|  |  |
| --- | --- |
| Logo, icon  Description automatically generatedLogo, company name  Description automatically generated  Military College of Signals  National University of Sciences & Technology | |
| **Complex variable transformation** | |
| **Submitted to: Brig(R) Dr. Abdul Razzaque** | |
| **Assignment** | 1 |
| **Submission Date:** | 7March,2025 |

**Group Members Details**

|  |  |
| --- | --- |
| **Sr.** | **Name of members** |
| **1** | NC Danish Sakander |

**Assignment 01: Basic Operations on Complex Numbers Using Python**

|  |
| --- |
| **Code** |
| # **Class represents a complex number with real and imaginary parts** class ComplexNumber:  def \_\_init\_\_(self, real, imaginary):  self.real = real # **Stores the real part of the complex number**  self.imaginary = imaginary # **Stores the imaginary part of the complex number**   # **This method controls how the object is printed**  def \_\_str\_\_(self):  return f"{self.real} + {self.imaginary}i" if self.imaginary >= 0 else f"{self.real} - {-self.imaginary}i"   # **Addition of Complex Numbers**  def \_\_add\_\_(self, other):  return ComplexNumber(self.real + other.real, self.imaginary + other.imaginary)   # **Subtraction of Complex Numbers**  def \_\_sub\_\_(self, other):  return ComplexNumber(self.real - other.real, self.imaginary - other.imaginary)  # **Multiplication of Complex Numbers**  def \_\_mul\_\_(self, other):  real\_part = self.real \* other.real - self.imaginary \* other.imaginary  imaginary\_part = self.real \* other.imaginary + self.imaginary \* other.real  return ComplexNumber(real\_part, imaginary\_part)   # **Division of Complex Numbers**  def \_\_truediv\_\_(self, other):  denominator = other.real \*\* 2 + other.imaginary \*\* 2  if denominator == 0:  return "Division by zero is not allowed"  real\_part = (self.real \* other.real + self.imaginary \* other.imaginary) / denominator  imaginary\_part = (self.imaginary \* other.real - self.real \* other.imaginary) / denominator  return ComplexNumber(real\_part, imaginary\_part)   # **Get user input for a complex number**  @staticmethod  def get\_user\_input():  real = float(input("Enter the real part: "))  imaginary = float(input("Enter the imaginary part: "))  return ComplexNumber(real, imaginary)   # **Get user input** print("Enter the first complex number:") c1 = ComplexNumber.get\_user\_input()  print("Enter the second complex number:") c2 = ComplexNumber.get\_user\_input()  # **Perform operations and display results** print("\nResults:") print("Addition:", c1 + c2) print("Subtraction:", c1 - c2) print("Multiplication:", c1 \* c2) print("Division:", c1 / c2) |

**Output**A screenshot of a computer program

AI-generated content may be incorrect.