VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted by

Aarush Garg (1BM23CS004)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
COMPUTER SCIENCE AND ENGINEERING

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)

BENGALURU-560019
Sep-2024 to Jan-2025

B.M.S. College of Engineering,

Bull Temple Road, Bangalore 560019
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Lab work entitled "Object Oriented Java Programming (23CS3PCOOJ)" carried out by **Aarush Garg (1BM23CS004),** who is bonafide student of **B.M.S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum. The Lab report has been approved as it satisfies the academic requirements in respect of an Object Oriented Java Programming (23CS3PCOOJ) work prescribed for the said degree.

Geetha N Assistant Professor Department of CSE, BMSCE	Dr. Jyothi S Nayak Professor & HOD Department of CSE, BMSCE

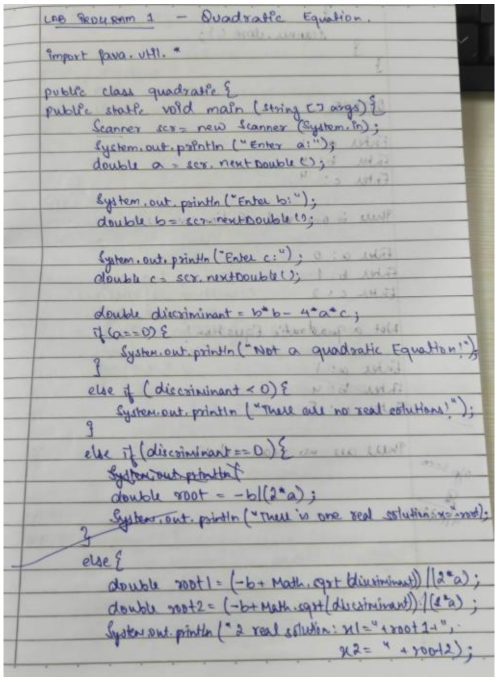
Index

GitHub Link:

Program 1

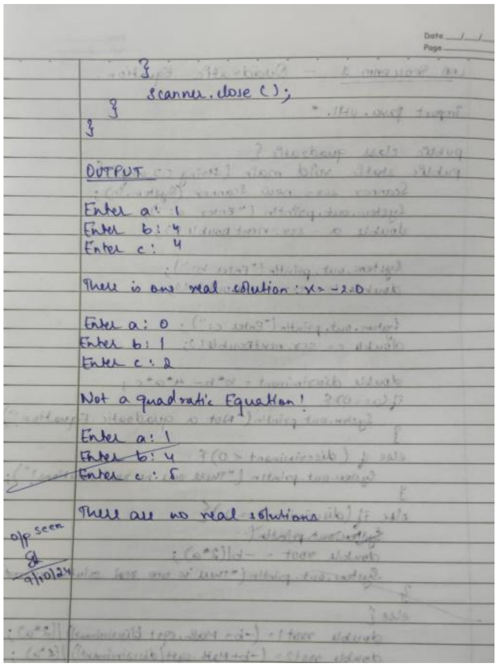
Implement Quadratic Equation

Algorithm:



LAB PROGRAM 1 — Quadratic Equation.

```
import java.util.*

public class quadratic {
public static void main (String [] args) {
        Scanner scr = new Scanner (System.in);
        System.out.println ("Enter a:");
        double a = scr.nextDouble ();

        System.out.println ("Enter b:");
        double b = scr.nextDouble ();

        System.out.println ("Enter c:");
        double c = scr.nextDouble ();

        double discriminant = b*b - 4*a*c;
        if (a==0) {
            System.out.println ("Not a quadratic Equation!");
        }
        else if (discriminant < 0) {
            System.out.println ("There are no real solutions!");
        }
        else if (discriminant == 0) {
            ~~System.out.printlnf~~
            double root = -b/(2*a);
            System.out.println ("There is one real solution: x="+root);
        }
        else {
            double root1 = (-b + Math.sqrt (discriminant)) / (2*a);
            double root2 = (-b + Math.sqrt (discriminant)) / (2*a);
            System.out.println (" 2 real solution: x1 = "+root1+",
                                          x2 = " +root2);
```



Date __/__/__
Page ____

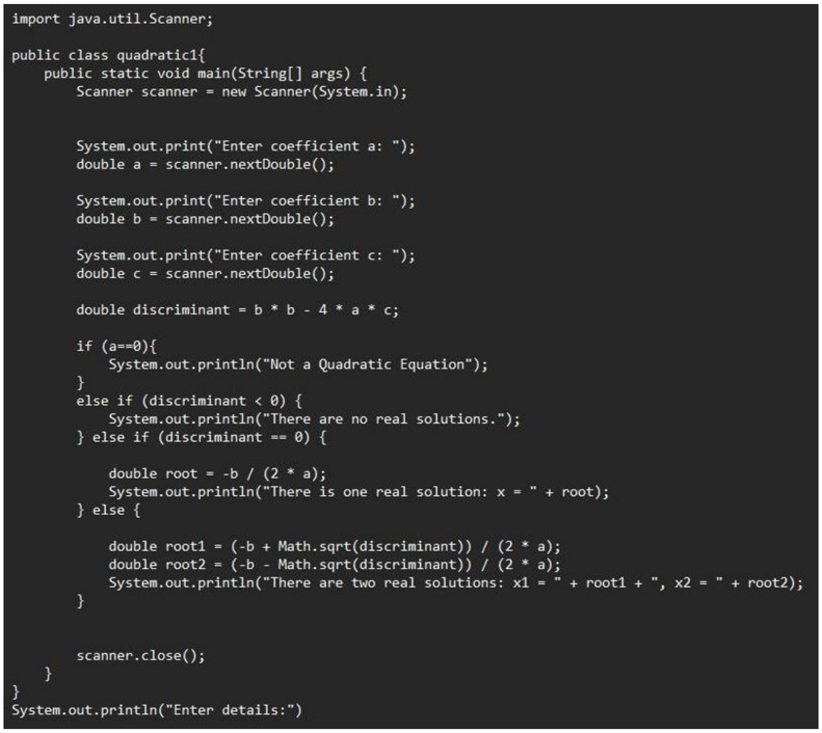
```
        }
        Scanner.close ();
    }
}
```

OUTPUT

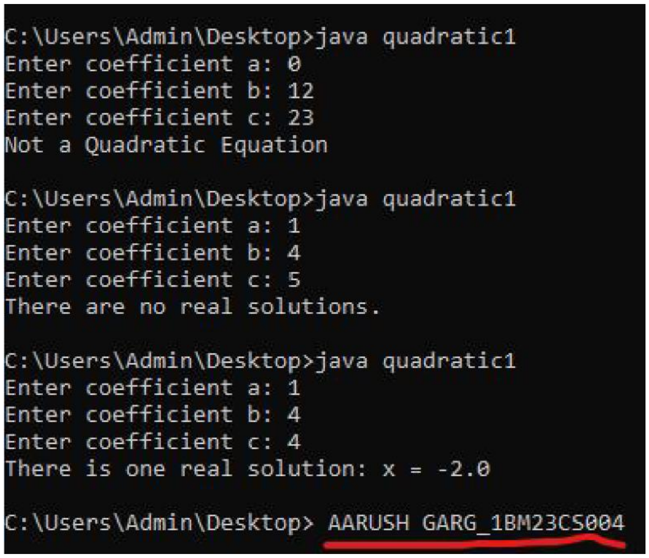
```
Enter a: 1
Enter b: 4
Enter c: 4

There is one real solution: x= -2.0

Enter a: 0
Enter b: 1
Enter c: 2

Not a quadratic Equation!

Enter a: 1
Enter b: 4
Enter c: 5

There are no real solutions
```

o/p seen
9/10/24

Code:



```
import java.util.Scanner;

public class quadratic1{
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);


        System.out.print("Enter coefficient a: ");
        double a = scanner.nextDouble();

        System.out.print("Enter coefficient b: ");
        double b = scanner.nextDouble();

        System.out.print("Enter coefficient c: ");
        double c = scanner.nextDouble();

        double discriminant = b * b - 4 * a * c;

        if (a==0){
            System.out.println("Not a Quadratic Equation");
        }
        else if (discriminant < 0) {
            System.out.println("There are no real solutions.");
        } else if (discriminant == 0) {

            double root = -b / (2 * a);
            System.out.println("There is one real solution: x = " + root);
        } else {

            double root1 = (-b + Math.sqrt(discriminant)) / (2 * a);
            double root2 = (-b - Math.sqrt(discriminant)) / (2 * a);
            System.out.println("There are two real solutions: x1 = " + root1 + ", x2 = " + root2);
        }


        scanner.close();
    }
}
System.out.println("Enter details:")
```

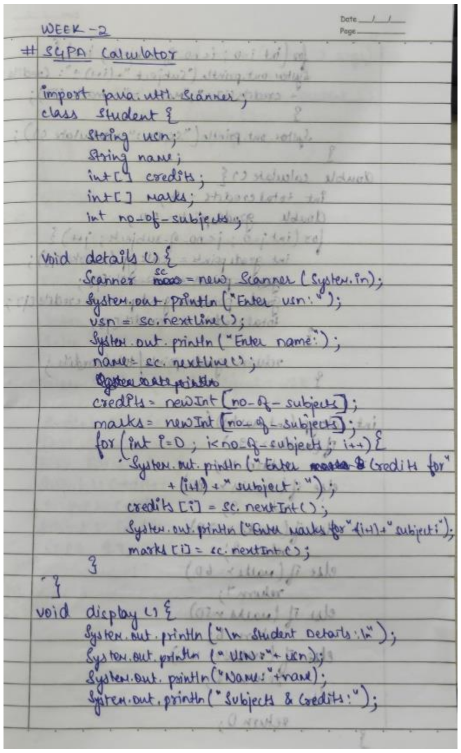


```
C:\Users\Admin\Desktop>java quadratic1
Enter coefficient a: 0
Enter coefficient b: 12
Enter coefficient c: 23
Not a Quadratic Equation

C:\Users\Admin\Desktop>java quadratic1
Enter coefficient a: 1
Enter coefficient b: 4
Enter coefficient c: 5
There are no real solutions.

C:\Users\Admin\Desktop>java quadratic1
Enter coefficient a: 1
Enter coefficient b: 4
Enter coefficient c: 4
There is one real solution: x = -2.0

C:\Users\Admin\Desktop> AARUSH GARG_1BM23CS004
```

<u>Program 2</u>

SGPA Calculator

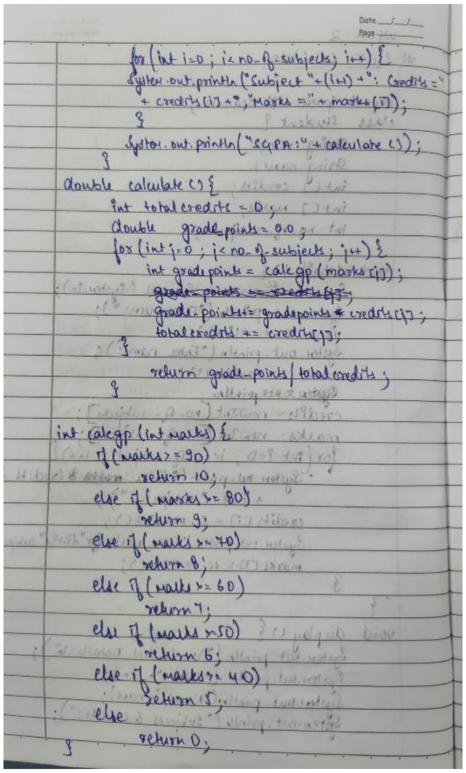
Algorithm:



WEEK - 2

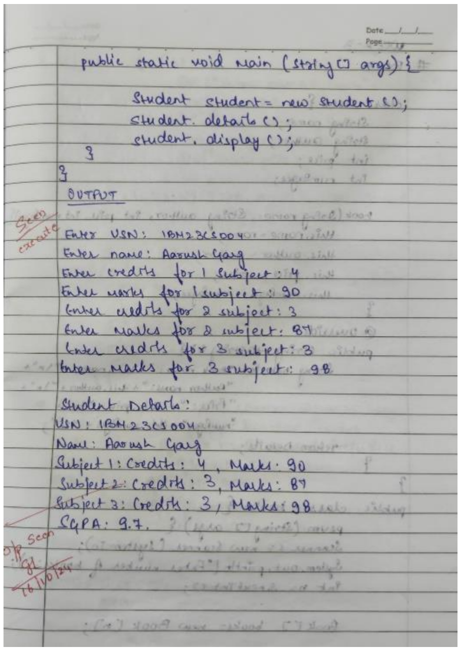
Date __/__/__
Page ____

```
# SGPA Calculator

import java.util.Scanner;
class Student {
        String usn;
        String name;
        int[] credits;
        int[] marks;
        int no-of-subjects;

void details () {
      Scanner ~~maas~~ sc = new Scanner (System.in);
      System.out.println ("Enter usn:");
      usn = sc.nextLine();
      System.out.println ("Enter name:");
      name = sc.nextLine();
      ~~System.out.println~~
      credits = new Int [no-of-subjects];
      marks = new Int [no-of-subjects];
      for (int i=0; i<no-of-subjects; i++) {
          System.out.println ("Enter ~~marks &~~ Credits for"
                + (i+1) + " subject:");
          credits [i] = sc.nextInt();
          System.out.println ("Enter marks for" + (i+1) + "subject:");
          marks [i] = sc.nextInt();
      }
  }
void display () {
      System.out.println ("\n Student Details:\n");
      System.out.println (" USN:" + usn);
      System.out.println ("Name:" + name);
      System.out.println ("Subjects & Credits:");
```



