VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted by

Rohit Kumawat (1BF24CS256)

in partial fulfillment for the award of the degree of
BACHELOR OF ENGINEERING
in
Computer Science & Engineering

B.M.S. COLLEGE OF ENGINEERING
(Autonomous Institution under VTU)
BENGALURU-560019
Aug-2025 to Jan-2026

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 **Rohit Kumawat(1BF24CS256),** who is bonafide student of **B.M.S. College of Engineering.** It is in partial fulfilment 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.

Dr. Seema Patil Associate Professor Department of CSE, BMSCE	Dr. Kavitha Sooda Professor & HOD Department of CSE, BMSCE

Index

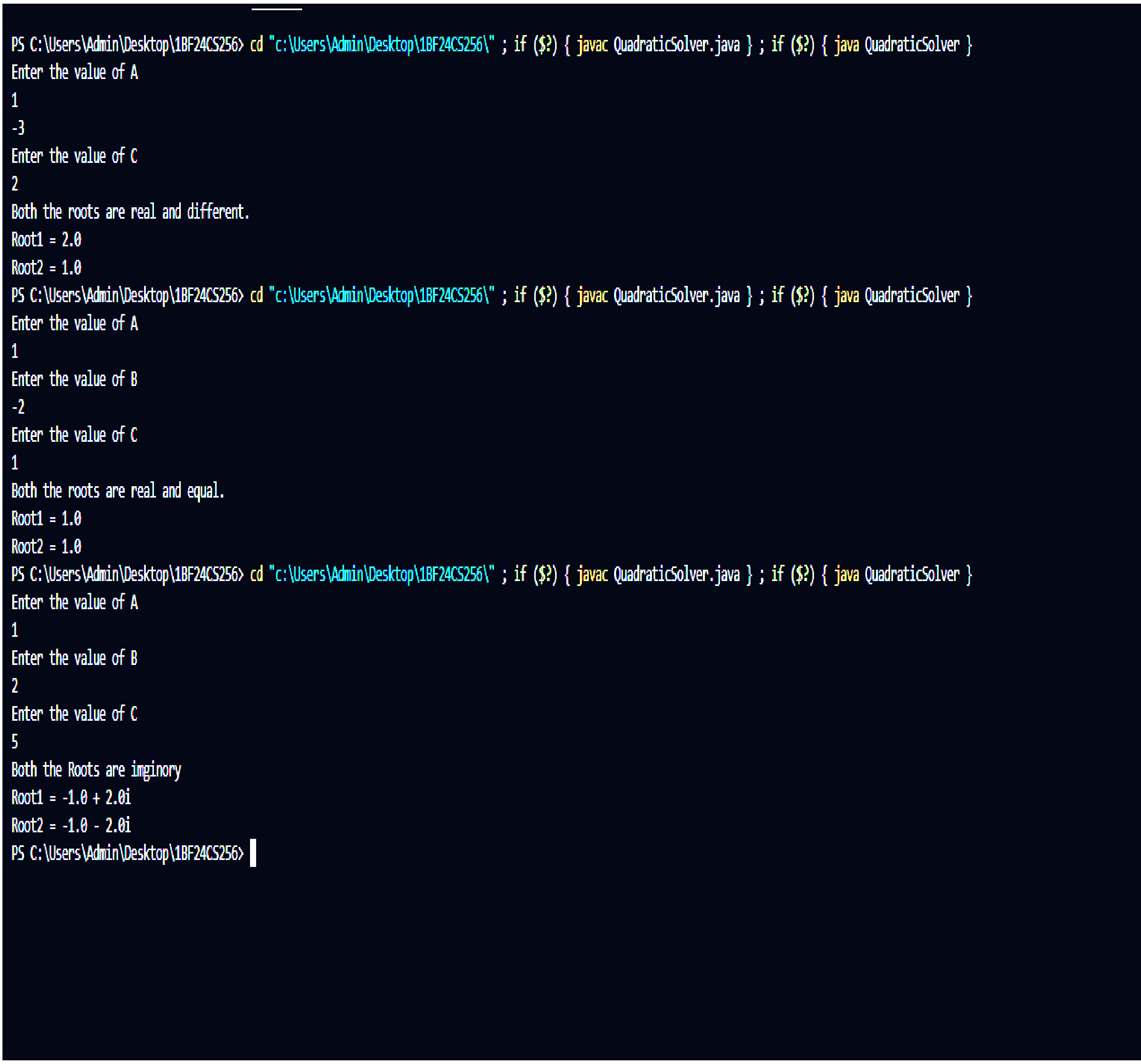
Github Link: https://github.com/Coder-Rohit29/java

Program 1

```
import java.util.Scanner;
```

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

        double a,b,c;
        double d ,r1 ,r2;

