Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

SKU: 9781337102100-TEST-BANK

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Name:_____ Class:_____ Date:_____

Chapter 02: Using Data

1. The equal sign (=) is the C# assignment operator.
   a. True
   b. False

ANSWER: True

FEEDBACK:
- *Correct* Correct. The equal sign (=) is the assignment operator. Any value to the right of the assignment operator is assigned to the identifier to the left.
- *Incorrect* Incorrect. The equal sign (=) is the assignment operator. Any value to the right of the assignment operator is assigned to the identifier to the left.

POINTS: 1
REFERENCES: Declaring Variables
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.01 - Declare variables
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

2. You can concatenate two strings with the plus (+) sign.
   a. True
   b. False

ANSWER: True

FEEDBACK:
- *Correct* Correct. When you concatenate a string with another value, you join the values with a plus sign.
- *Incorrect* Incorrect. When you concatenate a string with another value, you join the values with a plus sign.

POINTS: 1
REFERENCES: Displaying Variable Values
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.02 - Display variable values
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

3. Operator precedence rules ensure that addition and subtraction always take place prior to multiplication, division, or remainder in an expression.
   a. True
   b. False

ANSWER: False

FEEDBACK:
- *Correct* Correct. When you combine mathematical operations in a single statement, you must understand operator precedence, or the rules that determine the order in which parts of a mathematical expression are evaluated. Multiplication, division, and remainder always take place prior to addition or subtraction in an expression.
- *Incorrect* Incorrect. When you combine mathematical operations in a single statement, you must understand operator precedence, or the rules that determine the order in which parts of a mathematical expression are evaluated. Multiplication, division,

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

and remainder always take place prior to addition or subtraction in an expression.

POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

4. `someValue++` is an example of the use of the prefix increment operator.

a. True
b. False

ANSWER: False
FEEDBACK:
Correct Correct. To use a prefix increment operator, you type two plus signs before the variable name.
Incorrect Incorrect. To use a prefix increment operator, you type two plus signs before the variable name.

POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

5. An escape sequence is used to represent special characters such as a carriage return or a tab, and always begins with a backslash.

a. True
b. False

ANSWER: True
FEEDBACK:
Correct Correct. You can store any character—including nonprinting characters such as a backspace or a tab—in a `char` variable. To store these characters, you use two symbols in an escape sequence, which always begins with a backslash. The pair of symbols represents a single character.
Incorrect Incorrect. You can store any character—including nonprinting characters such as a backspace or a tab—in a `char` variable. To store these characters, you use two symbols in an escape sequence, which always begins with a backslash. The pair of symbols represents a single character.

POINTS: 1
REFERENCES: Using the char Data Type
QUESTION TYPE: True / False
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.08 - Use the char data type

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

6. What is the term for a named location in computer memory that can hold different values at different points in time?

a. variable
b. constant
c. literal constant
d. alias

ANSWER: a
FEEDBACK:

a. Correct. A variable is a named location in computer memory that can hold different values at different points in time.
b. Incorrect. A data item is constant when it cannot be changed after a program is compiled—in other words, when it cannot vary.
c. Incorrect. A literal constant is a value that is taken literally at each use.
d. Incorrect. An alias is another name for something.

POINTS: 1
REFERENCES: Declaring Variables
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.01 - Declare variables
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

7. An assignment that is made when a variable is declared is known as what?

a. A definition of the variable
b. An allocation of the variable
c. An assignment to the variable
d. An initialization of the variable

ANSWER: d
FEEDBACK:

a. Incorrect. By definition, a variable is a named location in computer memory that can hold different values at different points in time.
b. Incorrect. An allocation is the distribution of something; the term does not have a conventional meaning with variables.
c. Incorrect. The assignment after the declaration of a variable is simply an assignment. Thus, an assignment does not include the data type, as it is only used in a declaration.
d. Correct. An assignment made when a variable is declared is an initialization.

