a social media site has a set of login credentials.
Incorrect A programmer can often discover classification patterns in the attributes that describe any person or object. For example, the series of attributes that define each user's login credentials have a pattern of similarities. Each user has three attributes: a user ID, a password, and a mobile number. By recognizing this pattern, a programmer can create a template for any user's login credentials.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.3 Pattern Identification
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.3.5 - Provide examples of pattern identification in the real world and technology applications.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

24. Which term refers to a technique that hides details, simplifies complexity, and substitutes a generalization for something specific?

a. computational thinking

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

b. pattern identification
c. abstraction
d. decomposition

ANSWER: c

FEEDBACK:
a. Incorrect. Computational thinking is a set of techniques designed to formulate problems and their solutions.
b. Incorrect. Pattern identification refers to the process of finding similarities in procedures and tasks.
c. Correct. Abstraction is a key element of computational thinking and helps programmers in a multitude of ways.
d. Incorrect. Decomposition refers to the process of dividing an extensive app into smaller pieces.

POINTS: 1
DIFFICULTY: Easy
REFERENCES: 1.4 Abstraction
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.4.1 - Define the term "abstraction" as a technique for generalization and simplifying levels of complexity.
KEYWORDS: Bloom's: Remember
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

25. Which statement describes why abstraction is an important computer science concept?
a. It provides a means to hide implementation details from the user to simplify subsequent development and promotes software reuse.
b. It allows a complex application to be divided into smaller modules.
c. It ensures a computer program will continue to work efficiently as the size of the data set grows.
d. It ensures the correct application classes are assigned to the correct attribute.

ANSWER: a

FEEDBACK:
a. Correct. An algorithm that has inputs or processing steps that are based on changing values or multiple inputs can use abstraction to hide the details of performing the task. Abstraction allows an algorithm to work for multiple inputs.
b. Incorrect. Dividing a complex application into smaller modules refers to decomposition.
c. Incorrect. Abstraction has no direct connection to the size of a data set.
d. Incorrect. Classes are not assigned to attributes; rather, a class is an abstraction that contains a set of attributes.

POINTS: 1
DIFFICULTY: Difficult
REFERENCES: 1.4 Abstraction
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.4.2 - Explain why abstraction is an important computer science concept.

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

26. Which example illustrates how abstraction can help identify variables?
    a. Abstraction can be used in functional design to identify the classes related to variables.
    b. Abstraction can be used in functional design to identify the attributes related to variables.
    c. Abstraction can be used in object-oriented design to identify the objects related to variables.
    d. Abstraction can be used in pattern recognition to identify possible variables.

ANSWER: d
FEEDBACK:
a. Incorrect. Identifying the classes related to variables is used in object-oriented decomposition.
b. Incorrect. Attributes are not related to variables.
c. Incorrect. In object-oriented design, one identifies objects that represent people, places, or things. Variables are symbols that represent values and are not directly relevant to identifying objects.
d. Correct. When pattern recognition is applied to the Amaze-Your-Friends math trick, the resulting blank line(s) within an algorithm can represent an abstraction that represents a number in a sequence. This is because by definition, an abstraction hides details, simplifies complexity, substitutes a generalization for something specific, and allows an algorithm to work for multiple inputs.

POINTS: 1
DIFFICULTY: Difficult
REFERENCES: 1.4 Abstraction
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.4.3 - Provide an example illustrating how abstraction can help identify variables.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

27. Logging on to an online social media website is an example of using an abstracted technology application.
    a. True
    b. False

ANSWER: True
FEEDBACK:
Correct Each user who needs to log in to a social media website provides the same general information that is unique to them. What the application does with the entered user data and how it is processed are unknown to the user. The user simply wants to obtain access to the application.
Incorrect An abstracted technology application processes hidden steps that apply to every user when a user interacts with the application. The user cannot see the details within the online social media website and instead only knows their unique user identification information required to gain access to the website.

POINTS: 1
DIFFICULTY: Medium

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

Module 01: Computational Thinking

REFERENCES: 1.4 Abstraction
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.4.4 - Provide examples of technology applications that have abstracted or hidden details.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

28. Which example illustrates how a class can be used as an abstraction?
    a. A programmer creates a class to represent the temperature required to boil water in degrees Celsius.
    b. A programmer creates a class to represent the temperature required to freeze water in degrees Celsius.
    c. A programmer creates a class to handle a recognized pattern of repetitive social media login credential items.
    d. A programmer creates a class to handle the value of the mathematical construct pi.

ANSWER: c
FEEDBACK:
    a. Incorrect. A class is an abstraction that contains a set of attributes. The temperature required to boil water is a single value.
    b. Incorrect. A class is an abstraction that contains a set of attributes. The temperature required to freeze water is a single value.
    c. Correct. To utilize a class as an abstraction of a set of objects, a set of objects has to be identified. Logging into a social media site requires several unique objects, making it a good candidate for abstraction.
    d. Incorrect. A class is an abstraction that contains a set of attributes. The mathematical construct pi is a single value.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.4 Abstraction
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.4.5 - Provide an example illustrating the use of a class as an abstraction of a set of objects.
KEYWORDS: Bloom's: Apply
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

29. Which statement describes how the black box concept is an implementation of abstraction?
    a. It produces an output by using unique templates that utilize abstract variables.
    b. It produces an output by creating a classification pattern.
    c. It produces an output while hiding the inner workings of a process.
    d. It produces an output by implementing pattern identification.

ANSWER: c
FEEDBACK:
    a. Incorrect. The concept of using unique templates that utilize abstract variables is not part of a black box.
    b. Incorrect. The concept of creating classification patterns is not part of a black box.

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank

c. Correct. In a black box, the user or program provides known input and receives known output. Exactly what the program or task does with the input is not visible or known by the user. It is hidden.

d. Incorrect. The concept of implementing pattern identification is not part of a black box.

POINTS: 1
DIFFICULTY: Difficult
REFERENCES: 1.4 Abstraction
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.4.6 - Explain how the black box concept is an implementation of abstraction.
KEYWORDS: Bloom's: Understand
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

30. Neglecting abstraction can produce programs that are too detailed to work with a wide variety of data.

a. True

b. False

ANSWER: False
FEEDBACK:
Correct The amount of detail hidden or exposed within a program has to be carefully judged and monitored. The key to abstraction is that it is largely to the benefit of the programmer(s), not the program. Too much abstraction in the component parts can make an implementation inflexible but exposing too much can make it confusing and more vulnerable to the introduction of errors. It is a balance, perhaps that is what is meant by "experience is the best teacher.

Incorrect The level of abstraction relates to the amount of detail that is hidden. Abstracting out too much detail can make a program too generalized. If a program does not hide enough details, or neglects abstraction, the required input will have to use very specific data and a wide variety of data cannot be accepted.

POINTS: 1
DIFFICULTY: Medium
REFERENCES: 1.4 Abstraction
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: PC++.PARS.21.1.4.7 - Identify appropriate levels of abstraction.
KEYWORDS: Bloom's: Apply
DATE CREATED: 1/11/2021 7:32 AM
DATE MODIFIED: 1/11/2021 7:32 AM

Copyright Cengage Learning. Powered by Cognero.

Programming with C++ 1st edition Mcmullen Test Bank