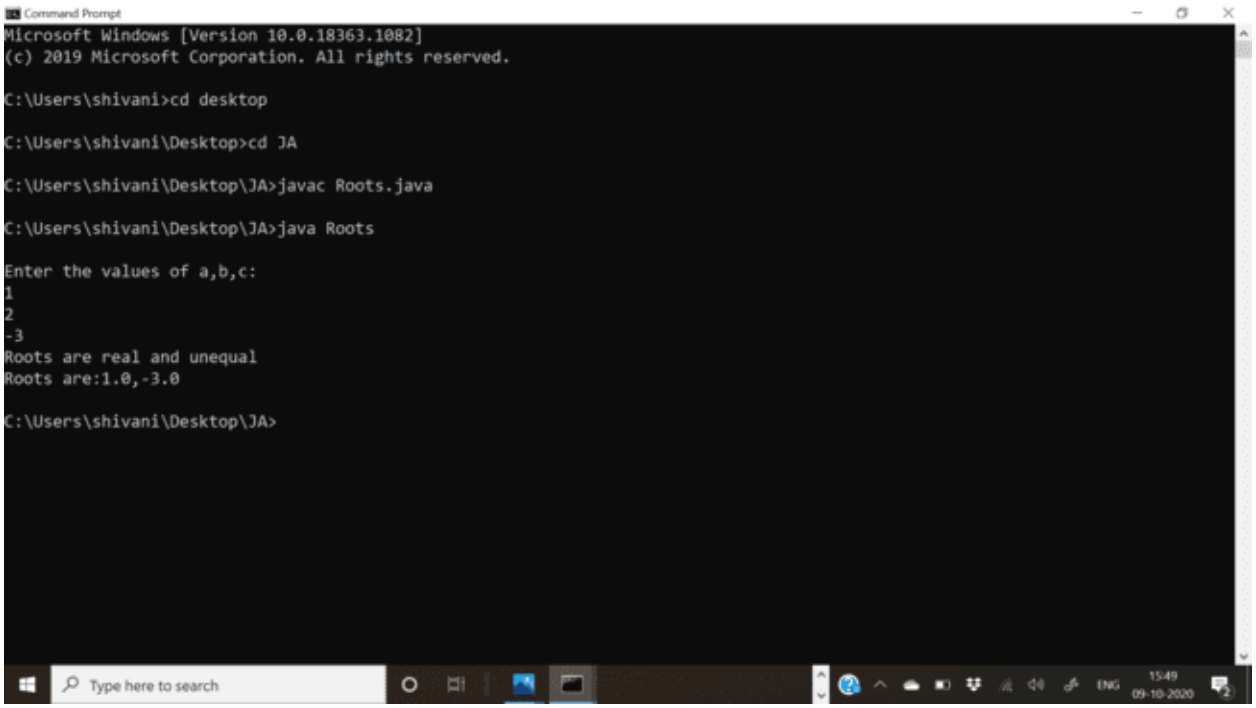
LAB PROGRAM 1 : Develop a Java program that prints all real solutions to the quadratic equation ax2 +bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminant b2-4ac is negative, display a message stating that there are no real solutions.

```
import java.util.*;
class Roots
{
  public static void main(String[] args){

  int a,b,c,f=0;
  double D;
  Scanner sc=new Scanner(System.in);
  System.out.println("\nEnter the values of a,b,c:");
  a=sc.nextInt();
  b=sc.nextInt();
  c=sc.nextInt();
  D=(b*b)-(4*a*c);
  if(D==0)
 {
  System.out.println("Roots are real and equal");
  f=1;
 }
 else if(D>0)
 {
  System.out.println("Roots are real and unequal");
  f=1;
 }
 else if(D<0)
 {
  System.out.println("Roots are imaginary");
 }
if(f==1)
{
  double r1=((-b+Math.sqrt(D))/(2*a));
  double r2=((-b-Math.sqrt(D))/(2*a));
  System.out.println("Roots are:"+r1+","+r2);
}
}
}
```



