**Exercise 8: Rational numbers**

**Deadline: 14th May 2021**

A *rational number* is a number such as −3/7 that can be expressed as the quotient or fraction p/q of two integers, a numerator p and a non-zero denominator q. Every integer is a rational number: for example, 5 = 5/1. The set of all rational numbers is usually denoted by Q.

**Task**

Define the class `rational`, representing the rational number as a pair of integers: numerator and denominator.

```
class rational {
    int num; // numerator
    int den; // denominator
// ...
};
```

Make sure that the denominator is always positive and the greatest common divisor of both numerator and denominator is always 1. Also provide getters for reading numerator and denominator. Place the class definition in the `calc` namespace.

The `rational` class should be equipped with a constructor with a numerator and denominator and a constructor converting from values of `int` type.

In the `rational` class, define the binary and unary arithmetic operators: addition (`+`), subtraction (−), multiplication (`*`), division (`/`), reverse sign (−), and reciprocal (`!`). All of these operators should be friends with the class `rational`. Also define a cast operator to type `double` and an explicit cast operator to type `int` (rounding to the nearest integer) as a member functions in the class.

In constructors, operators, and member functions, throw exceptions if the arguments or the context of these operations are invalid.

Throw exceptions in constructors, operators, and member functions if only the arguments or the context of these operations are invalid. So design the exception class hierarchy for rational numbers, starting with the base class `rational_error` inheriting from `std::logic_error`.

Also define a stream operator to write the rational number to the output stream as a real number with a periodic fraction.

```
class rational {
// ...
    friend ostream& operator << (ostream &out, const rational &w);
};
```

Finally, write a program that reliably test all the functionality in the rational numbers.