        System.out.println("Enter the value of A");
        a = scanner.nextDouble();

        while(a==0){
            System.out.println("Enter a non-zero no , not a quadratic eq");
            a=scanner.nextDouble();
        }

        System.out.println("Enter the value of B");
        b= scanner.nextDouble();

        System.out.println("Enter the value of C");
        c= scanner.nextDouble();

        d=b*b-4*a*c;

        if(d>0){
            r1 = (-b + Math.sqrt(d))/(2*a);
            r2 = (-b - Math.sqrt(d))/(2*a);
            System.out.println("Both the roots are real and different.");
            System.out.println("Root1 = "+ r1);
            System.out.println("Root2 = "+ r2);
        }
        else if(d==0){
            r1 =r2 = -b/(2*a);
            System.out.println("Both the roots are real and equal.");
            System.out.println("Root1 = "+ r1);
            System.out.println("Root2 = "+ r2);
        }
        else{
            double realPart = -b/(2*a);
            double imaginoryPart = Math.sqrt(-d)/(2*a);
            System.out.println("Both the Roots are imginary");
            System.out.println("Root1 = "+ realPart + " + " + imaginoryPart + "i");
            System.out.println("Root2 = " + realPart + " - " + imaginoryPart + "i");


        }
        scanner.close();
    }
}
```

Output:



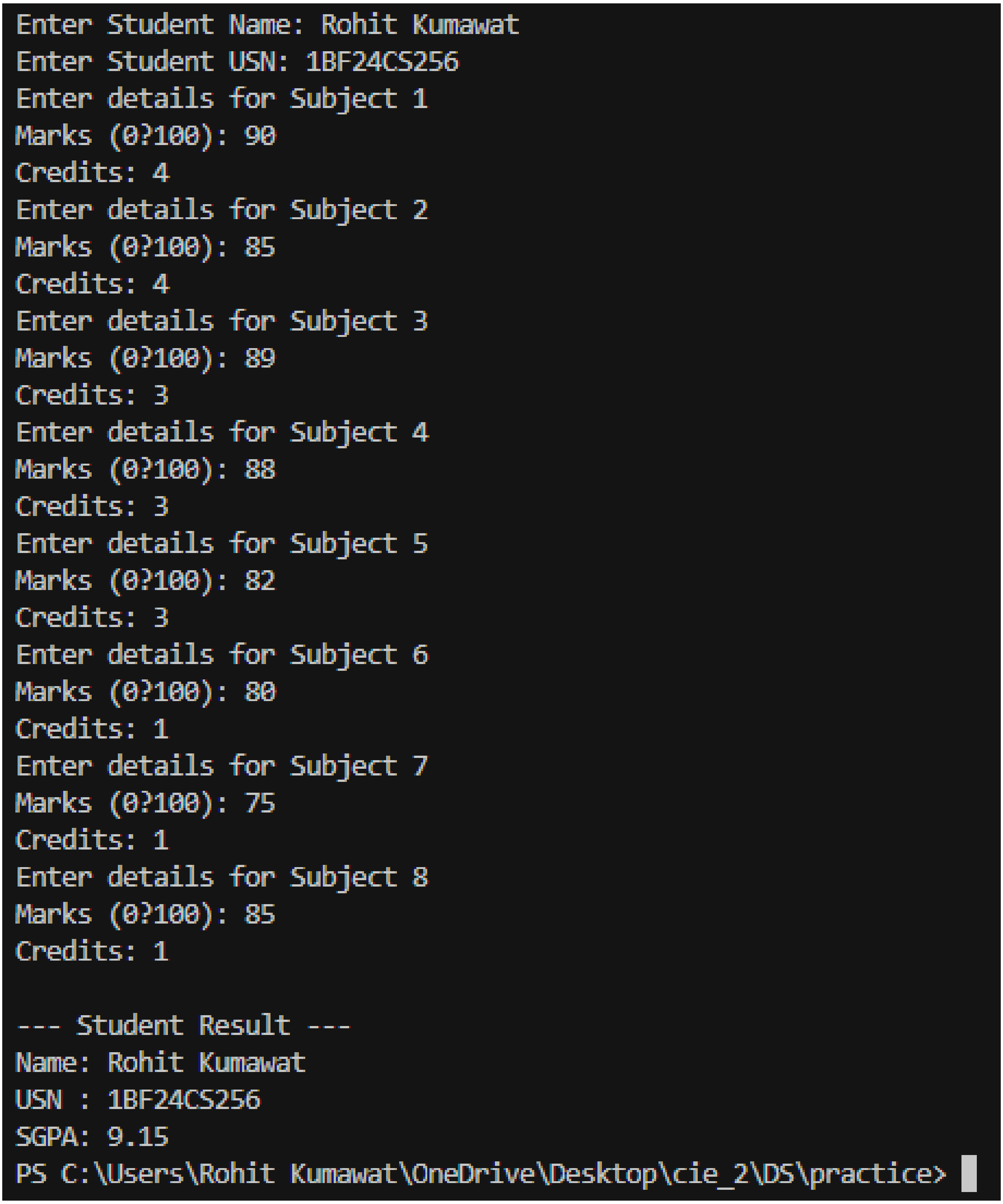
```
PS C:\Users\Admin\Desktop\1BF24CS256> cd "c:\Users\Admin\Desktop\1BF24CS256\" ; if ($?) { javac QuadraticSolver.java } ; if ($?) { java QuadraticSolver }
Enter the value of A
1
-3
Enter the value of C
2
Both the roots are real and different.
Root1 = 2.0
Root2 = 1.0
PS C:\Users\Admin\Desktop\1BF24CS256> cd "c:\Users\Admin\Desktop\1BF24CS256\" ; if ($?) { javac QuadraticSolver.java } ; if ($?) { java QuadraticSolver }
Enter the value of A
1
Enter the value of B
-2
Enter the value of C
1
Both the roots are real and equal.
Root1 = 1.0
Root2 = 1.0
PS C:\Users\Admin\Desktop\1BF24CS256> cd "c:\Users\Admin\Desktop\1BF24CS256\" ; if ($?) { javac QuadraticSolver.java } ; if ($?) { java QuadraticSolver }
Enter the value of A
1
Enter the value of B
2
Enter the value of C
5
Both the Roots are imginory
Root1 = -1.0 + 2.0i
Root2 = -1.0 - 2.0i
PS C:\Users\Admin\Desktop\1BF24CS256>
```

Program 2: SGPA Calculator

Code:

```
import java.util.Scanner;

class Subject {
    int subjectMarks;
    int credits;
    int grade;
}

   // HERE IS STEP 3RD AND 4TH
class Student {
    String name;
    String usn;
    double SGPA;
    Subject[] subject = new Subject[8]; // CREATED ARRAY
    Scanner s;

    // CONSTRUCTOR
    Student() {
        s = new Scanner(System.in);
        for (int i = 0; i < subject.length; i++) {
            subject[i] = new Subject();
        }
    }

    // TAKE STUDENT DETAILS
    void getStudentDetails() {
        System.out.print("Enter Student Name: ");
        name = s.nextLine();
        System.out.print("Enter Student USN: ");
        usn = s.nextLine();
    }

    // Step 7: Accept marks and credits, compute grade
    void getMarks() {
        for (int i = 0; i < subject.length; i++) {
            System.out.println("Enter details for Subject " + (i + 1));
            System.out.print("Marks (0–100): ");
            subject[i].subjectMarks = s.nextInt();

            System.out.print("Credits: ");
            subject[i].credits = s.nextInt();

            // Compute grade
            subject[i].grade = (subject[i].subjectMarks / 10) + 1;

            if (subject[i].grade > 10)
                subject[i].grade = 10;
            if (subject[i].subjectMarks < 4)
                subject[i].grade = 0;
        }
    }


    void computeSGPA() {
        int effectiveScore = 0, totalCredits = 0;

        for (int i = 0; i < subject.length; i++) {
            effectiveScore += subject[i].grade * subject[i].credits;
            totalCredits += subject[i].credits;
        }

        if (totalCredits != 0)
            SGPA = (double) effectiveScore / totalCredits;
        else
            SGPA = 0.0;
    }

    void displayResult() {
        System.out.println("\n--- Student Result ---");
        System.out.println("Name: " + name);
        System.out.println("USN : " + usn);
        System.out.println("SGPA: "+SGPA);
    }
}

public class Main {
    public static void main(String[] args) {
        Student student = new Student();
        student.getStudentDetails();
        student.getMarks();
        student.computeSGPA();
        student.displayResult();
    }
}
```

Output:



```
Enter Student Name: Rohit Kumawat
Enter Student USN: 1BF24CS256
Enter details for Subject 1
Marks (0?100): 90
Credits: 4
Enter details for Subject 2
Marks (0?100): 85
Credits: 4
Enter details for Subject 3
Marks (0?100): 89
Credits: 3
Enter details for Subject 4
Marks (0?100): 88
Credits: 3
Enter details for Subject 5
Marks (0?100): 82
Credits: 3
Enter details for Subject 6
Marks (0?100): 80
Credits: 1
Enter details for Subject 7
Marks (0?100): 75
Credits: 1
Enter details for Subject 8
Marks (0?100): 85
Credits: 1

