CS 304
Program 1: C++ Classes

Due Wednesday, Sept. 11, 10:00 pm
(but see due date for "Show me that you got this far" on page 2)

This project is worth a total of 80 points, or 8% of your course grade

Goals

- Design, implement and test an Abstract Data Type class in C++, making appropriate use of member functions and variables, constructors, namespaces, header files. Document class value semantics, and class invariants
- Demonstrate appropriate use of various C++ language facilities, including **`const`**, **`bool`**, overloaded binary infix operators and input/output operators, **`friend`** functions

Overview

This programming project is based on one described in your textbook (page 90).

Implement a statistician class that accepts a sequence of integers. The statistician class has member functions to provide information about the sequence of numbers: length of the sequence, the sum of the numbers in the sequence, the arithmetic mean of the numbers, the smallest and largest numbers in the sequence. There is also a function to reset (clear) the statistician.

The statistician class also has functions (friend and non-member) to overload the infix operators +, *, ==., and <<

- Two statisticians are equivalent if their lengths, mean, smallest, and largest numbers are the same.
- Adding two statisticians together results in a new statistician object that is equivalent to a statistician that has received the numbers in the sequences seen by its arguments.
- Multiplying a statistician by a scaling factor results in a new statistician that is equivalent to one that has seen the numbers seen by its argument statistician, except that each has been multiplied by the scaling factor.
- The output operator << has been overloaded to print a string representation of the statistician.

Details

A header file is supplied (on Moodle), except that the instance variables are left for you to specify. Complete and test the implementation of the statistician class.

Details and Grading

Header file (5 points total)

Modify the header file to

- Define the instance variables (5 points)
  Note: do not attempt to save all of the values seen. Instead, just maintain the information needed to implement the specified methods.

Basic member functions (10 points total)

Create and test an implementation file with the following functions as specified in the header file:
- No-argument constructor (5 points)
- next (5 points)

Show me that you got this far (3 points total)

Bring your running, tested program to my office (or ask me to come to your computer in the lab) and show me that you have gotten this far no later than **Thursday, Sept. 5, 4:00 pm**.
Note: you must talk to me in person; e-mailed submissions will not be accepted unless prior arrangements have been made. (3 points)

Remaining member functions (12 points total)

Create and test an implementation file with the following functions as specified in the header file:
- reset (2 points)
- length (2 points)
- sum (2 points)
- mean (2 points)
- minimum (2 points)
- maximum (2 points)

Overloaded infix operators (35 points total)
- operator + (10 points)
- operator * (5 points)
- operator == (10 points)
- operator << (10 points)

Quality (15 points total)
- Code conforms to all C++ coding standards (15 points)

Submitting

Zip up your VS project and submit it through Moodle.