B.M.S College of Engineering

(Autonomous Institution affiliated to VTU, Belagavi)
Bengaluru – 20

Department of Computer Science and Engineering

Report on

OOJP LAB PROGRAMS

Course Title: Object Oriented Java Programming

Course Code: 19CS3PCOOJ

(Autonomous Scheme 2020)

Submitted by

Name: AVIRATH MANOHAR HEGDE
USN: 1BM19CS195

LAB PROGRAM 1

QUESTION

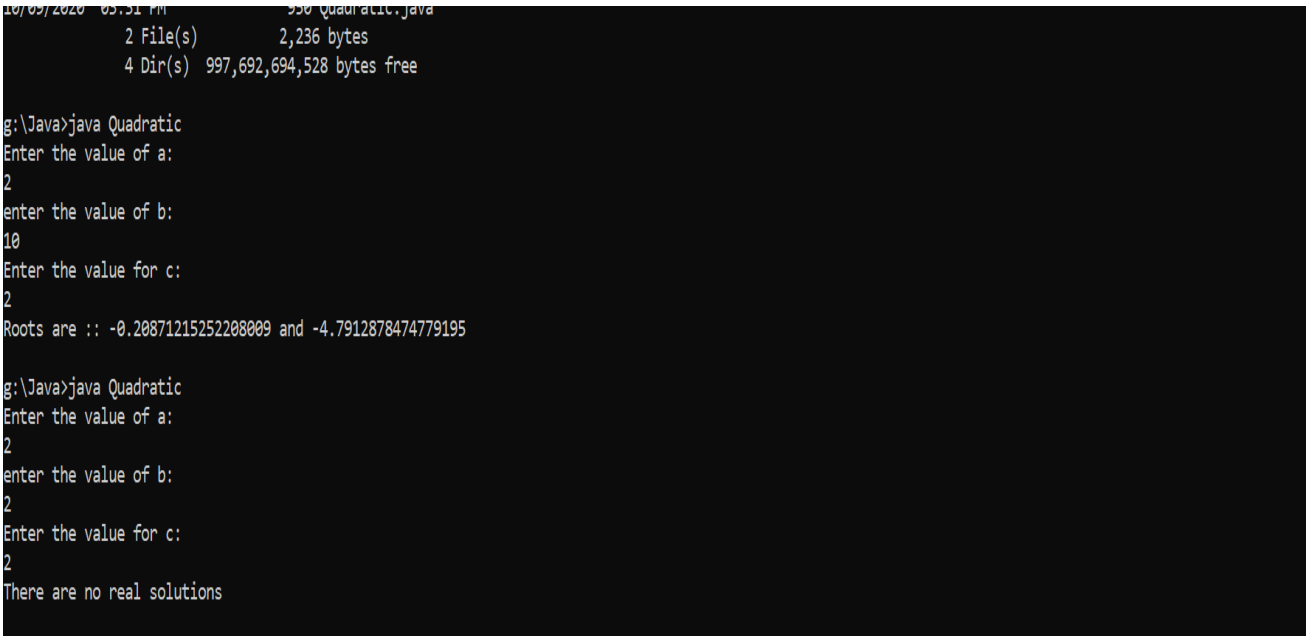
Develop a Java program that prints all real solutions to the quadratic equation $ax^2+bx+c = 0$. Read in a, b, c and use the quadratic formula. If the discriminate b^2-4ac is negative, display a message stating that there are no real solutions.

CODE

```
import java.util.Scanner;
public class Quadratic
{
   public static void main(String args[])
   {
      double x=0, y=0;
      double a,b,c;
      double det;
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter the value of a:");
      a = sc.nextDouble();

      System.out.println("enter the value of b:");
      b  = sc.nextDouble();

      System.out.println("Enter the value for c:");
      c = sc.nextDouble();

      det= (b*b)-(4*a*c);
      double sqrt = Math.sqrt(det);
       if(det>0)
      {
         x = (-b+sqrt)/(2*a);
         y = (-b-sqrt)/(2*a);
         System.out.println("Roots are :: "+x+" and "+y);
      }
      else if(det==0)
      {
          System.out.println("Root is :: "+(-b + sqrt)/(2*a));
      }
      else
```

```
        System.out.println("There are no real solutions");
    }
}
```

OUTPUT



LAB PROGRAM 2

QUESTION

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.

