

The techniques used in each of the files:

1.complex_overloading.h:

Class Declaration:

Declares the Complex class with private data members and public member functions to represent complex numbers.

Operator Overloading:

Overloads various operators such as arithmetic operators (+, -, *, /), comparison operators (==, !=), and compound assignment operators (=, +=, -=) as member functions or friend functions.

Friend Functions:

Declares friend functions for non-member operator overloads to enable operations between a complex number object and another complex number object or a scalar value.

Stream I/O Overloading:

Declares friend functions for stream input (>>) and output (<<) to enable easy input and output of complex numbers using stream objects.

Encapsulation:

Uses private data members and member functions to encapsulate the complex number's real and imaginary parts, providing controlled access to the data.

2.complex_overloading.cpp:

Constructor Implementation:

Implements constructors for the Complex class to initialize complex numbers with given real and imaginary parts.

Member Function Definitions:

Defines member functions of the Complex class, including getters, setters, absolute value calculation, unary minus operation, and compound assignment operators.

Friend Function Definitions:

Defines friend functions for operator overloading and stream I/O operations declared in the header file.

3.quadratic_equation_verifier.cpp:

Input/Output Operations:

Uses input/output stream operations (cin, cout) to interactively input coefficients of a quadratic equation and output the roots.

Mathematical Calculations:

Calculates the discriminant of the quadratic equation to determine the nature of its roots (real or complex).

Conditional Statements:

Uses conditional statements to handle different cases based on the discriminant value (real or complex roots).

Complex Number Usage:

Utilizes the Complex class and its operations to handle complex roots when the discriminant is negative.

Function Calls:

Calls the solveQuadratic function to solve the quadratic equation and verify its roots, printing the results to the console.

These techniques collectively enable the implementation of complex number operations, quadratic equation solving, and verification functionalities in the respective files.