POINTS: 1
REFERENCES: Declaring Variables
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.01 - Declare variables

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

8. What is the name for a series of characters used to control the appearance of text output?
   a. A declared namespace
   b. A set placeholder
   c. A defined format string
   d. Carefully placed whitespace

ANSWER: c
FEEDBACK:
   a. Incorrect. A namespace is a construct that acts like a container to provide a way to group similar classes.
   b. Incorrect. A placeholder holds a position for a variable value within a string. It consists of a pair of curly braces containing a number that indicates the desired variable's position in a list that follows the string.
   c. Correct. A format string is a string of characters that controls the appearance of output.
   d. Incorrect. Whitespace is any combination of spaces, tabs, and carriage returns (blank lines). You use whitespace to organize your program code and make it easier to read.

POINTS: 1
REFERENCES: Displaying Variable Values
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.02 - Display variable values
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

9. What uses curly braces containing a number that indicates the desired position for a variable in a list that follows a string used for formatting purposes?
   a. format string
   b. whitespace
   c. namespace
   d. placeholder

ANSWER: d
FEEDBACK:
   a. Incorrect. A format string is a string of characters that controls the appearance of output. It optionally contains fixed text and contains one or more format items or placeholders for variable values.
   b. Incorrect. Whitespace is any combination of spaces, tabs, and carriage returns (blank lines). You use whitespace to organize your program code and make it easier to read.
   c. Incorrect. A namespace is a construct that acts like a container to provide a way to group similar classes.
   d. Correct. A placeholder holds a position for a variable value within a string. It consists of a pair of curly braces containing a number that indicates the desired variable's position in a list that follows the string.

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

POINTS: 1
REFERENCES: Displaying Variable Values
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.02 - Display variable values
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

10. What type of number can be described as having varying numbers of significant digits, which contains decimal positions?

a. integer
b. whole
c. floating-point
d. `int`

ANSWER: c
FEEDBACK:

a. Incorrect. Integers are whole numbers without any fractional parts.
b. Incorrect. A whole number is an integer without any fractional parts.
c. Correct. A floating-point number is one that contains decimal positions. Floating-point numbers are described as having varying numbers of significant digits.
d. Incorrect. The data type `int` holds a whole number without any fractional parts.

POINTS: 1
REFERENCES: Using Floating-Point Data Types
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.04 - Use floating-point data types
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

11. In the C# statement `moneyString = someMoney.ToString("F3")`, what is `"F3"` an example of?

a. literal constant
b. standard numeric format string
c. named constant
d. intrinsic type

ANSWER: b
FEEDBACK:

a. Incorrect. A literal constant is a value that is taken literally at each use.
b. Correct. Standard numeric format strings are strings of characters expressed within double quotation marks that indicate a format for output.
c. Incorrect. A named constant is an identifier whose value must be assigned upon declaration and whose contents cannot change.
d. Incorrect. Intrinsic types of data are basic types. C# provides 15 intrinsic types. The most commonly used are `int`, `double`, `decimal`, `char`, `string`, and `bool`.

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

POINTS: 1
REFERENCES: Using Floating-Point Data Types
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.04 - Use floating-point data types
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

12. What type of arithmetic operator has one argument to the left and another argument to the right of the operator?

a. binary
b. complex
c. composite
d. unary

ANSWER: a
FEEDBACK:

a. Correct. Binary operators use two operands—one value to the left of the operator and another value to the right of it.
b. Incorrect. There are no complex operators.
c. Incorrect. A composite operator is not an arithmetic operator.
d. Incorrect. Unary operators are operators used with one operand.

POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

13. In C#, += adds the operand on the right to the operand on the left and assigns it to the operand on the left in one step. What is the name for this operator?

a. prefix increment
b. postfix increment
c. add and assign
d. increment and assign

ANSWER: c
FEEDBACK:

a. Incorrect. The prefix increment operator (++ before a variable) increases the variable's value by 1 and then evaluates it.
b. Incorrect. The postfix increment operator (++ after a variable) evaluates a variable and then adds 1 to it.
c. Correct. The += operator is the add and assign operator. It adds the operand on the right to the operand on the left and assigns the result to the operand on the left in one step.
d. Incorrect. Incrementing a variable increases the value by 1.