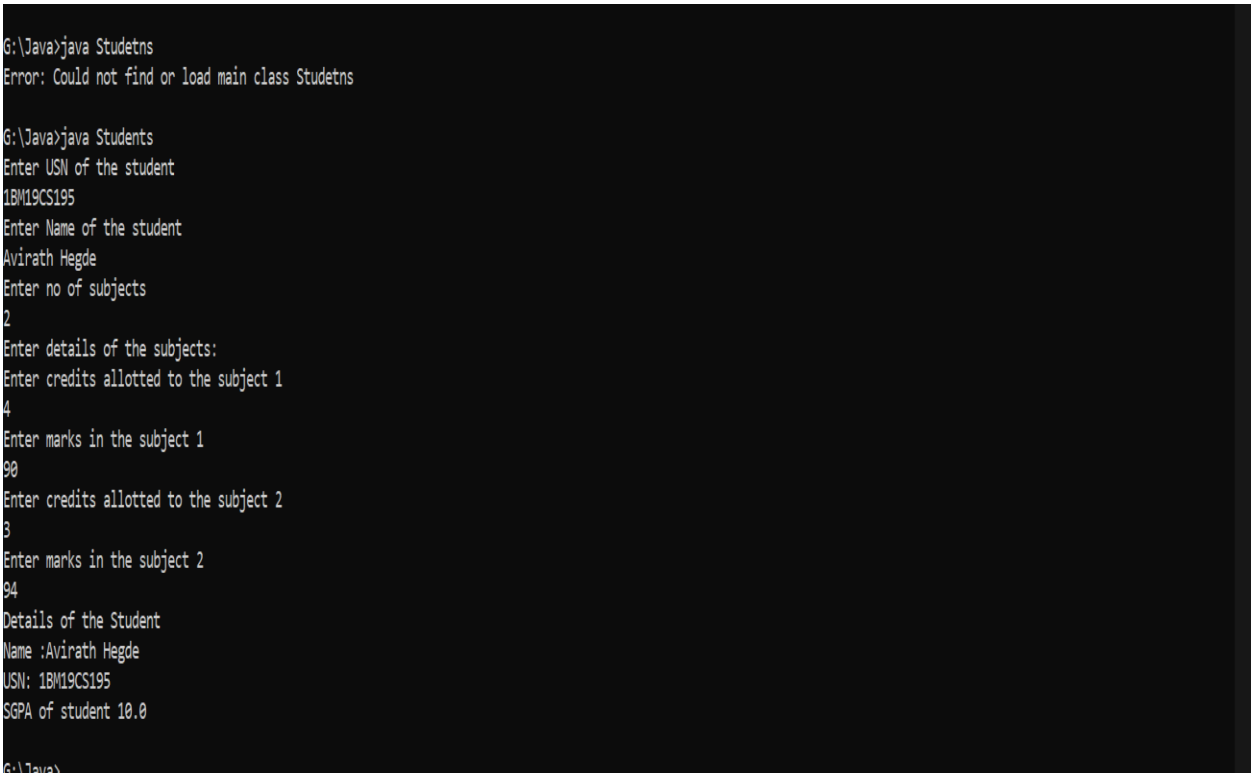
CODE

```
import java.util.Scanner;
class Student
{
```

```
private String USN;
private String name;
private int n;
private double SGPA = 0;
private int totalCredits = 0;
Scanner ss = new Scanner(System.in);

void Details()
{
System.out.println("Enter USN of the student");
USN = ss.nextLine();
System.out.println("Enter Name of the student");
name = ss.nextLine();
System.out.println("Enter no of subjects");
n = ss.nextInt();
int credits[] = new int[n];
double marks[] = new double[n];
System.out.println("Enter details of the subjects:");
for(int i=0;i<n;i++)
{
 System.out.println("Enter credits allotted to the subject "+(i+1));
 credits[i] = ss.nextInt();
 System.out.println("Enter marks in the subject "+(i+1));
 marks[i] = ss.nextInt();
 Calculate(credits[i],marks[i],i);
}
   }
   void Calculate(int credit,double mark,int j)
   {
 totalCredits = totalCredits + credit;
 if(mark>=90&&mark<=100)
  SGPA = SGPA + (10*credit);
 else if(mark>=80 && mark<=89)
  SGPA = SGPA + (9*credit);
 else if(mark>=70&&mark<=79)
  SGPA = SGPA + (8*credit);
 else if(mark>=60&&mark<=69)
  SGPA = SGPA + (7*credit);
 else if(mark>=50 && mark<=59)
  SGPA = SGPA + (6*credit);
 else if(mark>=40&&mark<=49)
  SGPA = SGPA + (5*credit);
 else
  System.out.println("Failed in subject "+(j+1));
```

```
 }
 void Display()
 {
  System.out.println("Details of the Student");
  System.out.println("Name :"+name);
  System.out.println("USN: "+USN);
  System.out.println("SGPA of student "+(SGPA/totalCredits));
 }
}

public class Students
{
 public static void main(String args[])
 {
  Student s1 = new Student();
  s1.Details();
  s1.Display();
 }
}
```

