

Program – 3.1

AIM: Write C++ program to overload Unary and Binary operator in member function.

PROGRAM:

```
#include<iostream>

using namespace std;

class unary

{
    int x,y,z;

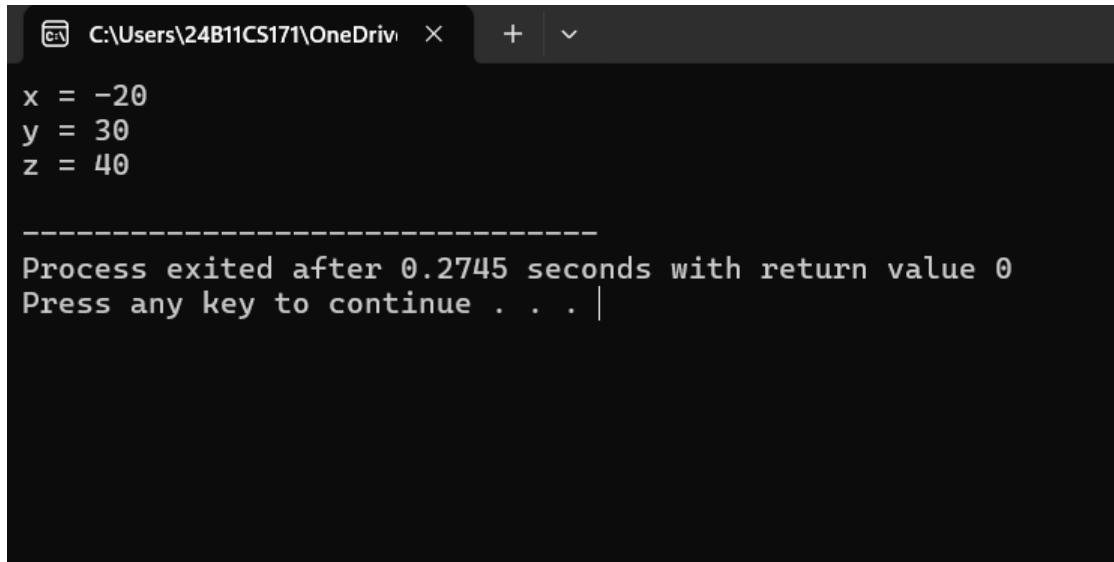
public:
    void get()
    {
        x=20;
        y=-30;
        z=-40;
    }

    void show()
    {
        cout<<"x="<<x<<endl;
        cout<<"y="<<y<<endl;
        cout<<"z="<<z;
    }

    void operator -()
    {
        x=-x;
        y=-y;
        z=-z;
    }
}
```

```
    }  
};  
  
int main()  
{  
    unary u;  
  
    u.get();  
    -u;  
    u.show();  
  
    return 0;  
}
```

OUTPUT:



```
C:\Users\24B11CS171\OneDrive  X  +  ▾  
x = -20  
y = 30  
z = 40  
-----  
Process exited after 0.2745 seconds with return value 0  
Press any key to continue . . . |
```

AIM: Write C++ program to overload Unary and Binary operator in member function.

PROGRAM:

```
#include<iostream>
using namespace std;
class complex
{
    int real,imag;
public:
    void data()
    {
        cout<<"enter real,imag values";
        cin>>real>>imag;
    }
    void operator+(complex c2)
    {
        cout<<real+c2.real<<"+ "<<imag+c2.imag<<"i";
    }
};
int main()
{
    complex c1,c2;
    c1.data();
    c2.data();
    c1+c2;
    return 0;
}
```

OUTPUT:

```
C:\Users\24B11CS171\OneDrive\ + ▾
Enter real and imaginary values: 10 20
Enter real and imaginary values: 30 40
Result = 40 + 60i

-----
Process exited after 8.588 seconds with return value 0
Press any key to continue . . . |
```

Program – 3.2

AIM: Write C++ program to overload Unary and Binary operators in friend function.

PROGRAM:

```
#include<iostream>
```

```
using namespace std;
```

```
class unary
```

```
{
```

```
    int x,y,z;
```

```
public:
```

```
    void get()
```

```
{
```

```
    x=20;
```

```
    y=-30;
```

```
    z=-40;
```

```
}
```

```
    void show()
```

```
{
```

```
        cout<<"x="<<x<<endl;
```

```
        cout<<"y="<<y<<endl;
```

```
        cout<<"z="<<z;
```

```
}
```

```
friend void operator -(unary &u);
```

```
};
```