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

14. What type of operator can be used to reduce a variable's value by 1 either at time of evaluation, or after it is evaluated, depending on placement?

a. increment operator
b. sub and assign operator
c. add and assign operator
d. decrement operator

ANSWER: d
FEEDBACK:

a. Incorrect. An increment operator increases a variable's value by 1.
b. Incorrect. The −= operator is the subtract and assign operator. It subtracts the operand on the right from the operand on the left and assigns the result to the operand on the left in one step.
c. Incorrect. The += operator is the add and assign operator. It adds the operand on the right to the operand on the left and assigns the result to the operand on the left in one step.
d. Correct. A prefix or postfix decrement operator (– –) reduces a variable's value by 1.

POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

15. How is a Boolean variable declared in a program?

a. A `char` type variable must be declared, optionally followed by a `bool` value.
b. A `bool` type variable must be declared, optionally followed by a `bool` value.
c. An `int` type variable must be declared, optionally followed by a `bool` value.
d. A `boolean` type variable must be declared, optionally followed by a `bool` value.

ANSWER: b
FEEDBACK:

a. Incorrect. A `char` type cannot hold a `bool` value because a `char` can only hold single characters and a `boolean` value is either `true` or `false`.
b. Correct. A `bool` type holds `bool` values.
c. Incorrect. An `int` type cannot hold a `bool` value because an int can only hold

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

integers and a `boolean` value is either `true` or `false`.

d. Incorrect. The data type used to declare a Boolean variable is a `bool` type, not a `boolean` type.

POINTS: 1
REFERENCES: Using the bool Data Type
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.06 - Use the bool data type
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

16. What type of expression has a Boolean value as a result?

a. decrement
b. comparison
c. assignment
d. unifying

ANSWER: b
FEEDBACK:

a. Incorrect. A decrement expression reduces a variable's value by 1. Thus, a decrement cannot return a `Boolean` value of `true` or `false`.
b. Correct. A comparison operator compares two items. An expression containing a comparison operator has a Boolean value.
c. Incorrect. An assignment expression assigns a value to a variable.
d. Incorrect. A unifying type is the type chosen for an arithmetic result when operands are of dissimilar types.

POINTS: 1
REFERENCES: Using the bool Data Type
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.06 - Use the bool data type
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

17. When using arithmetic operations with operands of dissimilar types, an automatic conversion of nonconforming operands to a unifying type occurs. What is this conversion process called?

a. implicit cast
b. explicit cast
c. manual cast
d. level cast

ANSWER: a
FEEDBACK:

a. Correct. An implicit cast is the automatic transformation that occurs when a value is assigned to a type with higher precedence.
b. Incorrect. An explicit cast purposefully assigns a value to a different data type. It

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

involves placing the desired result type in parentheses followed by the variable or constant to be cast.

c. Incorrect. This is manual, not automatic.

d. Incorrect. *Level cast* is not a term used by C# programmers.

POINTS: 1
REFERENCES: Understanding Numeric Type Conversion
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.07 - Describe numeric type conversion
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

18. You are performing an arithmetic operation and have placed the desired result type in parentheses followed by a variable to be cast with the result. What is this an example of?

a. implicit casting
b. automatic casting
c. intrinsic casting
d. explicit casting

ANSWER: d
FEEDBACK:

a. Incorrect. An implicit cast is the automatic transformation that occurs when a value is assigned to a type with higher precedence.

b. Incorrect. Automatic casting is known as implicit casting, which is the automatic transformation that occurs when a value is assigned to a type with higher precedence.

c. Incorrect. Intrinsic is a basic type of data. The ones most commonly used are `int`, `double`, `decimal`, `char`, `string`, and `bool`.

d. Correct. An explicit cast purposefully assigns a value to a different data type. It involves placing the desired result type in parentheses followed by the variable or constant to be cast.

POINTS: 1
REFERENCES: Understanding Numeric Type Conversion
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.07 - Describe numeric type conversion
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

19. What data type can be used to hold any single character, including numbers and non-printing characters?

a. `char`
b. `string`
c. `bool`
d. `int`

ANSWER: a

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

FEEDBACK:

a. Correct. The `char` data type can hold any single character, such as `A` or `2`.
b. Incorrect. A `string` data type holds a series of characters. For example, `"Hello"` is a `string` data type.
c. Incorrect. A `bool` data type can hold only one of two values, `true` or `false`.
d. Incorrect. An `int`data type can hold whole numbers. For example, the number 33 is an `int` data type.

