

Source File: ~/1337/57/lab57.(C|CPP|cpp|c++|cc|cxx|cp)
Input: Under control of main function
Output: Under control of main function
Value: 3

A *rational number* is any number that can be expressed as the quotient of two integers (division by zero excluded). In this assignment you are to write functions for validating a rational number, normalizing a rational number, and functions to facilitate the input and output of rational numbers. A header file containing the definition of the structure as well as the prototypes of the various functions is shown in Figure 1.

```
1  /*
2   * Prototypes for operations on rational numbers.
3   */
4
5  #ifndef LAB57_H
6  #define LAB57_H
7
8  #include <iostream>
9
10 using namespace std;
11
12 typedef struct
13 {
14     int numerator;
15     int denominator;
16 } Rational;
17
18 // Function rationalValid returns true if rational represents a valid
19 // rational number and false otherwise.
20 bool rationalValid(const Rational& rational);
21
22 // Function rationalNormalize reduces rational to lowest
23 // terms. Negative rationals should be manipulated so that the
24 // minus sign goes with the numerator. Zero is represented by zero
25 // (0) as the numerator and one (1) as the denominator.
26 void rationalNormalize(Rational& rational);
27
28 // Function rationalInput initializes rational from input stream in
29 istream& rationalInput(istream& in, Rational& rational);
30
31 // Function rationalOutput writes rational to output stream out in the
32 // format rational.numerator/rational.denominator
33 ostream& rationalOutput(ostream& out, const Rational& rational);
34
35 // Function gcd returns the greatest common divisor of u & v
36 int gcd(int u, int v);
37
38 #endif
```

Figure 1. /usr/local/1337/include/lab57.h

All input, output, and intermediate values will be within the range of values representable in an `int`. Only fractional forms are allowed (i.e., the numerator may be greater than the denominator). Function `rationalNormalize` should not only reduce to lowest terms, but also negative values should be manipulated so that the minus sign goes with the numerator. Zero is represented by zero (0) as the numerator and one (1) as the denominator.

If u and v are integers, not both zero, we say that their greatest common divisor, $\text{gcd}(u, v)$, is the largest positive integer that evenly divides both u and v . When u and v are both zero, every integer evenly divides zero, so it is convenient to set $\text{gcd}(0, 0) = 0$. When either u or v is zero, define $\text{gcd}(u, 0) = |u|$ and $\text{gcd}(0, v) = |v|$.

A sample `main` function for testing these functions is shown in Figure 2. A sample execution sequence is shown in Figure 3. To use the `Makefile` as distributed in class, add a target of `lab57` to `targets2srcfiles`.

```
1 #include <iostream>
2 #include <lab57.h>
3
4 using namespace std;
5
6 int main()
7 {
8     Rational rational;
9
10    while (rationalInput(cin, rational))
11    {
12        if (rationalValid(rational))
13        {
14            rationalOutput(cout, rational);
15            cout << " = ";
16            rationalNormalize(rational);
17            rationalOutput(cout, rational);
18            cout << endl;
19        }
20        else
21        {
22            rationalOutput(cout, rational);
23            cout << " is invalid " << endl;
24        }
25    }
26
27    return 0;
28 }
```

Figure 2. `/usr/local/1337/src/lab57main.C`

```
1 newuser@csunix ~> cd 1337
2 newuser@csunix ~/1337> mkdir 57
3 newuser@csunix ~/1337> cd 57
4 newuser@csunix ~/1337/57> cp /usr/local/1337/data/57/* .
5 newuser@csunix ~/1337/57> cp /usr/local/1337/include/lab57.h .
6 newuser@csunix ~/1337/57> cp /usr/local/1337/src/lab57main.C .
7 newuser@csunix ~/1337/57> cp /usr/local/1337/src/Makefile .
8 newuser@csunix ~/1337/57> touch lab57.cpp
9 newuser@csunix ~/1337/57> # Edit Makefile and lab57.cpp
10 newuser@csunix ~/1337/57> make lab57
11 g++ -g -Wall -std=c++11 -c lab57main.C -I/usr/local/1337/include -I.
12 g++ -g -Wall -std=c++11 -c lab57.cpp -I/usr/local/1337/include -I.
13 g++ -o lab57 lab57main.o lab57.o -L/usr/local/1337/lib -lm -lbits
14 newuser@csunix ~/1337/57> cat 01.dat
15 1651 1079
16 -3 4
17 3 4
18 0 5
19 0 7
20 1 1
21 5 4
22 25 45
23 8 99
24 3 -2
25 -1 -4
26 361 -551
27 1 0
28 0 0
29 newuser@csunix ~/1337/57> cat 01.dat | ./lab57
30 1651/1079 = 127/83
31 -3/4 = -3/4
32 3/4 = 3/4
33 0/5 = 0/1
34 0/7 = 0/1
35 1/1 = 1/1
36 5/4 = 5/4
37 25/45 = 5/9
38 8/99 = 8/99
39 3/-2 = -3/2
40 -1/-4 = 1/4
41 361/-551 = -19/29
42 1/0 is invalid
43 0/0 is invalid
44 newuser@csunix ~/1337/57> cat 01.dat | ./lab57 > my.out
45 newuser@csunix ~/1337/57> diff 01.out my.out
46 newuser@csunix ~/1337/57> cat 05.dat | ./lab57 > my.out
47 newuser@csunix ~/1337/57> diff 05.out my.out
48 newuser@csunix ~/1337/57>
```

Figure 3. Commands to Compile, Link, & Run Lab 57