--- Student Result ---
Name: Rohit Kumawat
USN : 1BF24CS256
SGPA: 9.15
PS C:\Users\Rohit Kumawat\OneDrive\Desktop\cie_2\DS\practice>
```

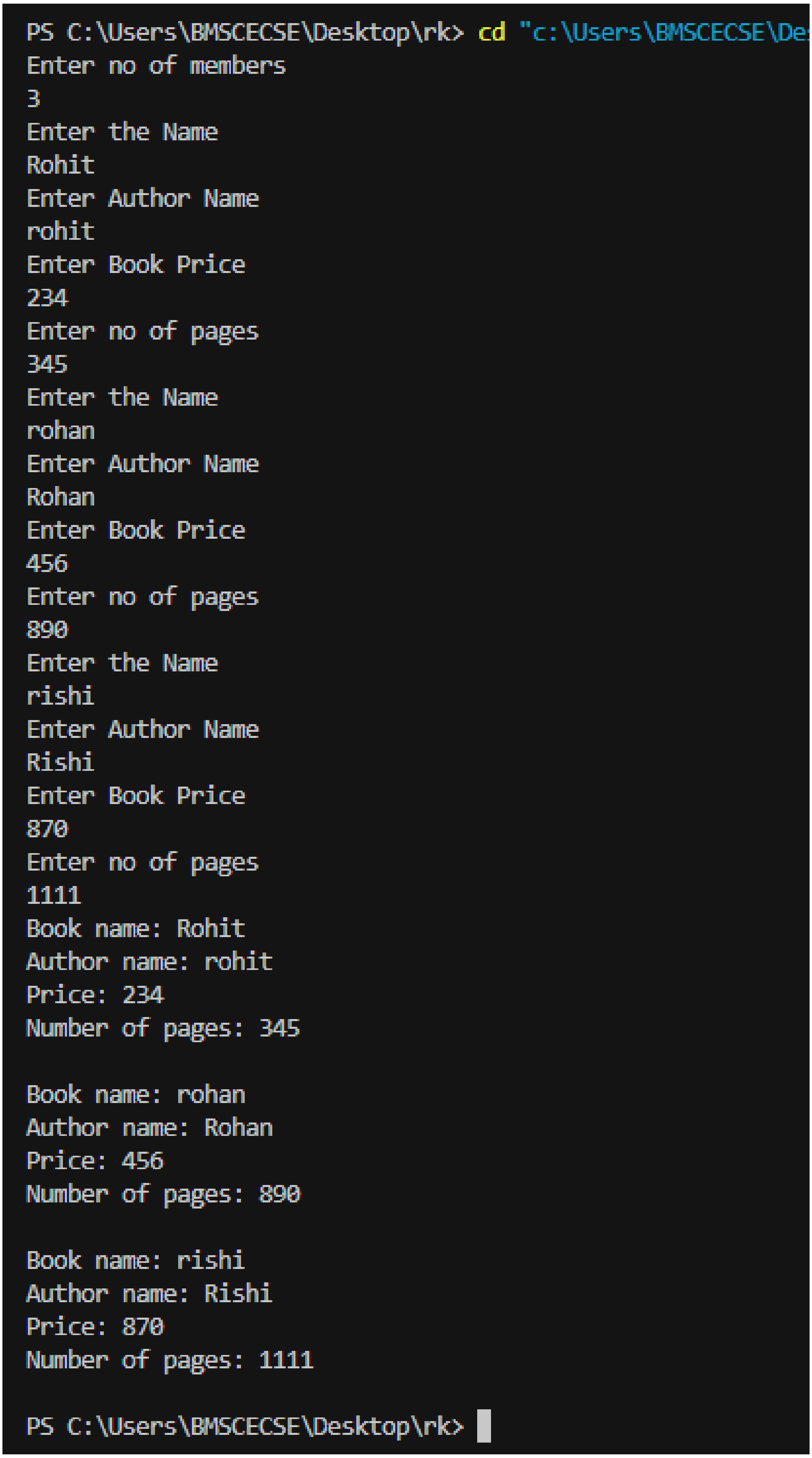
Program 3: Bookstore Program

Code:

```
import java.util.Scanner;

class Books {
   String name;
   String author;
   int price;
   int numPages;

   Books(String name, String author, int price, int numPages) {

      this.name = name;
      this.author = author;
      this.price = price;
      this.numPages = numPages;

   }

   public String toString() {

      String name, author, price, numPages;
      name = "Book name: " + this.name + "\n";
      author = "Author name: " + this.author + "\n";
      price = "Price: " + this.price + "\n";
      numPages = "Number of pages: " + this.numPages + "\n";
      return name + author + price + numPages;
   }

}

public class labprogramn_3{
   public static void main(String arg[]) {
      System.out.println("Enter no of members");
      Scanner s = new Scanner(System.in);
      int n;
      String name;
      String author;
      int price;
      int numPages;
      n = s.nextInt();
      Books b[];
      b = new Books[n];
      for (int i = 0; i < n; i++) {

