190905513 **MOHAMMAD DANISH EQBAL**

Session I

Part II

Lab No. 3: Classes and Objects

Lab Exercises

1. Define a class to represent a complex number called Complex. Provide the following methods:

i) To assign initial values to the Complex object.

ii) To display a complex number in a+ib format.

iii) To add 2 complex numbers. (the return type should be Complex)

iv) To subtract 2 complex numbers.

Write a main method to test the class.

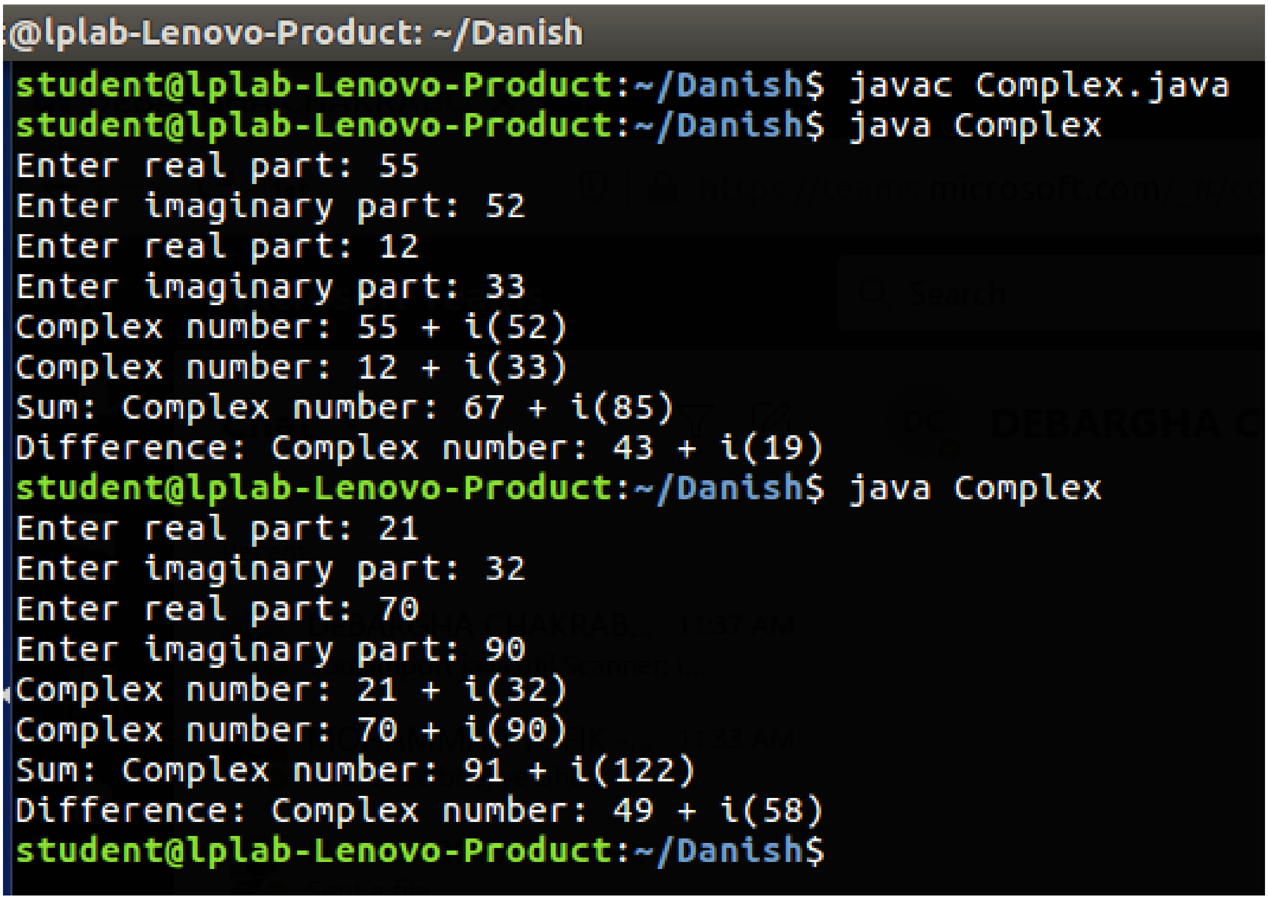
Solution:

```
import java.util.Scanner;
import java.lang.Math;

public class Complex
{
	int real, imaginary;
	public Complex(int re, int im)
	{
		this.real=re;
		this.imaginary=im;
	}

	public static Complex InitComplex()
	{
		Complex a = new Complex(0, 0);
		Scanner sc=new Scanner(System.in);
```

```
    System.out.print("Enter real part: ");
    a.real=sc.nextInt();
    System.out.print("Enter imaginary part: ");
    a.imaginary=sc.nextInt();

    return a;
}

public void display()
{
    System.out.println("Complex number: "+this.real+" + i("+this.imaginary+")");
}

public Complex add(Complex a)
{
    Complex c = new Complex(0, 0);
    c.real=a.real+this.real;
    c.imaginary=a.imaginary+this.imaginary;
    return c;
}

public Complex subtract(Complex a)
{
    Complex c = new Complex(0, 0);
    c.real=Math.abs(this.real-a.real);
    c.imaginary=Math.abs(this.imaginary-a.imaginary);
    return c;
}

public static void main(String[] arg)
{
```

```
                Complex obj1 = InitComplex();

                Complex obj2 = InitComplex();

                obj1.display();

                obj2.display();

                System.out.print("Sum: ");

                (obj1.add(obj2)).display();

                System.out.print("Difference: ");

                (obj1.subtract(obj2)).display();


        }

}
```

