Lab 5

1. Create a class PositiveInteger that represents a positive integer. Implement an overloaded unary operator - to perform negation on a positive integer and return its negated value. The PositiveInteger class should ensure that the value remains positive at all times, even after negation. Use friend function to overload unary operator -.
2. Create a class TimeCounter to represent a time counter in seconds. Implement the prefix and postfix increment operators ++ to increase the time counter by one second. The class provides separate functions for prefix and postfix increment operations to showcase the difference in behavior between the operators. The example usage demonstrates the time counter's functionality with appropriate test cases, displaying the updated time counter after each increment operation.
3. You are given a Rational class that represents rational numbers (fractions). The class has two private member variables: numerator (to store the numerator) and denominator (to store the denominator) of the rational number. Your task is to overload the binary operators +, -, *, and / for this class to perform arithmetic operations with rational numbers. Make sure to simplify the rational numbers after performing arithmetic operations to represent them in their simplest form (i.e., with the smallest possible integer numerator and denominator). You may assume that the input rational numbers will have valid values, and the denominator will never be zero.
4. Write a program to compare magnitude of complex numbers by overloading <,>,== and != operators