

**Don Bosco Institute of Technology, Kurla(W)**  
**Department of Electronics and Tele-Communication Engineering**  
**ECL304 - Skill Lab: C++ and Java Programming**  
**Sem III**  
**2021-22**

<b>Lab Number:</b>	<b>5</b>
<b>Student Name:</b>	<b>Simran Santosh Koparkar</b>
<b>Roll No :</b>	<b>41</b>

**Title:**

To perform Operator Overloading using C++ for

- adding 2 complex numbers
- adding matrices

**Learning Objective:**

- Students will be able to perform user-defined overloading of built-in operators.

**Learning Outcome:**

- Understanding the overloading concept on built-in operators.

**Course Outcome:**

<b>ECL304.2</b>	Comprehend building blocks of OOPs language, inheritance, package and interfaces
-----------------	--

**Theory:**

Explain about operator overloading with respect to:

- constructor,
- methods and
- operators

We can change the way operators work for user-defined types like objects and structures. This is known as operator overloading.

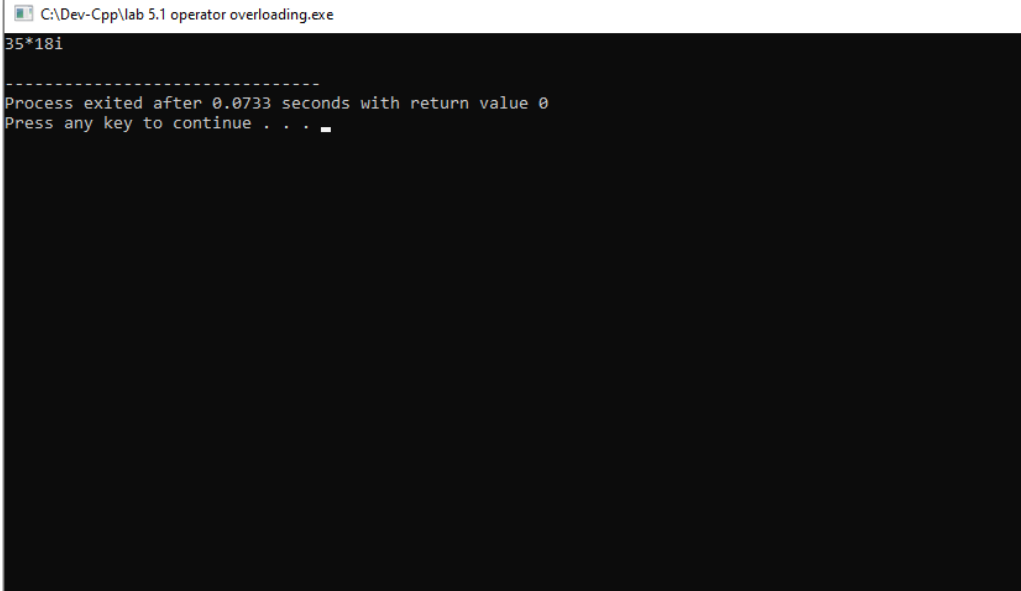
<b>Program</b> :	<pre>#include&lt;iostream&gt;  using namespace std ;  class complexno {     public :</pre>

**Faculty: Ms. Deepali Kayande**

**Don Bosco Institute of Technology, Kurla(W)**  
**Department of Electronics and Tele-Communication Engineering**  
**ECL304 - Skill Lab: C++ and Java Programming**  
**Sem III**  
**2021-22**

	<pre> int real, imag; complexno() {     real = 0;     imag = 0;  } complexno(int r, int i) {     real = r;     imag = i; } void display() {     cout &lt;&lt; real &lt;&lt; "*" &lt;&lt; imag &lt;&lt;     "i" &lt;&lt; endl; } complexno operator *(complexno c) {     complexno temp;     temp.real = real * c.real;     temp.imag =    imag * c.imag;      return temp; }  };  int main() {     complexno c3;     complexno c1(7,2);     complexno c2(5,9);     c3 = c1*c2;     c3.display();     return 0; } </pre>
<b>Input given:</b>	-

