Homework 7

Deadline: 2015/05/22 09:00

Shin-Jie Lee（李信杰）
Assistant Professor
Computer and Network Center
Department of Computer Science and Information Engineering
National Cheng Kung University

Problem Description

- Write a program that can do some operations on big integers including addition, subtraction and comparison.
  - Big integer: an integer that is greater than $2^{31}-1$ or less than -2^{31}
- Declare an interface `IOperation` with only one operation:
  - `Object perform(Object o1, Object o2)`
- Create three classes that implement `IOperation`
  - `Addition`
    - The `perform` method takes two arguments and returns the sum of them.
  - `Subtraction`
    - The `perform` method takes two arguments and returns the difference of them.
  - `Comparison`
    - The `perform` method takes two arguments and returns -1, 0, or 1 if the first argument is less than, equal to, or greater than the second argument, respectively.
- Create a class `BigIntegerCalculator` that holds a reference to `IOperation`, and use this class to solve the problem.

Problem Description

- For example, if you want to calculate 123 + 456:
  1. Create a `BigIntegerCalculator` instance.
  2. Create an `Addition` instance and assign to the `IOperation` variable which `BigIntegerCalculator` holds.
  3. Use the `IOperation` object to do the operation; that is, you should invoke the object's `perform` method.

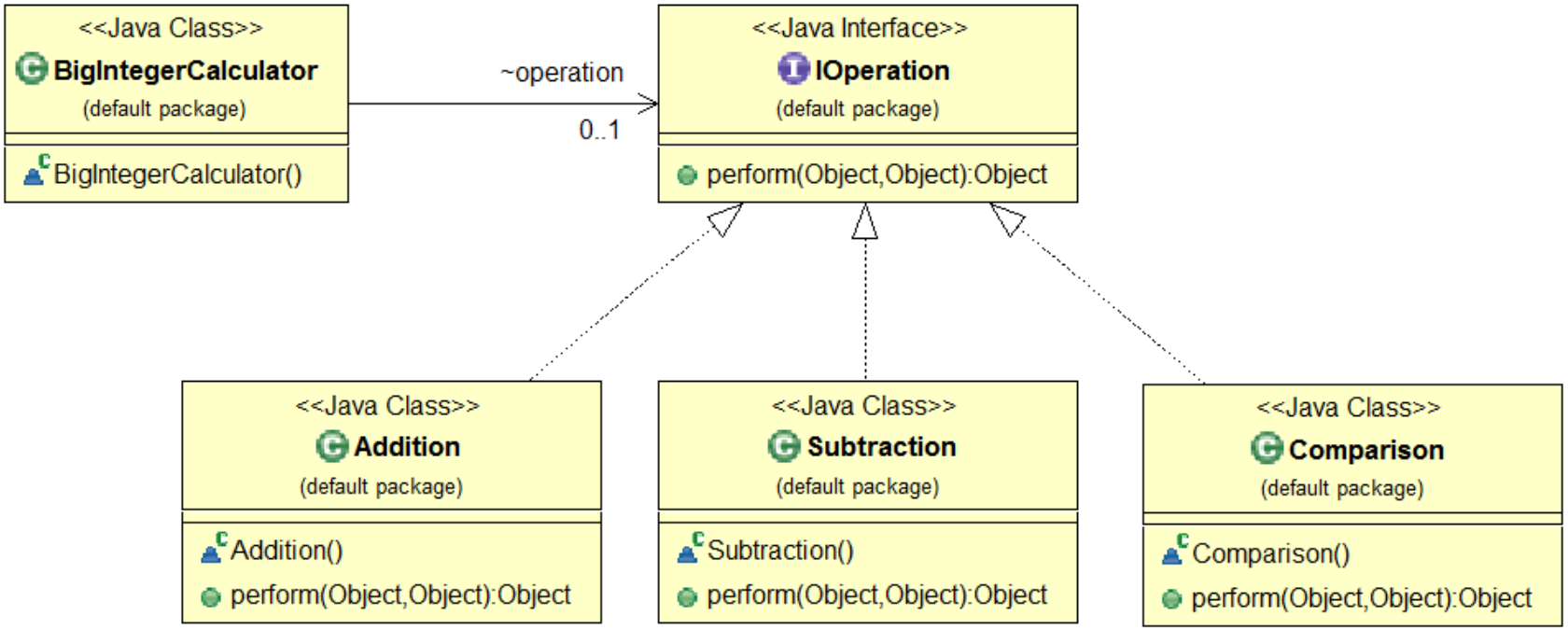
```
// Calculate 123 + 456
BigIntegerCalculator calculator = new BigIntegerCalculator();
calculator.operation = new Addition();
Object result = calculator.operation.perform("123", "456");
```

```
// Determine whether 123 > 456
BigIntegerCalculator calculator = new BigIntegerCalculator();
calculator.operation = new Comparison();
Object result = calculator.operation.perform("123", "456");
```

Problem Description

- The input file is given from the program argument `args[0]`. Each line contains an expression to be evaluated.
- The expression only contains the following elements:
  - Integers (May be greater than 2^{31}-1 or less than -2^{31})
  - Operators (Type-A): `+, -`
  - Operators (Type-B): `>, <, =`
  - Whitespaces
- Type-A and Type-B operators will not appear simultaneously.
- Note that the following expressions may be possible:
  - `123 + -456` (You should evaluate it to `-333`)
  - `-123 - -456` (You should evaluate it to `333`)

Class Design



Sample Input and Output

Input	123 + 456 123 - 456 123 > 456 123 < 456 123 = 456 99999 + 99999 0 - 0 + 0 - 0 + 0 - 0 -123456789 + 987654321 - -123456789 9876543210123456789876543210123456789 + 123 -135792468123456789876543210123456789 < -135792468012345678987654321012345678 9
Output	579 -333 false true false 199998 0 987654321 9876543210123456789876543210123456912 false

Scoring Criteria and Rules

- Correctness: 80%
  - There will be 20 test cases. (Each for 4%)
- Coding standards: 20%
- Plagiarism is strictly forbidden

- You MUST follow the class design illustrated in page 4.
- The following APIs are NOT ALLOWED in your program:
  - java.math.BigInteger
  - java.math.BigDecimal

Submission

- Please upload your source code to Moodle
  - Put all classes in one java file
- The file name should be `{STUDENT_ID}_hw7.java`
- Deadline: 2015/05/22 09:00
- No late submission is accepted

If you have any problem about this homework,
please email to: OpenXavierX@gmail.com（林孝融）