ITSE 1430 (DL)- Fall 2016
Quiz 1- Chapters 1-6
Total Points: 70

Due: Saturday, October 1st @ 11:59PM. Look at Syllabus/ICR on late work.

Directions: For Questions 1-23, clearly mark answers on a separate word (or notepad) document. See sample file/directions provided by your professor and submit to the appropriate location on the MyTCC (BlackBoard) site.

– Assume all variables are properly declared- unless otherwise mentioned.

Multiple Choice. Mark the one best answer for each question. **(2 pts. each)**

1. Delegate declarations resembles ____ declarations.
   A. method
   B. variable
   C. object
   D. data

2. Which one of the following statements initializes a variable with a double literal?
   A. `string modelNumber = "R2C32";`
   B. `double percentage = 23.59;`
   C. `int width = 150;`
   D. `decimal bonus = 0.99m;`

3. `void` and `static` are examples of C# predefined ____.
   A. classes
   B. attributes
   C. objects
   D. keywords

4. The starting point of a C# program is the _____ method.
   B. `Start`
   C. `Open`
   D. None of the above

5. How can you produce the following results in one line of code?

```
Welcome
to
C#
 Programming!
```

   A. `Console.Write("Welcome\nto\nC#\nProgramming!\n");`
   B. `Console.WriteLine("Welcome\nto\nC#\nProgramming!");`
   C. `Console.Write("Welcome \n to \n C# Programming!");`

6. Which of the following generates a syntax error?
   A. `c *= 3;`
   B. `c %= 2;`
   C. `c /= 4;`

7. Before a C# application can be run, it must be compiled into a language called
   A. Microsoft Common Language
   B. Microsoft Intermediate Language
   C. Microsoft Runtime Language
   D. Microsoft Solution Language

8. Two properties that are common to both forms and controls are the
   A. `Name` and `AcceptButton` properties
   B. `Name` and `TabIndex` properties
   C. `Name` and `Text` properties
   D. `TabIndex` and `AcceptButton` properties

9. To change the file name for a form, project, or solution you use the
   A. Form Designer
   B. Solution Explorer
   C. Code Editor
   D. Toolbox

10. To say that a C# application is event-driven means that it responds to
   A. user events only
   B. user events and other types of events
   C. application events only
   D. class events only

11. Which of the following statements is true?
   A. You can't use C# keywords in a comment.
   B. Comments must appear at the beginning of a line.
   C. Comments can be inaccurate or out of date.
   D. Comments make the program run slower.

12. Which of the following statements results in a string that looks like this when displayed?

```
c:\murach\files
```

   A. `string s = "c:\murach\files";`
   B. `string s = "c:\\murach\\files";`
   C. `string s = "c:\\murach\\files\";`
   D. `string s = "c:\murach\files\";`

13. If a decimal variable named `total` has a value of 1234.56, what string will result from the following statement?

```
string s = total.ToString("c2");
```

   A. `$1,234.56`
   B. `1,234.56`
   C. `1234.56`
   D. `1235`

14. When you call a method with a parameter list, the arguments in the argument list
   A. must be coded in the same sequence as the parameters
   B. must be declared with data types that are compatible with the parameters
   C. must have the same names as the parameters
   D. must be coded in the same sequence as the parameters and have data types that are compatible with the parameters

15. How can the compound conditional expression `((average > 79 && average <= 89))` be written to eliminate the logical operator - without changing the logic?
   A. remove the `&&` and create a nested `if` statement
   B. `(average > 79 <= 89)`
   C. by replacing the `&&` with `||`
   D. `(79 > average <= 89)`

16. Looking at the example below, what happens if the `break` is omitted?

```
switch (phoneDigit)
{
    case 1: num = 1;
     break;
    case 2: num = 2;
     break;
    case 3: num = 3;
       break;
    case 4: num = 4;
       break;
    default: num = 0;
       break;
}
```

A. a syntax error is generated
B. `num` is always assigned 0
C. `num` is never assigned a value
D. `num` is assigned 1

17. The only posttest loop structure available in C# is _____.

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

18. Which of the following loops will print out the numbers 5, 10, 15, 20, 25?

I.
```
int i = 5;
while(i <= 25)
{
  Console.WriteLine(i);
  i = i + 5;
}
```

II.
```
for(int i = 5;i <= 25;i = i + 5)
{
  Console.WriteLine(i);
}
```

III.
```
for(int i = 4;i < 24;i = i + 5)
{
  Console.WriteLine(i + 1);
}
```

A. I only
B. II only
D. I and III only

19. What is the following code attempting to calculate?

```
static int go (int x)
{
   int ans = 0;
   while(x > 0)
   {
      ans += x % 10;
      x /= 10;
   }
   return ans;
}
```

A. The code is counting the number of 0s in the number.
B. The code is counting the number of 1s in the number.
C. The code is summing all of the digits in the number.
D. The code is counting the number of digits in the number.

