Delegate Practice

We’ve talked about delegates a fair amount. What makes them useful is that they’re a SUPER easy mechanism for customizing and expressing *any* publisher-subscriber relationship! So, let’s try declaring and using a delegate in a simple exercise.

# Requirements:

* Create a new console project for this exercise.
* Create a brand-new delegate that can handle methods that return void and may take in any number of ints.
* Create a ParamMath class.
* In the ParamMath class, create a public static method for each of the 5 major math operations. Each of these methods should be written to fit the delegate. When the method runs, it performs its math function on each of the ints it was given to form a total result, then prints that result with a proper label to the console.
  + Example: Imagine the Add method is called with arguments (1, 2, 3, 4, 5). It would perform addition across all arguments, starting with 1 and ending with 5. The total in this case would be 15. The method would then print to the console the result with a useful label, like “Sum: 15”.
* In the Program class, write the code necessary to do the following:
  + Give Program a public, static, automatic property of the delegate type you declared earlier.
  + Subscribe each of the math methods in ParamMath to Program’s delegate.
  + Prompt the user for a collection of numbers (separated by some character or sequence of your choosing).
    - Make sure to give the user clear instructions on how to correctly enter the values for your application.
  + Once the user has entered an arbitrary collection of ints, invoke the delegate passing the user’s arguments.
  + You may allow the program to terminate at this point.
* NOTE: If the user gives you a zero at any point in the process (other than the first argument in the number sequence), you may allow the application to throw an exception for dividing by zero. That’s perfectly acceptable
* Do NOT abuse the Main method. This is a real project and you are expected to code in a professional way.

# What you’ll need

* Delegates
* The “params” keyword

# You might like to check out

* BigInteger in the System.Numerics dll

# Rubric

This assignment is graded in binary fashion. Either you met all the requirements and earn all the points, or you don’t and you get a 0.