OUTPUT



LAB PROGRAM 3

QUESTION

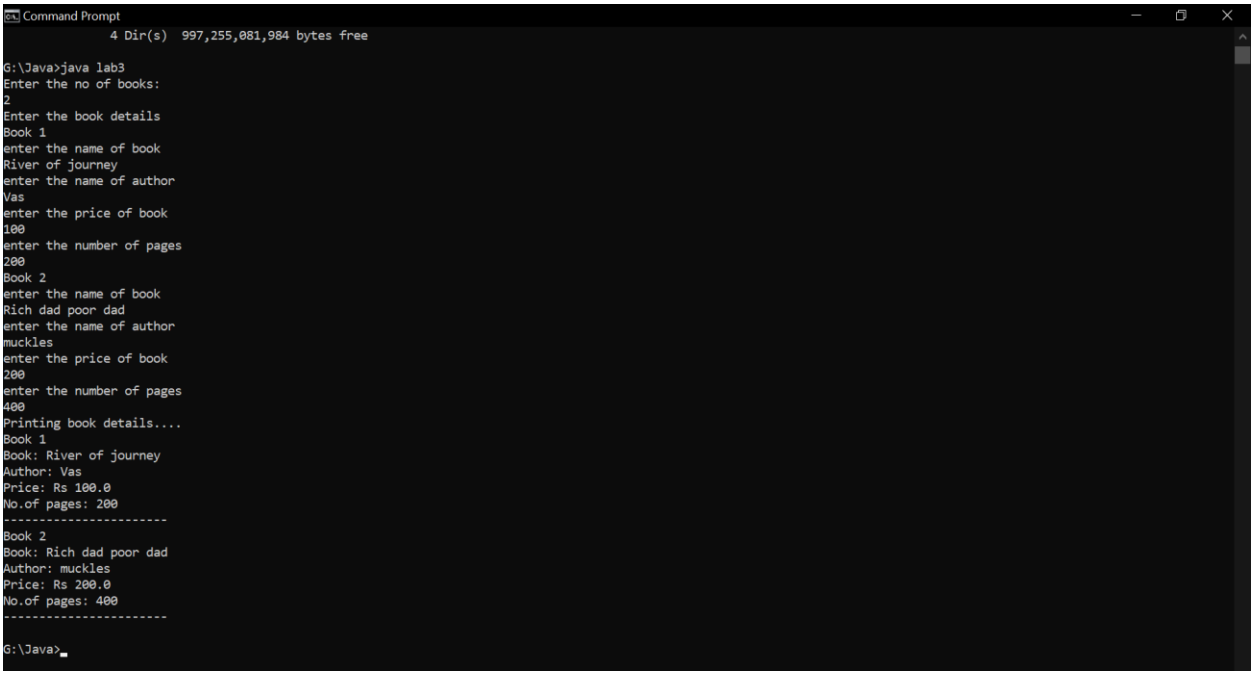
Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program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

CODE

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

	Book()
	{
		name="xyz";
		author="abc";
		price= 0.0;
		num_pages=10;
	}
	void getdata()
	{
		Scanner sc= new Scanner(System.in);
		System.out.println("enter the name of book");
		name = sc.nextLine();
		System.out.println("enter the name of author");
		author = sc.nextLine();
		System.out.println("enter the price of book");
		price = sc.nextDouble();
		System.out.println("enter the number of pages");
		num_pages = sc.nextInt();
	}
```

```
	public String toString()
	{
		return("Book: "+name+"\nAuthor: "+author+"\nPrice: Rs "+price+"\nNo.of pages:
"+num_pages);
	}
}
class lab4
{
	public static void main(String ss[])
	{
	  Scanner xx=new Scanner(System.in);
	  System.out.println("Enter the no of books:");
	  int n=xx.nextInt();
	  Book b[]=new Book[n];
	  int i;
	  System.out.println("Enter the book details");
	  for(i=0;i<n;i++)
	  {
	    System.out.println("Book "+(i+1));
	    b[i]=new Book();
		b[i].getdata();
	  }
	  System.out.println("Printing book details....");
	  for(i=0;i<n;i++)
	  {
	    System.out.println("Book "+(i+1));
		System.out.println(b[i]);
		System.out.println("-----------------------");

