Engr 123 **January 25, 2016**
Practice Problems – Loops

1. Write a C# console application to produce a multiplication table for x times y where x goes from 1 to lastX and y goes from 1 to lastY. Input lastX and lastY from the user. For example, your program output might look like this:

```
Enter the last value of x... 3
Enter the last value of y... 2
Multiplication table for x*y
1 * 1 = 1
1 * 2 = 2
2 * 1 = 2
2 * 2 = 4
3 * 1 = 3
3 * 2 = 6
```

2. Write a program to prompt the user to input a sequence of integers. Continue inputting integers until the user enters a zero. When the user enters a zero to indicate the end of the list your program should print the minimum and maximum integer and the average value of all of the integers entered. For example, your program output might look like this:

```
Enter an integer... 4
Enter an integer... 5
Enter an integer... 8
Enter an integer... 13
Enter an integer... 2
Enter an integer... 7
Enter an integer... 0
The maximum integer entered was 13.
The minimum integer entered was 2.
The average of all of the integers was 6.50.
```

3. Write a program to evaluate the equation $y = 3x^3 + 2x^2 - 4x + 10$ for values of x starting at 0 and incrementing in steps of 0.1 until the value of y exceeds 1000. Print only the first value of x and the corresponding value of y for which y exceeds 1000.

4. Write a program to find the integer square root of a number input from the user. The integer square root is the largest integer whose square is less than or equal to the number. Use a loop and do this program with an exhaustive search.

5. Write a program consisting of 2 nested for loops that produces the following output line: `0001021011122022122`

6. Write a program to using nested loops to print a table of all possible 4 bit binary numbers in order. Your output should look like this:

```
0000
0001
0010
0011
0100
0101
0110
0111
1000
1001
1010
1011
1100
1101
1110
1111
```

7. Write a program in C# which will compute the value of e from the infinite series given by

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$$

Using this series you can calculate e by allowing x to equal 1. The more terms you use the more accuracy you get. Your program should calculate and print the approximate value of e using the series equation with 1 to 20 terms. The output of your program will look something like this:

```
Terms    e          error
  1    1.00000    1.718281828459045
  2    2.00000    0.718281828459045
  3    2.50000    0.218281828459045
  4    2.66667    ...
       ...
 20    ...        ...
```

To calculate the error declare a constant e = 2.718281828459045 and subtract your approximation to e from this constant.