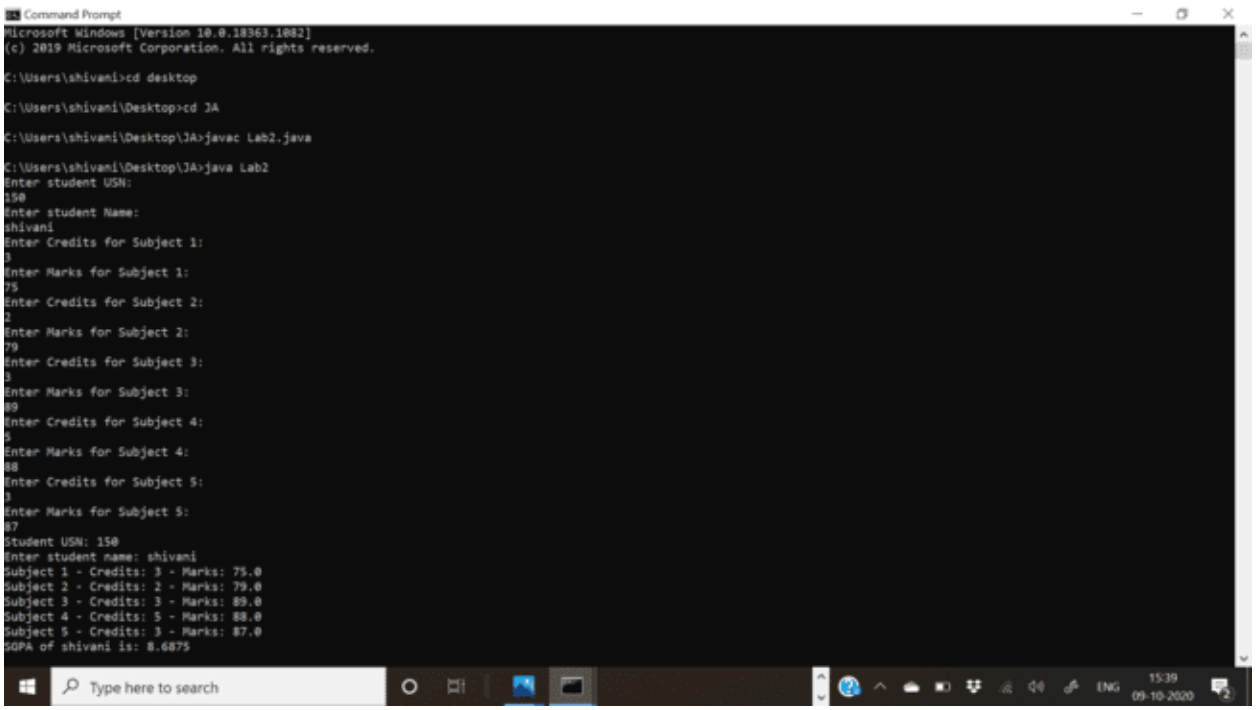
LAB PROGRAM 2 :

Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student

```
import java.util.Scanner;


class Student{
    Scanner sc = new Scanner(System.in);
    String USN;
    String Name;
    int credits[] = new int[5];
    float marks[] = new float[5];
    int points[] = new int[5];
    float SGPA;
    int totalCredits = 0;

    void getDetails(){
        System.out.println("Enter student USN: ");
        USN = sc.nextLine();
        System.out.println("Enter student Name: ");
```

```
        Name = sc.nextLine();
        for(int i=0;i<5;i++){
            System.out.println("Enter Credits for Subject " + (i+1) + ": ");
            credits[i] = sc.nextInt();
            totalCredits += credits[i];
            System.out.println("Enter Marks for Subject " + (i+1) + ": ");
            marks[i] = sc.nextFloat();
        }
    }

    void showDetails(){
        System.out.println("Student USN: " + USN);
        System.out.println("Enter student name: " + Name);
        for(int i=0;i<5;i++){
            System.out.println("Subject " + (i+1) + " - Credits: " + credits[i] + " - Marks: " +
marks[i]);
        }
        System.out.println("SGPA of " + Name + " is: " + (float)(SGPA/totalCredits));
    }

    void calcSGPA(){


        for(int i = 0;i<5;i++){
            if(marks[i] > 100){
                System.out.println("Error: Marks are above 100");
                return;
            }else if(marks[i] >= 90){
                points[i] = 10;
            }else if(marks[i] >= 80){
                points[i] = 9;
            }else if(marks[i] >= 70){
                points[i] = 8;
            }else if(marks[i] >= 60){
                points[i] = 7;
            }else if(marks[i] >= 50){
                points[i] = 5;
            }else if(marks[i] >= 40){
                points[i] = 4;
            }else{
```

```
            points[i] = 0;
        }

        SGPA += (points[i]*credits[i]);
    }
  }


}

public class Lab2 {
  public static void main(String args[]) {
    Student stu1 = new Student();
    stu1.getDetails();
    stu1.calcSGPA();
    stu1.showDetails();

  }
}
```