	  }
	 }
}
```

OUTPUT



LAB PROGRAM 4

QUESTION

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.

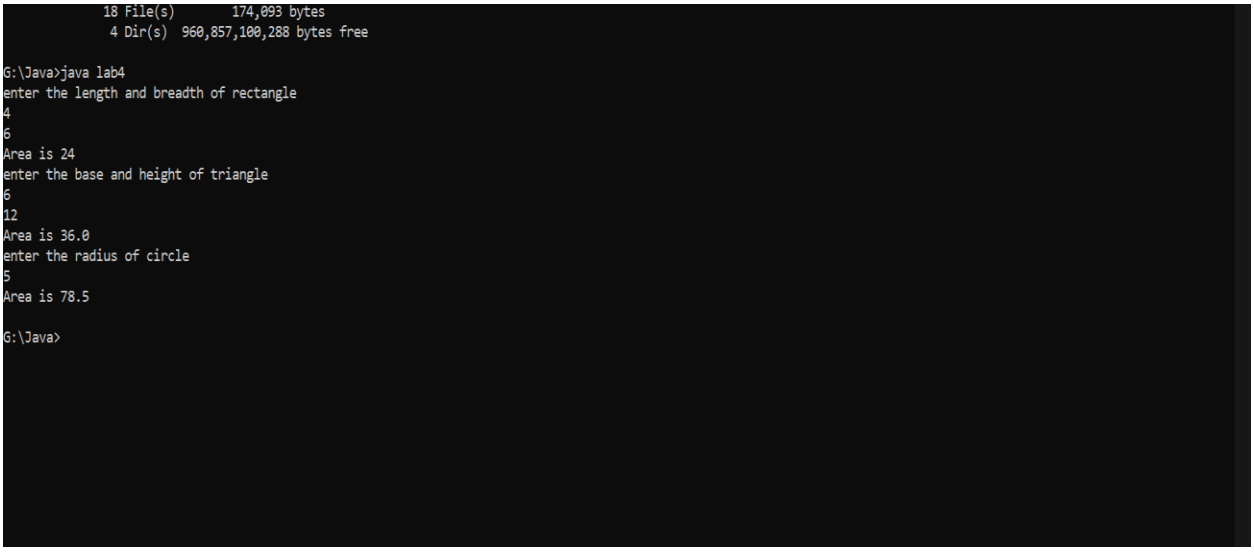
CODE

```
import java.util.*;
abstract class Shape
{
        int a;
        int b;

        abstract void printArea();
}
```

```
class Rectangle extends Shape
{
	Rectangle(int x, int y)
	{
		a=x;
		b=y;
	}

	void printArea()
	{
		System.out.println("Area is "+(a*b));
	}
}

class Triangle extends Shape
{
	Triangle(int x, int y)
	{
		a=x;
		b=y;
	}
	void printArea()
	{
		System.out.println("Area is "+(a*b*0.5));
	}
}
class Circle extends Shape
{
	Circle(int x)
	{
		a=x;
	}
	void printArea()
	{
		System.out.println("Area is "+(a*a*3.14));
	}
}

class lab4
{
	public static void main(String ss[])
	{
		int l,b,ba,h,ra;
		Scanner sc = new Scanner(System.in);
```

```
                System.out.println("enter the length and breadth of rectangle");
                l= sc.nextInt();
                b= sc.nextInt();
                Rectangle r= new Rectangle(l,b);
                r.printArea();

                System.out.println("enter the base and height of triangle");
                ba= sc.nextInt();
                h= sc.nextInt();
                Triangle t = new Triangle(ba,h);
                t.printArea();

                System.out.println("enter the radius of circle");
                ra= sc.nextInt();
                Circle c = new Circle(ra);
                c.printArea();
        }
}
```

OUTPUT



LAB PROGRAM 5

QUESTION

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.

CODE

```
import java.util.*;
import java.lang.Math;
class Account
{
        String name;
        int acctno;
```

```
        char type;
        double balance;
        double dep;
        boolean cheq;

        void get(char c)
        {
                type = c;
                if(c=='s' || c == 'S')
                cheq=false;
                else cheq=true;
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter your name");
                name = sc.nextLine();
                System.out.println("Enter the account number");
                acctno = sc.nextInt();
                System.out.println("Enter the current available balance in your account");
                balance= sc.nextDouble();
        }

        void putd()
        {
                System.out.println("Account details");
                System.out.println("Name: "+name);
                System.out.println("Account number: "+acctno);
                System.out.println("Account type :"+type);
                System.out.println("balance: "+balance);
        }
        void dep()
        {
                                Scanner ss = new Scanner(System.in);
                                System.out.println("Enter the amount to be deposited");
                                dep= ss.nextDouble();
                                balance=balance +dep;
                                System.out.println("Amount has been deposited and balance has been
updated");
        }
        void display()
        {