Output:



```
:@lplab-Lenovo-Product: ~/Danish
student@lplab-Lenovo-Product:~/Danish$ javac Complex.java
student@lplab-Lenovo-Product:~/Danish$ java Complex
Enter real part: 55
Enter imaginary part: 52
Enter real part: 12
Enter imaginary part: 33
Complex number: 55 + i(52)
Complex number: 12 + i(33)
Sum: Complex number: 67 + i(85)
Difference: Complex number: 43 + i(19)
student@lplab-Lenovo-Product:~/Danish$ java Complex
Enter real part: 21
Enter imaginary part: 32
Enter real part: 70
Enter imaginary part: 90
Complex number: 21 + i(32)
Complex number: 70 + i(90)
Sum: Complex number: 91 + i(122)
Difference: Complex number: 49 + i(58)
student@lplab-Lenovo-Product:~/Danish$
```

2. Create a class called Time that has instance variables to represent hours, minutes and seconds. Provide the following methods:

i) To assign initial values to the Time object.

ii) To display a Time object in the form of hh:mm:ss {24 hours format}

iii) To add 2 Time objects (the return type should be a Time)

iv) To subtract 2 Time objects (the return type should be a Time)

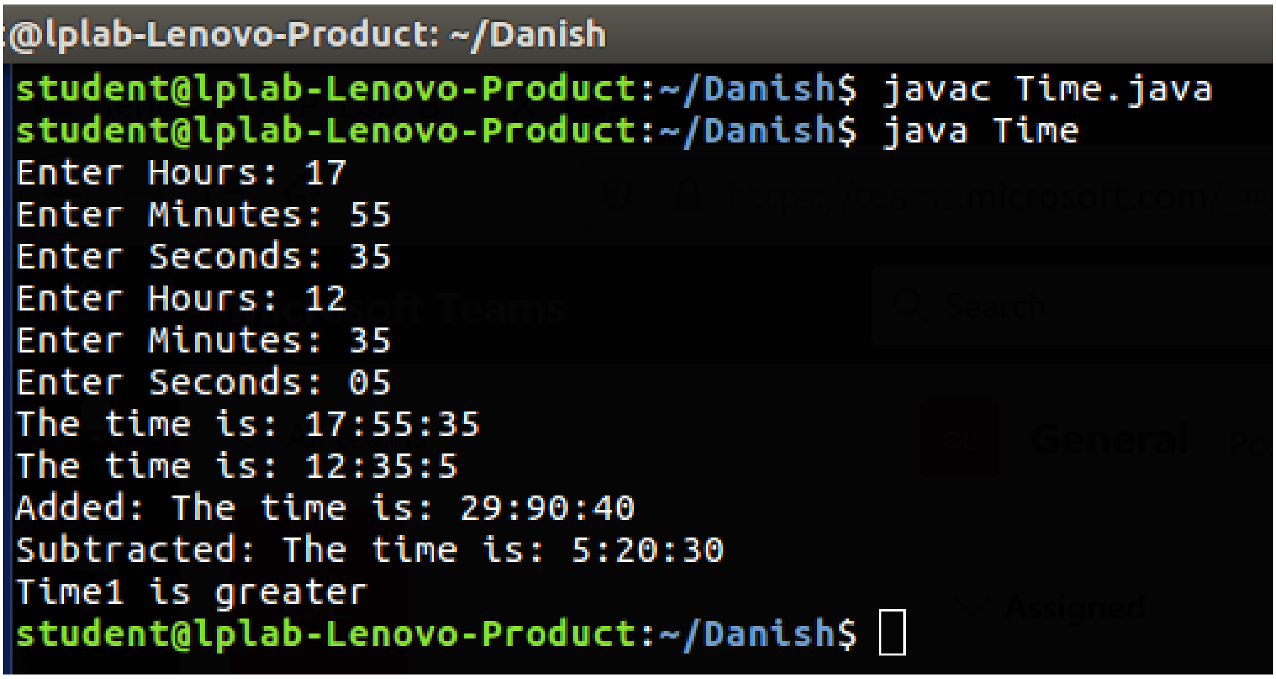
v) To compare 2 Time objects and to determine if they are equal or if the first is greater or smaller than the second one.