POINTS: 1
REFERENCES: Using the char Data Type
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.08 - Use the char data type
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

20. What data type can be used to hold a series of characters?

a. `int`
b. `string`
c. `sbyte`
d. `char`

ANSWER: b
FEEDBACK:

a. Incorrect. An `int` data type can hold whole numbers. For example, the number 15 is an `int` data type.
b. Correct. A `string` data type holds a series of characters. For example, `"Hello"` is a string data type.
c. Incorrect. An `sbyte` is an integral data type that can hold a signed numeric value from –128 through 127.
d. Incorrect. The `char` data type can hold any single character, such as `A` or `2`.

POINTS: 1
REFERENCES: Using the string Data Type
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.09 - Use the string data type
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

21. What keyword can be used to create a named identifier for a memory location whose contents cannot change?

a. `final`
b. `static`
c. `const`
d. `fix`

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

ANSWER: c

FEEDBACK:

a. Incorrect. The keyword `final` is not an identifier.

b. Incorrect. The keyword `static` indicates that a method will be executed through a class, not by a variety of objects.

c. Correct. You create a named constant by adding the keyword `const` before the data type in a declaration.

d. Incorrect. The word `fix` is not a C# keyword.

POINTS: 1
REFERENCES: Defining Named Constants
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.10 - Define named constants and enumerations
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

22. What type of program specifically allows and accepts user input?

a. interactive program
b. static program
c. constant program
d. procedural program

ANSWER: a

FEEDBACK:

a. Correct. A program that allows user input is an interactive program.

b. Incorrect. The keyword `static` indicates a method is used with a class and not an object.

c. Incorrect.The term *constant* is not used to describe programs. A constant is a named memory location whose value never varies.

d. Incorrect. A procedural program is created by writing a series of steps or operations to manipulate values.

POINTS: 1
REFERENCES: Accepting Console Input
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.11 - Accept console input
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

23. What method can be used to accept user input from the keyboard on the console?

a. `Next()`
b. `NextLine()`
c. `ReadLine()`
d. `Input()`

ANSWER: c

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

FEEDBACK:
a. Incorrect. `Next()` is a method that returns a non-negative random integer.
b. Incorrect. `NextLine()` is not a built-in C# method.
c. Correct. You can use the `ReadLine()` method to create an interactive program that accepts user input from the keyboard.
d. Incorrect. `Input()` is not a built-in C# method.

POINTS: 1
REFERENCES: Accepting Console Input
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.11 - Accept console input
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

24. When running a program, an instruction to the user to enter data for the program to process is known by what term?
a. line
b. prompt
c. cue
d. hint

ANSWER: b
FEEDBACK:
a. Incorrect. A line is a line of code.
b. Correct. A prompt is an instruction to the user to enter data.
c. Incorrect. An instruction to the user to enter data is not referred to as a cue.
d. Incorrect. A hint is a suggestion and does not instruct a user to enter data.

POINTS: 1
REFERENCES: Accepting Console Input
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.11 - Accept console input
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

25. What method from the `Convert` class can be used to convert a specified value to an 8-bit unsigned integer?
a. `ToDecimal()`
b. `ToSingle()`
c. `ToInt16()`
d. `ToByte()`

ANSWER: d
FEEDBACK:
a. Incorrect. `ToDecimal()` converts a specified value to a decimal number.

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

b. Incorrect. `ToSingle()` converts a specified value to a single-precision floating-point number.
c. Incorrect. `ToInt16()` converts a specified value to a 16-bit signed integer (a `short`).
d. Correct. `ToByte()` converts a specified value to an 8-bit unsigned integer.

POINTS: 1
REFERENCES: Accepting Console Input
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.11 - Accept console input
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

26. What method can be used as an alternative to the `Convert` class methods to change a string into a number?
a. `Cast()`
b. `Parse()`
c. `Assign()`
d. `Write()`

ANSWER: b
FEEDBACK:
a. Incorrect. Cast represents a conversion, but `Cast()` is not a built-in method.
b. Correct. The `Parse()` method converts a string to a number.
c. Incorrect. `Assign()` is not a built-in method.
d. Incorrect. The `Write()` method is used to display variable values.