                                System.out.println("Balance amount is "+balance);
        }

        void check()
```

```
	{
				if(cheq==false)
				System.out.println("Cheque book facility is not available");
				else
				System.out.println("Cheque book facility is available");

	}


}

class Saving extends Account
{

	double rate;

	double s_with;
	int n;

	int ch;
	double amt;
	double term;
	double pr;



	void ci()
	{
					Scanner ss = new Scanner(System.in);
					System.out.println("Enter principal deposit amount");
					pr = ss.nextDouble();
					System.out.println("Enter the rate of interest");
					rate = ss.nextDouble();
					System.out.println("Enter the term(years)");
					term = ss.nextDouble();
					System.out.println("Enter the number of times interest in
compounded annually");
					n = ss.nextInt();
					amt = pr* Math.pow((1+(rate/100)),(n*term));
					balance+= amt;
					System.out.println("Interest is compounded and deposited;
balance is updated");

	}
```

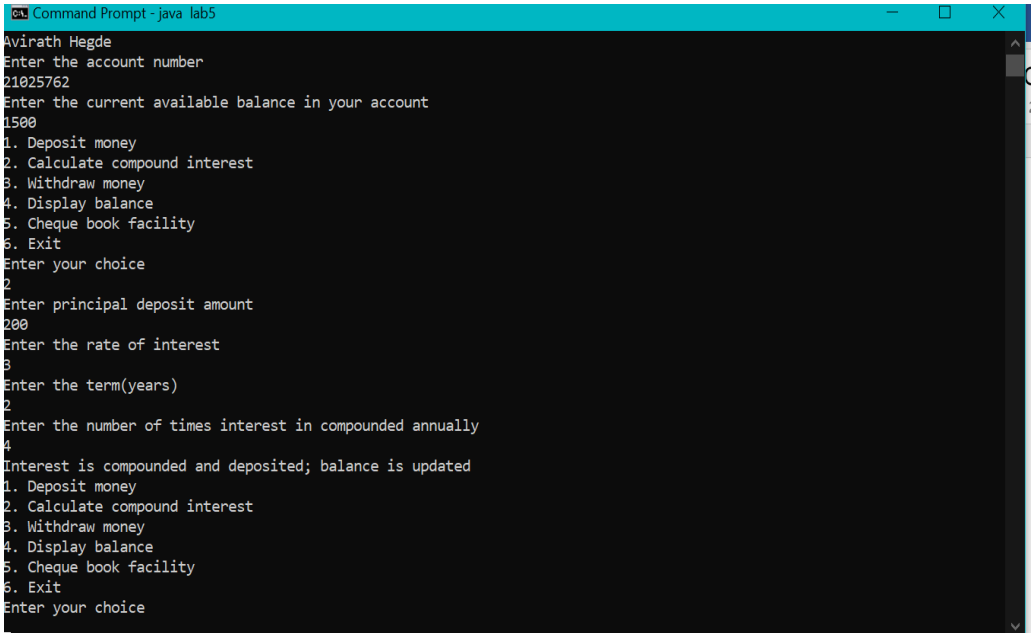
```
	void with_s()
	{
                                Scanner ss = new Scanner(System.in);
                                System.out.println("Enter the amount of money to be withdrawn");
                                s_with = ss.nextDouble();
                                if(s_with>balance)
                                System.out.println("Insufficient balance");
                                else
                                {balance= balance - s_with;
                                System.out.println("Money has been withdrawn and balance has been
updated");}
	}




}

class Current extends Account
{

	double c_with;
	double pen;
	double min;
	Current()
	{
		pen=100;
		min=500;
	}

	void with_c()
	{
                                Scanner xx = new Scanner(System.in);
                                System.out.println("Enter the amount to be withdrawn");
                                c_with= xx.nextDouble();
                                if(c_with>balance)
                                {System.out.println("Insufficient funds!");
                                return;}
                                else
                                {balance= balance- c_with;
```

```
                                        System.out.println("Amount has been withdrawn and balance has been
updated");}
                                        if(balance<min)
                                        {
                                                System.out.println("Balance is below the minimum threshold.
Service penalty charge = 100/- .");
                                                if(balance<pen)
                                                System.out.println("Due to insufficient funds, penalty charge
will be deducted from account after replenishing. Current balance is "+balance);
                                                else
                                                {
                                                        balance= balance-pen;
                                                        System.out.println("Penalty charge has been deducted
from account balance. Current balance is "+balance);
                                                }
                                        }
        }


