

HW9 (Due 2018/05/21)

1. Write a C++ program to perform the on-line computation for arithmetic expressions and print out the result by a fraction $\mathbf{a/b}$ in reduced form, where \mathbf{a} is the numerator and \mathbf{b} is the denominator. The arithmetic expression is composed of n -digit decimal integer operands and operators $+$, $-$, $*$, and $/$, where $1 \leq n \leq 3$.

A class RationalO is described as follows:

```
#include <iostream>
using namespace std;
class RationalO
{
    friend ostream &operator<<(ostream &, const RationalO &);
public:
    RationalO(long =0, long = 1);
    void setNumerator(long);
    void setDenominator(long);
    long getNumerator();
    long getDenominator();
    RationalO operator+( RationalO &);
    RationalO operator-(RationalO &);
    RationalO operator*(RationalO &);
    RationalO operator/(RationalO &);
    RationalO &operator=( const RationalO &);
    bool operator==( const RationalO &);
    bool operator!=( const RationalO &);
    bool operator<( const RationalO &);
    bool operator>( const RationalO &);
    void Reduce();
    long Gcd(long, long);
    long Lcm(long, long);
private:
    long Numerator;
    long Denominator;
}
```

A rational number is represented by an object of class RationalO.

Sample I/O 1: (The italics for program output and boldfaces for user input)

Please input an expression: **1+2+3/4**
The answer is 3/2.

Sample I/O 2: (The italics for program output and boldfaces for user input)

Please input an expression: **1-2- $\frac{3}{4}$**

The answer is -1/1.

Some member functions of class RationalO are described below:

```
ostream &operator<<(ostream &o, const RationalO &r)
```

```
{
    o << r.Numerator << " / " << r.Denominator;
    return o;
}
```

```
RationalO &RationalO::operator=( const RationalO &r)
```

```
{
    if (&r != this)
    {
        setNumerator(r.Numerator);
        setDenominator(r.Denominator);
    }
    return *this;
}
```