Solution:

```
import java.util.Scanner;

import java.lang.Math;


public class Time

{

        int hours, minutes, seconds;

        public Time(int h, int m, int s)

        {

                this.hours=h;

                this.minutes=m;

                this.seconds=s;

        }


        public static Time getTime()

        {

                Scanner sc=new Scanner(System.in);

                Time a=new Time(0, 0, 0);

                System.out.print("Enter Hours: ");

                a.hours=sc.nextInt();

                System.out.print("Enter Minutes: ");

                a.minutes=sc.nextInt();

                System.out.print("Enter Seconds: ");
        a.seconds=sc.nextInt();

        //Handling cases when seconds or minutes>60
        a.minutes+=a.seconds/60;
        a.seconds=a.seconds%60;
        a.hours+=a.minutes/60;
        a.minutes=a.minutes%60;

        return a;
    }

    public void display()
    {
        System.out.println("The time is: "+this.hours+":"+this.minutes+":"+this.seconds);
    }

    public Time add(Time a)
    {
        Time c=new Time(0, 0, 0);
        c.hours=this.hours+a.hours;
        c.minutes=this.minutes+a.minutes;
        c.seconds=this.seconds+a.seconds;

        return c;
    }

    public Time subtract(Time a)
    {
        Time c=new Time(0, 0, 0);
        c.hours=this.hours-a.hours;
        c.minutes=this.minutes-a.minutes;
        c.seconds=this.seconds-a.seconds;

        return c;
    }

    public int isGreaterThan(Time a)
    {

    if(this.hours*24+this.minutes*60+this.seconds>a.hours*24+a.minutes*60+a.seconds)
            return 1;
        else
if(this.hours*24+this.minutes*60+this.seconds==a.hours*24+a.minutes*60+a.seconds)
            return 0;
        else
            return -1;
    }

    public static void main(String arg[])
    {
        Time obj1=getTime();
        Time obj2=getTime();
        obj1.display();
        obj2.display();
        System.out.print("Added: ");
        (obj1.add(obj2)).display();
        System.out.print("Subtracted: ");
        (obj1.subtract(obj2)).display();
        //Comparing
        if(obj1.isGreaterThan(obj2)==1)
            System.out.println("Time1 is greater");
        if(obj1.isGreaterThan(obj2)==0)
```

```
                        System.out.println("They are equal");
                if(obj1.isGreaterThan(obj2)==-1)
                        System.out.println("Time2 is greater");
        }
}
```

Output:



```
@lplab-Lenovo-Product: ~/Danish
student@lplab-Lenovo-Product:~/Danish$ javac Time.java
student@lplab-Lenovo-Product:~/Danish$ java Time
Enter Hours: 17
Enter Minutes: 55
Enter Seconds: 35
Enter Hours: 12
Enter Minutes: 35
Enter Seconds: 05
The time is: 17:55:35
The time is: 12:35:5
Added: The time is: 29:90:40
Subtracted: The time is: 5:20:30
Time1 is greater
student@lplab-Lenovo-Product:~/Danish$ 
```

Lab No. 4: Constructors and Static Members

Lab Exercises

1. Consider the already defined Complex class. Provide a default constructor and

parameterized constructor to this class. Also provide a display method. Illustrate all the constructors as well as the display method by defining Complex objects.

