COM S 213 – Fall 2002

ASSIGNMENT #4:   **Big Numbers**

DATE GIVEN: **9/26/02**

DATE DUE: **10/3/02**

PURPOSE:
To get more experience in using either arrays or C++ strings and to make sure you understand the concept of operator overloading.

ASSIGNMENT:
Every once in a while you need to write a program that needs to deal with a wide range of data of a given type. Sometimes, this means you are faced with the prospect of having to limit the range of data you can work with because you are limited by the data types available in C++.

For example, you know that your program needs to work with big integers and you are worried you might need more capacity than an `unsigned long` can provide (a little over 4,000,000,000). Some compilers support the notion of a `long long`, but you can't count on it being there. And, while that gets you a bigger range there is still a limit on what you can store!

To deal with this problem, some programmers will seek out or implement themselves a "Big Number" class. This class will allow you to deal with an arbitrarily big number.

Your assignment is to implement, as best you can, a Big Number class. This class only needs to be able to deal with integer values, but arbitrarily large ones. A good way to do this is to store the integer representation of the number as a string (either C style or C++ style). The C++ string has benefits because it will expand as you need it to if your number starts getting really large!

You will need to overload operators to make sure that this big number class can be used just like and integer. I would like the following operations supported:

      Addition
      Subtraction

Assignment
Able to print out via "cout << bignum << endl;"
Comparisons (==,!=,>,<,>=,<=,!=)
Multiplication

NOTE: Multiplication is potentially hard and/or potentially very slow. All I'm looking for is that you get it implemented no matter how slow it is ☺

SUGGESTIONS:
I'm beginning to get more abstract with the assignments and leaving a lot of the details up to you. Please flesh out this assignment sooner rather than later as it will suddenly appear daunting if left to the last minute. Review lecture material and the book for help and come to office hours as needed!