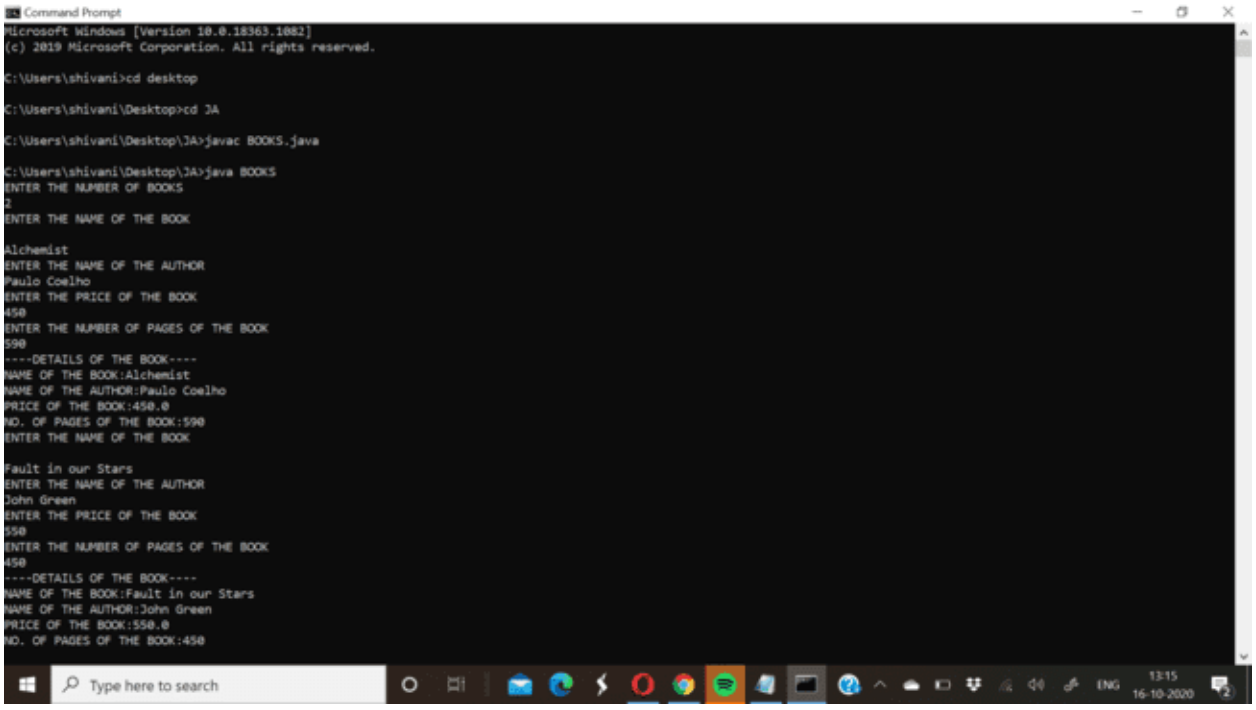


LAB PROGRAM 3:

Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.

```
import java.util.*;
class BOOKS
{
   String name,author;
   double price;
   int num_pages;

      public BOOKS()
      {
       this.name="";
       this.author="";
       this.price=0.0;
       this.num_pages=0;
      }

       public void DETAILS()
      {
       Scanner ob=new Scanner(System.in);
       System.out.println("ENTER THE NAME OF THE BOOK\n");
       name=ob.nextLine();
       System.out.println("ENTER THE NAME OF THE AUTHOR");
       author=ob.nextLine();
       System.out.println("ENTER THE PRICE OF THE BOOK");
       price=ob.nextDouble();
       System.out.println("ENTER THE NUMBER OF PAGES OF THE BOOK");
       num_pages=ob.nextInt();
      }

     public void ToString()
    {
     System.out.println("----DETAILS OF THE BOOK----");
```

```
    System.out.println("NAME OF THE BOOK:"+name);
    System.out.println("NAME OF THE AUTHOR:"+author);
    System.out.println("PRICE OF THE BOOK:"+price);
    System.out.println("NO. OF PAGES OF THE BOOK:"+num_pages);
   }
    public static void main(String args[])
    {
      int i=0,n;
      BOOKS obj=new BOOKS();
      Scanner ob1=new Scanner(System.in);
      System.out.println("ENTER THE NUMBER OF BOOKS");
      n=ob1.nextInt();
      for(i=1;i<=n;i++)
      {
        obj.DETAILS();
        obj.ToString();
      }
    }
}
```