Date __/__/__
Page ____

```
          for (int i=0; i<no-of-subjects; i++) {
          System.out.println ("Subject " + (i+1) + ": Credits ="
           + credits[i] + ", Marks =" + marks[i]);
           }
          System.out.println ("SGPA:" + calculate ());
      }
double calculate () {
        int totalcredits = 0;
        double  grade_points = 0.0;
        for (int j=0; j<no-of-subjects; j++) {
              int gradepoints = calcgp (marks [j]);
              ~~grade_points += credits[j];~~
              grade_points += gradepoints * credits[j];
              totalcredits += credits[j];
        }
              return grade_points / totalcredits;
        }

int calcgp (int marks) {
     if (marks >= 90)
          return 10;
     else if (marks >= 80)
          return 9;
     else if (marks >= 70)
          return 8;
     else if (marks >= 60)
          return 7;
     else if (marks >= 50)
          return 6;
     else if (marks >= 40)
          return 5;
     else
          return 0;
 }
```



Date __/__/__
Page ____

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

        Student student= new Student ();
        student. details ();
        student. display ();
    }
}
```

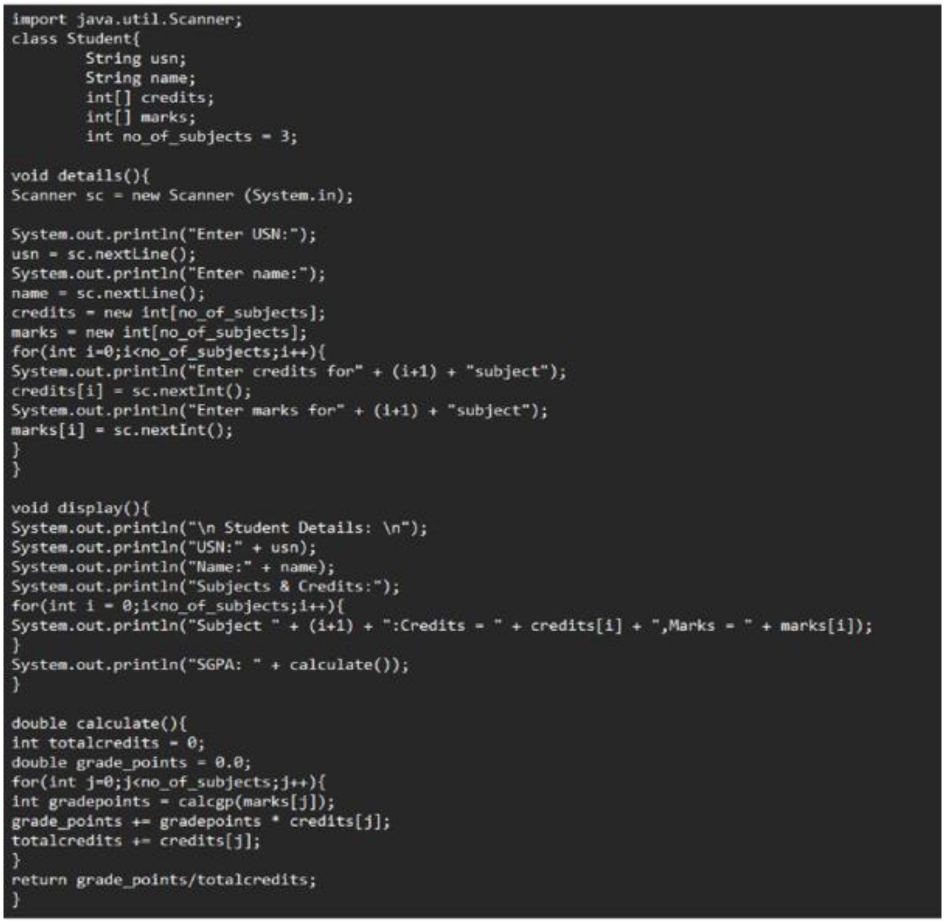
OUTPUT

Seen
execute

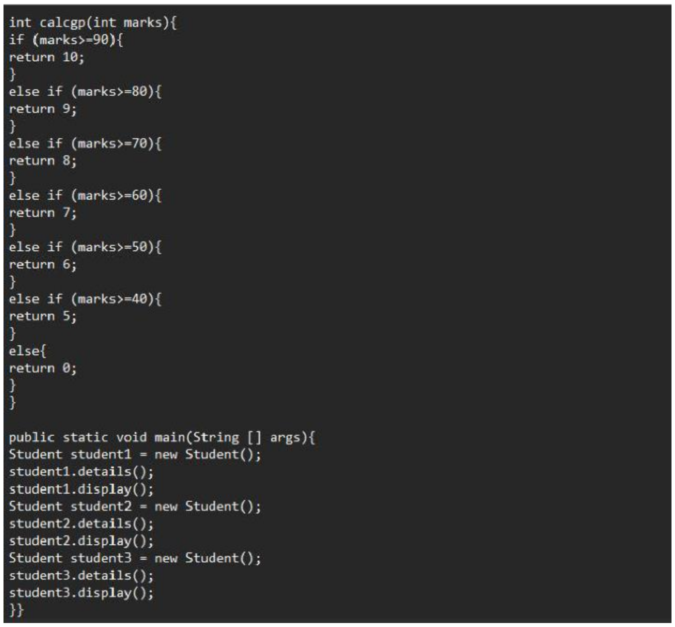
```
Enter USN: 1BM23CS004
Enter name: Aarush Garg
Enter credits for 1 Subject: 4
Enter marks for 1 subject: 90
Enter credits for 2 subject: 3
Enter marks for 2 subject: 87
Enter credits for 3 subject: 3
Enter marks for 3 subject: 98

Student Details:
USN: 1BM23CS004
Name: Aarush Garg
Subject 1: Credits: 4, Marks: 90
Subject 2: Credits: 3, Marks: 87
Subject 3: Credits: 3, Marks: 98
SGPA: 9.7.
```

o/p Seen
81
16/10/24

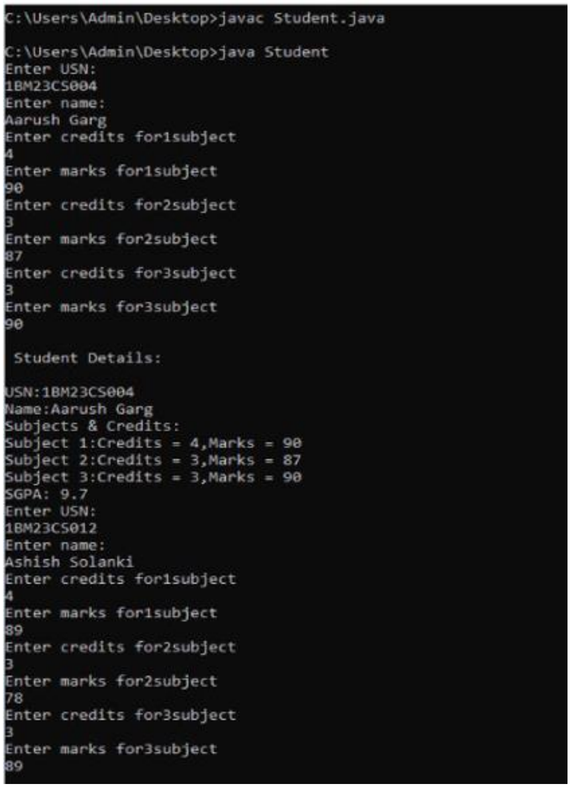
Code:



```
import java.util.Scanner;
class Student{
        String usn;
        String name;
        int[] credits;
        int[] marks;
        int no_of_subjects = 3;

void details(){
Scanner sc = new Scanner (System.in);

System.out.println("Enter USN:");
usn = sc.nextLine();
System.out.println("Enter name:");
name = sc.nextLine();
credits = new int[no_of_subjects];
marks = new int[no_of_subjects];
for(int i=0;i<no_of_subjects;i++){
System.out.println("Enter credits for" + (i+1) + "subject");
credits[i] = sc.nextInt();
System.out.println("Enter marks for" + (i+1) + "subject");
marks[i] = sc.nextInt();
}
}

void display(){
System.out.println("\n Student Details: \n");
System.out.println("USN:" + usn);
System.out.println("Name:" + name);
System.out.println("Subjects & Credits:");
for(int i = 0;i<no_of_subjects;i++){
System.out.println("Subject " + (i+1) + ":Credits = " + credits[i] + ",Marks = " + marks[i]);
}
System.out.println("SGPA: " + calculate());
}

double calculate(){
int totalcredits = 0;
double grade_points = 0.0;
for(int j=0;j<no_of_subjects;j++){
int gradepoints = calcgp(marks[j]);
grade_points += gradepoints * credits[j];
totalcredits += credits[j];
}
return grade_points/totalcredits;
}
```



```
int calcgp(int marks){
if (marks>=90){
return 10;
}
else if (marks>=80){
return 9;
}
else if (marks>=70){
return 8;
}
else if (marks>=60){
return 7;
}
else if (marks>=50){
return 6;
}
else if (marks>=40){
return 5;
}
else{
return 0;
}
}

public static void main(String [] args){
Student student1 = new Student();
student1.details();
student1.display();
Student student2 = new Student();
student2.details();
student2.display();
Student student3 = new Student();
student3.details();
student3.display();
}}
```

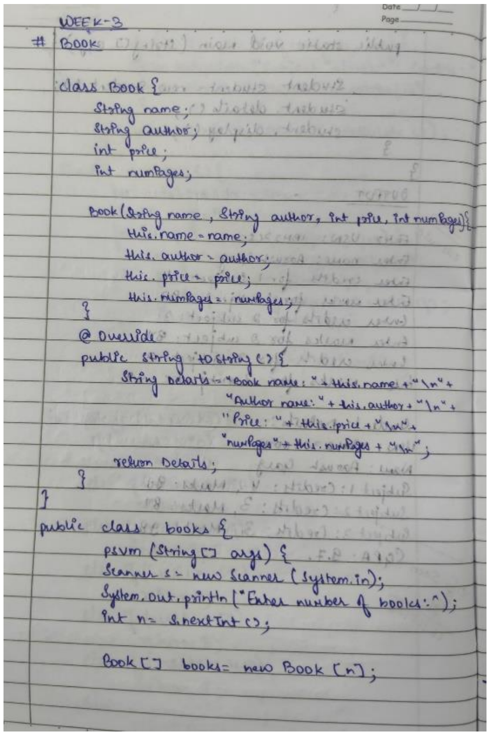


```
C:\Users\Admin\Desktop>javac Student.java

C:\Users\Admin\Desktop>java Student
Enter USN:
1BM23CS004
Enter name:
Aarush Garg
Enter credits for1subject
4
Enter marks for1subject
90
Enter credits for2subject
3
Enter marks for2subject
87
Enter credits for3subject
3
Enter marks for3subject
90

 Student Details:

USN:1BM23CS004
Name:Aarush Garg
Subjects & Credits:
Subject 1:Credits = 4,Marks = 90
Subject 2:Credits = 3,Marks = 87
Subject 3:Credits = 3,Marks = 90
SGPA: 9.7
Enter USN:
1BM23CS012
Enter name:
Ashish Solanki
Enter credits for1subject
4
Enter marks for1subject
89
Enter credits for2subject
3
Enter marks for2subject
78
Enter credits for3subject
3
Enter marks for3subject
89
```

Program 3

Book Class

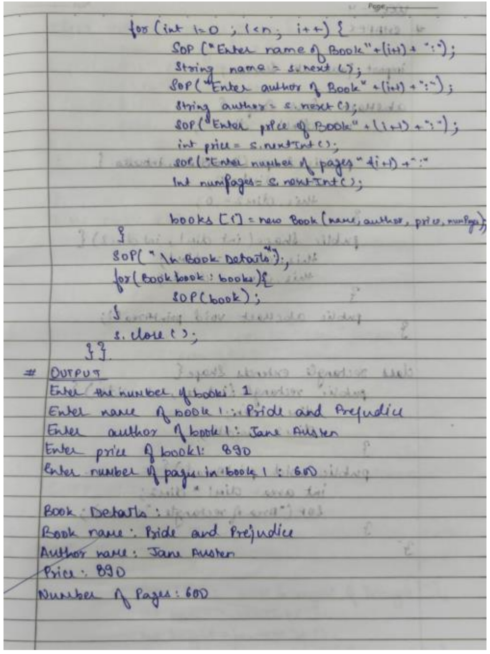
Algorithm:



WEEK-3

Book