         System.out.println("Enter the Name");
         name = s.next();
```

```
            System.out.println("Enter Author Name");
            author = s.next();
            System.out.println("Enter Book Price");
            price = s.nextInt();
            System.out.println("Enter no of pages");
            numPages = s.nextInt();
            b[i] = new Books(name, author, price, numPages);
        }
        for (int i = 0; i < n; i++) {
            System.out.println(b[i]);
        }
    }
}
```

Output:



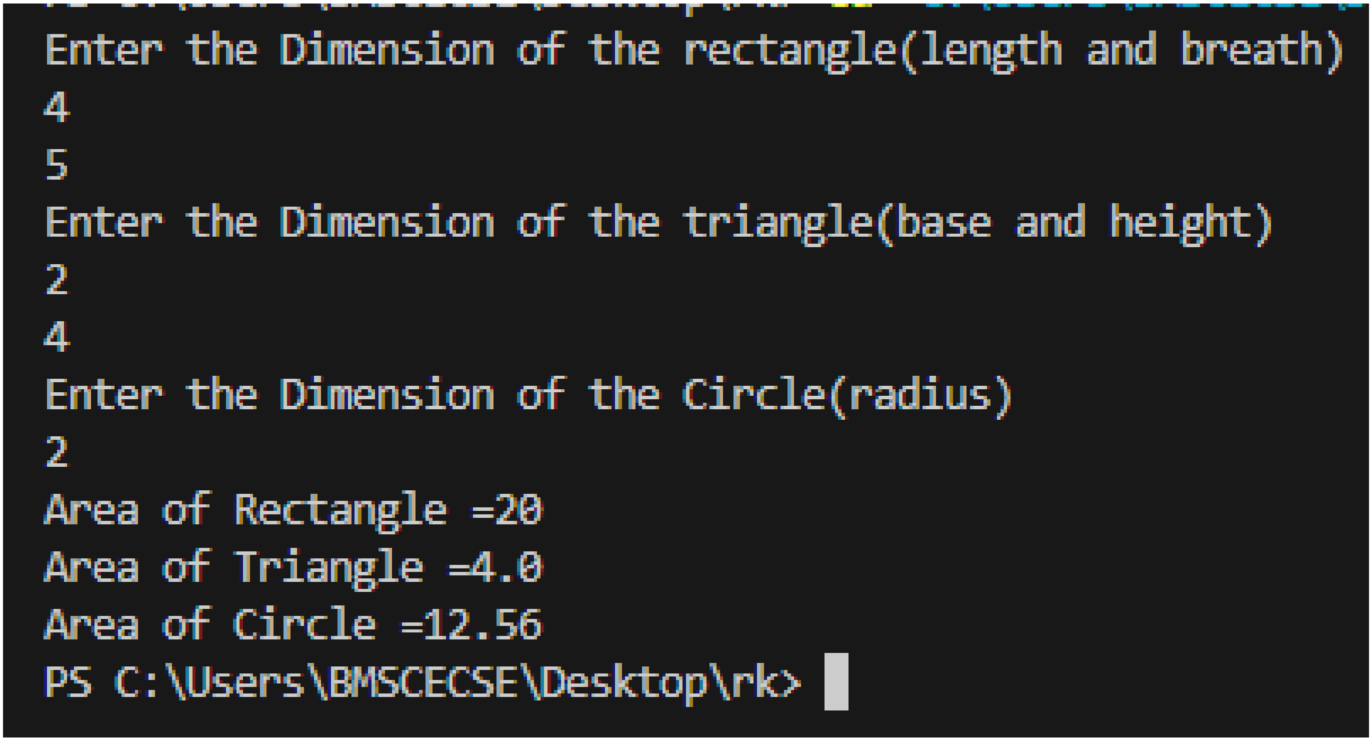
```
PS C:\Users\BMSCECSE\Desktop\rk> cd "c:\Users\BMSCECSE\De
Enter no of members
3
Enter the Name
Rohit
Enter Author Name
rohit
Enter Book Price
234
Enter no of pages
345
Enter the Name
rohan
Enter Author Name
Rohan
Enter Book Price
456
Enter no of pages
890
Enter the Name
rishi
Enter Author Name
Rishi
Enter Book Price
870
Enter no of pages
1111
Book name: Rohit
Author name: rohit
Price: 234
Number of pages: 345

Book name: rohan
Author name: Rohan
Price: 456
Number of pages: 890

Book name: rishi
Author name: Rishi
Price: 870
Number of pages: 1111

PS C:\Users\BMSCECSE\Desktop\rk> 
```

Program 4: Shapes Program

```
Code:import java.util.Scanner;
abstract class Shape{
   int a,b;
   Shape(int a,int b){
      this.a = a;
      this.b =b;
   }
   abstract void printArea();
}
class Rectangle extends Shape{
   Rectangle(int a,int b){
         super(a,b);
   }

