

```
class Complex(object):
    def __init__(self, real, imaginary):
        self.real = real
        self.imaginary = imaginary

    def __add__(self, no):
        real_sum = self.real + no.real
        imag_sum = self.imaginary + no.imaginary
        return Complex(real_sum, imag_sum)

    def __sub__(self, no):
        real_diff = self.real - no.real
        imag_diff = self.imaginary - no.imaginary
        return Complex(real_diff, imag_diff)

    def __mul__(self, no):
        real_prod = (self.real * no.real) - (self.imaginary * no.imaginary)
        imag_prod = (self.real * no.imaginary) + (self.imaginary * no.real)
        return Complex(real_prod, imag_prod)

    def __truediv__(self, no):
        denominator = no.real**2 + no.imaginary**2
        real_quot = ((self.real * no.real) + (self.imaginary * no.imaginary)) / denominator
        imag_quot = ((self.imaginary * no.real) - (self.real * no.imaginary)) / denominator
        return Complex(real_quot, imag_quot)

    def mod(self):
        modulus = math.sqrt(self.real**2 + self.imaginary**2)
        return Complex(modulus, 0)

    def __str__(self):
        if self.imaginary == 0:
            result = "%.2f+0.00i" % (self.real)
        elif self.real == 0:
            if self.imaginary >= 0:
                result = "0.00+%.2fi" % (self.imaginary)
            else:
                result = "0.00-%.2fi" % (abs(self.imaginary))
        elif self.imaginary > 0:
            result = "%.2f+%.2fi" % (self.real, self.imaginary)
        else:
            result = "%.2f-%.2fi" % (self.real, abs(self.imaginary))
        return result
```