**Question - Complex Numbers (C++) - Define a class Complex to represent complex numbers with member variables for real and imaginary parts. Overload the +, -, and \* operators for complex number addition, subtraction, and multiplication.**

**Code -**

#include <iostream>

class Complex {

private:

double real;

double imag;

public:

// Constructor

Complex(double real = 0, double imag = 0) : real(real), imag(imag) {}

// Overload + operator for complex number addition

Complex operator+(const Complex& other) const {

return Complex(real + other.real, imag + other.imag);

}

// Overload - operator for complex number subtraction

Complex operator-(const Complex& other) const {

return Complex(real - other.real, imag - other.imag);

}

// Overload \* operator for complex number multiplication

Complex operator\*(const Complex& other) const {

return Complex(real \* other.real - imag \* other.imag, real \* other.imag + imag \* other.real);

}

// Overload << operator for easy printing

friend std::ostream& operator<<(std::ostream& os, const Complex& c) {

os << c.real << " + " << c.imag << "i";

return os;

}

};

int main() {

Complex c1(3, 4); // 3 + 4i

Complex c2(2, 1); // 2 + 1i

Complex sum = c1 + c2; // Complex number addition

Complex diff = c1 - c2; // Complex number subtraction

Complex prod = c1 \* c2; // Complex number multiplication

std::cout << "Sum: " << sum << std::endl;

std::cout << "Difference: " << diff << std::endl;

std::cout << "Product: " << prod << std::endl;

return 0;

}