```
class Book {
    String name;
    String author;
    int price;
    int numPages;

    Book(String name , String author, int price, int numPages){
        this.name = name;
        this.author = author;
        this.price = price;
        this.numPages = numPages;
    }
    @ Override
    public String toString () {
        String Details = "Book name: " + this.name + "\n" +
                         "Author name: " + this.author + "\n" +
                         "Price: " + this.price + "\n" +
                         "numPages" + this.numPages + "\n";
        return Details;
    }
}
public class books {
    psvm (String [] args) {
    Scanner s = new Scanner (System.in);
    System.out.println ("Enter number of books:");
    int n= s.nextInt ();

    Book [] books= new Book [n];
```

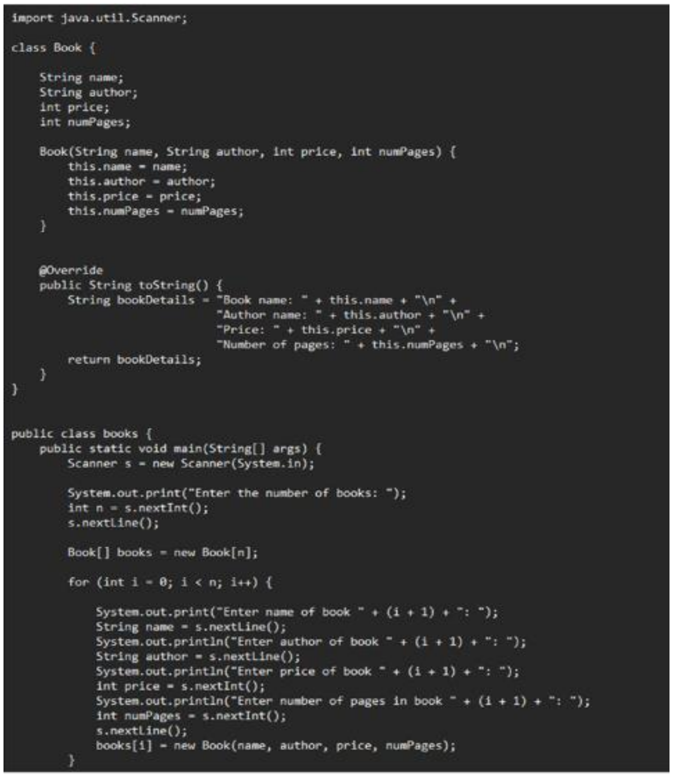


```
        for (int i=0 ; i<n; i++) {
            SOP ("Enter name of Book"+(i+1) + ":");
            String name = s.next ();
            SOP ("Enter author of Book" +(i+1) + ":");
            String author= s.next ();
            SOP ("Enter price of Book" +(i+1) + ":");
            int price= s.nextInt ();
            SOP ("Enter number of pages" +(i+1) + ":"
            int numPages= s.nextInt ();

            books [i] = new Book (name, author, price, numPages);
        }
        SOP (" \n Book Details");
        for(Book book : books){
            SOP(book);
        }
        s. close ();
    }}
```

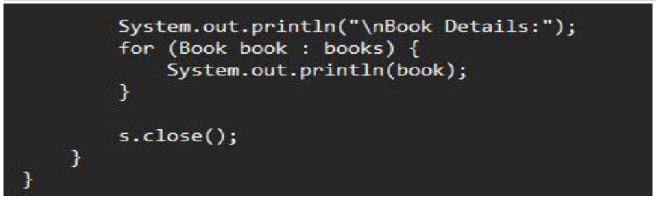
OUTPUT

```
Enter the number of books: 1
Enter name of book 1 :: Pride and Prejudice
Enter author of book 1: Jane Austen
Enter price of book1: 890
Enter number of pages in book 1 : 600

Book Details :
Book name : Pride and Prejudice
Author name : Jane Austen
Price : 890
Number of Pages : 600
```

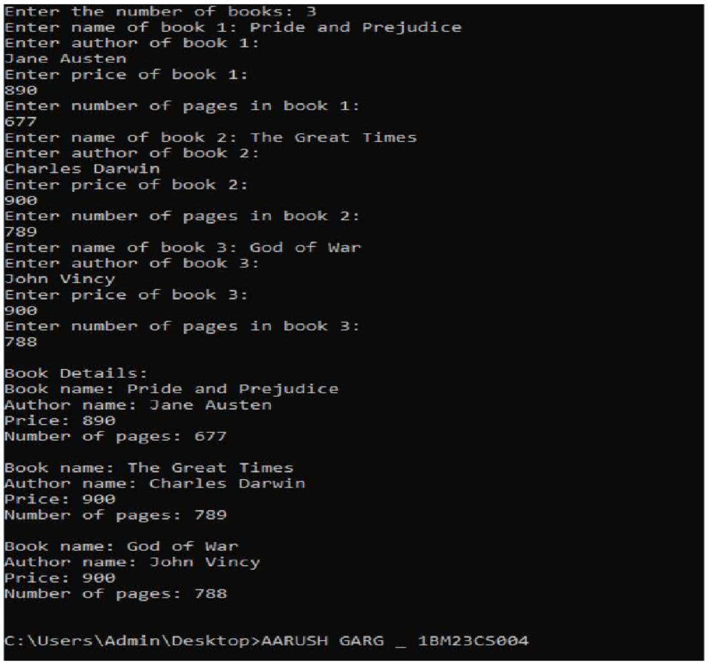
Code:



```
import java.util.Scanner;

class Book {

    String name;
    String author;
    int price;
    int numPages;

    Book(String name, String author, int price, int numPages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.numPages = numPages;
    }


    @Override
    public String toString() {
        String bookDetails = "Book name: " + this.name + "\n" +
                             "Author name: " + this.author + "\n" +
                             "Price: " + this.price + "\n" +
                             "Number of pages: " + this.numPages + "\n";
        return bookDetails;
    }
}


public class books {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);

        System.out.print("Enter the number of books: ");
        int n = s.nextInt();
        s.nextLine();

        Book[] books = new Book[n];

        for (int i = 0; i < n; i++) {

            System.out.print("Enter name of book " + (i + 1) + ": ");
            String name = s.nextLine();
            System.out.println("Enter author of book " + (i + 1) + ": ");
            String author = s.nextLine();
            System.out.println("Enter price of book " + (i + 1) + ": ");
            int price = s.nextInt();
            System.out.println("Enter number of pages in book " + (i + 1) + ": ");
            int numPages = s.nextInt();
            s.nextLine();
            books[i] = new Book(name, author, price, numPages);
        }
```



```
        System.out.println("\nBook Details:");
        for (Book book : books) {
            System.out.println(book);
        }

        s.close();
    }
}
```

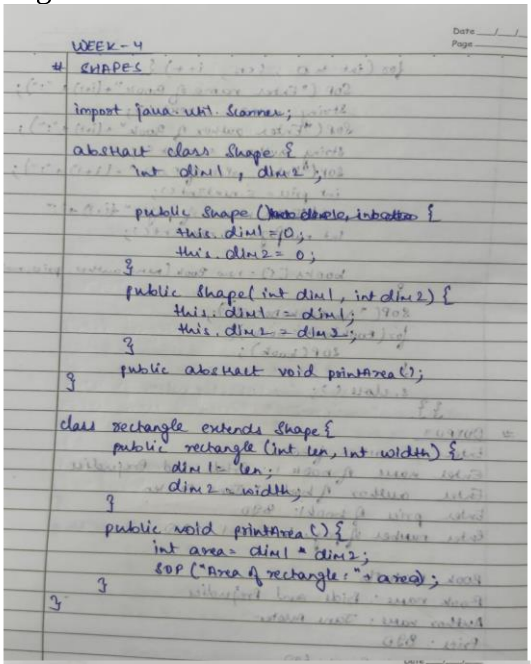


```
Enter the number of books: 3
Enter name of book 1: Pride and Prejudice
Enter author of book 1:
Jane Austen
Enter price of book 1:
890
Enter number of pages in book 1:
677
Enter name of book 2: The Great Times
Enter author of book 2:
Charles Darwin
Enter price of book 2:
900
Enter number of pages in book 2:
789
Enter name of book 3: God of War
Enter author of book 3:
John Vincy
Enter price of book 3:
900
Enter number of pages in book 3:
788

Book Details:
Book name: Pride and Prejudice
Author name: Jane Austen
Price: 890
Number of pages: 677

Book name: The Great Times
Author name: Charles Darwin
Price: 900
Number of pages: 789

Book name: God of War
Author name: John Vincy
Price: 900
Number of pages: 788


C:\Users\Admin\Desktop>AARUSH GARG _ 1BM23CS004
```

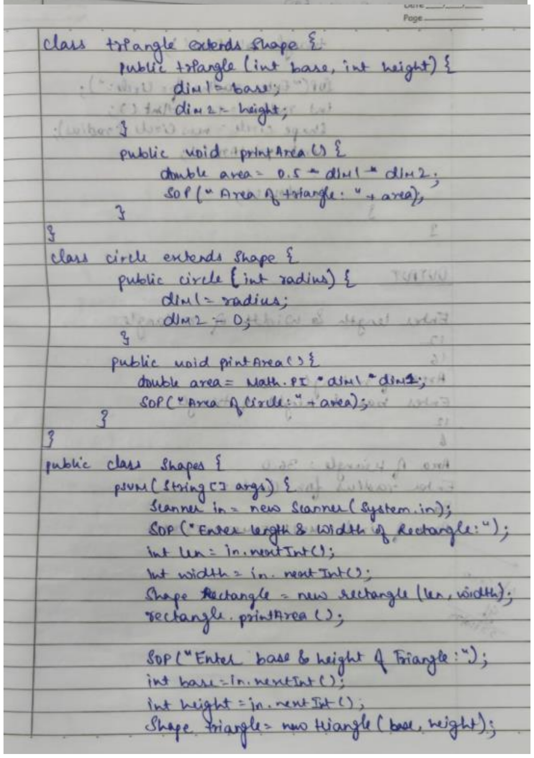
Program 4

Abstract class Shape

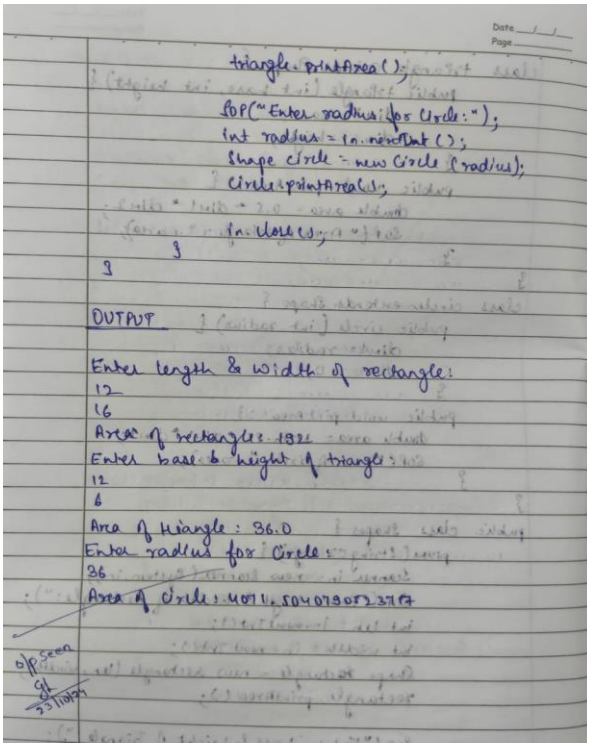
Algorithm:



```
WEEK-4
# SHAPES

import java.util.Scanner;

abstract class Shape {
    int dim1, dim2;

    public Shape () ~~[illegible]~~ {
        this.dim1 = 0;
        this.dim2 = 0;
    }
    public Shape(int dim1, int dim2) {
        this.dim1 = dim1;
        this.dim2 = dim2;
    }
    public abstract void printArea();
}

class rectangle extends Shape {
    public rectangle (int len, int width) {
        dim1 = len;
        dim2 = width;
    }
    public void printArea () {
        int area = dim1 * dim2;
        SOP ("Area of rectangle : " + area);
    }
}
```



```
class triangle extends Shape {
    public triangle (int base, int height) {
        dim1 = base;
        dim2 = height;
    }
    public void printArea () {
        double area = 0.5 * dim1 * dim2;
        SOP (" Area of triangle : " + area);
    }
}
class circle extends Shape {
    public circle (int radius) {
        dim1 = radius;
        dim2 = 0;
    }
    public void printArea () {
        double area = Math.PI * dim1 * dim1;
        SOP (" Area of circle : " + area);
    }
}
public class Shapes {
    psvm (String [] args) {
        Scanner in = new Scanner(System.in);
        SOP ("Enter length & Width of Rectangle: ");
        int len = in.nextInt();
        int width = in.nextInt();
        Shape Rectangle = new rectangle (len, width);
        rectangle.printArea ();

