

Philip Pesic

Week 15

November 27 2022

Week 15 Prog 3

Add the ++ operator to the complex class and test.

```
//
```

```
// main.cpp
```

```
// Week 15 Prog 3
```

```
//
```

```
// Created by Pippo Pesic on 11/26/22.
```

```
//
```

```
#include <iostream>
```

```
using namespace std;
```

```
class Complex
```

```
{
```

```
    private:
```

```
        float real;
```

```
        float imag;
```

```
    public:
```

```
        Complex(): real(0), imag(0){ }
```

```
        void input()
```

```
        {
```

Philip Pesic

Week 15

November 27 2022

Week 15 Prog 3

```
cout<<"Enter real and imaginary parts respectively: ";

cin>>real;

cin>>imag;

}

Complex operator - (Complex c2)    /* Operator Function */

{

Complex temp;

temp.real=real-c2.real;

temp.imag=imag-c2.imag;

return temp;

}

Complex operator + (Complex a2)    /* Operator Function */

{

Complex temp;

temp.real=real+a2.real;

temp.imag=imag+a2.imag;

return temp;

}

Complex operator * (Complex m2)    /* Operator Function */

{
```

Philip Pesic

Week 15

November 27 2022

Week 15 Prog 3

```
Complex temp;  
  
temp.real=real*m2.real;  
  
temp.imag=imag*m2.imag;  
  
return temp;  
  
}
```

```
Complex operator / (Complex d2)    /* Operator Function */  
  
{  
  
Complex temp;  
  
temp.real=real/d2.real;  
  
temp.imag=imag/d2.imag;  
  
return temp;  
  
}
```

```
Complex operator ++ ()    /* Operator Function */  
  
{  
  
Complex temp;  
  
temp.real=real++;  
  
temp.imag=imag++;  
  
return temp;  
  
}
```

Philip Pesic

Week 15

November 27 2022

Week 15 Prog 3

```
void output()
{
    if(imag<0)
        cout<<"Output Complex number: "<<real<<imag<<"i" << endl;
    else
        cout<<"Output Complex number: "<<real<<"+"<<imag<<"i" << endl;
}

};

int main()
{
    Complex c1, c2, a1, a2, m1, m2, d1, d2, i1, i2, result;

    cout<<"Enter first complex number:\n";

    c1.input();

    cout<<"Enter second complex number:\n";

    c2.input();

    result=c1-c2; /* c2 is furnished as an argument to the operator function. */

    result.output();

    cout<<"Enter first complex number:\n";

    a1.input();
```

Philip Pesic

Week 15

November 27 2022

Week 15 Prog 3

```
cout<<"Enter second complex number:\n";  
  
a2.input();  
  
result=a1+a2; /* c2 is furnished as an argument to the operator function. */  
  
result.output();
```

```
cout<<"Enter first complex number:\n";  
  
m1.input();  
  
cout<<"Enter second complex number:\n";  
  
m2.input();  
  
result=m1*m2; /* c2 is furnished as an argument to the operator function. */  
  
result.output();
```

```
cout<<"Enter first complex number:\n";  
  
d1.input();  
  
cout<<"Enter second complex number:\n";  
  
d2.input();  
  
result=a1/d2; /* c2 is furnished as an argument to the operator function. */  
  
result.output();
```

```
cout<<"Enter first complex number:\n";  
  
i1.input();
```

Philip Pesic

Week 15

November 27 2022

Week 15 Prog 3

```
        cout<<"Enter second complex number:\n";

        i2.input();

        result=i1+i2; /* c2 is furnished as an argument to the operator function. */

        result.output();

/* In case of operator overloading of binary operators in C++ programming, the object on right
hand side of operator is always assumed as argument by compiler. */

        cout << "Philip Pesic 11/27/22" << endl;

        return 0;

}
```

