
Project - 1 MA517M-Basic Programming Laboratory	Last Date: 14 November 2026
Name	Roll No.: MA26M108

Split-Complex Numbers using C++ Classes and Operator Overloading

Objective: To design a C++ program that implements split-complex numbers using classes, supports arithmetic operations, demonstrates operator overloading, and explores the algebraic properties of a number system in which the imaginary-like unit satisfies $j^2 = 1$.

A split-complex number, also called a hyperbolic number, is a number of the form

$$z = a + bj, \quad a, b \in \mathbb{R},$$

where the hyperbolic unit j satisfies

$$j^2 = 1.$$

This differs from the usual complex number, for which

$$i^2 = -1.$$

For two split-complex numbers

$$z_1 = a + bj, \quad z_2 = c + dj,$$

addition and multiplication are defined using the rule $j^2 = 1$.

Problem Description

Design a class `SplitComplex` that represents split-complex numbers of the form

$$z = a + bj.$$

The class should support addition, subtraction, multiplication, conjugation, norm calculation, equality comparison, and display using operator overloading.

Students should also investigate an important difference from ordinary complex numbers: some non-zero split-complex numbers can have zero norm and can act as **zero divisors**.

Class Specification

- **Class Name:** SplitComplex
- **Private Data Members:**
 - double real;
 - double hyperbolic;
- **Public Member Functions:**
 - SplitComplex(double a=0, double b=0);
 - SplitComplex conjugate() const;
 - double norm() const;
 - void display() const;

Split-Complex Arithmetic

For

$$z_1 = a + bj$$

and

$$z_2 = c + dj,$$

use

$$j^2 = 1.$$

Addition

$$z_1 + z_2 = (a + c) + (b + d)j.$$

Subtraction

$$z_1 - z_2 = (a - c) + (b - d)j.$$

Multiplication

Using $j^2 = 1$,

$$\begin{aligned} z_1 z_2 &= (a + bj)(c + dj) \\ &= ac + adj + bcj + bdj^2 \\ &= (ac + bd) + (ad + bc)j. \end{aligned}$$

Therefore,

$$\boxed{(a + bj)(c + dj) = (ac + bd) + (ad + bc)j}$$

Note that multiplication is commutative:

$$z_1 z_2 = z_2 z_1.$$

Conjugate

The conjugate of

$$z = a + bj$$

is

$$\boxed{\bar{z} = a - bj.}$$

Norm

For split-complex numbers, define the quadratic norm by

$$\boxed{N(z) = a^2 - b^2.}$$

Equivalently,

$$z\bar{z} = (a + bj)(a - bj) = a^2 - b^2.$$

Unlike the ordinary complex modulus, the split-complex norm can be:

$$N(z) > 0, \quad N(z) < 0, \quad \text{or} \quad N(z) = 0.$$

Operator Overloading

The following operators should be overloaded:

- `operator +()` Split-complex addition
- `operator -()` Split-complex subtraction
- `operator *()` Split-complex multiplication
- `operator ==( ), !=()` Checks equality or inequality
- `operator <<()` Displays the split-complex number in standard form

The following member functions should also be implemented:

- `conjugate()` — returns the conjugate
- `norm()` — returns $a^2 - b^2$

Tasks

1. Create two split-complex numbers

$$z_1 = 3 + 2j$$

and

$$z_2 = 4 - 5j.$$

2. Compute

$$z_1 + z_2, \quad z_1 - z_2.$$

3. Compute

$$z_1 z_2$$

using the overloaded multiplication operator.

4. Compute the conjugates

$$\overline{z_1}, \quad \overline{z_2}.$$

5. Compute the norms

$$N(z_1), \quad N(z_2).$$

6. Verify the identity

$$z\overline{z} = N(z)$$

for z_1 and z_2 .

7. Verify that

$$N(z_1 z_2) = N(z_1)N(z_2).$$

8. Compare z_1 and z_2 using the overloaded `==` and `!=` operators.

9. Display all results using the overloaded `<<` operator.

Expected Output Example

```
z1 = 3 + 2j
```

```
z2 = 4 - 5j
```

```
z1 + z2 = 7 - 3j
```

```
z1 - z2 = -1 + 7j
```

```
z1 * z2 = 2 - 7j
```

```
z1 == z2 : False
```

```
z1 != z2 : True
```

```
conjugate(z1) = 3 - 2j
```

```
conjugate(z2) = 4 + 5j
```

```
N(z1) = 5
```

$$N(z_2) = -9$$

$$N(z_1 * z_2) = -45$$

$$N(z_1) * N(z_2) = -45$$

Additional Challenge: Zero Divisors

Consider the two non-zero split-complex numbers

$$u = 1 + j$$

and

$$v = 1 - j.$$

Compute their product:

$$\begin{aligned} uv &= (1 + j)(1 - j) \\ &= 1 - j + j - j^2 \\ &= 1 - 1 \\ &= 0. \end{aligned}$$

Thus,

$$(1 + j)(1 - j) = 0$$

even though

$$1 + j \neq 0, \quad 1 - j \neq 0.$$

This demonstrates that the split-complex numbers contain **zero divisors**.

Students should verify this result using the overloaded **operator** `*`.

Further Challenge: Division

For

$$z = a + bj,$$

if

$$N(z) = a^2 - b^2 \neq 0,$$

then the multiplicative inverse is

$$z^{-1} = \frac{\bar{z}}{N(z)} = \frac{a - bj}{a^2 - b^2}.$$

Extend the class by implementing

inverse()

and then overload

operator $/()$.

For example,

$$z = 3 + 2j$$

has

$$N(z) = 3^2 - 2^2 = 5,$$

so

$$z^{-1} = \frac{3 - 2j}{5} = \frac{3}{5} - \frac{2}{5}j.$$

Verify that

$$zz^{-1} = 1.$$

Conceptual Questions

1. What is the difference between split-complex numbers and ordinary complex numbers?
2. What changes in the multiplication formula when

$$j^2 = 1$$

instead of

$$i^2 = -1?$$

3. Why can the norm

$$N(a + bj) = a^2 - b^2$$

be negative?

4. Find a non-zero split-complex number whose norm is zero.
5. Explain why

$$(1 + j)(1 - j) = 0$$

does not contradict the usual rule that a product is zero only when one of the factors is zero.

6. For which split-complex numbers does an inverse exist?
7. Verify

$$N(z_1 z_2) = N(z_1)N(z_2).$$

8. Compare the algebraic properties of

$$a + bi, \quad a + bj.$$

Optional Extension: Idempotent Representation

Define

$$e_+ = \frac{1+j}{2}, \quad e_- = \frac{1-j}{2}.$$

Show that

$$e_+^2 = e_+, \quad e_-^2 = e_-, \quad e_+e_- = 0.$$

Then show that every split-complex number can be written as

$$a + bj = (a + b)e_+ + (a - b)e_-.$$

This provides an interesting connection between split-complex numbers, algebraic structures, and zero divisors.

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()`