20. What is output by the code below?

```
int w = 100, x = 88;
if(w > 90)
   if(x > 80)
      Console.Write("def");
   else
      Console.Write("xyz");
else
   Console.Write("ghi");
Console.Write("fun");
```

A. `xyzdef`
B. `xyz`
C. `xyzfun`
D. `deffun`

Short Answer. Clearly mark answers as directed. Partial Credit will be given. Point values to the right of the question.

21. Suppose `x`, `y`, and `z` are `double` variables and `x = 2.5, y = 6.9`, and `z = 10.0`. What is value of each of the variables after each statement is executed. Fill in the remaining values of the table below. Use original declaration for each question. **(10 @ 2 each)**

	x	**y**	**z**
A. `z *= y++ % 7;`	2.5	7.9	_____
B. `x = z / 3 * --y;`	_____	5.9	10.0
C. `z /= (int) y / x;`	_____	6.9	4.167
D. `z = x + y / 4;`	2.5	6.9	_____
E. `y = (z- y) * 2 + --y;`	2.5	_____	10.0

22. Given the following method definition what would be a valid call. Mark as valid or invalid. **(10 @ 2 each)**

```
public static int GetData(out int aValue, ref int bValue)
```

A. `someIntValue = GetData(aValue, bValue);`

B. `someIntValue = GetData(out aValue, ref bValue);`

C. `someIntValue = GetData(out, ref);`

D. `someIntValue = GetData(int out aValue, int ref bValue);`

E. `GetData(out aValue, ref bValue);`

23. Write a C# program to test a method that will return the day within the year for a given date. For example, May 19, 1981 was the 31 + 28 + 31 +30 + 19 = 139th day of that year. Output should look similar to below.

Sample Run:

```
Enter a month: 5
Enter a day: 19
Enter a year: 1981

This is the 139 day of the year.
```

Fill in the missing parts of the program below to solve the problem as stated above. Do not add any additional lines of code. (**10 @ 2 each)**

```
//DayInYear.cs

using System;
class DayInYear {
   static void Main() {
      int day, month, year, inYear;

      Console.Write("Enter a month: ");
      _____________________________                                    //1

      Console.Write("Enter a day: ");
      month = Convert.ToInt32(Console.ReadLine());

      Console.Write("Enter a year: ");
      year = Convert.ToInt32(Console.ReadLine());

      _____________________________                                    //2

      Console.WriteLine();
      Console.WriteLine("This is the {0} day of the year.", inYear);
  }
```

```
static int dayInYear(int day, int month, int year){
        int dayNumber;

        _____________________________                                    //3
        if (month > 2) {
        dayNumber = dayNumber - ((4 * month + 23) / 10);
          if (isLeapYear(year))
            _________________                                            //4
     }

        return dayNumber;
 }

static bool isLeapYear(int year)
 {
    ____________________________                                         //5
 }
}
```

Note: An easier way to solve this problem is using `DateTime` class methods.

Pre Test- This part has already been taken. Your score will be added to the quiz. **(10 pts.)**

==========

Extra Credit: Implement the following program. Follows same program guidelines and graded on the same scale as program sets. Submit only your `.java` file- no test runs required. Partial credit given. (**10 points)**

A rational number is one that can be expressed as a fraction `k/m`, where `k` and `m` are integers and `m` is nonzero. It turns out that these are also the numbers than can be expressed in repeating decimal form. That is, every rational number can also be written in the form

$$p.d_1\ d_2\ d_3\ \ldots\ d_{n-r}\ (d_{n-r+1}\ d_{n-r+2}\ \ldots\ d_n)$$

where `p` is an integer, each d_i is a digit between 0 and 9, and the parentheses around the last `r` digits indicate that that sequence of digits repeats forever. Write a C# console application that takes as input two nonnegative integers `k` and `m` (a numerator and denominator) and produces as output the simplest repeating decimal representation of the rational number `k/m`. Simplest here means the number of digits appearing after the decimal point is minimum. For example, the rational number `13/6` is represented by each of `2.1(6)`, `2.16(6)`, and `2.1(66)`, but only the first of these is in simplest form. Another example, the simplest repeating decimal representation of `40/4` is `10.(0)`. The program should terminate when the user enters zero for the denominator. To simplify the task of producing output, your results (labeled) are to be written in the form $p.d_1\ d_2 \ldots d_n$ as well as `r` the length of the repeating sequence of digits. Assume any rational number used in testing your program is such that its simplest repeating decimal representation has no more than thirty digits appearing after the decimal point. Output should look similar to below.

Sample Run:

```
Enter numerator: 131
Enter denominator: 350

Decimal Representation: 0.37428571
Length of Repeating digits: 6

Enter numerator: 56
Enter denominator: 0
```

Name the program: `RepeatDecimalsXX.cs`, where XX are your initials.