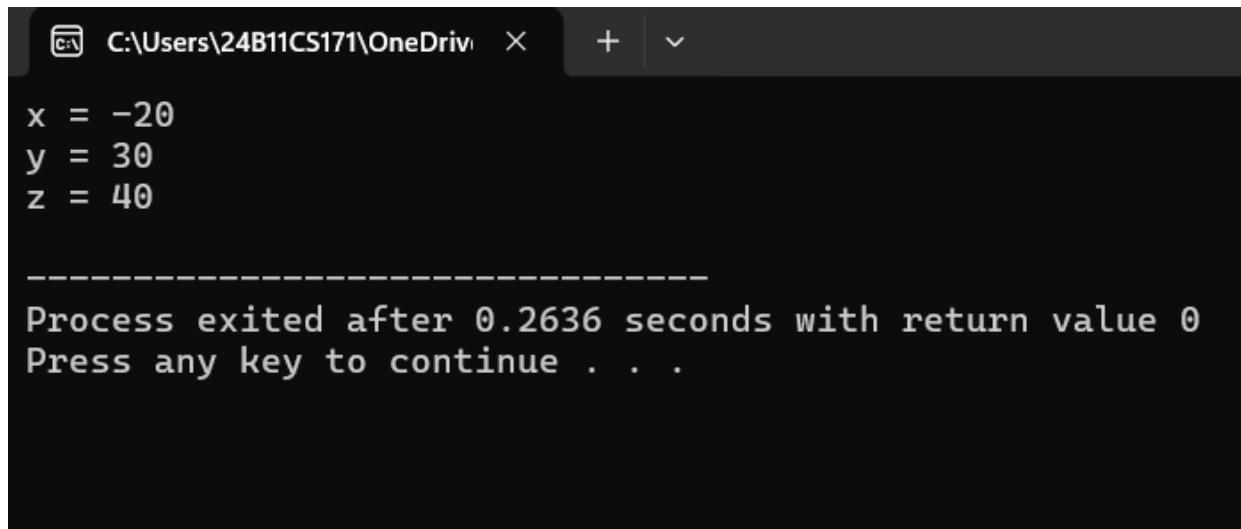
```
void operator -(unary &u)
```

```
{
```

```
    u.x=-u.x;
```

```
    u.y=-u.y;  
    u.z=-u.z;  
}  
  
int main()  
{  
    unary u;  
    u.get();  
    -u;  
    u.show();  
    return 0;  
}
```

OUTPUT:



```
C:\Users\24B11CS171\OneDrive\ + | v  
x = -20  
y = 30  
z = 40  
-----  
Process exited after 0.2636 seconds with return value 0  
Press any key to continue . . .
```

AIM: Write C++ program to overload Unary and Binary operators in friend function.

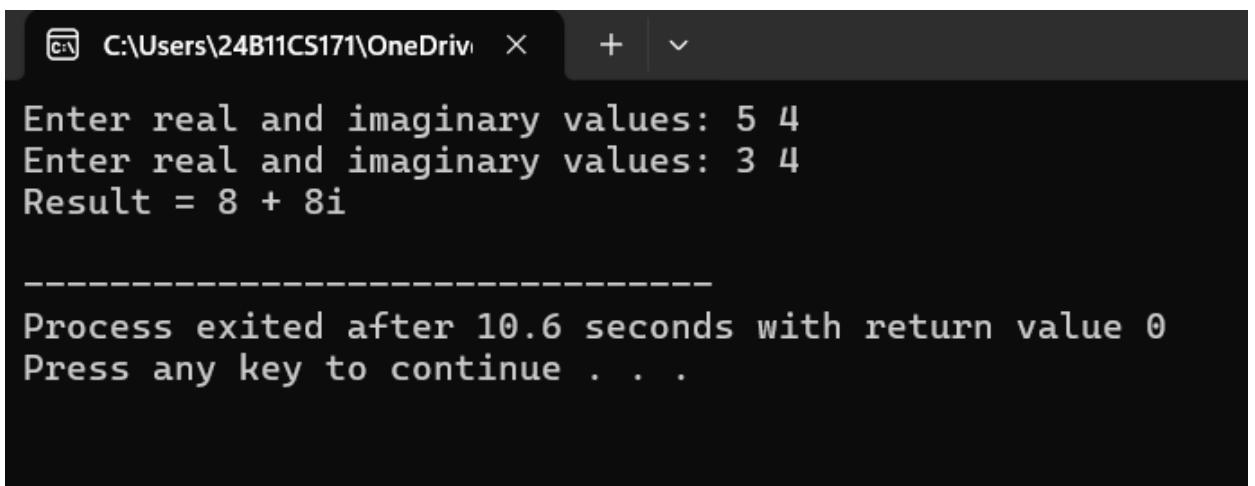
PROGRAM:

```
#include<iostream>
using namespace std;
class complex
{
    int real,imag;
public:
    void data()
    {
        cout<<"enter real,imag values";
        cin>>real>>imag;
    }
    friend void operator +(complex &c1,complex
    &c2);
};

void operator +(complex &c1,complex &c2)
{
    cout<<c1.real+c2.real<<"+"<<c1.imag+c2.imag<<"i";
}

int main()
{
    complex c1,c2;
    c1.data();
    c2.data();
    c1+c2;
```

```
    return 0;  
}  
  
OUTPUT:
```



The screenshot shows a terminal window with a dark background and light-colored text. At the top, it displays the path 'C:\Users\24B11CS171\OneDrive\'. Below this, there are three lines of user input: 'Enter real and imaginary values: 5 4', 'Enter real and imaginary values: 3 4', and 'Result = 8 + 8i'. A horizontal dashed line follows, and then the terminal outputs 'Process exited after 10.6 seconds with return value 0' and 'Press any key to continue . . .'. The window has standard operating system controls (minimize, maximize, close) at the top right.

```
Enter real and imaginary values: 5 4  
Enter real and imaginary values: 3 4  
Result = 8 + 8i  
-----  
Process exited after 10.6 seconds with return value 0  
Press any key to continue . . .
```