

**MS-3812**  
**C++ Program #2**  
**Simple Classes**

**Date Assigned:**      **Wednesday, December 6, 2006**  
**Date Due:**            **Wednesday, December 21, 2006**

A rational number is any quantity that can be expressed as  $i/j$  where both  $i$  and  $j$  are integers. Rational addition, subtraction, multiplication, and division are defined as follows:

$$\frac{a}{b} + \frac{c}{d} = \frac{ad + bc}{bd}$$
$$\frac{a}{b} - \frac{c}{d} = \frac{ad - bc}{bd}$$
$$\frac{a}{b} \times \frac{c}{d} = \frac{ac}{bd}$$
$$\frac{a/b}{c/d} = \frac{ad}{bc}$$

In the previous assignment, you were asked to implement only the basic object-oriented functionality of the Rational class. Now that we've covered all the topics below, add the following features to your Rational abstraction (or Abstract Data Type):

- Default values for parameters
- Adding global functions to the interface (i.e. reduce() )
- Proper placement of the reduce() function call
- C'tor initialization lists and simple in-lining
- Accessors and Mutators (2 forms: overloaded vs. getters/setters)
- The concept of const objects:
  - o The const keyword
  - o Const member functions
  - o Const objects
  - o Using const objects vs. non-const objects
  - o Returning const objects vs. non-const objects
  - o Places for the const keyword:
    - Leading context for return types
    - Trailing context for modifying member fns.
    - In context for parameters
    - Using const reference effectively (see below)
- The concept of References
- The idea of const references for safety and efficiency

(The last implementation of the Rational class will be to add operator overloading to the mix.)