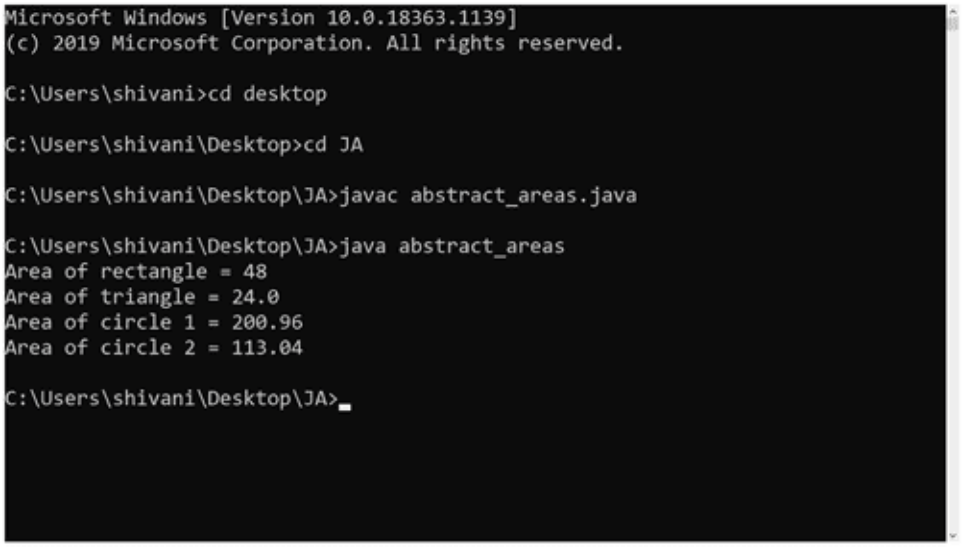


LAB PROGRAM 4:

Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea( ). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.

```
abstract class Shape
{
 int a=8, b=6;
 abstract void printArea();
}

class Rectangle extends Shape
{
 int area_rectangle;
 void printArea()
 {
  area_rectangle = a*b;
  System.out.println("Area of rectangle = " + area_rectangle);
 }
}

class Triangle extends Shape
{
 float area_triangle;
 void printArea()
 {
  area_triangle = (float)(0.5*a*b);
  System.out.println("Area of triangle = " + area_triangle);
 }
}

class Circle extends Shape
{
 float area_circle_1, area_circle_2;
 void printArea()
 {
```

```
  area_circle_1 = (float)(3.14*a*a);
  area_circle_2 = (float)(3.14*b*b);
  System.out.println("Area of circle 1 = " + area_circle_1);
  System.out.println("Area of circle 2 = " + area_circle_2);
 }
}

class abstract_areas
{
 public static void main(String args[])
 {
  Rectangle r = new Rectangle();
  r.printArea();
  Triangle t = new Triangle();
  t.printArea();
  Circle c = new Circle();
  c.printArea();
 }
}
```



LAB PROGRAM-5 Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility.

The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks: • Accept deposit from customer and update the balance. • Display the balance. • Compute and deposit interest • Permit withdrawal and update the balance • Check for the minimum balance, impose penalty if necessary and update the balance

```
import java.util.*;
class Account
{
	String name,type;
	int acc_no;
	double amount;
	Scanner in=new Scanner(System.in);
	void type(int choice)
	{
		if(choice==1)
			type="Savings Account";
		if(choice==2)
			type="Current Account";
	}
	void input()
	{
		System.out.println("Enter the Name,Account number and Balance:");
		name=in.next();
		acc_no=in.nextInt();
		amount=in.nextDouble();
	}
	void deposit()
	{
		System.out.println("Enter the amount to be deposited:");
		double x=in.nextDouble();
		amount=amount+x;
```

```
	}
	void display()
	{
		System.out.println("Name:"+name);
		System.out.println("Account number:"+acc_no);
		System.out.println("Type:"+type);
		System.out.println("balance:"+amount);
	}
}
class Savings_acc extends Account
{
	double a,cinterest;
	int r,t;
	Scanner in=new Scanner(System.in);
	void withdrawal()
	{
		System.out.println("Enter amount to be withdrawn:");
		double amtw=in.nextDouble();
		if(amtw<=amount)
			amount=amount-amtw;
		else
			System.out.println("Invalid amount");
	}
	void cmp_interest()
	{
		System.out.println("Enter the rate and time:");
		r=in.nextInt();
		t=in.nextInt();
		a=amount* Math.pow(1 + (r *0.01),t);
		cinterest= a - amount;

	}
	 void display()
	{
		super.display();
		System.out.println("Compound Interest after " + t + " years:
"+cinterest);
```

```
            System.out.println("Amount after " + t + " years: "+a);
        }
}
class Current_acc extends Account
{
        double min=10000;
        void input()
        {
            super.input();
        }
        void service_charge()
        {
            if(amount<min)
            amount=amount-500;
        }
        void display()
        {
            super.display();
        }
}
class bankdemo
{
        public static void main(String args[])
        {
            Scanner in=new Scanner(System.in);
            System.out.println("Choose type of account.");
            System.out.println("1.Savings account.");
            System.out.println("2.Current account.");
            int choice=in.nextInt();

            if(choice==1)
            {
                Savings_acc b=new Savings_acc();
                b.type(choice);
                b.input();
                System.out.println("Do you want to deposit or
withdraw?\n1.Deposit.\n2.Withdraw\n");
```

```
                int ch=in.nextInt();
                if(ch==1)
                b.deposit();
                else if(ch==2)
                b.withdrawal();
                else
                System.out.println("Invalid choice");
                b.cmp_interest();
                b.display();
            }
            else if(choice==2)
            {
                Current_acc b=new Current_acc();
                b.type(choice);
                 b.input();
                b.deposit();
                b.service_charge();
                b.display();
            }
            else
                System.out.println("Invalid choice");

        }
}
```