        SOP (" Enter base & height of Triangle : ");
        int base = in.nextInt ();
        int height = in.nextInt ();
        Shape triangle = new triangle (base, height);
```



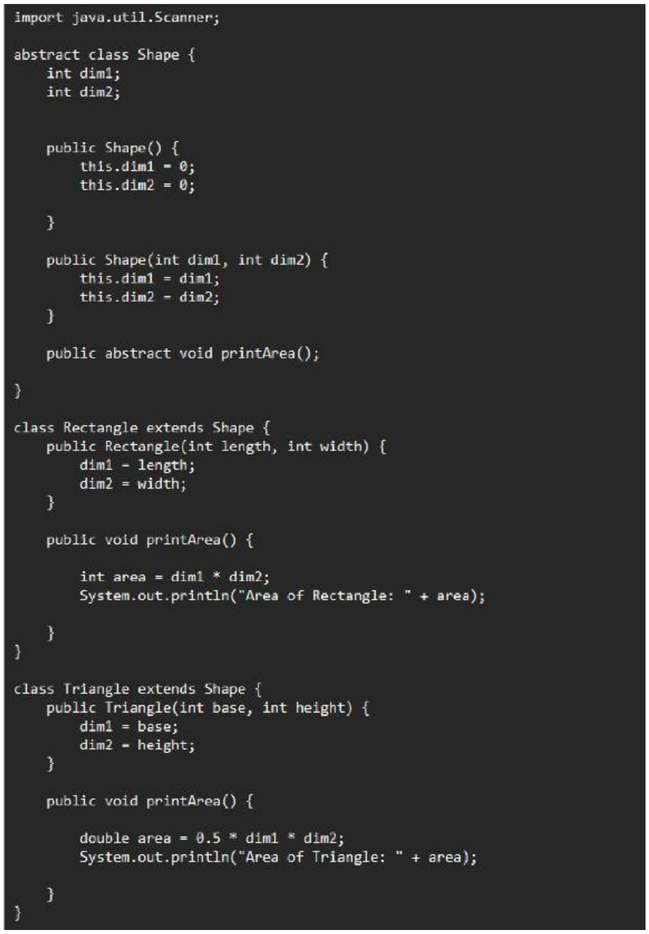
```
        triangle.printArea();

        SOP("Enter radius for Circle:");
        int radius = in.nextInt();
        Shape circle = new Circle (radius);
        Circle.printArea();

        in.close();
    }
}
```

OUTPUT

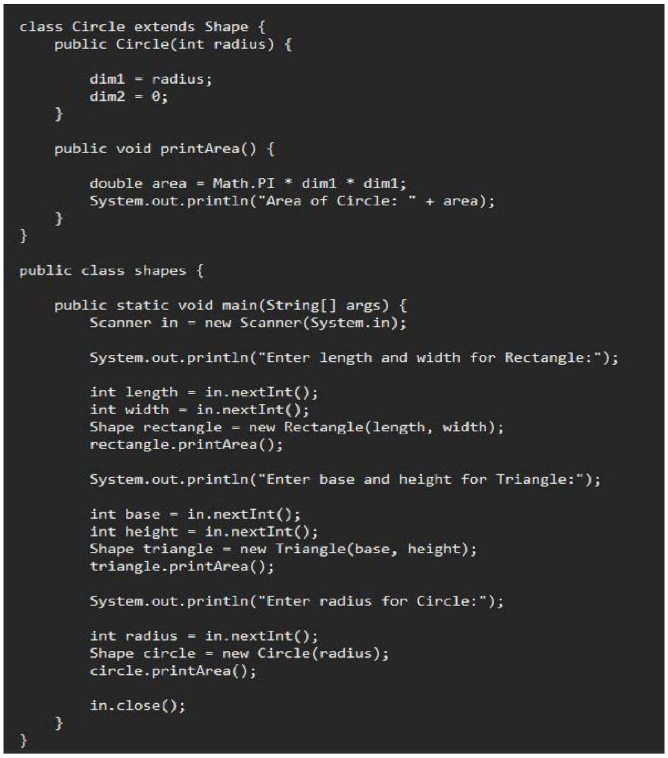
```
Enter length & width of rectangle:
12
16
Area of rectangle: 192
Enter base & height of triangle:
12
6
Area of triangle : 36.0
Enter radius for Circle:
36
Area of Circle: 4071.5040790523717
```

o/p seen
GL
23/10/24

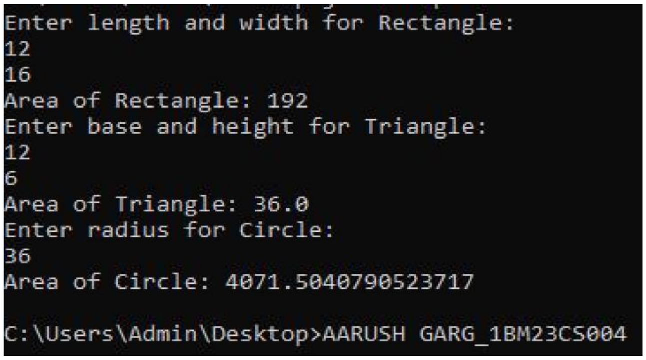
Code:



```
import java.util.Scanner;

abstract class Shape {
    int dim1;
    int dim2;


    public Shape() {
        this.dim1 = 0;
        this.dim2 = 0;

    }

    public Shape(int dim1, int dim2) {
        this.dim1 = dim1;
        this.dim2 = dim2;
    }

    public abstract void printArea();

}

class Rectangle extends Shape {
    public Rectangle(int length, int width) {
        dim1 = length;
        dim2 = width;
    }

    public void printArea() {

        int area = dim1 * dim2;
        System.out.println("Area of Rectangle: " + area);

    }
}

class Triangle extends Shape {
    public Triangle(int base, int height) {
        dim1 = base;
        dim2 = height;
    }

    public void printArea() {

        double area = 0.5 * dim1 * dim2;
        System.out.println("Area of Triangle: " + area);

    }
}
```



```
class Circle extends Shape {
    public Circle(int radius) {

        dim1 = radius;
        dim2 = 0;
    }

    public void printArea() {

        double area = Math.PI * dim1 * dim1;
        System.out.println("Area of Circle: " + area);
    }
}

public class shapes {

    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);

        System.out.println("Enter length and width for Rectangle:");

        int length = in.nextInt();
        int width = in.nextInt();
        Shape rectangle = new Rectangle(length, width);
        rectangle.printArea();

        System.out.println("Enter base and height for Triangle:");

        int base = in.nextInt();
        int height = in.nextInt();
        Shape triangle = new Triangle(base, height);
        triangle.printArea();

        System.out.println("Enter radius for Circle:");

        int radius = in.nextInt();
        Shape circle = new Circle(radius);
        circle.printArea();

        in.close();
    }
}
```

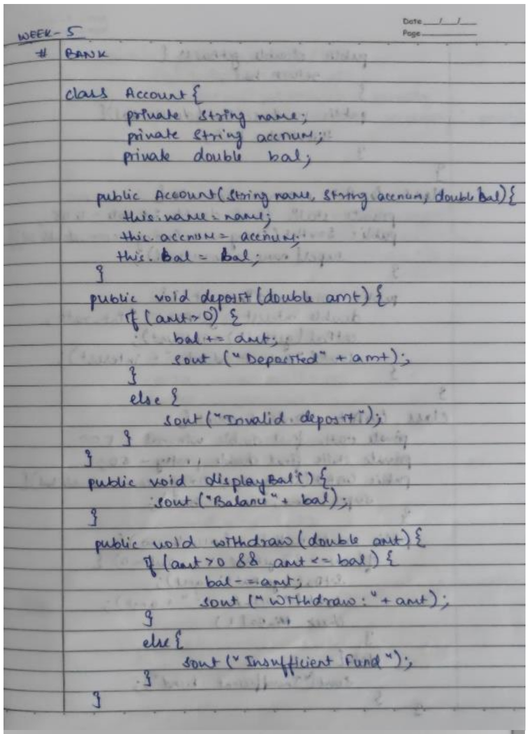


```
Enter length and width for Rectangle:
12
16
Area of Rectangle: 192
Enter base and height for Triangle:
12
6
Area of Triangle: 36.0
Enter radius for Circle:
36
Area of Circle: 4071.5040790523717

C:\Users\Admin\Desktop>AARUSH GARG_1BM23CS004
```

Program 5

Bank Class

Algorithm:

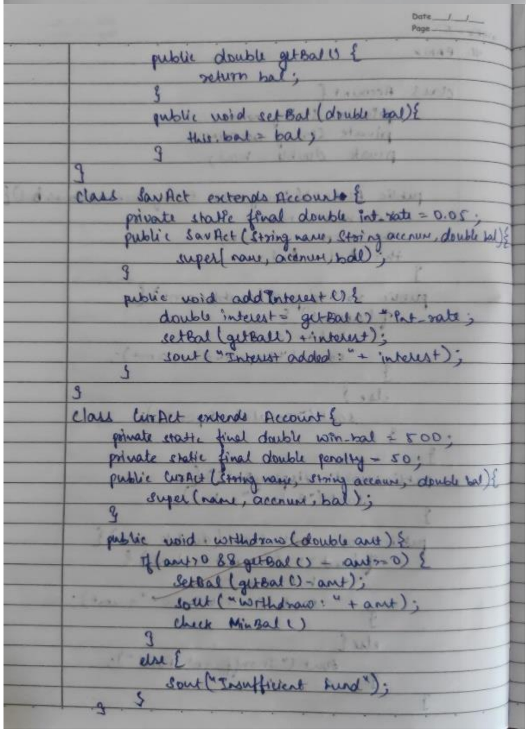


Date __/__/__
Page ____

WEEK-5

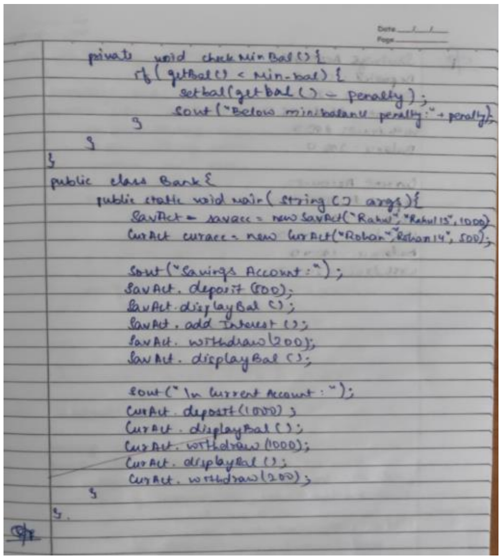
BANK

```
class Account{
    private String name;
    private String accnum;
    private double bal;

  public Account(String name, String accnum, double Bal){
    this.name = name;
    this.accnum = accnum;
    this.Bal = Bal;
  }
  public void deposit(double amt){
     if (amt>0) {
        bal += amt;
        sout ("Deposited" + amt);
     }
     else {
        sout("Invalid deposit");
     }
  }
  public void displayBal(){
       sout ("Balance"+ bal);
  }
  public void withdraw(double amt){
       if (amt>0 && amt <= bal) {
           bal -= amt;
           sout ("Withdraw:"+ amt);
       }
       else{
         sout ("Insufficient Fund");
       }
  }
```

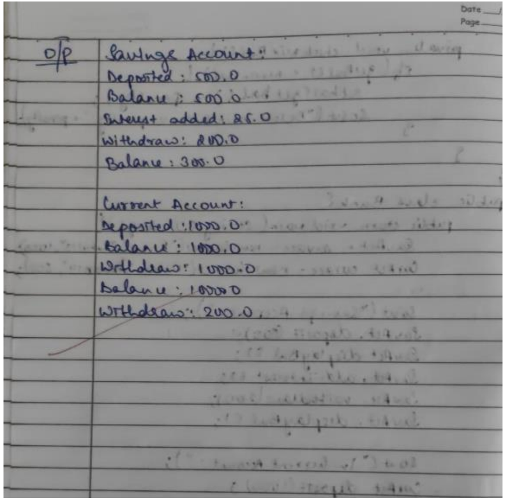


Date __/__/__
Page ____