}

class lab5
{
        public static void main(String sss[])
        {
                int cch, chh;
                Scanner sx = new Scanner(System.in);
                System.out.println("--------Welcome----------");
                System.out.println("Savings account or current account? 1- Savings; 2- Current");
                int ch= sx.nextInt();
                if(ch==1)
                {
                        Saving s = new Saving();
                        s.get('S');
                        do{
                        System.out.println("1. Deposit money\n2. Calculate compound interest\n3.
Withdraw money\n4. Display balance\n5. Cheque book facility\n6. Exit");
                        System.out.println("Enter your choice");
                        chh= sx.nextInt();
                        switch(chh)
                        {
                                case 1:
                                s.dep();
                                break;
```

```
                    case 2:
                    s.ci();
                    break;

                    case 3:
                    s.with_s();
                    break;

                    case 4:
                    s.display();
                    break;

                    case 5:
                    s.check();
                    break;

                    case 6:
                    break;

                    default:
                    System.out.println("Wrong option.");
                    break;
                }
                }while(chh!=6);

        }
        else if(ch==2)
        {
                Current cr = new Current();
                cr.get('C');
                do{
                System.out.println("1. Deposit money\n2. Chequebook facility\n3. Withdraw
money\n4. Display balance\n5. Exit");
                cch= sx.nextInt();
                switch(cch)
                {
                    case 1:
                    cr.dep();
                    break;

                    case 2:
                    cr.check();
                    break;
```

```
                        case 3:
                        cr.with_c();
                        break;

                        case 4:
                        cr.display();
                        break;

                        case 5:
                        break;

                        default:
                        System.out.println("Wrong option.");
                        break;
                    }
                }while(cch!=5);



                }
                else System.out.println("Wrong!");
            }
}
```

OUTPUT



LAB PROGRAM 6

QUESTION

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.

CODE

Internals.java

```
package CIE;
import java.util.Scanner;
public class Internals extends CIE.Student
{
   public int ciem[]=new int[5];
   Scanner xx =new Scanner (System.in);
   public void accept()
   {
      for(int i=0;i<5;i++)
      {
         System.out.println("Enter the cie marks of subject"+(i+1)+" out of 50");
         {
            ciem[i]=xx.nextInt();
         }
      }
   }


}
```

Externals.java

```
package SEE;
import CIE.*;
import java.util.Scanner;

public class Externals extends CIE.Student
{
   public int seem[]=new int[5];
   Scanner xx =new Scanner (System.in);
   public void accept()
   {
      for(int i=0;i<5;i++)
      {
         System.out.println("Enter the see marks of subject"+(i+1)+" out of 100");
         {
            seem[i]=xx.nextInt();
         }
      }
   }

}
```

Student.java

```
package CIE;
import java.util.Scanner;

public class Student
{
   String name,usn;
   int sem;
   Scanner xx=new Scanner(System.in);
   public void accept()
   {
      System.out.println("Enter name:");
      name=xx.nextLine();
      System.out.println("Enter usn:");
      usn=xx.next();
      System.out.println("Enter sem:");
      sem=xx.nextInt();
   }
   public void display()
```

```
    {
        System.out.println("Name :"+name);
        System.out.println("Usn :"+usn);
        System.out.println("Sem :"+sem);
    }
}
```

TotalMarks.java

```
import CIE.*;
import SEE.*;
import java.util.*;

class TotalMarks
{
    public static void main(String sss[])
    {
        int i,j,n;
        int total[]=new int[5];
        Scanner xx=new Scanner(System.in);
        System.out.println("Enter the number of students");
        n=xx.nextInt();
        CIE.Student s[]=new CIE.Student[n];
        CIE.Internals ci[]= new CIE.Internals[n];
                SEE.Externals se[]=new SEE.Externals[n];
        for(i=0;i<n;i++)
        {
            System.out.println("ENTER STUDENT"+(i+1)+" DETAILS");
            s[i]=new CIE.Student();
            s[i].accept();
            ci[i]=new CIE.Internals();
            ci[i].accept();
            se[i]=new SEE.Externals();
            se[i].accept();
        }
        for(i=0;i<n;i++)
        {
            System.out.println("DETAILS OF STUDENT "+(i+1));
            s[i].display();
```

```
        for(j=0;j<5;j++)
        {
           total[j]=ci[i].ciem[j]+(se[i].seem[j]/2);
           System.out.println("Total marks in subject"+(j+1)+" is "+total[j]);
        }
     }
  }
}
```

OUTPUT

LAB PROGRAM 7

QUESTION

Write a program to demonstrate generics with multiple object parameters.

CODE

