

7. Write a rational number class. This problem will be revisited in Chapter 11, where operator overloading will make the problem much easier. For now we will use member functions `add`, `sub`, `mul`, `div`, and `less` that each carry out the operations `+`, `-`, `*`, `/`, and `<`. For example, `a + b` will be written `a.add(b)`, and `a < b` will be written `a.less(b)`.

Define a class for rational numbers. A rational number is a "ratio-nal" number, composed of two integers with division indicated. The division is not carried out, it is only indicated, as in `1/2`, `2/3`, `15/32`, `65/4`, `16/5`. You should represent rational numbers by two `int` values, numerator and denominator.

A principle of abstract data type construction is that constructors must be present to create objects with any legal values. You should provide constructors to make objects out of pairs of `int` values; this is a constructor with two `int` parameters. Since every `int` is also a rational number, as in `2/1` or `17/1`, you should provide a constructor with a single `int` parameter.

Provide member functions `input` and `output` that take an `istream` and `ostream` argument, respectively, and `fetch` or `write` rational numbers in the form `2/3` or `37/51` to or from the keyboard (and to or from a file).



Provide member functions `add`, `sub`, `mul`, and `div` that return a rational value. Provide a function `less` that returns a *bool* value. These functions should do the operation suggested by the name. Provide a member function `neg` that has no parameters and returns the negative of the calling object.

Provide a `main` function that thoroughly tests your class implementation. The following formulas will be useful in defining functions.

$$a/b + c/d = (a * d + b * c) / (b * d)$$

$$a/b - c/d = (a * d - b * c) / (b * d)$$

$$(a/b) * (c/d) = (a * c) / (b * d)$$

$$(a/b) / (c/d) = (a * d) / (c * b)$$

$$-(a/b) = (-a/b)$$

$$(a/b) < (c/d) \text{ means } (a * d) < (c * b)$$

$$(a/b) == (c/d) \text{ means } (a * d) == (c * b)$$

Let any sign be carried by the numerator; keep the denominator positive.