```
      public double getBal() {
           return bal;
      }
      public void setBal(double bal){
          this.bal = bal;
      }
}
class SavAct extends Account {
    private static final double int_rate = 0.05;
    public SavAct(String name, String accnum, double bal){
         super(name, accnum, bal);
    }
    public void addInterest(){
        double interest = getBal() * int_rate;
        setBal(getBal() + interest);
        sout("Interest added:"+ interest);
    }
}
class CurAct extends Account{
   private static final double min_bal = 500;
   private static final double penalty = 50;
   public CurAct(String name, String accnum, double bal){
       super(name, accnum, bal);
   }
   public void withdraw(double amt){
       if(amt>0 && getBal() - amt>=0) {
           setBal(getBal() - amt);
           sout("Withdraw:" + amt);
           check MinBal()
       }
       else {
         sout("Insufficient fund");
       }
  }
```



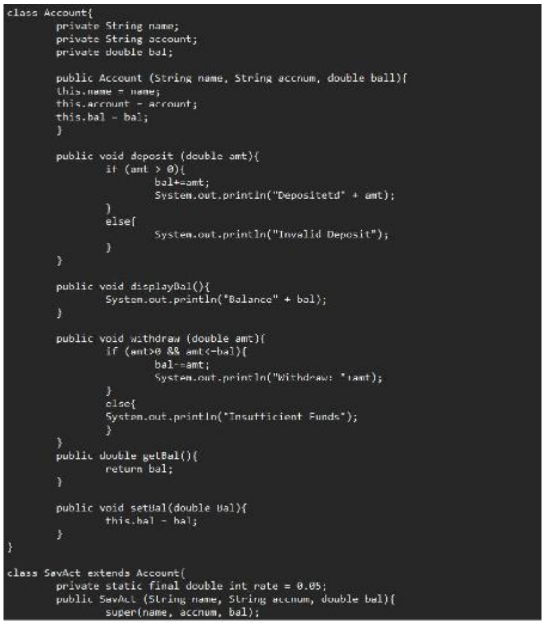
```
    private void checkMinBal(){
        if(getBal() < Min-bal){
            setbal(getbal() - penalty);
            sout("Below minibalance penalty:"+penalty);
        }
    }
}
public class Bank{
    public static void main(String[] args){
        SavAct savacc = new SavAct("Rahul","Rahul15",1000);
        CurAct curacc = new CurAct("Rohan","Rohan14",500);

        sout("Savings Account:");
        SavAct.deposit(500);
        SavAct.displayBal();
        SavAct.add Interest();
        SavAct.withdraw(200);
        SavAct.displayBal();

        sout("\n Current Account:");
        CurAct.deposit(1000);
        CurAct.displayBal();
        CurAct.withdraw(1000);
        CurAct.displayBal();
        CurAct.withdraw(200);
    }
}.
```



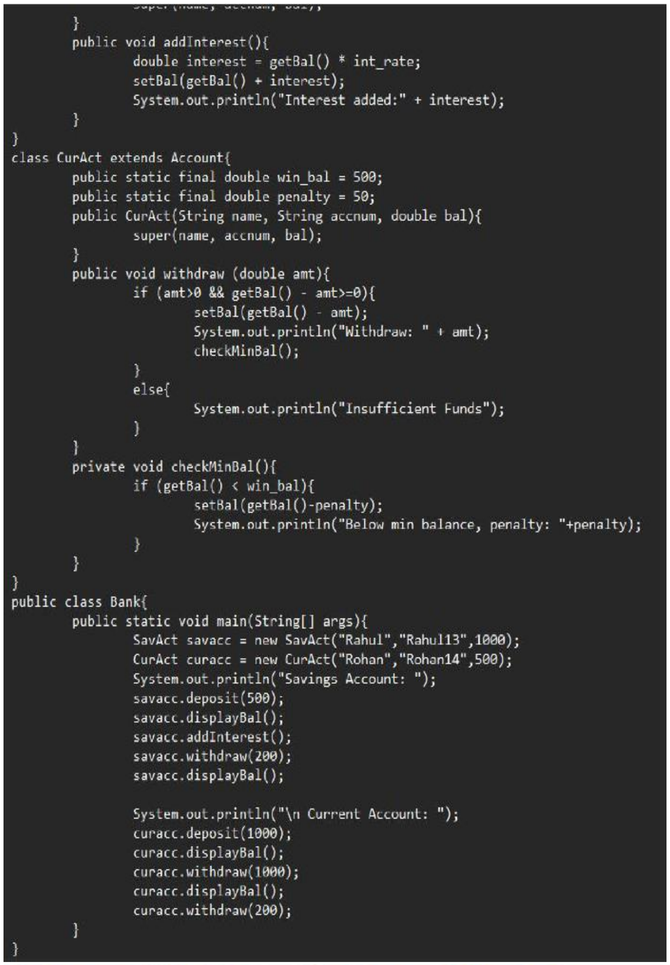
O/P

```
Savings Account:
Deposited: 500.0
Balance: 500.0
Interest added: 25.0
Withdraw: 200.0
Balance: 300.0

Current Account:
Deposited: 1000.0
Balance: 1000.0
Withdraw: 1000.0
Balance: 1000.0
Withdraw: 200.0
```

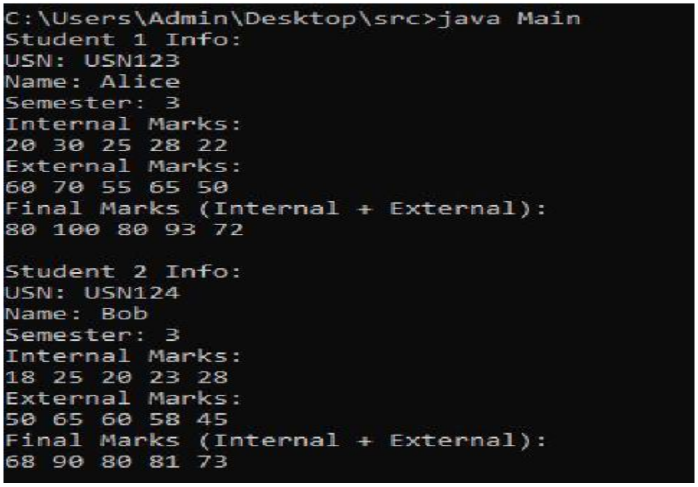
Code:



```
class Account{
        private String name;
        private String account;
        private double bal;

        public Account (String name, String accnum, double ball){
        this.name = name;
        this.account = account;
        this.bal = bal;
        }

        public void deposit (double amt){
                if (amt > 0){
                        bal+=amt;
                        System.out.println("Depositetd" + amt);
                }
                else{
                        System.out.println("Invalid Deposit");
                }
        }

        public void displayBal(){
                System.out.println("Balance" + bal);
        }

        public void withdraw (double amt){
                if (amt>0 && amt<=bal){
                        bal-=amt;
                        System.out.println("Withdraw: "+amt);
                }
                else{
                System.out.println("Insufficient Funds");
                }
        }
        public double getBal(){
                return bal;
        }

        public void setBal(double bal){
                this.bal = bal;
        }
}

class SavAct extends Account{
        private static final double int rate = 0.05;
        public SavAct (String name, String accnum, double bal){
                super(name, accnum, bal);
```



```
                super(name, accnum, bal);
        }
        public void addInterest(){
                double interest = getBal() * int_rate;
                setBal(getBal() + interest);
                System.out.println("Interest added:" + interest);
        }
}
class CurAct extends Account{
        public static final double win_bal = 500;
        public static final double penalty = 50;
        public CurAct(String name, String accnum, double bal){
                super(name, accnum, bal);
        }
        public void withdraw (double amt){
                if (amt>0 && getBal() - amt>=0){
                        setBal(getBal() - amt);
                        System.out.println("Withdraw: " + amt);
                        checkMinBal();
                }
                else{
                        System.out.println("Insufficient Funds");
                }
        }
        private void checkMinBal(){
                if (getBal() < win_bal){
                        setBal(getBal()-penalty);
                        System.out.println("Below min balance, penalty: "+penalty);
                }
        }
}
public class Bank{
        public static void main(String[] args){
                SavAct savacc = new SavAct("Rahul","Rahul13",1000);
                CurAct curacc = new CurAct("Rohan","Rohan14",500);
                System.out.println("Savings Account: ");
                savacc.deposit(500);
                savacc.displayBal();
                savacc.addInterest();
                savacc.withdraw(200);
                savacc.displayBal();

                System.out.println("\n Current Account: ");
                curacc.deposit(1000);
                curacc.displayBal();
                curacc.withdraw(1000);
                curacc.displayBal();
                curacc.withdraw(200);
        }
}
```

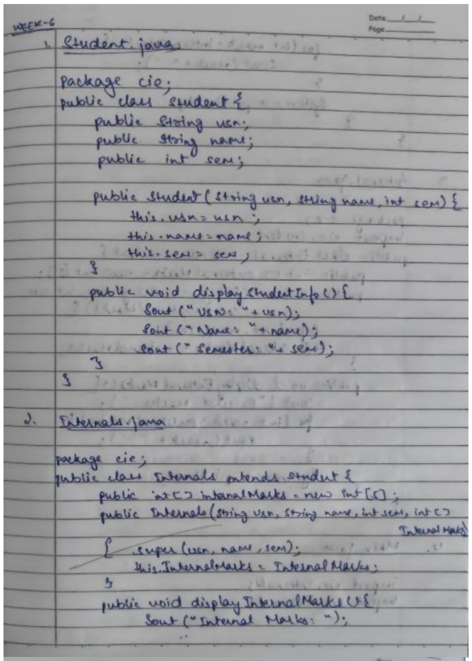


```
C:\Users\Admin\Desktop\src>java Main
Student 1 Info:
USN: USN123
Name: Alice
Semester: 3
Internal Marks:
20 30 25 28 22
External Marks:
60 70 55 65 50
Final Marks (Internal + External):
80 100 80 93 72

Student 2 Info:
USN: USN124
Name: Bob
Semester: 3
Internal Marks:
18 25 20 23 28
External Marks:
50 65 60 58 45
Final Marks (Internal + External):
68 90 80 81 73
```

Program 6

Packages CIE and SEE

Algorithm:



WEEK-6

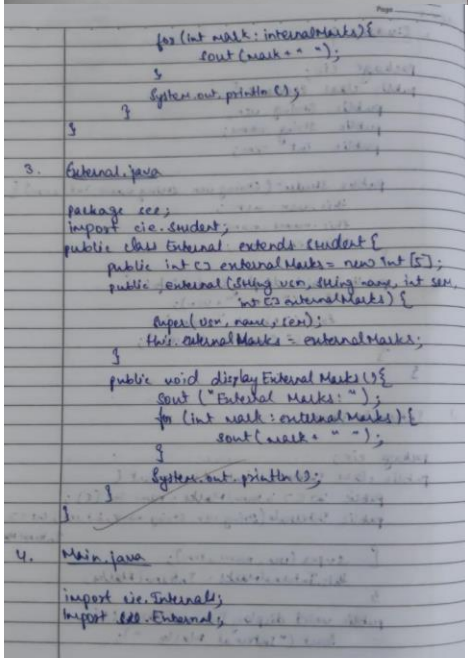
Date __/__/__
Page ____

1. 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 ;
    }
    public void displayStudentInfo () {
        Sout (" USN: " + usn);
        Sout (" Name: " + name);
        Sout (" Semester: " + sem);
    }
}
```

2. Internals.java

```
package cie;
public class Internals extends Student {
    public int [] internalMarks = new int [5];
    public Internals (String usn, String name, int sem, int []
                                                  InternalMarks)
    {   super (usn, name, sem);
        this.InternalMarks = InternalMarks;
    }
    public void displayInternalMarks () {
        Sout ("Internal Marks: ");
```



Page ____

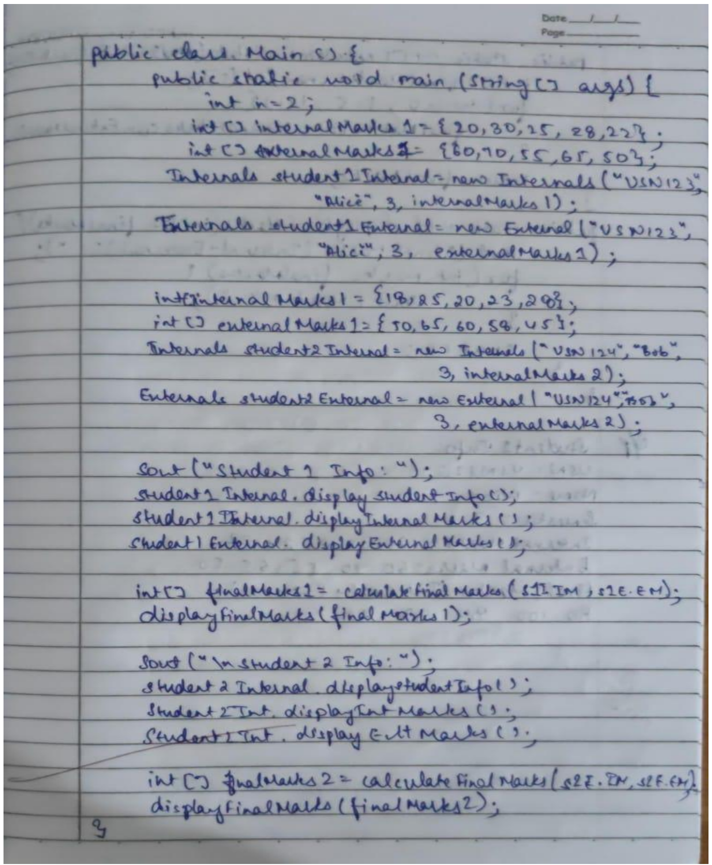
```
        for (int mark : internalMarks) {
            Sout (mark + " ");
        }
        System.out.println ();
    }
}
```