POINTS: 1
REFERENCES: Accepting Console Input
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.11 - Accept console input
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

27. What happens when "string1" is added to "string2" using the + sign in the C# programming language?
a. The strings are concatenated together.
b. The strings are converted to integer equivalents and then added together.
c. The result is returned as a Boolean value.
d. A type error occurs, and the program is halted.

ANSWER: a
FEEDBACK:
a. Correct. When you concatenate a string with another value, you join the values with a plus sign.
b. Incorrect. The plus sign does not indicate that the values should be added together. The plus sign between strings concatenates them.

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

c. Incorrect. The plus sign does not indicate a comparison between the two strings, and so a Boolean value is not returned.
d. Incorrect. The plus sign will not result in an error because it is used to combine two values.

POINTS: 1
REFERENCES: Displaying Variable Values
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.02 - Display variable values
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

28. What is NOT part of a variable declaration in the C# programming language?
a. The data type the variable will store.
b. The variable's name.
c. An optional assignment operator and assigned value.
d. The variable's associated parent class.

ANSWER: d
FEEDBACK:
a. Incorrect. When declaring a variable, the programmer must determine the data type that the variable will store.
b. Incorrect. When declaring a variable, the programmer must name the variable, which is its identifier.
c. Incorrect. When declaring a variable, the programmer can include the assignment operator (=) and an assigned value (but is not required to do so).
d. Correct. When declaring a variable, a parent class is not required.

POINTS: 1
REFERENCES: Declaring Variables
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.01 - Declare variables
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

29. What operator is used for exponentials in the C# programming language?
a. ^
b. %
c. *
d. None of the above.

ANSWER: d
FEEDBACK:
a. Incorrect. The operator ^ is a logical XOR operator.
b. Incorrect. The operator % determines the remainder in integer division. For example, the result of 45%2 is 1.

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

c. Incorrect. The operator `*` represent multiplication. For example, the result of 3*2 is 6.

d. Correct. C# does not have an exponential operator. Instead, you can use the built-in method `Math.Pow()`.

POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

30. What statement regarding operator precedence is accurate?

a. Multiplication, division, and addition always take place prior to the calculation of a remainder.

b. Operations that are placed in parentheses are evaluated first.

c. Operations are always evaluated left to right, regardless of the expressions.

d. Addition and subtraction operations are evaluated first.

ANSWER: b
FEEDBACK:

a. Incorrect. Multiplication, division, and remainder always take place prior to addition or subtraction in an expression.

b. Correct. The order of operations for arithmetic operators indicates that parentheses should be evaluate first.

c. Incorrect. First, evaluate expressions within parentheses. Second, evaluate multiplication, division, and remainder from left to right. Finally, evaluate addition and subtraction from left to right.

d. Incorrect. Addition and subtraction should be evaluated last unless they are enclosed in parentheses.

POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: Multiple Choice
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

Match each item with a statement below.

a. data type
b. integral data type
c. precision specifier
d. culture
e. operator precedence
f. Boolean variable
g. comparison operator

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

h. Unicode

i. named constant

j. unary operator

REFERENCES: Glossary
QUESTION TYPE: Matching
HAS VARIABLES: False
LEARNING OBJECTIVES: MVC#.FARR.18.02.01 - Declare variables
MVC#.FARR.18.02.03 - Use integral data types
MVC#.FARR.18.02.04 - Use floating-point data types
MVC#.FARR.18.02.05 - Use arithmetic operators
MVC#.FARR.18.02.06 - Use the bool data type
MVC#.FARR.18.02.08 - Use the char data type
MVC#.FARR.18.02.09 - Use the string data type
MVC#.FARR.18.02.10 - Define named constants and enumerations
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

31. Describes the format and size (amount of memory occupied) of a data item and defines a set of relevant operations
ANSWER: a
POINTS: 1

32. Used to store a whole number
ANSWER: b
POINTS: 1

33. Controls the number of significant digits or zeros to the right of a decimal point
ANSWER: c
POINTS: 1

34. A set of rules that determines how to format values such as money and dates that can be represented differently based on locale and tradition
ANSWER: d
POINTS: 1

