D1265315

In the head and file, I defines a Complex class to represent complex numbers, including constructors, getter/setter methods, and overloaded operators for arithmetic operations, comparison, and stream input/output. This allows natural handling of complex numbers in C++ using familiar operators like +, -, *, and /.

In the overloading.cpp file, I implements the complex class methods and overloaded operators. It defines constructors, arithmetic operations, comparison operators, and stream input/output handling for complex numbers, enabling natural manipulation and display of complex numbers.

The verifier is the main program that uses the `Complex` class to solve quadratic equations of the form \(ax^2 + bx + c = 0\). The program performs the following steps:

1. **Input Coefficients**: It prompts the user to enter the coefficients \(a\), \(b\), and \(c\). If \(a\) is zero, it repeatedly asks for valid input since a quadratic equation requires \(a \neq 0\).

2. **Calculate Discriminant**: It calculates the discriminant \(b^2 - 4ac\) to determine the nature of the roots (real or complex).

3. **Compute Roots**: Based on the discriminant:
    - If the discriminant is non-negative, it computes two real roots.
    - If the discriminant is negative, it computes two complex roots.

4. **Output Results**: It prints the quadratic equation and the calculated roots.

5. **Verification**: It verifies the roots by plugging them back into the original equation and checking if the results are close to zero, indicating correctness.

Overall, this program demonstrates solving quadratic equations using complex arithmetic and verifies the solutions' accuracy.