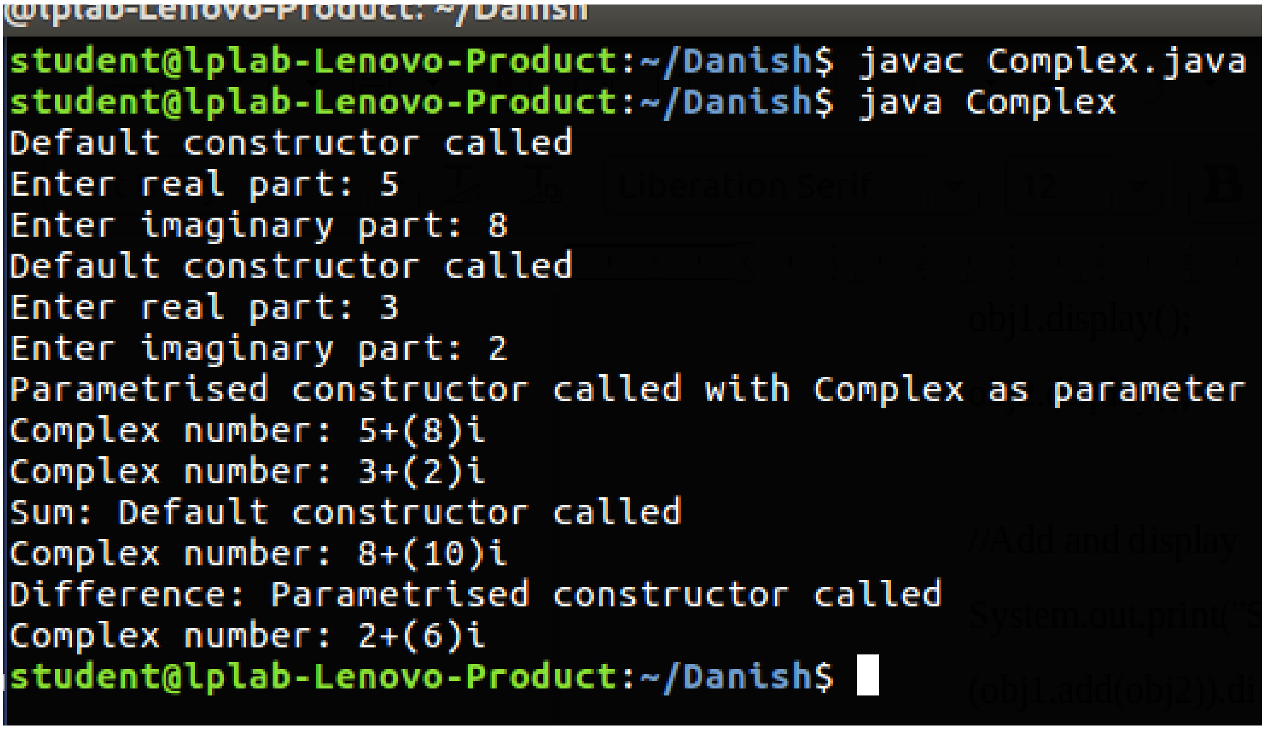
Solution:

```
import java.util.Scanner;

import java.lang.Math;


public class Complex

{

        int real, imaginary;


        //Default constructor

        public Complex()

        {

                System.out.println("Default constructor called");

                this.real=0;

                this.imaginary=0;

        }


        //Parameterised Constructor

        public Complex(int re, int im)

        {

                System.out.println("Parametrised constructor called");

                this.real=re;

                this.imaginary=im;

        }


        //Parameterised Constructor with Object as Parameter
```

```
public Complex (Complex c)
{
        System.out.println("Parametrised constructor called with Complex as parameter");
        this.real=c.real;
        this.imaginary=c.imaginary;
}

public static Complex getComplex()
{
        Complex a = new Complex();
        Scanner sc=new Scanner(System.in);
        System.out.print("Enter real part: ");
        a.real=sc.nextInt();
        System.out.print("Enter imaginary part: ");
        a.imaginary=sc.nextInt();

        return a;
}

public void display()
{
        System.out.println("Complex number: "+this.real+"+("+this.imaginary+")i");
}

public Complex add(Complex a)
{
        Complex c = new Complex();//default constructor used
        c.real=a.real+this.real;
        c.imaginary=a.imaginary+this.imaginary;
        return c;
}
```

```
    }

    public Complex subtract(Complex a)
    {
        Complex c = new Complex(0, 0);//first parameterised constructor used
        c.real=this.real-a.real;
        c.imaginary=this.imaginary-a.imaginary;
        return c;
    }

    public static void main(String[] arg)
    {
        //Add
        Complex obj1 = getComplex();
        Complex obj2 = new Complex(getComplex());     //giving Complex object as
parameter for constructor

        //Display method
        obj1.display();
        obj2.display();

        //Add and display
        System.out.print("Sum: ");
        (obj1.add(obj2)).display();

        //Subtract and display
        System.out.print("Difference: ");
        (obj1.subtract(obj2)).display();

    }
}
```

