

Source File: ~/1337/66/lab66.(C|CPP|cpp|c++|cc|cxx|cp)
Input: Under control of `main` function
Output: Under control of `main` function
Value: 3

Extend the `Rational` class from Lab 64 to provide the following additional member functions:

- Addition of two `Rational` numbers. The result is returned in reduced form.
- Additive inverse of a `Rational` number. The rational number `a/b` is returned as `-a/b`.
- Subtraction of two `Rational` numbers. The result should be determined by calling the addition and additive inverse member functions. The result is returned in reduced form.
- Multiplication of two `Rational` numbers. The result is returned in reduced form.
- Multiplicative inverse of a `Rational` number. The rational number `a/b` is returned as `b/a`.
- Division of two `Rational` numbers. The result should be determined by calling the multiply and multiplicative inverse member functions. The result is returned in reduced form.
- Printing `Rational` numbers in the form `a/b` where `a` is the numerator and `b` is the denominator. The output should be written to the given output stream.
- Reading `Rational` numbers. The function should read two `ints` from the given input stream, where the first represents the numerator and the second the denominator. Use the “set” functions to perform the initialization of the `private` members.

The least common multiple of two integers u and v , written $\text{lcm}(u, v)$, is the smallest non-negative integer that is a multiple of (i.e., evenly divisible by) both u and v ; and $\text{lcm}(0, 0) = 0$. For non-zero values of u and v , define the least common multiple as

$$\text{lcm}(u, v) = \frac{uv}{\text{gcd}(u, v)}$$

where $\text{gcd}(u, v)$ is the greatest common divisor of u and v .

A header file is shown in Figure 1, a sample `main` function for testing your implementation is shown in Figure 2, and a sample execution sequence is shown in Figure 3. To use the `Makefile` as distributed in class, add a target of `lab66` to `targets2srcfileswithlibrary`.

```
1  #ifndef LAB66_H
2  #define LAB66_H
3
4  #include <iostream>
5
6  using namespace std;
7
8  class Rational
9  {
10 public:
11     Rational(); // default constructor
12     Rational(int num, int denom); // additional constructor
13     void setNumerator(int num); // set numerator to num
14     void setDenominator(int denom); // set denominator to denom
15     int getNumerator() const; // returns numerator
16     int getDenominator() const; // returns denominator
17     void reduce(); // reduce to lowest terms
18                     // and normalize
19     Rational add(const Rational& addend) const; // addition
20     Rational additiveInverse() const; // given a/b, returns -a/b
21     Rational subtract(const Rational& subtrahend) const; // subtraction
22     Rational multiply(const Rational& multiplicand) const; // multiplication
23     Rational multiplicativeInverse() const; // given a/b, returns b/a
24     Rational divide(const Rational& divisor) const; // division
25     ostream& print(ostream& os) const; // print Rational to output
26                                         // stream os
27     istream& read(istream& is); // read Rational from input
28                                   // stream is
29 private:
30     int numerator;
31     int denominator;
32     int gcd(int u, int v) const; // returns the greatest
33                                   // common divisor of u
34                                   // and v
35     int lcm(int u, int v) const; // returns the least common
36                                   // multiple of u and v
37 };
38
39 #endif
```

Figure 1. /usr/local/1337/include/lab66.h

```
1 #include <lab66.h>
2 #include <string>
3 #include <cstdlib>
4
5 using namespace std;
6
7 int main()
8 {
9     unsigned i;
10    Rational first(1, -2), second(-3, 0), result;
11    string operators = "+-*/";
12
13    first.print(cout);
14    cout << ' ';
15    second.print(cout);
16    cout << ' ';
17    result.print(cout);
18    cout << endl;
19
20    while (first.read(cin) && second.read(cin))
21    {
22        for (i = 0; i < operators.length(); ++i)
23        {
24            first.print(cout);
25            cout << ' ' << operators[i] << ' ';
26            second.print(cout);
27            cout << " = ";
28
29            if (operators[i] == '/' && second.getNumerator() == 0)
30                cout << "Division by zero is not allowed";
31            else
32            {
33                switch (operators[i])
34                {
35                    case '+': result = first.add(second); break;
36                    case '-': result = first.subtract(second); break;
37                    case '*': result = first.multiply(second); break;
38                    case '/': result = first.divide(second); break;
39                    default : cerr << "Unknown op"; exit(EXIT_FAILURE);
40                }
41                result.print(cout);
42            }
43            cout << endl;
44        }
45    }
46
47    return EXIT_SUCCESS;
48 }
```