Philip Pesic

Week 15

November 27 2022

Week 15 Prog 3

The screenshot shows a C++ IDE with a project named "Week 15 Prog 3". The main file, "main.cpp", contains the following code:

```
11 class Complex
31 {
32     Complex operator + (Complex a2) /* Operator Function */
33     {
34         Complex temp;
35         temp.real=real+m2.real;
36         temp.imag=imag+m2.imag;
37         return temp;
38     }
39     Complex operator * (Complex m2) /* Operator Function */
40     {
41         Complex temp;
42         temp.real=real*m2.real;
43         temp.imag=imag*m2.imag;
44         return temp;
45     }
46     Complex operator / (Complex d2) /* Operator Function */
47     {
48         Complex temp;
49         temp.real=real/d2.real;
50         temp.imag=imag/d2.imag;
51         return temp;
52     }
53 }
54
55 Complex operator ++ () /* Operator Function */
56 {
57     Complex temp;
58     temp.real=real++;
59     temp.imag=imag++;
60     return temp;
61 }
62
63 void output()
64 {
65     if(imag<0)
66         cout<<"Output Complex number: "<<real<<imag<<"i" << endl;
67     else
68         cout<<"Output Complex number: "<<real<<"+ "<<imag<<"i" << endl;
69 }
70
71 int main()
72 {
73     Complex c1, c2, a1, a2, m1, m2, d1, d2, i1, i2, result;
74     cout<<"Enter first complex number:\n";
75     c1.input();
76     cout<<"Enter second complex number:\n";
77     c2.input();
78     result=c1-c2; /* c2 is furnished as an argument to the operator function. */
79 }
```

The IDE interface includes a sidebar on the left with a file explorer showing the project structure. On the right, there is a "Properties" panel with sections for "Identity and Type", "On Demand Resource Tags", "Target Membership", and "Text Settings". The "Text Settings" section shows "Text Encoding" as "No Explicit Encoding", "Line Endings" as "No Explicit Line Endings", "Indent Using" as "Spaces", and "Wrap lines" checked.

Philip Pesic

Week 15

November 27 2022

Week 15 Prog 3

The screenshot shows a C++ IDE with a dark theme. The left sidebar displays a project tree with 'Week 15 Prog 3' and a sub-folder 'main'. The main editor window shows the source code for 'main.cpp'. The code defines a complex number structure and implements arithmetic operators. It includes comments explaining the use of the 'c2' argument in the operator functions. The output window at the bottom shows the program's execution, including prompts for entering complex numbers and the resulting complex number outputs. The IDE's right sidebar shows the 'Identity and Type' panel for 'main.cpp', indicating it is a C++ source file. The 'Text Settings' panel shows the text encoding as 'No Explicit Encoding', line endings as 'No Explicit Line Endings', and indent settings as 'Spaces' with a width of 4.

```
89 m1.input();
90 cout<<"Enter second complex number:\n";
91 m2.input();
92 result=m1+m2; /* c2 is furnished as an argument to the operator function. */
93 result.output();
94
95 cout<<"Enter first complex number:\n";
96 d1.input();
97 cout<<"Enter second complex number:\n";
98 d2.input();
99 result=a1/d2; /* c2 is furnished as an argument to the operator function. */
```

Enter first complex number:
Enter real and imaginary parts respectively: 5
5
Enter second complex number:
Enter real and imaginary parts respectively: 5
5
Output Complex number: 0+0i
Enter first complex number:
Enter real and imaginary parts respectively: 5
5
Enter second complex number:
Enter real and imaginary parts respectively: 5
5
Output Complex number: 10+10i
Enter first complex number:
Enter real and imaginary parts respectively: 5
5
Enter second complex number:
Enter real and imaginary parts respectively: 5
5
Output Complex number: 25+25i
Enter first complex number:
Enter real and imaginary parts respectively: 5
5
Enter second complex number:
Enter real and imaginary parts respectively: 5
5
Output Complex number: 14+1i
Enter first complex number:
Enter real and imaginary parts respectively: 5
5
Enter second complex number:
Enter real and imaginary parts respectively: 5
5
Output Complex number: 10+10i
Philip Pesic 11/27/22
Program ended with exit code: 0

I learned: how to use the ++ operator