Lab 5.3.6 Modelling fractions: part 2

Objectives

Familiarize the student with:

- interactions between objects of the same type;
- providing meaningful and helpful representations of objects.

Scenario

Let's continue working on our fraction representations.

Here's what we'll be working on in this lab:

- the ability to add, subtract, multiply and divide fractions;
- reducing fractions to their lowest terms.

Your program should read two fractions, and then perform all the basic operations on that pair.

When displaying fractions, they should now be reduced to their lowest terms, so "2/4" should be replaced with "1/2".

To reduce the terms, it might be useful to first read about the Euclidean algorithm for finding the greatest common divisor.

```cpp
#include <iostream>
#include <string>

using namespace std;

class Fraction{
public:
  Fraction(int numerator, int denominator);
  string toString();
  double toDouble();
  // the functions below do not change the object
  // on which they are called, they produce a new object
  Fraction plus(Fraction that);
  Fraction minus(Fraction that);
  Fraction times(Fraction that);
  Fraction by(Fraction that);
private:
  int numerator;
  int denominator;
  void reduce();
};

Fraction::times(Fraction that)
{
  int num = this->numerator * that.denominator;
  int den = this->denominator * that.denominator;
  Fraction result(num, den);
  // we could call result.reduce() here
  // or use it just before it is necessary
  return result;

// implement other Fraction methods
```

Example input

```
3 / 4
1 / 3
```

Example output

```
3/4 + 1/3 = 1 1/12
3/4 - 1/3 = 5/12
3/4 * 1/3 = 1/4
3/4 / 1/3 = 2 1/4
```

Example input

```
7 / 4
-6 / 10
```

Example output

```
1 3/4 + -6/10 = 1 3/20
1 3/4 - -6/10 = 2 7/10
1 3/4 * -6/10 = -1 1/20
1 3/4 / -6/10 = 2 11/12
```