35. Rules that determine the order in which parts of a mathematical expression are evaluated
ANSWER: e
POINTS: 1

36. Can hold only one of two values: true or false
ANSWER: f
POINTS: 1

37. The use of <= or similar operators, for example.
ANSWER: g

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

POINTS: 1

38. A 16-bit coding scheme for characters
ANSWER: h
POINTS: 1

39. An identifier whose contents cannot change
ANSWER: i
POINTS: 1

40. The prefix and postfix increment operators are both examples of this type of operator.
ANSWER: j
POINTS: 1

41. What is a variable declaration, what is its purpose, and what does the declaration include?
ANSWER: A variable declaration is the statement that names a variable and reserves storage for it. The declaration includes:
* The data type that the variable will store
* The variable name (its identifier)
* An optional assignment operator and assigned value when you want a variable to contain an initial value
* An ending semicolon
POINTS: 1
REFERENCES: Declaring Variables
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.01 - Declare variables
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

42. What are the nine integral data types in C# and how do you decide which is the most appropriate to use in a particular situation?
ANSWER: The nine types are `byte`, `sbyte`, `short`, `ushort`, `int`, `uint`, `long`, `ulong`, and `char`. The first eight always represent whole numbers, and the ninth type, `char`, is used for characters like 'A' or 'a'.
The most basic of the integral types is `int`. An `int` uses four bytes of memory and can hold any whole number value ranging from 2,147,483,647 down to -2,147,483,648. Many programmers use `int` for most whole numbers. If you want to save memory and know you need only a small value, you can use one of the shorter integer types-`byte`, `sbyte` (which stands for signed byte), `short` (short `int`), or `ushort` (unsigned `short int`). For example, a payroll program might contain a variable named `numberOfDependents` that is declared as type `byte`, because `numberOfDependents` will never need to hold a negative value or a value

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

exceeding 255; for that reason, you can allocate just one byte of storage to hold the value. If you use a type that is too large, you waste storage. If you use a type that is too small, your program won't compile.

POINTS: 1
REFERENCES: Using the Integral Data Types
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.03 - Use integral data types
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

43. What are the three floating-point data types in C#? Briefly describe the characteristics of each type.

ANSWER: A floating-point number is one that contains decimal positions. C# supports three floating-point data types: `float`, `double`, and `decimal`. A `float` data type can hold as many as seven significant digits of accuracy. A `double` data type can hold 15 or 16 significant digits of accuracy. Compared to `float`s and `double`s, the `decimal` type has a greater precision and a smaller range, which makes it suitable for financial and monetary calculations. For example, a `decimal` given the value 123456789.987654321 will appear as 123456789.987654321 (notice that it is accurate to the rightmost digit). A `decimal` cannot hold as large a value as a `double` can, but it can be accurate to more decimal places.

POINTS: 1
REFERENCES: Using Floating-Point Data Types
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.04 - Use floating-point data types
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

44. Explain how to use the C# shortcut arithmetic operators `-=`, `*=`, and `/=`.

ANSWER: Each of these operators is used to perform an operation and assign the result in one step. For example:

```
balanceDue -= payment
```

subtracts a payment from `balanceDue` and assigns the result to `balanceDue`.

```
rate *= 100
```

multiplies rate by 100 and assigns the result to `rate`. For example, this could be used to convert a fractional value stored in rate, such as 0.27, to a whole number, such as 27.

```
payment /= 12
```

divides `payment` by 12 and assigns the result to payment. This could be used to change a payment value from an annual amount to a monthly amount due.