LAB PROGRAM 6

Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.

In CIE package, Student.java

```
package CIE;
public class Student
{
 public String usn;
 public String name;
 public int sem;
 public Student(String usn, String name, int sem)
 {
 this.usn = usn;
 this.name = name;
```

```
 this.sem = sem;
 }
}
```

In CIE package, Internals.java

```
package CIE;
import java.util.Scanner;
public class Internals extends Student
{
 Scanner in = new Scanner(System.in);
 public int[] cie = new int[5];
 public Internals(String usn, String name, int sem)
 {
 super(usn,name,sem);
 }
 public void get_data()
 {
 for(int i=0;i<5;i++)
 {
 System.out.print("Enter CIE marks of the student in Subject " + (i+1) + " (out
of 50) : ");
 cie[i] = in.nextInt();
 }
 }
}
```

In SEE package, External.java

```
package SEE;
import java.util.Scanner;
public class External extends CIE.Student
{
 public External(String usn, String name, int sem)
 {
 super(usn,name,sem);
 }
 Scanner in = new Scanner(System.in);
 public int[] see = new int[5];
 public void get_data()
```

```
 {
 for(int i=0;i<5;i++)
 {
 System.out.print("Enter SEE marks of the student in Subject " + (i+1) + "  (out
of 100) : ");
 see[i] = in.nextInt();
 }
 }
}

 In default package, final_marks.java
import CIE.*;
import SEE.*;
import java.util.Scanner;
class final_marks
{
 public static void main(String args[])
 {
 Scanner in = new Scanner(System.in);
 System.out.print("Enter the number of students : ");
 int n = in.nextInt();
 CIE.Internals obj[] = new CIE.Internals[5];
 SEE.External obj1[] = new SEE.External[5];
 int total=0;
 for(int i=0;i<n;i++)
 {
 System.out.println("\nEnter details of Student " + (i+1) + " :");
System.out.print("USN : ");
 String usn = in.next();
 System.out.print("Name : ");
 String name = in.next();
 System.out.print("Semester : ");
 int sem = in.nextInt();
 obj[i] = new CIE.Internals(usn,name,sem);
 obj[i].get_data();
 obj1[i] = new SEE.External(usn,name,sem);
 obj1[i].get_data();
 }
```

```
 for(int i=0;i<n;i++)
 {
 System.out.println("\nTOTAL MARKS OF STUDENT " + (i+1) + " (out of  100)
in :");
 for(int j=0;j<5;j++)
 {
 total = (obj[i].cie[j]) + ((obj1[i].see[j])/2);
 System.out.println("Subject " + (j+1) + " : " + total);  }
 }
 }
}
```

