Problem Set 06 - Methods

Complete each task below. Remember to include all header files in the accompanying cpp file. Additionally, the accompanying header file contains the function `GCD()` that returns the greatest common divisor of two integers.

Tasks:

1. Create a header file named `"Currency.h"` and define the class *Currency.h* within the namespace *PS6*. The class must contain

   - ☐ a public double field named *value*.
   - ☐ a public default constructor that assigns 0 to *value*.
   - ☐ a public overloaded constructor that takes a double parameter and assigns the parameter to *value*.
   - ☐ a public copy constructor.
   - ☐ a public assignment operator.
   - ☐ a public empty destructor.
   - ☐ a public string constant method named `ToString()` that takes no parameters. It returns a string in the format
     $$\begin{cases} \texttt{\$}\,y & \text{if } x \geq 0 \\ \texttt{\$(}y\texttt{)} & \text{if } x < 0 \end{cases}$$
     where x is the value of the *value* field and y is the absolute value of x with two decimal places.
   - ☐ an ostream operator that returns an outcome in the same format as `ToString()`.

   Afterward, within the accompanying cpp file, within the main function, initialize two *Currency* objects such that one is assigned a positive value and the other is assigned a negative value; and then, display both objects.

2. Create a header file named `"Boolean.h"` and define the class *Boolean* within the namespace *PS6*. The class must contain

   - ☐ a public bool field named *value*.
   - ☐ a public default constructor that assigns false to *value*.
   - ☐ a public overloaded constructor that takes a bool parameter and assigns the parameter to *value*.
   - ☐ a public copy constructor that performs a deep copy.
   - ☐ a public assignment operator that performs a deep copy.
   - ☐ a public empty destructor.
   - ☐ a public string constant method named `ToString()` that takes no parameters. It returns a string in the format
     $$\begin{cases} \texttt{true} & \text{if } x = \texttt{true} \\ \texttt{false} & \text{if } x = \texttt{false} \end{cases}$$
     where x is the value of the *value* field.
   - ☐ an ostream operator that returns an outcome in the same format as `ToString()`.

   Afterward, within the accompanying cpp file, within the main function, initialize two *Boolean* objects such that one is assigned `true` and the other is assigned `false`; and then, display both objects.

3. Create a header file named `"Rational.h"` and define the class *Rational* within the namespace *PS6*. The class must contain

   - ☐ a public int array field named *values* with a size of 2.
   - ☐ a public default constructor that assigns {0,1} to *values*.
   - ☐ a public overloaded constructor that takes two int parameters and assigns the parameters to the elements of *values* in order.
   - ☐ a public copy constructor.
   - ☐ a public assignment operator.
   - ☐ a public empty destructor.
   - ☐ a public string constant method named `ToString()` that takes no parameters. It returns a string in the format

$$\begin{cases} x\texttt{/}y & \text{if } |y| = 1 \\ \texttt{undef} & \text{if } y = 0 \\ \texttt{0} & \text{if } x = 0 \text{ and } y \neq 0 \\ a\texttt{/}b & \text{if } x \cdot y > 0 \\ \texttt{-}a\texttt{/}b & \text{if } x \cdot y < 0 \end{cases}$$

     where x and y are the value of the first and second elements of the *values* field respectively, a is the absolute value of x divided by the gcd of the absolute values of x and y, and b is the absolute value of y divided by the gcd of the absolute values of x and y.
   - ☐ an ostream operator that returns an outcome in the same format as `ToString()`.

   Afterward, within the accompanying cpp file, within the main function, initialize five *Rational* objects such that each will produce a different outcome format; and then, display all objects.