Homework 1
Java Programming on Eclipse
Due Monday, September 30, 2019

1. Problems:

Develop a Java class Rational on Eclipse Java development environment. The specification of the class Rational is as follows:

```
class Rational {
    private int nu;   /* numerator */
    private int de;   /* denominator */
    public Rational()
    public Rational(int n, int d) throws IllegalDenominatorException;
    public int getNu();
    public int getDe();
    public boolean equals(Rational r);
    public Rational add(Rational r);
    public Rational sub(Rational r);
    public Rational mul(Rational r);
    public Rational div(Rational r) throws DivideByZeroException;
    public Rational neg();
    public Rational recip() throws IllegalNumeratorException;
    public String toString();
}
```

Each object of class Rational denotes a rational number and contains two fields: the numerator nu and the denominator de of the rational number. The numerator nu and the denominator de should be coprime. If a rational number is positive, both the numerator and the denominator should be positive. If a rational number is negative, only the numerator should be negative. If a rational number is zero, the numerator should be 0 and the denominator should be 1.

The constructor Rational() creates an object of class Rational with the numerator 0 and the denominator 1.

The constructor Rational(int n, int d) creates an object of class Rational with the numerator n and the denominator d. The constructor initializes the fields nu and de of the object with the parameters of the constructor, and the fields nu and de should be reduced to be coprime. The constructor should throw an IllegalDenominatorException if the denominator d is zero.

The two getters return the values of the two fields, respectively.

The method equals(Rational r) checks if the value of the object equals to the value of the argument r.

The method add(Rational r) returns a rational number whose value is the result of the addition of the value of the object and the value of the argument r.

The method sub(Rational r) returns a rational number whose value is the result of the subtraction of the value of the object and the value of the argument r.

The method mul(Rational r) returns a rational number whose value is the result of the multiplication of the value of the object and the value of the argument r.

The method div(Rational r) returns a rational number whose value is the result of the division of the value of the object and the value of the argument r. The method should throw a DivideByZeroException if the argument r is zero;

The method neg() returns a rational number whose value is the result of the negation of the value of the object.

The method recip() returns a rational number whose value is the result of the reciprocal of the value of the object. The method should throw an IllegalNumeratorException if the field nu is zero;

The method toString() returns a string representation of a rational number. For example, a rational number with the numerator 2 and the denominator 3 is represented as “2/3”.

You can write a main method to test the methods in Rational.

2. Handing in your homework:

You should upload a compressed file hw1.zip that contains the package that contains the source class to the eCourse2 website.