

Emphasis on: Overloading Operator Functions

For program 6, you are to revisit program 5 where we created a Fraction class. The fraction class contained methods for performing arithmetic operations such as add, sub, mult and div. The fraction class also contained comparison operators such as equal, greaterthan and lessthan. For program 6, you are to rewrite these methods as overloaded operator methods. Change the definition of Fraction.h to user overloaded operators and also change implementations of these member functions in Fraction.cpp to reflect the changes. You should also rewrite the FractionTestDriver to demonstrate the use of the overloaded operators on the Fraction class.

**Program 6 Requirements:**

- 1) Rewrite the arithmetic class methods add, sub, mult and div using overloaded operators
- 2) Rewrite the comparison operators equal, greaterthan and lessthan also as overloaded operators
- 3) You must change the Fraction.h declarations as well as the implementations of the new overloaded operators in Fraction.cpp
- 4) Rewrite the FractionTestDriver to use and test the overloaded arithmetic and comparison operators.

**Extra Credit:**

The print class method, as written in program 5, writes a representation of the fraction on cout standard output stream. It is conventional not to have classes send output directly to streams, but leave input and output up to the client/user program. For 10 extra credit points, remove the print() member function and instead provide and overloaded << (insertion) member function that will insert a representation of the fraction into a stream. See the book chapter 15, pg 822 for an example of overloading the stream insertion operator.