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

class Triangle extends Shape{
   Triangle(int a,int b){
         super(a,b);
   }

   void printArea(){
      double area = (0.5)*a*b;
     System.out.println("Area of Triangle =" +  area );
   }
}

class Circle extends Shape{
   Circle(int a){
         super(a,0);
   }

   void printArea(){
      double area = (3.14*a*a);
     System.out.println("Area of Circle =" + area );
   }
}

public class labprogramn4{
   public static void main(String [] args){
      Scanner s = new Scanner(System.in);
      System.out.println("Enter the Dimension of the rectangle(length and breath)");
      int len = s.nextInt();
```

```
    int bre = s.nextInt();
    Rectangle ob = new Rectangle(len,bre);

    System.out.println("Enter the Dimension of the triangle(base and height)");
    int base = s.nextInt();
    int height = s.nextInt();
    Triangle ob1 = new Triangle(base,height);

    System.out.println("Enter the Dimension of the Circle(radius)");
    int radius = s.nextInt();
    Circle ob2 = new Circle(radius);

    ob.printArea();
    ob1.printArea();
    ob2.printArea();
  }
}
```

Output:



```
Enter the Dimension of the rectangle(length and breath)
4
5
Enter the Dimension of the triangle(base and height)
2
4
Enter the Dimension of the Circle(radius)
2
Area of Rectangle =20
Area of Triangle =4.0
Area of Circle =12.56
PS C:\Users\BMSCECSE\Desktop\rk>
```

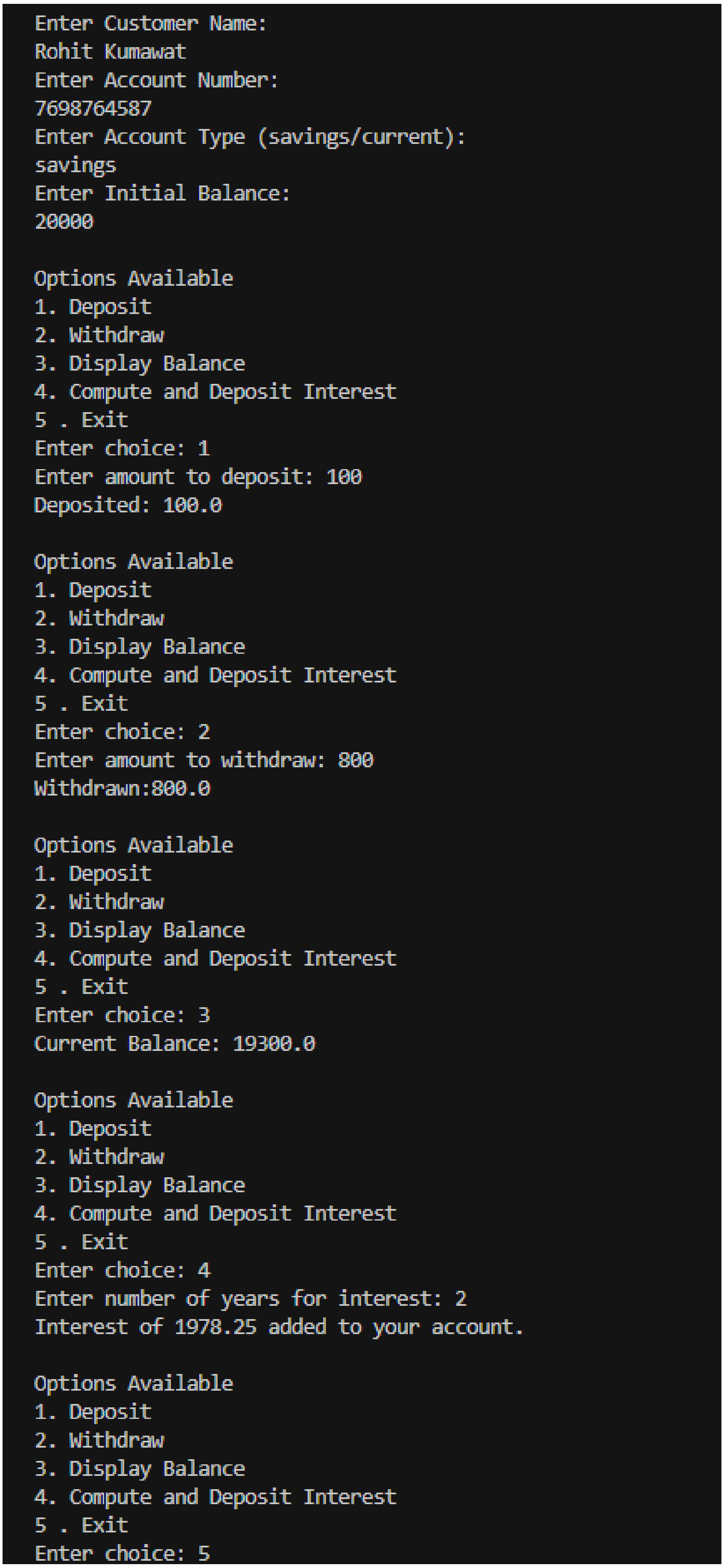
Program 5: Bank Program

Code:
```
import java.util.Scanner;

class Account
{
   String customerName;
   String accountNumber;
   String accountType;
   double balance;