Figure 2. /usr/local/1337/src/lab66main.C

```

1  newuser@csunix ~> cd 1337
2  newuser@csunix ~/1337> mkdir 66
3  newuser@csunix ~/1337> cd 66
4  newuser@csunix ~/1337/66> cp /usr/local/1337/data/66/* .
5  newuser@csunix ~/1337/66> cp /usr/local/1337/include/lab66.h .
6  newuser@csunix ~/1337/66> cp /usr/local/1337/src/lab66main.C .
7  newuser@csunix ~/1337/66> cp /usr/local/1337/src/Makefile .
8  newuser@csunix ~/1337/66> touch lab66.cpp
9  newuser@csunix ~/1337/66> # Edit Makefile and lab66.cpp
10 newuser@csunix ~/1337/66> make lab66
11 g++ -g -Wall -std=c++11 -c lab66main.C -I/usr/local/1337/include -I.
12 g++ -g -Wall -std=c++11 -c lab66.cpp -I/usr/local/1337/include -I.
13 g++ -o lab66 lab66main.o lab66.o -L/usr/local/1337/lib -lm -lbits \
14 -Wl,-whole-archive -llab66 -Wl,-no-whole-archive

15 newuser@csunix ~/1337/66> cat 01.dat
16 -3 4 3 4
17 3 -4 -3 -4
18 25 45 8 99
19 1 0 2 0
20 129 6579 1935 249
21 1331 1651 2301 1079
22 3 1260 6 198
23 43 1935 207 6579
24 5 7 -25 -35
25 -83 1651 127 -1079
26 1079 1651 -1651 1079
27 newuser@csunix ~/1337/66> cat 01.dat | ./lab66
28 1/-2 -3/1 0/1
29 -3/4 + 3/4 = 0/1
30 -3/4 - 3/4 = -3/2
31 -3/4 * 3/4 = -9/16
32 -3/4 / 3/4 = -1/1
33 3/-4 + -3/-4 = 0/1
34 3/-4 - -3/-4 = -3/2
35 3/-4 * -3/-4 = -9/16
36 3/-4 / -3/-4 = -1/1
37 25/45 + 8/99 = 7/11
38 25/45 - 8/99 = 47/99
39 25/45 * 8/99 = 40/891
40 25/45 / 8/99 = 55/8
41 1/1 + 2/1 = 3/1
42 1/1 - 2/1 = -1/1
43 1/1 * 2/1 = 2/1

44 1/1 / 2/1 = 1/2
45 129/6579 + 1935/249 = 32978/4233
46 129/6579 - 1935/249 = -32812/4233
47 129/6579 * 1935/249 = 215/1411
48 129/6579 / 1935/249 = 83/32895
49 1331/1651 + 2301/1079 = 402700/137033
50 1331/1651 - 2301/1079 = -181754/137033
51 1331/1651 * 2301/1079 = 235587/137033
52 1331/1651 / 2301/1079 = 110473/292227
53 3/1260 + 6/198 = 151/4620
54 3/1260 - 6/198 = -43/1540
55 3/1260 * 6/198 = 1/13860
56 3/1260 / 6/198 = 11/140
57 43/1935 + 207/6579 = 1766/32895
58 43/1935 - 207/6579 = -304/32895
59 43/1935 * 207/6579 = 23/32895
60 43/1935 / 207/6579 = 731/1035
61 5/7 + -25/-35 = 10/7
62 5/7 - -25/-35 = 0/1
63 5/7 * -25/-35 = 25/49
64 5/7 / -25/-35 = 1/1
65 -83/1651 + 127/-1079 = -23018/137033
66 -83/1651 - 127/-1079 = 9240/137033
67 -83/1651 * 127/-1079 = 1/169
68 -83/1651 / 127/-1079 = 6889/16129
69 1079/1651 + -1651/1079 = -9240/10541
70 1079/1651 - -1651/1079 = 23018/10541
71 1079/1651 * -1651/1079 = -1/1
72 1079/1651 / -1651/1079 = -6889/16129

73 newuser@csunix ~/1337/66> cat 01.dat | ./lab66 > my.out
74 newuser@csunix ~/1337/66> diff 01.out my.out
75 newuser@csunix ~/1337/66> cat 03.dat | ./lab66 > my.out
76 newuser@csunix ~/1337/66> diff 03.out my.out
77 newuser@csunix ~/1337/66>

```

Figure 3. Commands to Compile, Link, & Run Lab 66