**Exercise - Value Types and Arrays**

COURSE: ISTA 220  
Module Number: 5  
Supporting Material: Microsoft Visual C# - Step by Step, Chapters 9 - 10

The following exercises are open book and open note. You are free to use any written documentation you wish. These are individual exercises; please do not consult with each other in writing your programs. Submissions should made be no later than the start of the class day. Come to class prepared to discuss, ask questions, and present your solutions.

Description

In Chapter 10 you were introduced to the System.Random class which provides an interface to generate random numbers. By generating many random values, you can estimate the result of complicated mathematical equations via a technique known as the Monte Carlo method. In this exercise, you will estimate the value of using a Monte-Carlo method.

Envision a unit circle (radius = 1). Specifically, we will work with the top right quadrant where all valid points are between { x = 0, y = 0 } and { x = 1, y = 1 }. By randomly generating many x, y pairs and calculating the length of the hypotenuse of a triangle with sides of length x, y using the Pythagorean theorem, , you can classify the pairs into those that overlap the unit circle (, blue shaded area) and those that fall outside the unit circle (, red shaded area). The ratio of pairs that overlap the unit circle divided by the total number of pairs generated is the area of the unit circle in the top right quadrant.

Procedure

In a new project:

1. Create a struct to hold x, y coordinates as doubles. Provide a 2-parameter constructor which takes a new set of coordinates and a 1-parameter constructor which takes a Random object and uses the NextDouble method to initialize X and Y.
2. Create a function which takes an x, y coordinate struct and returns a double corresponding to the hypotenuse of a triangle with sides of lengths x, y.
3. Build a Main method which takes one int parameter from the command line and creates an array of that length containing randomly initialized coordinates.
4. Iterate over the array, incrementing a counter for each coordinate which overlaps the unit circle.
5. Using the Length parameter of the array, divide the number of coordinates overlapping the unit circle by the number of array elements. Multiply this value by 4.
6. Print the value from step #4 along with the absolute value of the difference between your estimate of and Math.Pi.
7. Run your program, passing 10, 100, 1000, and 10000 as the command line parameter. Record the output of your program.

Provide answers for the following:

1. Why do we multiply the value from step 5 above by 4?
2. What do you observe in the output when running your program with parameters of increasing size?
3. If you run the program multiple times with the same parameter, does the output remain the same? Why or why not?
4. Find a parameter that requires multiple seconds of run time. What is that parameter? How accurate is the estimated value of ?
5. Research one other use of Monte-Carlo methods. Record it in your exercise submission and be prepared to discuss it in class.