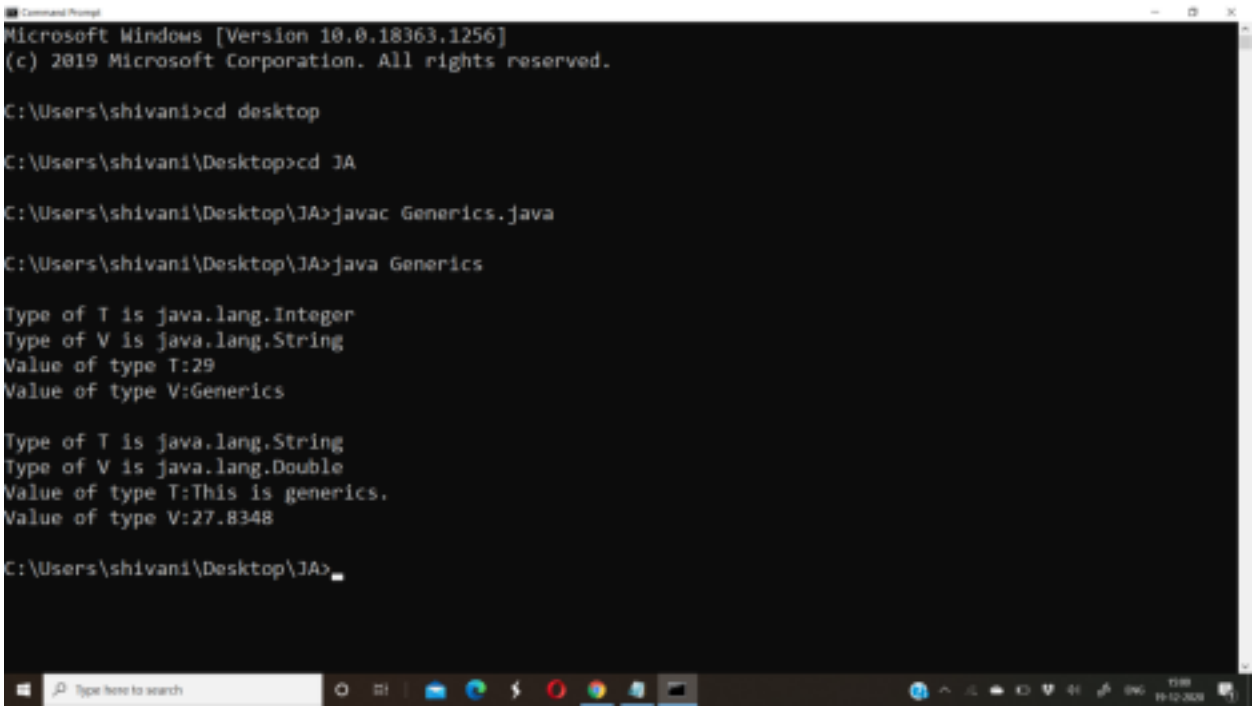
LAB PROGRAM 7

Write a program to demonstrate generics with multiple object parameters.

```
class TwoGen<T,V>

{

        T obj1;

        V obj2;

 TwoGen(T ob1, V ob2)

        {

 obj1 = ob1;

 obj2 = ob2;

        }

        void showTypes()

        {

 System.out.println("\nType of T is " + obj1.getClass().getName());

System.out.println("Type of V is " + obj2.getClass().getName()); }
```

```
        T getobj1()
        {
 return obj1;
        }
        V getobj2()
        {
 return obj2;
        }
}
 class Generics
{
 public static void main(String args[])
        {
 TwoGen<Integer, String> object1 = new TwoGen<Integer, String>(29, "Generics");
object1.showTypes();
 int i = object1.getobj1();
 System.out.println("Value of type T: " + i);
 String str = object1.getobj2();
 System.out.println("Value of type V: " + str);
 TwoGen<String, Double> object2 = new TwoGen<String, Double>("This is
generics.", 27.8348);
 object2.showTypes();
 String str1 = object2.getobj1();
 System.out.println("Value of type T: " + str1);
```

```
 double j = object2.getobj2();

 System.out.println("Value of type V: " + j);

        }

}
```



