CSC2111 Computer Science I Lab

Lab 15

Objectives

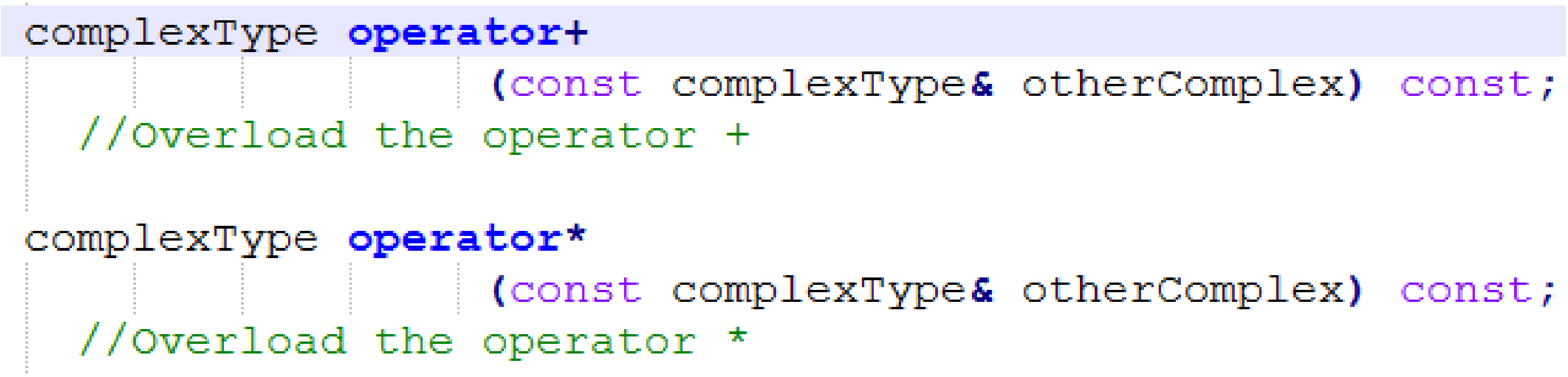
Implement operator overloading

Example

Implement operators for addition (+) and multiplication (*) for a complex number class.

Solution

Begin by adding prototypes the class declaration, then define the methods based on their mathematical operators.



```
complexType operator+
                (const complexType& otherComplex) const;
  //Overload the operator +

complexType operator*
                (const complexType& otherComplex) const;
  //Overload the operator *
```

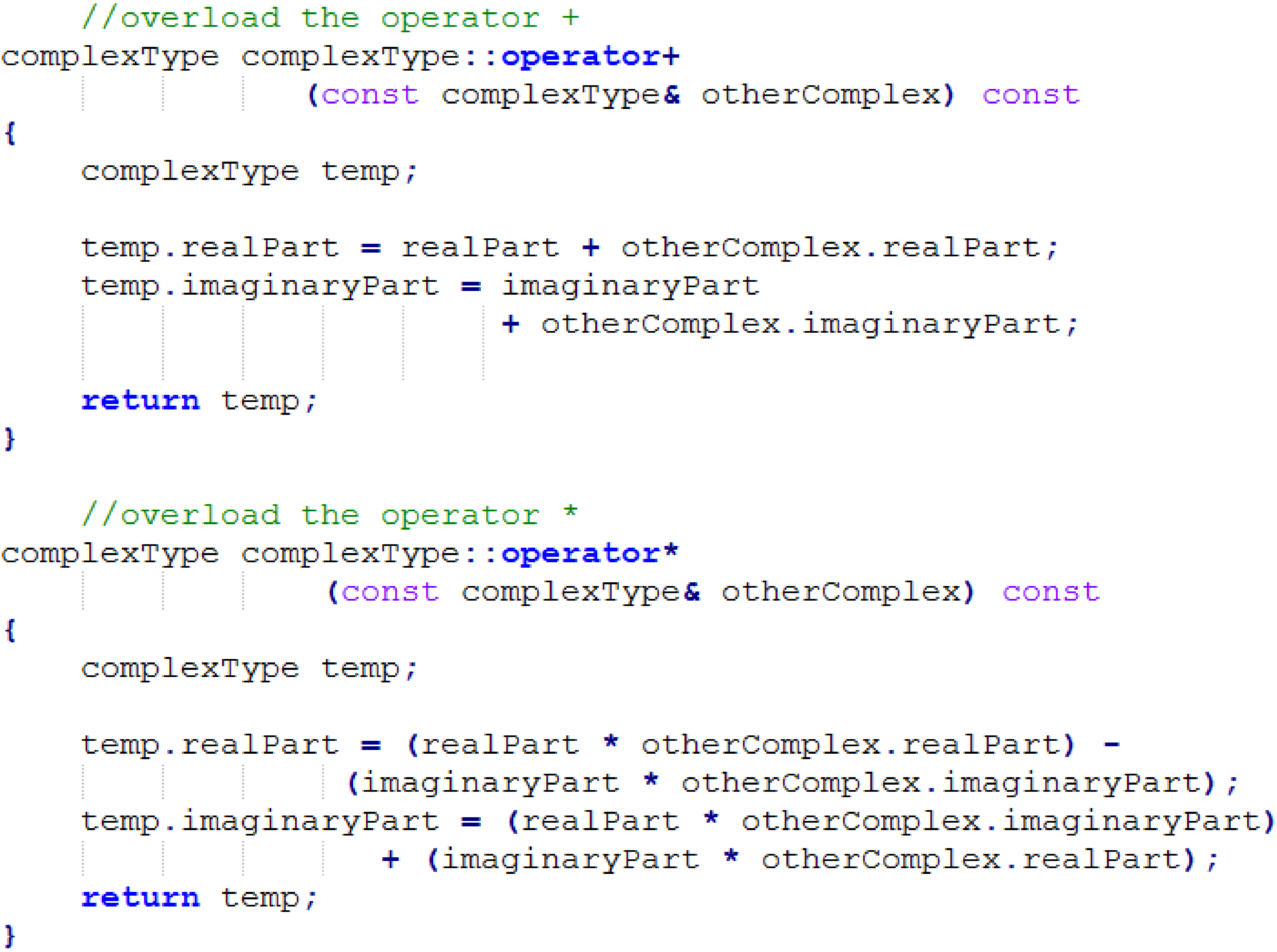
Addition of complex numbers

✓ $(a+bi)+(c+di)=(a+c)+(b+d)i$

Multiplication of complex numbers

✓ $(a+bi)\cdot(c+di)=(ac-bd)+(ad+bc)i$

Solution



```
    //overload the operator +
complexType complexType::operator+
               (const complexType& otherComplex) const
{
    complexType temp;

    temp.realPart = realPart + otherComplex.realPart;
    temp.imaginaryPart = imaginaryPart
                         + otherComplex.imaginaryPart;

    return temp;
}

    //overload the operator *
complexType complexType::operator*
                (const complexType& otherComplex) const
{
    complexType temp;

    temp.realPart = (realPart * otherComplex.realPart) -
                 (imaginaryPart * otherComplex.imaginaryPart);
    temp.imaginaryPart = (realPart * otherComplex.imaginaryPart)
                   + (imaginaryPart * otherComplex.realPart);
    return temp;
}
```