3. External.java

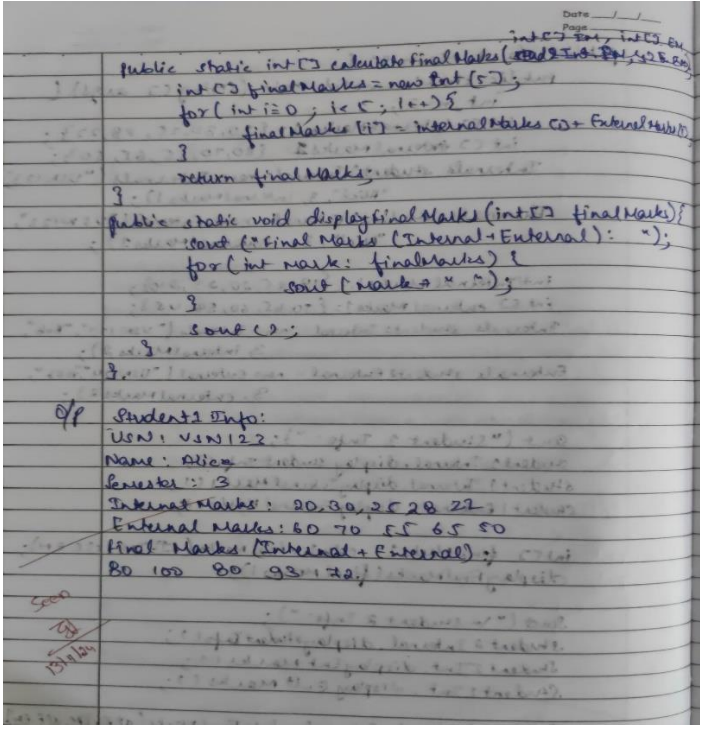
```
package see;
import cie.Student;
public class External extends Student {
    public int [] externalMarks = new int [5];
    public external (String usn, String name, int sem,
                        int [] externalMarks) {
        super (usn, name, sem);
        this.externalMarks = externalMarks;
    }
    public void displayExternalMarks () {
        Sout ("External Marks: ");
        for (int mark : externalMarks) {
            Sout (mark + " ");
        }
        System.out.println ();
    }
}
```

4. Main.java

```
import cie.Internals;
import see.External;
```

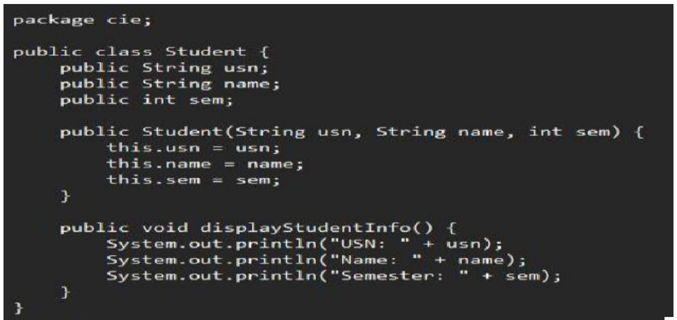


```
Date __/__/__
Page ______
public class Main () {
    public static void main (String [] args) {
        int n=2;
        int [] internalMarks 1= {20,30,25, 28,22};
        int [] ~~Ant~~ernalMarks ~~1~~= {60,70,55,65, 50};
      Internals student1Internal= new Internals ("USN123",
                          "Alice", 3, internalMarks1);
      Externals student1External= new External ("USN123",
                          "Alice", 3,  externalMarks1);

      int[]internal Marks1 = {18,25,20,23,28};
      int [] external Marks 1= {50,65, 60, 58, 45};
     Internals student2Internal = new Internals ("USN 124", "Bob",
                                          3, internalMarks 2);
     Externals student2 External = new External ( "USN124","Bob",
                                          3, externalMarks 2);

     sout ("Student 1 Info: ");
     student1 Internal. display student Info();
     student1 Internal. displayInternal Marks ();
     Student1 External.  displayExternal Marks ();

     int[]  finalMarks1 = calculate final Marks ( s1I.IM , s1E.EM);
     displayfinalMarks (final Marks1);

     Sout (" \n student 2 Info: ");
     student 2 Internal. displayStudentInfo();
     Student 2 Int. displayInt Marks ();
     Student2 Int . display Ext Marks ();

     int [] finalMarks 2 = calculate Final Marks ( s2I.IM, s2E.EM);
     displayFinalMarks (finalMarks2);
  }
```

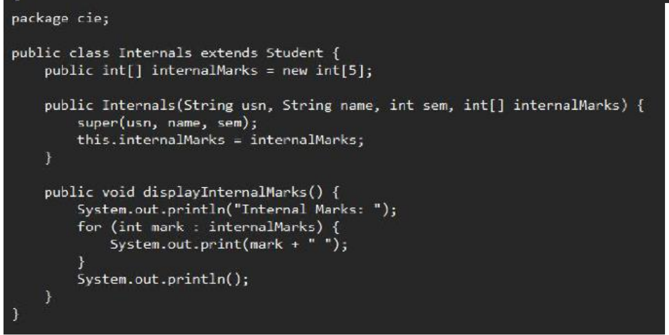


```
Date __/__/__
Page ______
                                         int [] IM, int [] EM
    public static int [] calculate final Marks( ~~stud2 Int. IM, s2E.EM~~);
        int [] finalMarks = new int [5];
        for( int i=0 ; i< 5; i++){
             finalMarks [i] = internalMarks [i]+ ExternalMarks[i];
        }
        return final Marks;
    }
   public static void displayfinal Marks (int[]  finalMarks){
        sout (" Final Marks (Internal + External):  ");
        for(int mark: finalMarks) {
                  sout (mark + " ");
        }
        sout ();
      }
    }.

O/P  Student1 Info:
    USN: USN123
    Name : Alice
    Semester : 3
    Internal Marks : 20,30, 25 28 22
    External Marks: 60 70 55 65 50
    Final Marks (Internal + External) ;
    80 100 80 93 72.

Seen
13/9/24
```

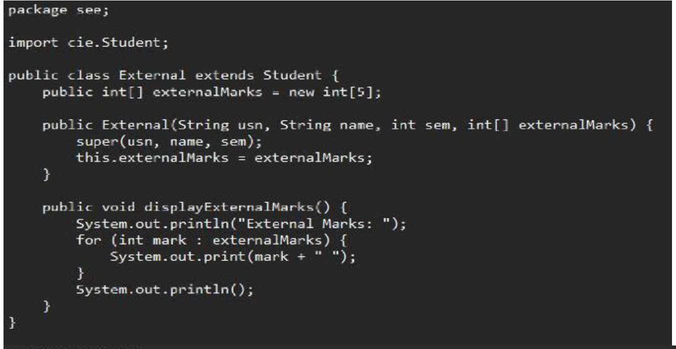
Code:



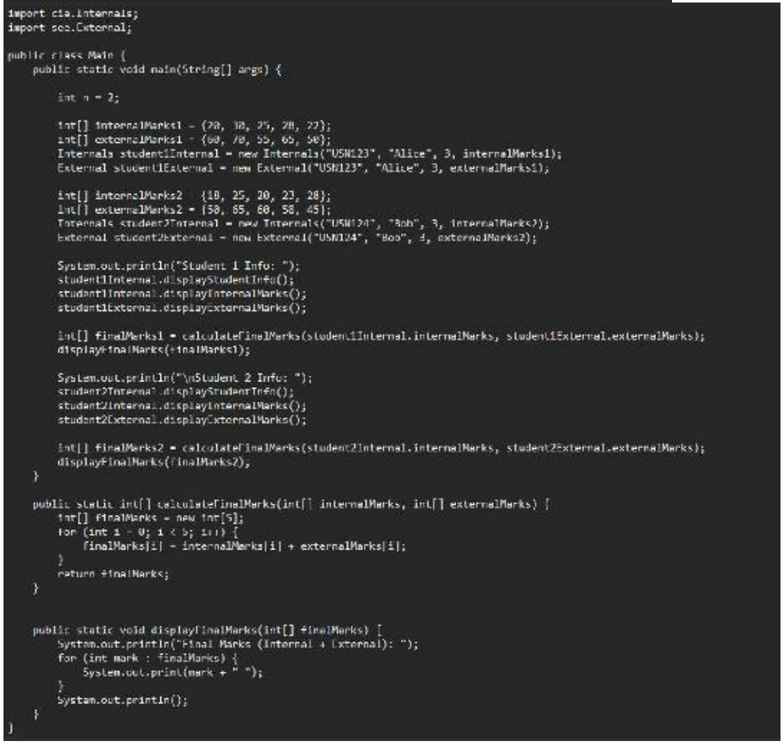
```
package cie;

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;
    }

    public void displayStudentInfo() {
        System.out.println("USN: " + usn);
        System.out.println("Name: " + name);
        System.out.println("Semester: " + sem);
    }
}
```



```
package cie;

public class Internals extends Student {
    public int[] internalMarks = new int[5];

    public Internals(String usn, String name, int sem, int[] internalMarks) {
        super(usn, name, sem);
        this.internalMarks = internalMarks;
    }

    public void displayInternalMarks() {
        System.out.println("Internal Marks: ");
        for (int mark : internalMarks) {
            System.out.print(mark + " ");
        }
        System.out.println();
    }
}
```



```
package see;

import cie.Student;

public class External extends Student {
    public int[] externalMarks = new int[5];

    public External(String usn, String name, int sem, int[] externalMarks) {
        super(usn, name, sem);
        this.externalMarks = externalMarks;
    }

    public void displayExternalMarks() {
        System.out.println("External Marks: ");
        for (int mark : externalMarks) {
            System.out.print(mark + " ");
        }
        System.out.println();
    }
}
```

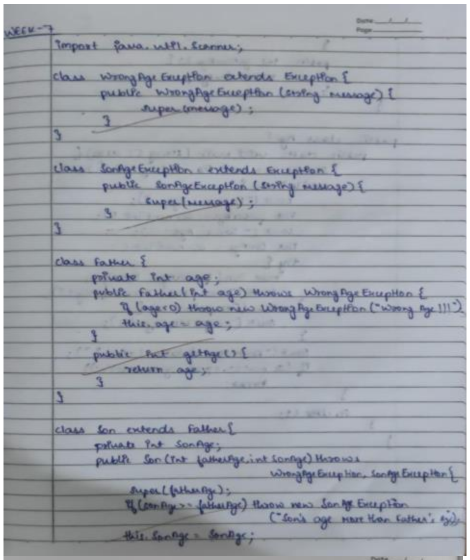


```
import cie.Internals;
import see.External;

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

        int n = 2;

        int[] internalMarks1 = {20, 18, 25, 28, 22};
        int[] externalMarks1 = {60, 70, 55, 65, 50};
        Internals student1Internal = new Internals("USN123", "Alice", 3, internalMarks1);
        External student1External = new External("USN123", "Alice", 3, externalMarks1);

        int[] internalMarks2 = {18, 25, 20, 23, 28};
        int[] externalMarks2 = {50, 65, 60, 58, 45};
        Internals student2Internal = new Internals("USN124", "Bob", 3, internalMarks2);
        External student2External = new External("USN124", "Bob", 3, externalMarks2);

        System.out.println("Student 1 Info: ");
        student1Internal.displayStudentInfo();
        student1Internal.displayInternalMarks();
        student1External.displayExternalMarks();

        int[] finalMarks1 = calculateFinalMarks(student1Internal.internalMarks, student1External.externalMarks);
        displayFinalMarks(finalMarks1);

        System.out.println("\nStudent 2 Info: ");
        student2Internal.displayStudentInfo();
        student2Internal.displayInternalMarks();
        student2External.displayExternalMarks();

        int[] finalMarks2 = calculateFinalMarks(student2Internal.internalMarks, student2External.externalMarks);
        displayFinalMarks(finalMarks2);
    }

    public static int[] calculateFinalMarks(int[] internalMarks, int[] externalMarks) {
        int[] finalMarks = new int[5];
        for (int i = 0; i < 5; i++) {
            finalMarks[i] = internalMarks[i] + externalMarks[i];
        }
        return finalMarks;
    }


    public static void displayFinalMarks(int[] finalMarks) {
        System.out.println("Final Marks (Internal + External): ");
        for (int mark : finalMarks) {
            System.out.print(mark + " ");
        }
        System.out.println();
    }
}
```

Program 7

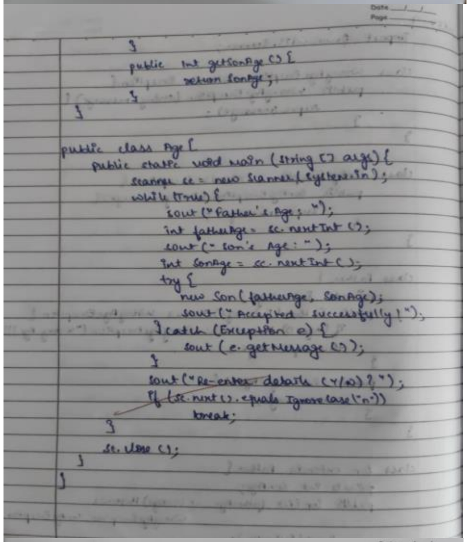
Exceptions in inheritance tree

Algorithm:

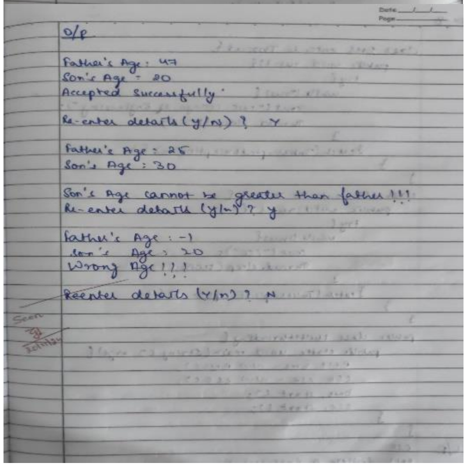


WEEK-7

```
import java.util.Scanner;

class WrongAgeException extends Exception{
        public WrongAgeException (String message) {
                super(message);
        }
}

class SonAgeException extends Exception {
        public SonAgeException (String message){
                super(message);
        }
}

class Father {
      private int age;
      public Father(int age) throws WrongAgeException {
            if (age<0) throw new WrongAgeException("Wrong Age!!!");
            this.age = age;
      }
      public int getAge() {
            return age;
      }
}

class Son extends Father{
      private int SonAge;
      public Son(int fatherAge,int SonAge) throws
                                 WrongAgeException, SonAgeException{
           super(fatherAge);
           if(SonAge>=fatherAge) throw new SonAgeException
                                 ("Son's age more than Father's age");
           this.SonAge = SonAge;
```



```
           }
           public int getSonAge() {
                  return SonAge;
           }
   }

public class Age {
     public static void main (String [] args){
           Scanner sc = new Scanner(System.in);
           while (True){
                 sout ("Father's Age: ");
                 int fatherAge = sc.nextInt ();
                 sout (" Son's Age: ");
                 int SonAge = sc.nextInt ();
                 try {
                    new Son(fatherAge, SonAge);
                    sout(" Accepted successfully!");
                 }catch (Exception e) {
                    sout (e.getMessage ());
                 }
                 sout("Re-enter details (Y/N)? ");
                 if (sc.next().equalsIgnoreCase("n"))
                      break;
           }
        sc.close ();
   }
}
```

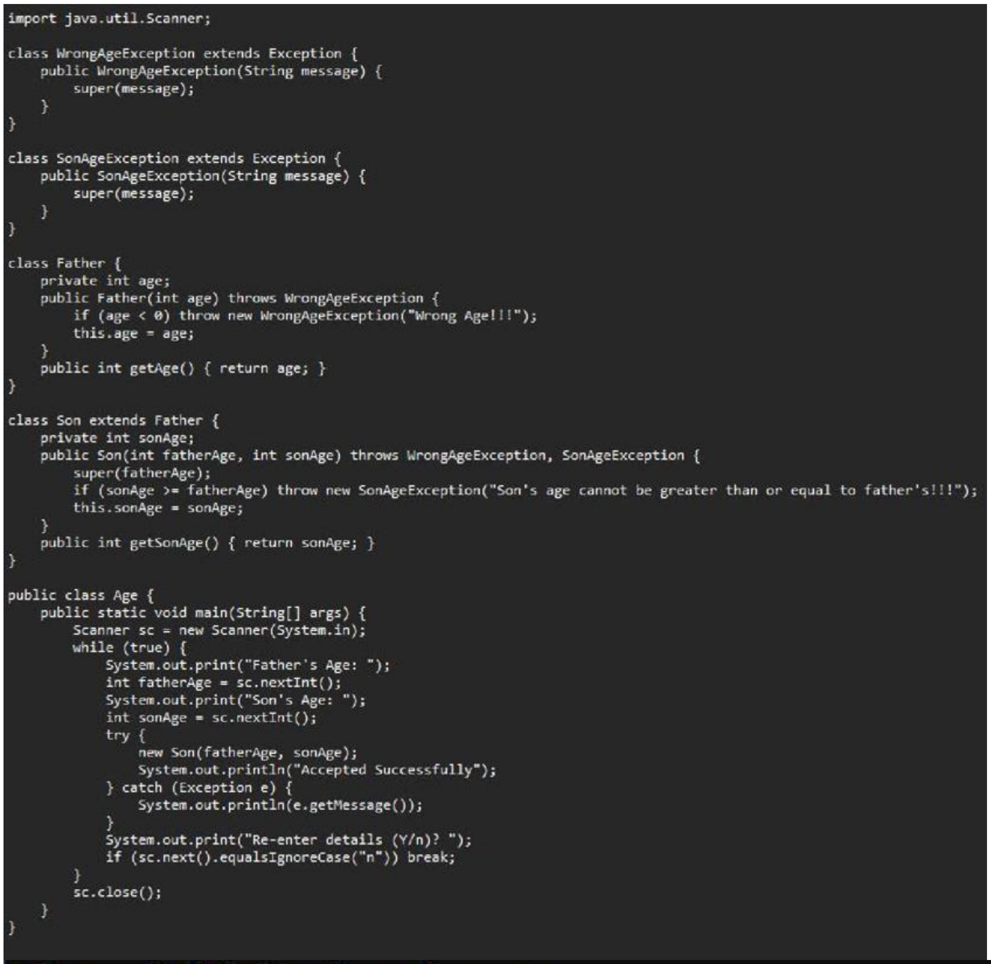


O/P

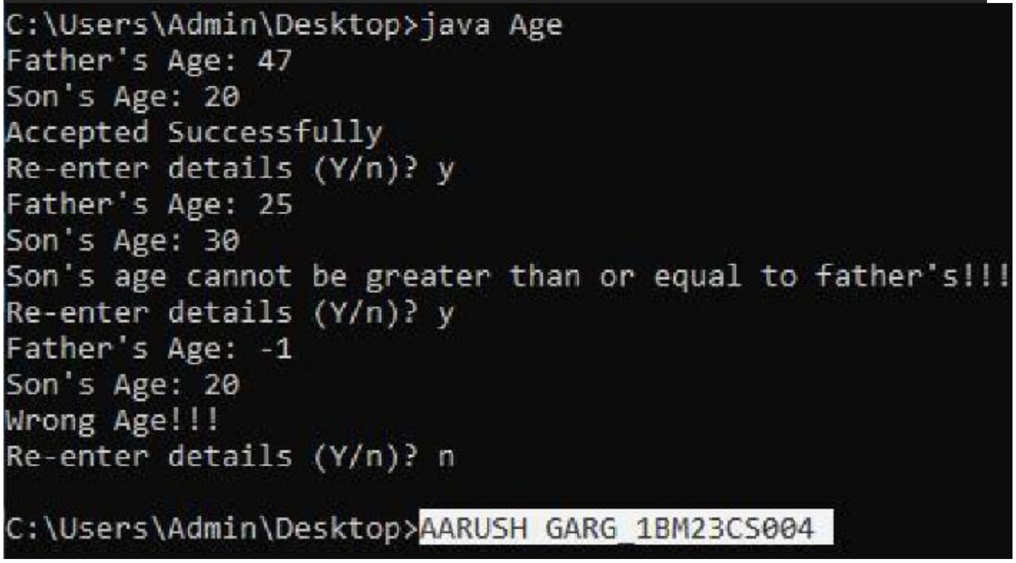
```
Father's Age: 47
Son's Age : 20
Accepted Successfully.

Re-enter details (y/n) ? Y

Father's Age : 25
Son's Age : 30

Son's Age cannot be greater than father!!!
Re-enter details (y/n)? y

Father's Age : -1
 Son's  Age : 20
Wrong Age!!!

Reenter details (Y/n) ? N
```

Seen

Code:



```
import java.util.Scanner;

class WrongAgeException extends Exception {
    public WrongAgeException(String message) {
        super(message);
    }
}

class SonAgeException extends Exception {
    public SonAgeException(String message) {
        super(message);
    }
}

class Father {
    private int age;
    public Father(int age) throws WrongAgeException {
        if (age < 0) throw new WrongAgeException("Wrong Age!!!");
        this.age = age;
    }
    public int getAge() { return age; }
}

class Son extends Father {
    private int sonAge;
    public Son(int fatherAge, int sonAge) throws WrongAgeException, SonAgeException {
        super(fatherAge);
        if (sonAge >= fatherAge) throw new SonAgeException("Son's age cannot be greater than or equal to father's!!!");
        this.sonAge = sonAge;
    }
    public int getSonAge() { return sonAge; }
}

public class Age {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        while (true) {
            System.out.print("Father's Age: ");
            int fatherAge = sc.nextInt();
            System.out.print("Son's Age: ");
            int sonAge = sc.nextInt();
            try {
                new Son(fatherAge, sonAge);
                System.out.println("Accepted Successfully");
            } catch (Exception e) {
                System.out.println(e.getMessage());
            }
            System.out.print("Re-enter details (Y/n)? ");
            if (sc.next().equalsIgnoreCase("n")) break;
        }
        sc.close();
    }
}
```

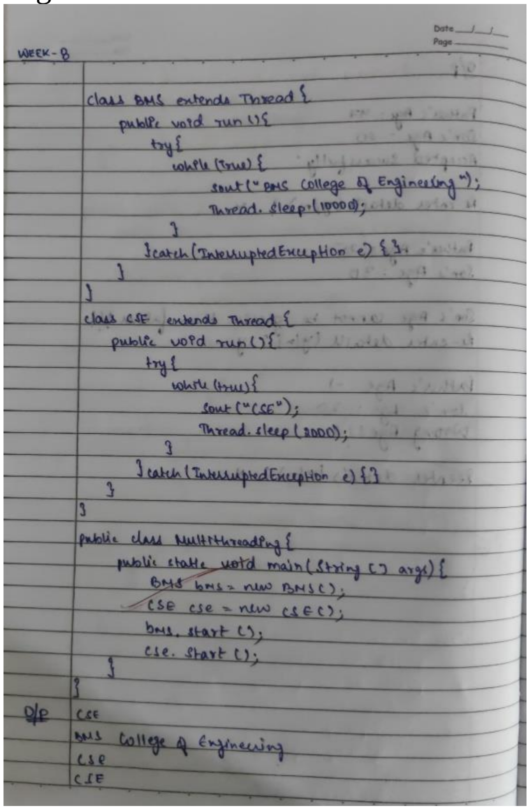


```
C:\Users\Admin\Desktop>java Age
Father's Age: 47
Son's Age: 20
Accepted Successfully
Re-enter details (Y/n)? y
Father's Age: 25
Son's Age: 30
Son's age cannot be greater than or equal to father's!!!
Re-enter details (Y/n)? y
Father's Age: -1
Son's Age: 20
Wrong Age!!!
Re-enter details (Y/n)? n

C:\Users\Admin\Desktop>AARUSH GARG 1BM23CS004
```

Program 8

Threads

Algorithm:



WEEK - 8

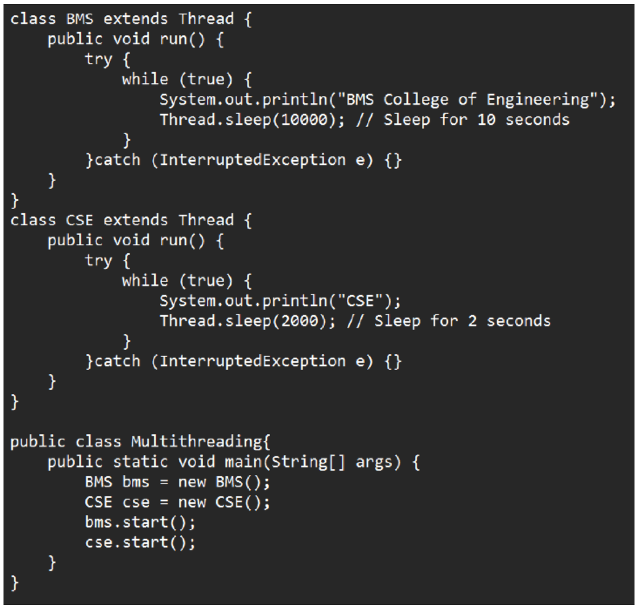
Date __/__/__
Page ____

```
class BMS extends Thread {
    public void run(){
        try{
            while (true) {
                sout("BMS College of Engineering");
                Thread. sleep (10000);
            }
        }catch(InterruptedException e) {}
    }
}
class CSE extends Thread {
    public void run(){
        try{
            while (true){
                sout ("CSE");
                Thread. sleep (2000);
            }
        }catch(InterruptedException e) {}
    }
}
public class Multithreading{
    public static void main(String [] args){
        BMS bms = new BMS();
        CSE cse = new CSE();
        bms. start ();
        cse. start ();
    }
}
```

