THINKING RATIONAL

Yesterday was your first day in your new university and you attended to your first MATH101 lesson. Your new teacher kept talking about how awesome the real numbers are and this made you a little bit upset since you are a true fan of the rational numbers. When you finally arrived to your home, you decided to honor the superior rational numbers in a proper way. Then you decided that the most "rational" way to do this would be using your programming knowledge, which you learned at the summer as a hard-working student, to write a rational number class.

You started to prepare a list of things to implement with excitement. Here is your list:

1) There should be two properties of this class: nominator and denominator. These properties should be integers as they are the second best numbers.
2) A default constructor which creates the number 1.
3) A constructor with two parameters for each property.
4) A constructor which takes a double number and converts it to a rational one.

```
Rational q = new Rational(0.5);
//sets the nominator to 5 and the denominator to 10
```

5) A function which simplifies the rational number. (You decided to use this function after every operation from now on since it would prevent overflow.)

```
q.simplify(); //sets the nominator to 1 and the dominator to 2
```

6) Arithmetic operation functions (add, sub, mul and div). (You decided to implement them as static methods.)

```
Rational q1 = new Rational(2, 3);
Rational q2 = new Rational(1, 3);
Rational q3 = Rational.add(q1, q2); //q3 is 1 over 1, not 3 over 3!
```

7) equals function. (As a very clever programmer you immediately made sure that this function was returning true when you compare 1/2 to 2/4.)

```
Rational q1 = new Rational(1, 2);
Rational q2 = new Rational(2, 4);
q1.equals(q2); //returns true
```

8) biggerThan and smallerThan functions. (Adding the equals function to the list but not adding the biggerThan and the smallerThan functions would be a shameful crime.)
9) (You thought that maybe you could add a power function during a fraction of a second and changed your mind immediately as that function could give you non-rational results.)
10) A function which checks if the number is an integer. Note to self: The modulus sign can be very useful here.

```
Rational q = new Rational(12, 3);
q.isInteger(); //returns true
```

11) toString function which returns the number in the form of "x over y"