OOPS ASSIGNMENT

NAME: ROHAN GHSOH

<u>ID:181001001122</u>

BATCH-BCS2B

1. Create a class complex that contains two double datamembers. Overload +,-, and * arithmetic operators, so that they can operate on the object of complex. Then find the expression a-b*c + d (where a, b, c, and d are complexobjects).

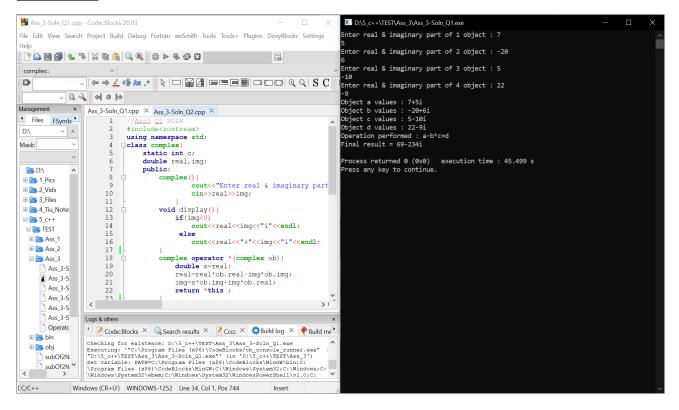
ANS:

```
#include<iostream>
using namespace std;
class complex{
  static int c;
  double real,img;
  public:
      complex(){
cout<="Enter real & imaginary part of "<<++c<=" object : ";
                           cin>>real>>img;
                    }
             void display(){
               if(img<0)
             cout<<real<<img<<"i"<<endl;
                else
cout<<real<<"+"<<img<<"i"<<endl;
             complex operator *(complex ob){
                double x=real:
```

```
real=real*ob.real-img*ob.img;
                    img=x*ob.img+img*ob.real;
                    return *this;
              }
              complex operator -(complex ob){
                    real=real-ob.real;
                    img=img-ob.img;
                    return *this;
              }
              complex operator +(complex ob){
                    real=real+ob.real;
                    img=img+ob.img;
                    return *this;
              }
};
int complex::c=0;
int main(){
      complex a,b,c,d;
      cout<< "Object a values : ";
       a.display();
       cout<< "Object b values: ";
       b.display();
      cout<< "Object c values : ";
       c.display();
      cout<< "Object d values : ";
      d.display();
       b=b*c;
       a=a-b;
```

```
d=d+a;
cout<< "Operation performed : a-b*c+d\nFinal result = ";
d.display();
};</pre>
```

OUTPUT



- 2. Create a class String then implement the following operation
- i. Overload the + operator for string concatenation.
- ii. Overload the operator for subtract the length of twostring.

ANS:

```
#include<iostream>
#include<cstring>
using namespace std;
class String{
    char str[80];
    int len;
    public:
```

```
String(){
cout<<"Enter string for concatenation:";
cin.getline(str, 80);
len=strlen(str);
         void display(){
cout<<str<<endl;
         }
         void display_len(){
cout<<"\nDifference of string : "<<len<<endl;</pre>
         }
         String operator +(String ob){
strcat(str,ob.str);
              return *this;
         }
         String operator -(String ob){
                     len=len-ob.len;
              return *this;
              }
};
//"return *this" is going to return the current class object.
//"return this" will return the object address of the current class.
int main(){
    String ob, ob1;
cout<<"\n1st String is: ";
ob.display();
cout<<"2nd String is: ";
    ob1.display();
ob=ob-ob1;
```

```
ob.display_len();
ob=ob+ob1;
cout<<"Concatenated String : ";
ob.display();
}</pre>
```

OUTPUT