```
class mulgen<T, V>
{
T ob1;
V ob2;

mulgen(T o1, V o2)
{
ob1 = o1;
ob2 = o2;
}
```

```
void show() {
System.out.println("Type of T is " +ob2.getClass().getName());
System.out.println("Type of V is " +ob1.getClass().getName());
}
T getob1() {
return ob1;
}
V getob2() {
return ob2;
}
}

class LAB7 {
public static void main(String args[]) {

mulgen<Integer, String> tgObj =new mulgen<Integer, String>(10, "Avirath");


tgObj.show();

String str = tgObj.getob2();
System.out.println("value: " + str);
int v = tgObj.getob1();
System.out.println("value: " + v);

System.out.println("       ");

mulgen<Integer, String> Obj =new mulgen<Integer, String>(101, "ok");

Obj.show();

str = Obj.getob2();
System.out.println("value: " + str);
v = Obj.getob1();
System.out.println("value: " + v);
}
}
```

OUTPUT

LAB PROGRAM 8

QUESTION

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 Wrong Age( ) when the input age<0. In Son class, implement a constructor that cases both father and son's age and throws an exception if son's age is >=father's age.

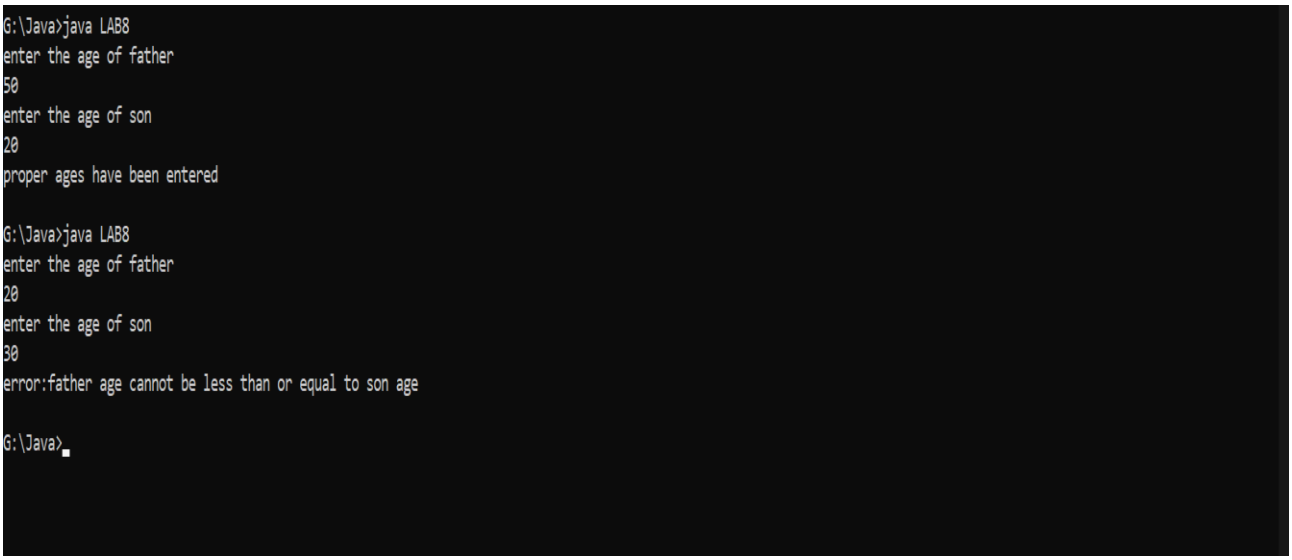
CODE

```
import java.util.*;
class WrongAge extends Exception
{
        private int a1,b1;
        WrongAge(int a,int b)
        {
```

```
                a1=a;
                b1=b;
        }
        public String toString()
        {
    if(a1<0||b1<0)
    return "input age cannot be less than 0";
    else if(a1<=b1)
        return "father age cannot be less than or equal to son age ";
    return "";
        }
}

class Father
{
        int fage,sage;
        Scanner sc=new Scanner(System.in);
  Father() throws WrongAge
  {
        System.out.println("enter the age of father");
        fage=sc.nextInt();
        System.out.println("enter the age of son");
        sage=sc.nextInt();
        if(fage<0||sage<0)
                throw new WrongAge(fage,sage);



  }
}

class Son extends Father
{

  Son() throws WrongAge
  {

    if(sage>=fage)
                throw new WrongAge(fage,sage);
        else
        System.out.println("proper ages have been entered");
  }
}
```

