Title/ Problem Statement: Implement a class complex which represents complex number data type. Implement the following operations. i)Constructor ii) Overloaded operator + to add two complex numbers iii)Overloaded operator \*to multiply two complex numbers iv)Overloaded << and >> to print and read complex numbers.

#include<iostream>

using namespace std;

class complex {

float real, imag;

public:

complex(float r = 0, float i = 0) {

real = r;

imag = i;

}

complex operator+ (complex &c1) {

complex c3;

c3.real = real + c1.real;

c3.imag = imag + c1.imag;

return c3;

}

complex operator\* (complex &c1) {

complex c3;

c3.real = real \* c1.real - imag \* c1.imag;

c3.imag = real \* c1.imag + imag \* c1.real;

return c3;

}

friend istream &operator>>(istream &in, complex &c2) {

cout << "Enter the real part: ";

in >> c2.real;

cout << "Enter the imaginary part: ";

in >> c2.imag;

return in;

}

friend ostream &operator<<(ostream &out, complex &c2) {

out << c2.real << "+" << c2.imag << "i";

return out;

}

};

int main() {

complex c1, c2, c3;

cout << "Enter the first complex number:" << endl;

cin >> c1;

cout << "Enter the second complex number:" << endl;

cin >> c2;

c3 = c1 + c2;

cout << "Sum: " << c3 << endl;

c3 = c1 \* c2;

cout << "Product: " << c3 << endl;

return 0;

}