**Name**

**Advanced Programming in C++**

**Lab Exercise 3/3/2020 Stardate: 73169.40**

In this exercise you will build a Fraction class. The Fraction class has the following:

1. Two private integer members; numerator and denominator.
2. A private double member that contains the decimal equivalent of the fraction.
3. A public get and set methods for all private fields.
4. It must have member function for add, subtract, multiply, and divide.
5. It also must have a member function reciprocate that takes the object and swaps the numerator and denominator as well as updating the decimal equivalent field.
6. The Fraction class also has a constructor that initializes the private members of the constructed object to some safe value. Initializing the denominator to 0 is not a real good idea.
7. It also must have a constructor that initializes the Fraction object to some program specified values ( i.e. Fraction one(2, 3) )
8. It must have a reduce member function that will reduce the fraction using Euclid’s Algorithm to find the GCD.

function gcd(a, b)

while b ≠ 0

t := b

b := a mod b

a := t

return a

1. It must have a print member function to print the fraction in the form n/d.

Be sure to write a main driver program to thoroughly test your class.

Print the source code for your class implementation as well as your main driver program and attach to this sheet to turn in.

Here is a copy of my fraction.h class definition file.

#ifndef FRACTION\_H

#define FRACTION\_H

class Fraction

{

private:

int numerator;

int denominator;

double decimalEquivalent;

void reduce();

void calcDecimalEquivalent();

int gcd();

public:

Fraction();

Fraction(int, int);

Fraction add(Fraction);

Fraction subtract(Fraction);

Fraction multiply(Fraction);

Fraction divide(Fraction);

void reciprocate();

int getNumerator();

int getDenominator();

double getDecimalEquivalent();

void setNumerator(int);

void setDenominator(int);

void print();

};

#endif