```
class LAB8
{
	public static void main(String args[])
	{

		try
		{
	Son s=new Son();
		}catch(WrongAge e){
			System.out.println("error:"+e);

		}
	}
}
```

OUTPUT



LAB PROGRAM 9

QUESTION

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.

CODE

```
class NewThread implements Runnable
{ Thread t;
  NewThread()
  {
    t = new Thread(this, "NThread");
          t.start();
  }

 public void run()
  {
            try
            {
                    for(int n=100;n>0;n--)
                    {
                            System.out.println("CSE");
                            Thread.sleep(2000);
                    }
            }
            catch(InterruptedException ie)
            {
                    System.out.println("Child Thread Interrupted");
            }

  }
}

class LAB9
{
          public static void main(String ss[])
          {
                  NewThread n1=new NewThread();
                  //n1.t.start();


                  try
                  {
                          for(int n=5;n>0;n--)
                          {
                                  System.out.println("BMS College of Engineering");
                                  Thread.sleep(10000);
```

```
                }
            }
            catch(InterruptedException ie)
            {
                System.out.println("Main Thread interrupted");
            }

        }
}
```

OUTPUT

LAB PROGRAM 10

QUESTION

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 Arithmetic Exception Display the exception in a message dialog box.

CODE

```
import java.awt.*;
import java.awt.event.*;

public class Lab10 extends Frame implements ActionListener{
	TextField t1,t2;
	String msg="";
	Button btn;
	Lab10(){
		Label l1 = new Label("First Number: ",Label.RIGHT);
		t1 = new TextField(10);
		Label l2 = new Label("Second Number: ",Label.RIGHT);

		t2 = new TextField(10);
		btn = new Button("Submit");
		//Label l = new Label("Updates:");
		l1.setBackground(Color.YELLOW);
		l2.setBackground(Color.YELLOW);
		//this.setResizable(false);
		this.add(l1);
		this.add(t1);
		this.add(l2);
		this.add(t2);
		//the following command will make sure that the input char is not visible to the user
		//(it has been added just to demonstrate). Can be used for passwords.
		//t1.setEchoChar('*');
		//t2.setEchoChar('#');
		this.add(btn,BorderLayout.CENTER);
		this.setVisible(true);
		this.setSize(600, 300);
		this.setLayout(new FlowLayout(FlowLayout.CENTER,20,10));
		//t1.addActionListener(this);
		btn.addActionListener(this);
		addWindowListener(new MyWindow());
		setBackground(Color.YELLOW);
		//System.out.println(BorderLayout.CENTER);
	}
	@Override
	public Insets getInsets() {
		return new Insets(50,10,10,20);

	}
```

```
	@Override
	public void actionPerformed(ActionEvent e) {

		String st1 = t1.getText();
		String st2 = t2.getText();
		double n1,n2;
		n1 = 0.0;
		n2 = 0.0;
		if(st1.equals("")||st2.equals("")) {

			msg="You cannot leave the text elements blank";
		}else{
			try {
				n1 = Double.parseDouble(st1);
				n2 = Double.parseDouble(st2);
				try {
					double res = n1/n2;
					msg = "Result of division: "+res;
				}catch(ArithmeticException e1) {
					msg = e1.toString();
				}
			}catch(NumberFormatException e2) {
				msg = "Enter only numbers and not other things";
			}
		}
		new MyDialog(this,"Result Dialog",false,msg,n1,n2);
	}
	public static void main(String[] args) {
		new Lab10();
	}
}

class MyDialog extends Dialog implements ActionListener{

	public MyDialog(Frame owner, String title, boolean modal,String msg, double n1, double n2) {
		super(owner, title, modal);
		this.setVisible(true);
		this.setSize(300, 400);
		this.setLayout(new FlowLayout());
		//System.out.println(owner);
		Label l1 = new Label("	Updates on the result:	");
		//l1.setSize(300, 20);
		this.add(l1);
```

```
		this.add(new Label("First Number: "+n1));
		this.add(new Label("Second Number: "+n2));
		this.add(new Label(msg));

		Button b = new Button("Close");
		this.add(b);
		b.addActionListener(this);
		this.addWindowListener(new WindowAdapter() {
			public void windowClosing(WindowEvent e) {
				dispose();
			}
		});
	}

	@Override
	public void actionPerformed(ActionEvent e) {
		dispose();

	}


}

class MyWindow extends WindowAdapter{
	public void windowClosing(WindowEvent e) {
		System.exit(0);
	}
}
```

OUTPUT