Assignment

Convert the member functions in your fraction class (lab #6) into overloaded operators as described below. Modify your calc program to utilize the overloaded operators.

Program Description

1. Convert the four arithmetic functions to overloaded operators. If you implement the operators as friend functions and provide an appropriate conversion constructor (described below) your operators can be invoked or called in three different ways. Assume that f1 and f2 are appropriately defined fraction objects:

   \[
   \text{fraction } f3 = f1 + f2; \\
   \text{cout} \ll f3 \ll " = " \ll f1 \ll " + " \ll f2 \ll \text{endl;} \\
   \text{fraction } f4 = f2 + 2; \\
   \text{cout} \ll f3 \ll " = " \ll f2 \ll " + " \ll 2 \ll \text{endl;} \\
   \text{fraction } f5 = 2 + f1; \\
   \text{cout} \ll f5 \ll " = " \ll 2 \ll " + " \ll f1 \ll \text{endl;} 
   \]

   The second and third addition operation automatically call the conversion constructor to convert the 2 into a fraction, which can then be added to the original fraction object.

2. A conversion constructor has one parameter and converts one data type to another (the parameter type to the class-type of the defining class). For example:

   \[
   \text{fraction(int n) : numerator(n), denominator(1) \{} \\
   \]

   is a conversion constructor that converts the integer n into the fraction n/1. By using default parameters, it often possible to make one constructor serve both as a “normal” constructor and as a conversion constructor. For example:

   \[
   \text{fraction(int n, int d = 1);} \\
   \]

   This constructor will construct a fraction from two integers, a numerator and a denominator, or will convert a single integer into a fraction (i.e., it will construct a fraction from a single integer). Adding a default value for n allows this version to also serve as the default constructor:

   \[
   \text{fraction(int n = 0, int d = 1);} \\
   \]
3. Replace the print function with operator<< (see pp. 616 - 620 and the Time example). You must add #include <iostream> and using namespace std; to the top of fraction.h (cout is an instance of ostream).

4. Replace the read function with operator>> (see the time example).

**Submitting Your Code**

Uploaded three files to WebCT: fraction.h, fraction.cpp, and calc.cpp (Note that you must make a new directory to avoid overwriting existing files – instructions are provided on the class web site above the lab assignments).