LAB #4 – Operator Overloading

Remember, you will not receive lab credit if you do not get checked off before leaving each lab. Once you have a zero on a lab, then it cannot be changed because we have no way to know if you were there or not! You must get a score of at least 5 or you cannot submit it later. If you need to get last week’s lab graded, you need to show your program to the TAs within 10 minutes of getting to this lab.

Define a class for rational numbers. A rational number is a number that can be represented as the quotient of two integers. For example, 1/2, 3/4, 64/2, and so forth are all rational numbers. (By 1/2 and so on we mean the everyday fraction, not the integer division this expression would produce in a C++ program.) If you don’t remember elementary school remember, mathisfun.com!

Call the class Rational. Represent rational numbers as two data member of type int, one for the numerator and one for the denominator. Include a constructor with two arguments that can be used to set the member variables. Also include a constructor that has only a single parameter of type int; call this single parameter wholeNumber and define the constructor so that the object will be initialized to the rational number wholeNumber /1. Include a default constructor that initializes an object to 0 (that is, to 0/1).

Overload the input and output operators >> and <<. Numbers are to be input and output in the form 1/2, 3/4, 64/2. Note that the numerator, the denominator, or both may contain a minus sign, so -1/2 , 3/-4, -64/-2 are also possible inputs. Get it working on positive numbers first. Overload all the following operators so that they correctly apply to the type Rational: ==, <, <=, >, >=, +, -, *, and /.

Hint: If you design your program correctly you will decompose these operations into a handful of functions. Then all you must do is use the correct function(s) possibly inserting or ignoring a minus sign, or possibly inverting a number.

Write a test program to test your class. What will your program do if the user enters a floating point value?

Hint: Two rational numbers a/b and c/d are equal if a*d equals c*b. If b and d are positive rational numbers, a/b is less than c/d provided a*d is less than c*b.

Hint: You should include a function to normalize the values stored so that, after normalization, the denominator is positive and the numerator and denominator are as small as possible. For example, after normalization 4/-8 would be represented the same as -1/2.