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

Chapter 02: Using Data

POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

45. What are the differences between the prefix increment operator and postfix increment operator?
ANSWER: When you only want to increase a variable's value by 1, there is no apparent difference between using the prefix and postfix increment operators. However, these operators function differently. When you use the prefix ++, the result is calculated and stored, and then the variable is used. In contrast, when you use the postfix ++, the variable is used, and then the result is calculated and stored.
POINTS: 1
REFERENCES: Using Arithmetic Operators
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.05 - Use arithmetic operators
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

46. Explain why implicit numeric conversions are necessary and how C# chooses operands to convert.
ANSWER: When you perform arithmetic with variables of the same type, the result of the arithmetic retains that type. For example, when you divide two `int`s, the result is an `int`. Often, however, you need to perform mathematical operations on different types. For example, you might need to multiply an `int` by a `double`. In this case, C# chooses a unifying type for the result and implicitly converts nonconforming operands to the unifying type, which is the type with the higher type precedence. In the case of multiplying an `int` by a `double`, the result will be a `double`.
POINTS: 1
REFERENCES: Understanding Numeric Type Conversion
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.07 - Describe numeric type conversion
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

47. How can you compare strings using the `Compare()` method?
ANSWER: The `Compare()` method requires two `string` arguments and returns an integer

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

result. When it returns 0, the two strings are equivalent; when it returns a positive number, the first string is greater than the second; and when it returns a negative number, the first string is less than the second. A string is considered equal to, greater than, or less than another string lexically, which in the case of letter values means alphabetically. That is, when you compare two strings, you compare each character in turn from left to right. If each Unicode value is the same, then the strings are equivalent. If any corresponding character values are different, the string that has the greater Unicode value earlier in the string is considered greater.

POINTS: 1
REFERENCES: Using the string Data Type
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.09 - Use the string data type
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

48. How is a named constant different from a variable? How would you create a named constant (and show an example)?

ANSWER: By definition, a variable's value can vary, or change. Sometimes you want to create a named constant (often simply called a *constant*), an identifier whose contents cannot change. You create a named constant similar to the way you create a named variable, but by using the keyword `const`. Although there is no requirement to do so, programmers usually name constants using all uppercase letters, inserting underscores for readability. This convention makes constant names stand out so that the reader is less likely to confuse them with changeable variable names. For example, the following declares a constant named `TAX_RATE` that is assigned a value of 0.06:

```
const double TAX_RATE = 0.06;
```

POINTS: 1
REFERENCES: Defining Named Constants
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.10 - Define named constants and enumerations
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

49. What is an enumeration and what are the advantages of creating an enumeration type? Show an example of an enumeration definition.

ANSWER: An enumeration is a set of constants represented by identifiers. An example of an enumeration definition is the following:

```
enum DayOfWeek
{
```


Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank

```
SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY,
SATURDAY
}
```

Creating an enumeration type provides several advantages. For example, the `DayOfWeek` type above improves a program as follows:
1. Only the seven allowed values can be assigned to a `DayOfWeek` object.
2. Using an enumeration type makes a program type-safe (for example, you couldn't use an integer method such as addition on an object of type `DayOfWeek`).
3. The `enum` constants provide clearer self-documentation (it is more apparent what you mean when you assign a variable the value `WEDNESDAY` rather than the number 4, for example).

POINTS: 1
REFERENCES: Defining Named Constants
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.10 - Define named constants and enumerations
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

50. What is the purpose of the `ReadLine()` method? How would you use it with a variable of type `string`? How would you use it with a variable of type `double`?

ANSWER: You use the `ReadLine()` method to accept user input from the keyboard. This method accepts all of the characters entered by a user until the user presses Enter. The characters can be assigned directly to a string. For example, the following statement accepts a user's input and stores it in the variable `myString`:

```
myString = ReadLine();
```

If you want to use the data as a `string`-for example, if the input is a word or a name-then you simply use the variable to which you assigned the value. If you want to use the data as a number, then you must use a conversion method to convert the input string to the proper type, as in:

```
itemPrice = Convert.ToDouble(myString);
```

POINTS: 1
REFERENCES: Accepting Console Input
QUESTION TYPE: Subjective Short Answer
HAS VARIABLES: False
STUDENT ENTRY MODE: Basic
LEARNING OBJECTIVES: MVC#.FARR.18.02.11 - Accept console input
DATE CREATED: 5/16/2017 1:18 PM
DATE MODIFIED: 11/15/2019 3:47 PM

Copyright Cengage Learning. Powered by Cognero.

Microsoft Visual C# 2017 An Introduction to Object-Oriented Programming 7th edition Farrell Test Bank