Output:



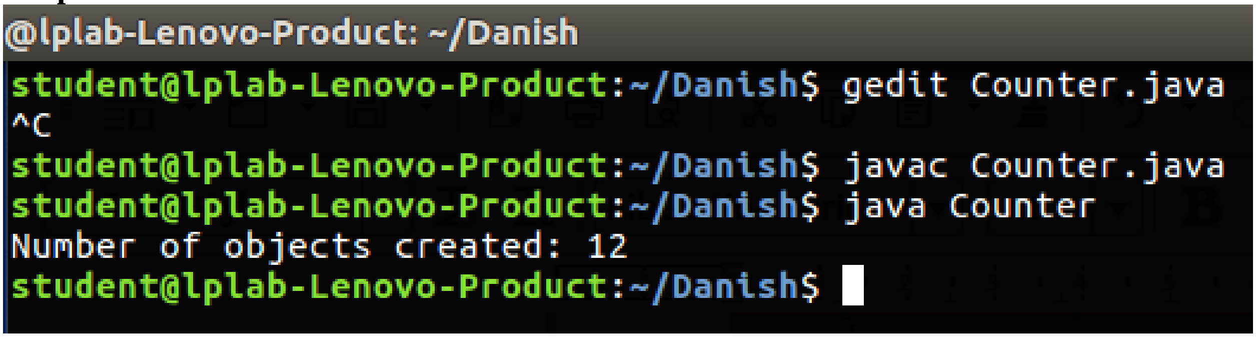
```
student@lplab-Lenovo-Product:~/Danish$ javac Complex.java
student@lplab-Lenovo-Product:~/Danish$ java Complex
Default constructor called
Enter real part: 5
Enter imaginary part: 8
Default constructor called
Enter real part: 3
Enter imaginary part: 2
Parametrised constructor called with Complex as parameter
Complex number: 5+(8)i
Complex number: 3+(2)i
Sum: Default constructor called
Complex number: 8+(10)i
Difference: Parametrised constructor called
Complex number: 2+(6)i
student@lplab-Lenovo-Product:~/Danish$ 
```

4. Create a class called Counter that contains a static data member to count the number of Counter objects being created. Also define a static member function called showCount() which displays the number of objects created at any given point of time. Illustrate this.

Solution:

```
import java.util.Scanner;
import java.lang.Math;

public class Counter
{
	static int counter=0;

	private static void updateCounter()
	{
		counter=counter+1;
	}

	public static void showCount()
	{
		System.out.println("Number of objects created: "+counter);
	}

	public Counter()
	{
		updateCounter();
	}

	public static void main(String[] arg)
	{
		Counter a=new Counter();
		Counter b=new Counter();
		Counter c=new Counter();
		Counter d=new Counter();
		Counter e=new Counter();
		Counter f=new Counter();
		Counter g=new Counter();
		Counter h=new Counter();
		Counter i=new Counter();
		Counter j=new Counter();
		Counter k=new Counter();
		Counter l=new Counter();
		showCount();
	}
}
```

Output:



```
student@lplab-Lenovo-Product:~/Danish$ gedit Counter.java
^C
student@lplab-Lenovo-Product:~/Danish$ javac Counter.java
student@lplab-Lenovo-Product:~/Danish$ java Counter
Number of objects created: 12
student@lplab-Lenovo-Product:~/Danish$ 
```