**Don Bosco Institute of Technology, Kurla(W)**  
**Department of Electronics and Tele-Communication Engineering**  
**ECL304 - Skill Lab: C++ and Java Programming**  
**Sem III**  
**2021-22**

<b>Output Screensh ot:</b>	 <p>The screenshot shows a command prompt window titled 'C:\Dev-Cpp\lab 5.1 operator overloading.exe'. The output displays '35*18i' followed by a separator line '-----'. Below the separator, it states 'Process exited after 0.0733 seconds with return value 0' and 'Press any key to continue . . . _'.</p>
------------------------------------	---

<b>Program :</b>	<pre># include&lt;iostream&gt;  using namespace std;  class matrices {  int a[2][2];  int b[2][2];  int c[2][2];  public:</pre>

**Faculty: Ms. Deepali Kayande**

**Don Bosco Institute of Technology, Kurla(W)**  
**Department of Electronics and Tele-Communication Engineering**  
**ECL304 - Skill Lab: C++ and Java Programming**  
**Sem III**  
**2021-22**

```
void get_elements(); //take numbers
from user

matrices operator +(matrices m2);
//operator overloading

void display(); //print the
result

};

//functions outside class, using scope resolution

void matrices::get_elements()
{
    cout<<"enter the elements";

    for(int i=0;i<2;i++) //for row
    {
        for(int j=0;j<2;j++) //for columns
            cin>>a[i][j];
    }
}

void matrices:: display()
{
    for(int i=0;i<2;i++)
```

**Don Bosco Institute of Technology, Kurla(W)**  
**Department of Electronics and Tele-Communication Engineering**  
**ECL304 - Skill Lab: C++ and Java Programming**  
**Sem III**  
**2021-22**

```
        {  
  
            for(int j=0;j<2;j++)  
  
                cout<<a[i][j]<<" ";  
  
            cout<<endl;  
  
        }  
  
    }  
  
    matrices matrices::operator+(matrices m2)  
  
    {  
  
        matrices m3;  
  
        for(int i=0;i<2;i++)  
  
        {  
  
            for(int j=0;j<2;j++)  
  
                m3.a[i][j]=a[i][j]+m2.a[i][j];  
  
        }  
  
        return(m3);  
  
    }  
  
    int main()  
  
    {  
  
        matrices ob1,ob2;
```

**Don Bosco Institute of Technology, Kurla(W)**  
**Department of Electronics and Tele-Communication Engineering**  
**ECL304 - Skill Lab: C++ and Java Programming**  
**Sem III**  
**2021-22**

		<pre> ob1.get_elements();  ob2.get_elements();  cout&lt;&lt;"\nMatrix 1:\n";  ob1.display();  cout&lt;&lt;"\nMatrix 2:\n";  ob2.display();  ob1=ob1+ob2;  cout&lt;&lt;"\nResult:\n";  ob1.display();  } </pre>
<b>Input given:</b>	4 3 2 1 9 8 7 6	

**Don Bosco Institute of Technology, Kurla(W)**  
**Department of Electronics and Tele-Communication Engineering**  
**ECL304 - Skill Lab: C++ and Java Programming**  
**Sem III**  
**2021-22**

**Output  
Screenshot:**

```
C:\Dev-Cpp\lab 5.1 operator overloading.exe
enter the elements
4
3
2
1
enter the elements
9
8
7
6

Matrix 1:
4 3
2 1

Matrix 2:
9 8
7 6

Result:
13 11
9 7

-----
Process exited after 35.52 seconds with return value 0
Press any key to continue . . .
```

**Faculty: Ms. Deepali Kayande**

**Don Bosco Institute of Technology, Kurla(W)**  
**Department of Electronics and Tele-Communication Engineering**  
**ECL304 - Skill Lab: C++ and Java Programming**  
**Sem III**  
**2021-22**

**Faculty: Ms. Deepali Kayande**