

Project - 1 MA517M-Basic Programming Laboratory Last Date: 14 November 2026
Name Roll No.: MA26M020

Complex Number Arithmetic using C++ Classes and Operator Overloading

Objective: To design a C++ program that implements complex numbers using classes, supports arithmetic operations, and demonstrates operator overloading.

A complex number is of the form $z = a + bi$, where $a, b \in \mathbb{R}$ and $i = \sqrt{-1}$. Arithmetic operations on complex numbers include addition, subtraction, multiplication, division, conjugation, and computation of modulus and argument.

Problem Description

Design a class `Complex` that represents complex numbers. The class should support basic arithmetic, comparisons, and display using operator overloading.

Class Specification

- **Class Name:** `Complex`
- **Private Data Members:**
 - `double real;` // real part
 - `double imag;` // imaginary part
- **Public Member Functions:**
 - `Complex(double r=0, double i=0);` // constructor
 - `double modulus() const;` // computes $|z|$
 - `double argument() const;` // computes $\arg(z)$
 - `Complex conjugate() const;` // returns $\bar{z}$
 - `void display() const;` // prints $a + bi$

Operator Overloading

- `operator +(), -, *, /` Supports complex addition, subtraction, multiplication, and division
- `operator ==()` Checks the equality of two complex numbers
- `operator <<()` Prints a complex number in the form $a + bi$

Tasks

1. Create complex numbers $z_1 = 3 + 4i$ and $z_2 = 1 - 2i$
2. Compute $z_1 + z_2$, $z_1 - z_2$, $z_1 * z_2$, z_1 / z_2
3. Compute modulus and argument of z_1 and z_2
4. Find the conjugate of z_1 and z_2
5. Compare z_1 and z_2 using the overloaded equality operator

Expected Output Example

$z_1 = 3 + 4i$

$z_2 = 1 - 2i$

$z_1 + z_2 = 4 + 2i$

$z_1 - z_2 = 2 + 6i$

$z_1 * z_2 = 11 - 2i$

$z_1 / z_2 = -1 + 2i$

Modulus $|z_1| = 5$

Argument $\arg(z_1) = 0.9273 \text{ rad}$

Conjugate of $z_1 = 3 - 4i$

$z_1 == z_2 : \text{False}$

Project - 2 : Word Shuffle Game Using C++ Classes

Problem Statement

Design and implement a **Word Shuffle Game** using **C++ classes**. The program should allow the user to unscramble letters to form meaningful words. The project should utilize object-oriented programming concepts such as classes, objects, encapsulation, and methods to handle word selection, shuffling, and user interaction.

Project Requirements

1. Create a **Word** class to represent a word in the game.
 - (a) Include a method to randomly select a word from a predefined list.
 - (b) Include a method to shuffle the letters of the word.
 - (c) Include a method to display the shuffled word to the user.
2. Create a **Game** class to manage gameplay.
 - (a) Display options for the user: **Play** or **Solution**.

- (b) If the user chooses **Play**, allow them to enter guesses.
 - (c) Validate user input and provide feedback if the guess is correct or incorrect.
 - (d) Keep track of the number of attempts.
 - (e) Allow the user to quit by entering a special command (e.g., **Q/q**) and confirm before exiting.
3. If the user chooses **Solution**, display the original word and the correct sequence of letters.
 4. Ensure proper encapsulation of word logic and gameplay operations within the respective classes.

Suggested Class Structure

1. **Word Class**:

- Data member: `string originalWord`, `string shuffledWord`
- Methods: `selectWord()`, `shuffleWord()`, `displayWord()`

2. **Game Class**:

- Data member: `Word` object, user choice, attempt counter
- Methods: `playGame()`, `showSolution()`, `processGuess(string guess)`, `confirmQuit()`