MTRE 2610 – Engineering algorithms and visualization – Dr. Kevin McFall

Homework – Operator overloading

1. Create a class for rational numbers using the following class prototype.

```
class Rational {
	int numer, denom;
public:
	Rational();
	Rational(int num, int den);
	Rational operator+(Rational r);
	Rational operator-(Rational r);
	Rational operator*(Rational r);
	Rational operator/(Rational r);
	Rational operator+(int x);
	Rational operator-(int x);
	Rational operator*(int x);
	Rational operator/(int x);
	int getNumer();
	int getDenom();
	void reduce();
};

ostream& operator<<(ostream& out, Rational r);
```

Each overloaded operator should end by calling `reduce` which reduces the fraction to its simplest form. Test the class implementation with the following code.

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

int main() {
	Rational a(1,2), b(5,2), c(1,4), d(5,3), e(3,2);
	cout << (a+1) + b*(c-2)*3 - d/e/2 << endl;
	return 0;
}
```