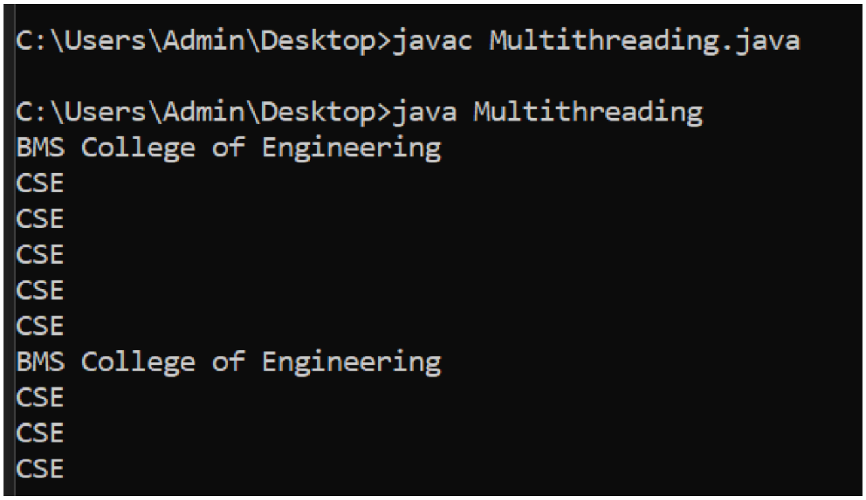
O/P

```
CSE
BMS College of Engineering
CSE
CSE
```

Code:



```
class BMS extends Thread {
    public void run() {
        try {
            while (true) {
                System.out.println("BMS College of Engineering");
                Thread.sleep(10000); // Sleep for 10 seconds
            }
        }catch (InterruptedException e) {}
    }
}
class CSE extends Thread {
    public void run() {
        try {
            while (true) {
                System.out.println("CSE");
                Thread.sleep(2000); // Sleep for 2 seconds
            }
        }catch (InterruptedException e) {}
    }
}

public class Multithreading{
    public static void main(String[] args) {
        BMS bms = new BMS();
        CSE cse = new CSE();
        bms.start();
        cse.start();
    }
}
```

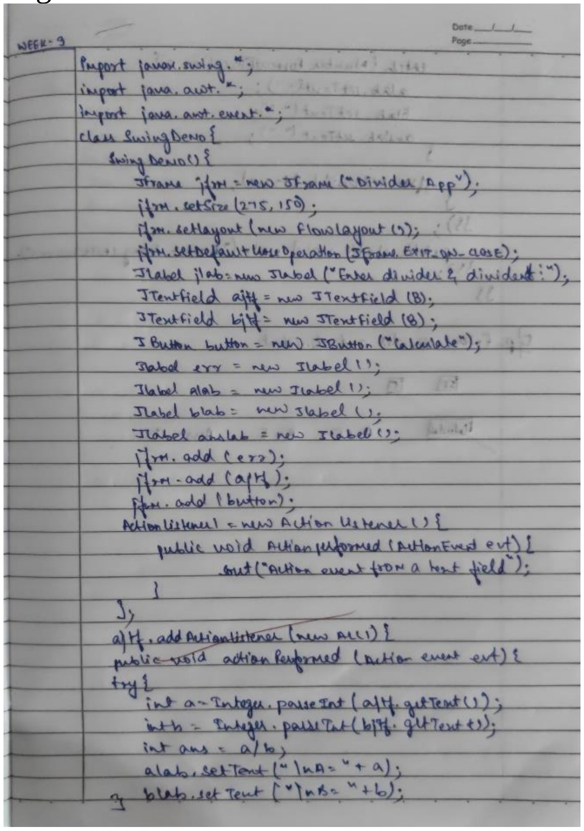


```
C:\Users\Admin\Desktop>javac Multithreading.java

C:\Users\Admin\Desktop>java Multithreading
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
```

Program 9

User interface for integer divisions

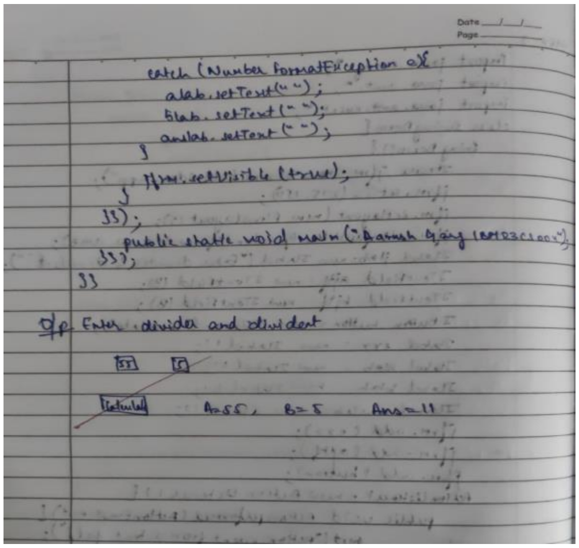
Algorithm:



WEEK-9

Date __/__/__
Page ______

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
class SwingDemo{
    SwingDemo(){
        JFrame jfrm = new JFrame("Divider App");
        jfrm.setSize(275, 150);
        jfrm.setLayout (new FlowLayout ());
        jfrm.setDefaultCloseOperation (JFrame.EXIT_ON_CLOSE);
        JLabel jlab=new JLabel ("Enter divider & divident:");
        JTextField ajtf = new JTextField (8);
        JTextField bjtf= new JTextField (8);
        JButton button = new JButton ("Calculate");
        JLabel err = new JLabel();
        JLabel alab = new JLabel ();
        JLabel blab = new JLabel ();
        JLabel anslab = new JLabel ();
        jfrm.add (err);
        jfrm.add (ajtf);
        jfrm.add (button);
      ActionListener l = new ActionListener (){
          public void Actionperformed (ActionEvent evt){
                  out("Action event from a text field");
          }
      };
     ajtf.add ActionListener (new AL()){
     public void actionPerformed (Action event evt){
     try{
         int a = Integer.parseInt (ajtf.getText());
         int b = Integer.parseInt(bjtf.getText());
         int ans = a/b;
         alab.setText ("\nA= "+ a);
     }   blab.set Text ("\nB= "+b);
```



```
Date __/__/__
Page ______

catch (NumberFormatException e){
alab.setText(" ");
blab.setText(" ");
anslab.setText(" ");
}
jfrm.setVisible(true);
}
}});
public static void main("Aarush Garg 1BM23CS00x");
}});
}}
```

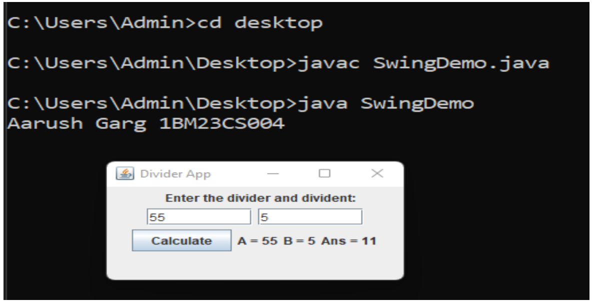
O/p Enter divider and divident

55 5

Calculate   A=55, B=5   Ans=11

Code:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
class SwingDemo{
SwingDemo(){
JFrame jfrm = new JFrame("Divider App");
jfrm.setSize(275, 150);
jfrm.setLayout(new FlowLayout());
jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
JLabel jlab = new JLabel("Enter the divider and divident:");
JTextField ajtf = new JTextField(8);
JTextField bjtf = new JTextField(8);
JButton button = new JButton("Calculate");
JLabel err = new JLabel();
JLabel alab = new JLabel();
JLabel blab = new JLabel();

JLabel anslab = new JLabel();
jfrm.add(err);
jfrm.add(jlab);
jfrm.add(ajtf);
jfrm.add(bjtf);
jfrm.add(button);
jfrm.add(alab);
jfrm.add(blab);
jfrm.add(anslab);
ActionListener l = new ActionListener() {
public void actionPerformed(ActionEvent evt) {
System.out.println("Action event from a text field");
}
};
ajtf.addActionListener(l);
bjtf.addActionListener(l);
button.addActionListener(new ActionListener() {
public void actionPerformed(ActionEvent evt) {
try{
int a = Integer.parseInt(ajtf.getText());
int b = Integer.parseInt(bjtf.getText());
int ans = a/b;
alab.setText("\nA = " + a);
blab.setText("\nB = " + b);
anslab.setText("\nAns = "+ ans);
}
catch(NumberFormatException e){
alab.setText("");
blab.setText("");
anslab.setText("");

err.setText("Enter Only Integers!");
}
catch(ArithmeticException e){
alab.setText("");
blab.setText("");
anslab.setText("");
err.setText("B should be NON zero!");
}
}
});
jfrm.setVisible(true);
}
public static void main(String args[]){
System.out.println("Aarush Garg 1BM23CS004");
SwingUtilities.invokeLater(new Runnable(){
public void run(){
new SwingDemo();
}
});
}
}
```



```
C:\Users\Admin>cd desktop

C:\Users\Admin\Desktop>javac SwingDemo.java

C:\Users\Admin\Desktop>java SwingDemo
Aarush Garg 1BM23CS004
```

Enter the divider and divident:

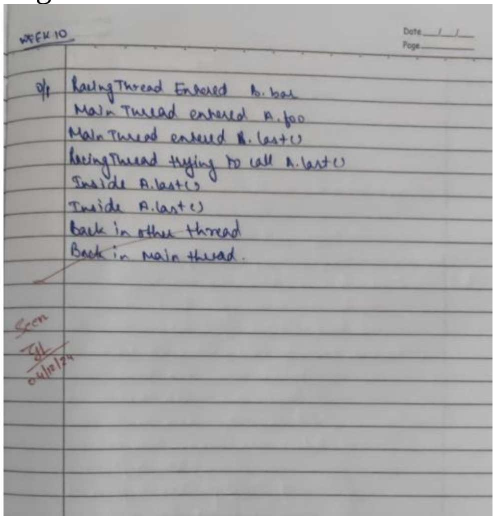
55 5

A = 55 B = 5 Ans = 11

Program 10

IPC and Deadlock

Algorithm:



WEEK 10

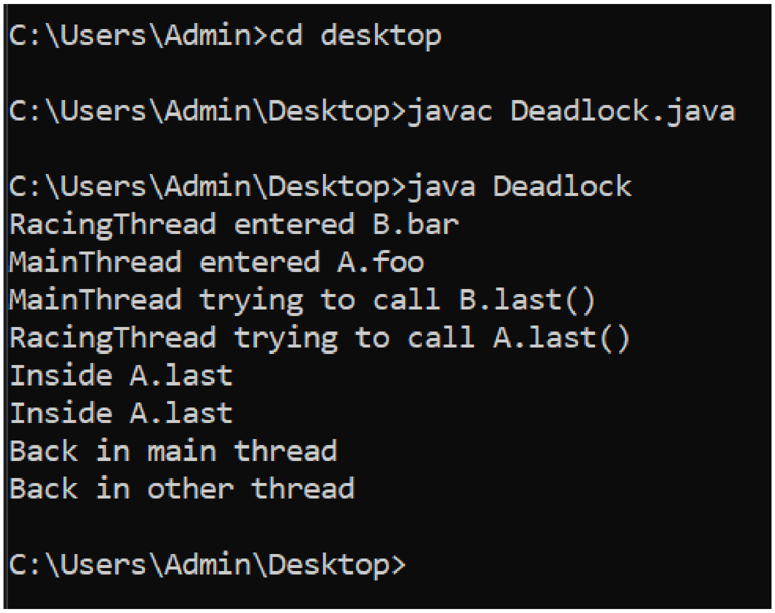
Date __/__/__
Page ____

O/P RacingThread Entered B.bar
Main Thread entered A.foo
Main Thread entered B.last()
RacingThread trying to call A.last()
Inside A.last()
Inside A.last()
back in other thread
Back in main thread.

Seen
04/12/24

Code:

```
class A {
synchronized void foo(B b) {
String name = Thread.currentThread().getName();
System.out.println(name + " entered A.foo");
try {
Thread.sleep(1000);
} catch(Exception e) {
System.out.println("A Interrupted");
}
System.out.println(name + " trying to call B.last()");
b.last();
}
void last() {
System.out.println("Inside A.last");
}
}
class B {
synchronized void bar(A a) {
String name = Thread.currentThread().getName();
System.out.println(name + " entered B.bar");
try {
Thread.sleep(1000);
} catch(Exception e) {
System.out.println("B Interrupted");
}
System.out.println(name + " trying to call A.last()");
a.last();
}
void last() {
System.out.println("Inside A.last");
}
}
class Deadlock implements Runnable
{
A a = new A();
B b = new B();
Deadlock() {
Thread.currentThread().setName("MainThread");
Thread t = new Thread(this,"RacingThread");
t.start();
a.foo(b); // get lock on a in thisthread.
System.out.println("Back in main thread");
}
public void run() {
b.bar(a); // get lock on b in other thread.
System.out.println("Back in other thread");
}
public static void main(String args[]) {
new Deadlock();
}
}
```



```
C:\Users\Admin>cd desktop

C:\Users\Admin\Desktop>javac Deadlock.java

C:\Users\Admin\Desktop>java Deadlock
RacingThread entered B.bar
MainThread entered A.foo
MainThread trying to call B.last()
RacingThread trying to call A.last()
Inside A.last
Inside A.last
Back in main thread
Back in other thread

C:\Users\Admin\Desktop>
```