LAB PROGRAM 8

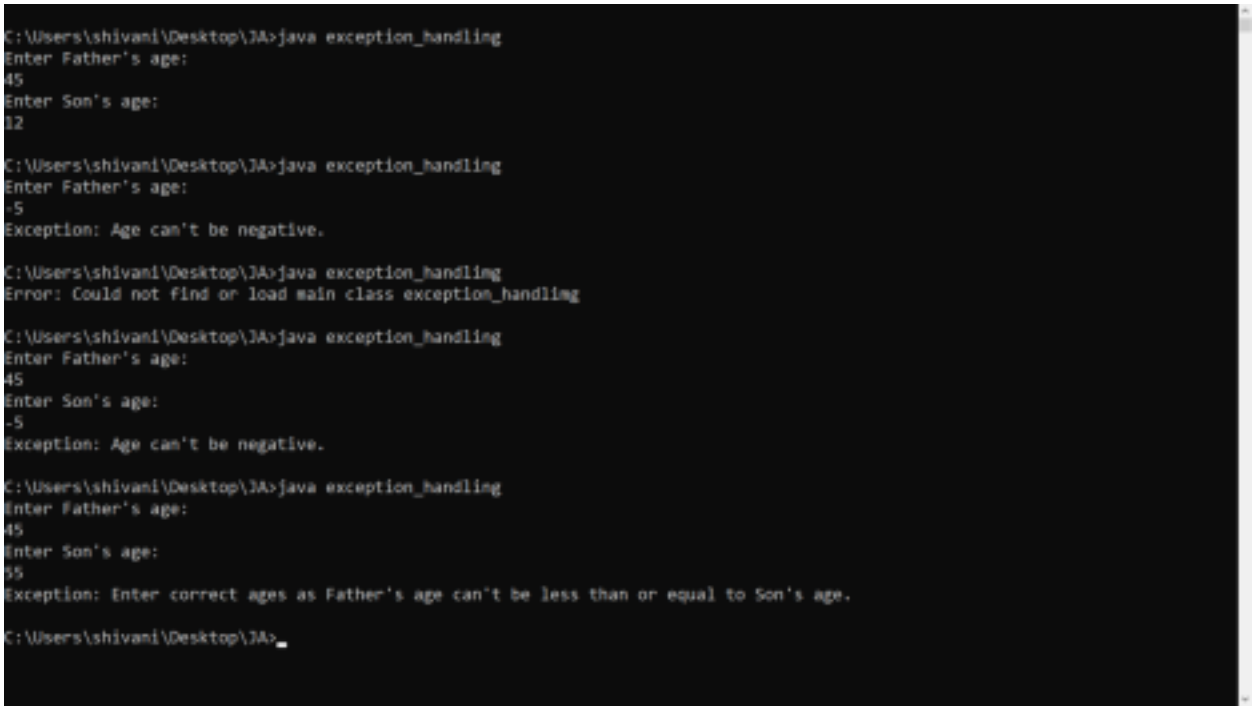
Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception NegativeAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws the exception WrongAge() if son’s age is >= father’s age.

```
import java.util.*;

class WrongAge extends Exception

{

        int f,s;
```

```
 WrongAge(int fAge,int sAge)
        {
f=fAge;
 s=sAge;
        }
 public String toString()
        {
 return "Enter correct ages as Father's age can't be less than or equal to Son's
age.";
        }
}
class NegativeAge extends Exception
{
        int x;
 NegativeAge(int fAge)
        {
 x=fAge;
        }
 public String toString()
        {
 return "Age can't be negative.";
        }
}
class Father
```

```
{
        int fAge;
 Scanner in = new Scanner(System.in);
Father() throws NegativeAge
        {
 System.out.println("Enter Father's age: ");
fAge=in.nextInt();
 if(fAge<0)
 {
 throw new NegativeAge(fAge);  }
        }
}
 class Son extends Father
{
        int sAge;
 Scanner in = new Scanner(System.in);
        Son() throws
        NegativeAge,WrongAge {
 super();
 System.out.println("Enter Son's age: ");
sAge=in.nextInt();
 if(sAge<0)
 {
 throw new NegativeAge(sAge);
```

```
 }
 if(sAge>=fAge)
 {
 throw new WrongAge(fAge,sAge);  }
          }
}
class exception_handling
{
 public static void main(String args[]) {
 try
 {
 Son s = new Son();
 }
 catch(NegativeAge n)
 {
 System.out.println("Exception: "+n);  }
 catch(WrongAge w)
 {
 System.out.println("Exception: "+w);  }
          }
}
```



