

CIS 230 – Spring 2006
Homework #7
Due Thursday, March 30, 2006 at 5:00 PM

Submit your files using the submission tool on the qs-server:
`/class-files/gilden/230submit`

Files to Submit:

- `fraction.h` (unmodified)
- `fraction.cpp` (with your modifications)
- `hw7a.cpp` (unmodified)
- `hw7b.cpp` (your modified version of `hw7a.cpp`)

The files `fraction.h`, `fraction.cpp` and `hw7a.cpp` are in my `class-files` directory on qs-server.

You may copy them to your current working directory by typing:

```
>cp /class-files/gilden/hw07/* .
```

(note the “.” at the end (after a space)– this tells the copy command to place all of the files you YOUR current working directory)

Do not modify the `fraction.h` file!! Do not modify `hw7a.cpp`.

Assignment:

The file `fraction.h` contains the declaration for the class *Fraction*:

The file `hw7a.cpp` contains several invocations upon member functions for *Fraction* objects.

The file `fraction.cpp` contains the beginnings of the class definition. It has the constructor functions and a few member/friend functions defined, as well as all the friend function headers. However, it leaves the friend function bodies for you to write!

Your job is to write the *Fraction* class friend functions `subtract()` and `multiply()`.

Once you have tested and compiled `hw7a.cpp` and `Fraction.cpp`, make a copy of `hw7a.cpp` to a new file `hw7b.cpp` by typing:

```
> cp hw7a.cpp hw7b.cpp
```

Now, modify `hw7b.cpp` to use operator functions for addition, subtraction, and multiplication (for example, instead of `z = add(x,y)`; you would write `z = x + y`). Also, write the friend functions `operator-()`, and `operator*()` in `fraction.cpp`

```

class Fraction
{
    private:
        int num, den;
    public:
        Fraction(int=0, int=1);
        void reduce();
        float decimal();
        void reveal(int &, int &);
        friend Fraction add(Fraction, Fraction);
        friend Fraction operator+(Fraction, Fraction);
        friend Fraction subtract(Fraction, Fraction);
        friend Fraction operator-(Fraction, Fraction);
        friend Fraction multiply(Fraction, Fraction);
        friend Fraction operator*(Fraction, Fraction);
};

#include <iostream>
#include "fraction.h"
using namespace std;

int main()
{
    int i, j;
    Fraction x(1,2);
    x.reveal(i, j);
    cout << "Fraction Addition: " << endl;
    cout << "Fraction x is " << i << "/" << j << endl;
    Fraction y(3,4);
    y.reveal(i, j);
    cout << "Fraction y is " << i << "/" << j << endl;
    Fraction z;

    z = add(x, y);
    z.reduce();

    z.reveal(i, j);
    cout << "Added together we get: " << i << "/" << j << endl << endl;

    z = subtract(x, y);
    z.reduce();
    z.reveal(i, j);
    cout << "Fraction Subtraction: " << endl;
    cout << "Subtracting y from x we get: " << i << "/" << j << endl;
    cout << endl;

    z = multiply(x, y);
    z.reduce();
    z.reveal(i, j);
    cout << "Fraction Multiplication: " << endl;
    cout << "Multiplying x times y we get: " << i << "/" << j << endl;
    cout << endl;
    return 0;
}

```