Data Structures and Algorithms I

Programming Assignment #1

Purposes:
Work with defining a C++ class, constructors and using a test driver to track down bugs in a class's implementation.

Description:
Implement and test a small class called Statistician with some member functions. The statistician is a class that is designed to keep track of simple statistics about a sequence of real numbers. Member functions are as follows:

- next_number – a number (*double*) is given to a statistician.
- length – returns the count of how many numbers have been given to the statistician.
- sum – returns the sum of all the numbers that have been given to the statistician.
- mean – returns the arithmetic mean (average) of all the numbers that have been given to the statistician.
- minimum – returns the smallest number.
- maximum – returns the largest number.
- reset – clear statistician as if no numbers had yet been given to it.

You will also need a default constructor that simply does any initialization needed for the statistician to start its work. Some of these functions are constant functions (accessors) and others modification functions (mutators). Think carefully what you need to store before deciding the private member variables. Note that you do not need to store all the sequence of numbers.

Write the class definition in the header file (stats.h) and the member function definitions in the implementation file (stats.cpp). Specify preconditions and postconditions for functions along with other documentations in the codes.

To validate preconditions of a member function, you may use *assert* library utility. Write a test driver and test your Statistician class.

Note that your implementation of statistician class should not produce any output to *cout*, nor expect any input from *cin*. That is, all the interaction with the member functions occurs through their parameters and return values.