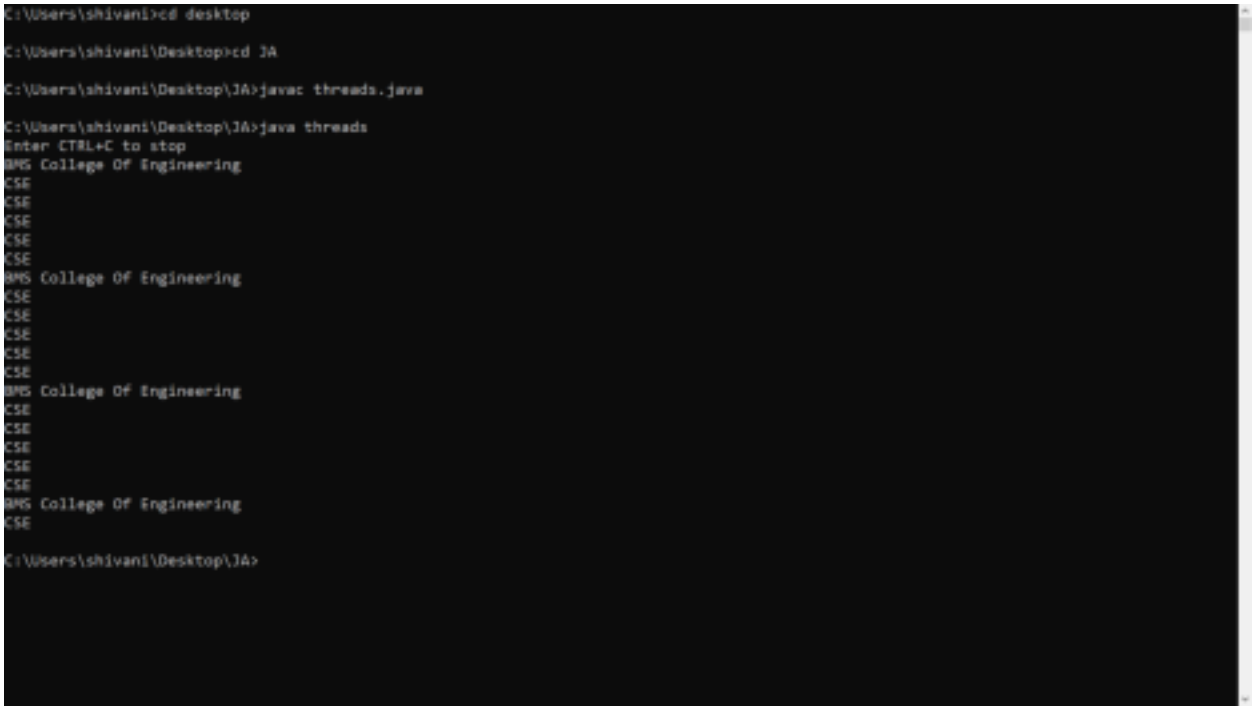
LAB PROGRAM 9

Write a program which creates two threads, one thread displaying "BMS College of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.

```
class thread1 implements Runnable
{
 Thread t1;
 thread1()
        {
 t1 = new Thread(this,"thread1");
 t1.start();
        }
 public void run()
 {
```

```
 for(;;)
 {
 try
 {
 System.out.println("BMS College Of Engineering");
Thread.sleep(10000);
 }
 catch(InterruptedException ie)
 {
 System.out.println("Interrupted");
 }
 }
        }
}
class thread2 implements Runnable
{
 Thread t2;
 thread2()
        {
 t2 = new Thread(this,"thread2");
 t2.start();
        }
 public void run()
 {
```

```
 for(;;)
 {
 try
 {
 System.out.println("CSE");
 Thread.sleep(2000);
 }
 catch(InterruptedException ie)  {
 System.out.println("Interrupted");  }
 }
          }
}
 class threads
{
 public static void main(String args[]) {
 System.out.println("Enter CTRL+C to stop");
 thread1 t1 = new thread1();
thread2 t2 = new thread2(); }
}
```



LAB PROGRAM 10

Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException. Display the exception in a message dialog box.

```
import java.awt.*;

import java.awt.event.*;

import javax.swing.*;

public class division extends Frame implements ActionListener{

 TextField n1,n2,res;

        Label ln1,ln2,lres;

 Button b;
 public division(){
```

```
 setLayout(new FlowLayout());
 Label ln1=new Label("NUMBER 1",Label.RIGHT);
Label ln2=new Label("NUMBER 2",Label.RIGHT);
Label lres=new Label("RESULT",Label.RIGHT);
n1=new TextField(12);
 n2=new TextField(8);
 res=new TextField(10);
 b=new Button("DIVIDE");
 add(ln1);
 add(n1);
 add(ln2);
 add(n2);
 add(b);
 add(lres);
 add(res);
 b.addActionListener(this);
 addWindowListener(new WindowAdapter1());
}
 public void actionPerformed(ActionEvent
ae) {
 if(ae.getSource()==b)
          {
 try{
  int num1=Integer.parseInt(n1.getText());   int
```

```
num2=Integer.parseInt(n2.getText());         int

num3=num1/num2;

 res.setText(String.valueOf(num3));

 }catch(NumberFormatException ne ){

 JOptionPane.showMessageDialog(this,ne,"ERROR",
JOptionPane.ERROR_MESSAGE);

          }

 catch(ArithmeticException a){

 JOptionPane.showMessageDialog(this,a,"ERROR",
JOptionPane.ERROR_MESSAGE);

          }

}

}

 public static void main(String args[])

{

 division i=new division();

 i.setSize(new Dimension(700,300));

 i.setTitle("DIVISION OF TWO INTEGERS");

i.setVisible(true);

}

 class WindowAdapter1 extends

WindowAdapter{  public void

windowClosing(WindowEvent we)

          {
```

```
    System.exit(0); }
  }
}
```