Question: 1

You need to gain a better understanding of the solution before writing the program. You decide to develop an algorithm that lists all necessary steps to perform an operation in the correct order. Any technique that you use should minimize complexity and ambiguity. Which of the following techniques should you use?

A. flowchart
B. decision table
C. C# program
D. A paragraph in English

Answer: A

Question: 2

Which of the following languages is not considered a high-level programming language?

A. C#
B. Visual Basic
C. Common Intermediate Language
D. C++

Answer: C

Question: 3

You are writing code for a business application by using C#. Write the following statement to declare an array:
int[] numbers = { 1, 2, 3, 4, 5, };
Now, you need to access the second item in this array (the number 2). Which of the following expression should you use?

A. numbers[0]
B. numbers[1]
C. numbers[2]
D. numbers[3]

Answer: B

Question: 4

You are developing a C# program. You write the following code:
int x = 10
int y = ++x
int x = y++;
What will be the variable z after all the above statements are executed?

A. 10
B. 11

C. 12
D. 13

Answer: B

Question: 5

You are writing a method named PrintReport that doesn't return a value to the calling code. Which keyword should you use in your method declaration to indicate this fact?

A. void
B. private
C. int
D. string

Answer: A

Question: 6

You need to provide complex multi-way branching in your C# program. You need to make sure that your code is easy to read and understand. Which of the following C# statements should you use?

A. case
B. break
C. if-else
D. switch

Answer: D

Question: 7

You are writing a C# program that iterates through a collection such as arrays and lists. You need to make sure that you process each item in the collection once. You also need to ensure that your code is easy to read and debug. Which of the following C# statements provide the best solution for this requirement?

A. while
B. for
C. foreach
D. do-while

Answer: C

Question: 8

You are developing a C# program that needs to perform 5 iterations. You write the following code:

```
01: int count = 0;
02: while (count <= 5)
03: {
```

```
04: Console.WriteLine("The value of count = {0}", count);
05: count++;
06: }
```

When you run the program, you notice that the loop does not iterate five times. What should you do to make sure that the loop is executed exactly five times?

A. Change the code in line 01 to
int count = 1;
B. Change the code in line 02 to:
while (count == 5)
C. Change the code in line 02 to
while (count >= 5)
D. Change the code in line 05 to
++count;

Answer: A

Question: 9

You are developing a C# program. You write the following code line:
int x = 6 + 4 * 4 / 2 - 1;
What will be the value of the variable x after this statement is executed?

A. 19
B. 13
C. 20
D. 14

Answer: B

Question: 10

You are writing a C# program that needs to manipulate very large integer values that may exceed 12 digits. The values can be positive or negative. Which data type should you use to store a variable like this?

A. int
B. float
C. double
D. long

Answer: D

Question: 11

You have written a C# method that opens a database connection by using the SqlConnect object. The method retrieves some information from the database and then closes the connection. You need to make sure that your code fails gracefully when there is a database error. To handle this situation, you wrap the database code in a try-catch-finally block. You use two catch blocks—one to catch the exceptions of type SqlException and the second to catch the exception of type Exception. Which of the following places should be the best choice for closing the SqlConnection object?

A. Inside the try block, before the first catch block
B. Inside the catch block that catches SqlException objects
C. Inside the catch block that catches Exception objects
D. Inside the finally block

Answer: D

Question: 12

You are assisting your colleague in solving a compiler error that his code is throwing. Following is the problematic portion of his code:

```
try
{
bool success = ApplyPicardoRotation(100, 0);
// additional code lines here
}
catch(DivideByZeroException dbze)
{
//exception handling code
}
catch(NotFiniteNumberException nfne)
{
//exception handling code
}
catch(ArithmeticException ae)
{
//exception handling code
}
catch(OverflowException oe)
{
//exception handling code
}
```

To remove the compilation error, which of the following ways should you suggest to rearrange the code?

A.
```
try
{
bool success = ApplyPicardoRotation(100, 0);
// additional code lines here
}
catch(DivideByZeroException dbze)
{
//exception handling code
}
catch(ArithmeticException ae)
{
//exception handling code
}
catch(OverflowException oe)
{
//exception handling code
}
```

B.
```
try
{
```

```
bool success = ApplyPicardoRotation(100, 0);
// additional code lines here
}
catch(DivideByZeroException dbze)
{
//exception handling code
}
catch(Exception e)
{
//exception handling code
}
catch(OverflowException oe)
{
//exception handling code
}
```

C.

```
try
{
bool success = ApplyPicardoRotation(100, 0);
// additional code lines here
}
catch(DivideByZeroException dbze)
{
//exception handling code
}
catch(NotFiniteNumberException nfne)
{
//exception handling code
}
catch(OverflowException oe)
{
//exception handling code
}
catch(ArithmeticException ae)
{
//exception handling code
}
```

D.

```
try
{
bool success = ApplyPicardoRotation(100, 0);
// additional code lines here
}
catch(DivideByZeroException dbze)
{
//exception handling code
}
catch(NotFiniteNumberException nfne)
{
//exception handling code
}
catch(Exception e)
{
//exception handling code
}
catch(ArithmeticException ae)
{
//exception handling code
```

```
}
```

Answer: C