   Account(String customerName, String accountNumber, String accountType, double initialBalance)
{
      this.customerName = customerName;
      this.accountNumber = accountNumber;
      this.accountType = accountType;
      this.balance = initialBalance;
   }

   void deposit(double amount)
   {
      if (amount > 0)
      {
         balance += amount;
         System.out.println("Deposited: " + amount);
      }
      else
      {
         System.out.println("Invalid deposit amount.");
      }
   }


   void displayBalance() {
      System.out.println("Current Balance: " + balance);
   }
}

class Saving extends Account
{
   double INTEREST_RATE = 0.05;

   Saving(String customerName, String accountNumber, double initialBalance)
   {
      super(customerName, accountNumber, "Savings", initialBalance);
   }

   void Interest(int years) {
        double interest = balance * Math.pow((1 + INTEREST_RATE), years) - balance;
        balance += interest;
        System.out.println("Interest of " + String.format("%.2f", interest) + " added to your account.");
    }
    void withdraw(double amount)
    {
        if (amount <= balance)
        {
            balance -= amount;
            System.out.println("Withdrawn:" + amount);
        } else
        {
            System.out.println("Insufficient balance!");
        }
    }
}

class Current extends Account {
        static final double MIN_BALANCE = 1000.0;
    static final double SERVICE_CHARGE = 50.0;

    public Current(String customerName, String accountNumber, double initialBalance)
    {
        super(customerName, accountNumber, "Current", initialBalance);
    }

void withdraw(double amount) {
        if (amount <= balance) {
            balance -= amount;
            System.out.println("Withdrawn: ₹" + amount);
            checkMinimumBalance();
        } else {
            System.out.println("Insufficient balance!");
        }
    }

    void checkMinimumBalance() {
        if (balance < MIN_BALANCE) {
            balance -= SERVICE_CHARGE;
            System.out.println("Balance below minimum! Service charge of ₹" + SERVICE_CHARGE +
" imposed.");
        }
    }
}

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

```
Scanner in = new Scanner(System.in);

System.out.println("Enter Customer Name:");
String name = in.nextLine();

System.out.println("Enter Account Number:");
String accNo = in.nextLine();

System.out.println("Enter Account Type (savings/current):");
String type = in.nextLine().toLowerCase();

System.out.println("Enter Initial Balance:");
double balance = in.nextDouble();

Account account;

if (type.equals("savings"))
{
    account = new Saving(name, accNo, balance);
}
else
{
    account = new Current(name, accNo, balance);
}

int choice;
do {
    System.out.println("\nOptions Available");
    System.out.println("1. Deposit");
    System.out.println("2. Withdraw");
    System.out.println("3. Display Balance");
    if (account instanceof Saving)
        System.out.println("4. Compute and Deposit Interest");
    System.out.println("5 . Exit");
    System.out.print("Enter choice: ");
    choice = in.nextInt();

    switch (choice) {
        case 1:
            System.out.print("Enter amount to deposit: ");
            double depositAmount = in.nextDouble();
            account.deposit(depositAmount);
            break;

        case 2:
            System.out.print("Enter amount to withdraw: ");
            double withdrawAmount = in.nextDouble();
```

```
                if (account instanceof Saving) {
                    ((Saving) account).withdraw(withdrawAmount);
                } else {
                    ((Current) account).withdraw(withdrawAmount);
                }
                break;

            case 3:
                account.displayBalance();
                break;

            case 4:
                if (account instanceof Saving) {
                    System.out.print("Enter number of years for interest: ");
                    int years = in.nextInt();
                    ((Saving) account).Interest(years);
                } else {
                    System.out.println("Interest computation not available for Current Account.");
                }
                break;

            case 5:
                System.out.println("Exiting Program ");
                return;

            default:
                System.out.println("Invalid choice. Try again.");
            }
        } while (choice != 0);

        in.close();
    }
}
```

Output:

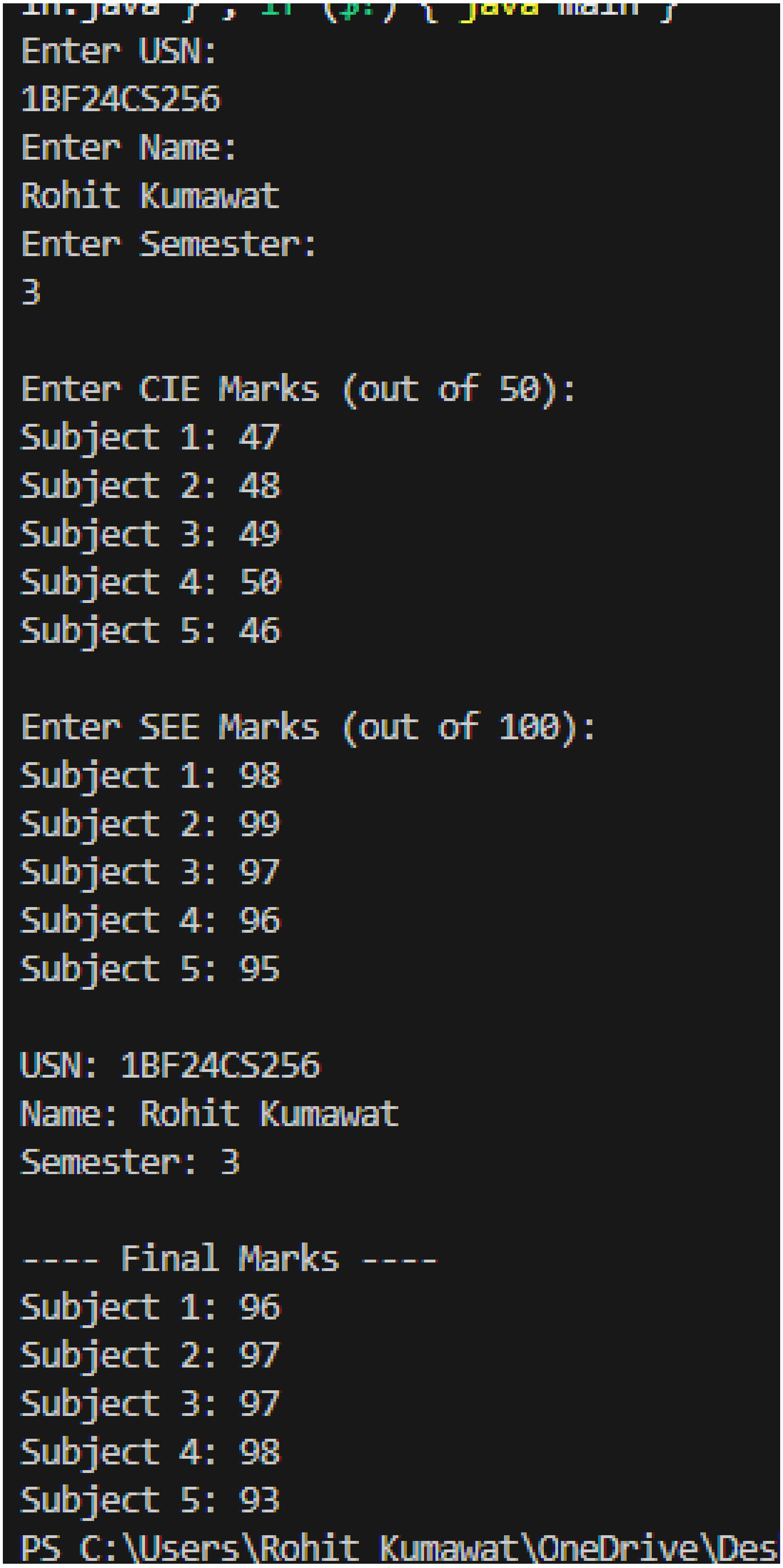


```
Enter Customer Name:
Rohit Kumawat
Enter Account Number:
7698764587
Enter Account Type (savings/current):
savings
Enter Initial Balance:
20000

Options Available
1. Deposit
2. Withdraw
3. Display Balance
4. Compute and Deposit Interest
5 . Exit
Enter choice: 1
Enter amount to deposit: 100
Deposited: 100.0

Options Available
1. Deposit
2. Withdraw
3. Display Balance
4. Compute and Deposit Interest
5 . Exit
Enter choice: 2
Enter amount to withdraw: 800
Withdrawn:800.0

Options Available
1. Deposit
2. Withdraw
3. Display Balance
4. Compute and Deposit Interest
5 . Exit
Enter choice: 3
Current Balance: 19300.0

Options Available
1. Deposit
2. Withdraw
3. Display Balance
4. Compute and Deposit Interest
5 . Exit
Enter choice: 4
Enter number of years for interest: 2
Interest of 1978.25 added to your account.

Options Available
1. Deposit
2. Withdraw
3. Display Balance
4. Compute and Deposit Interest
5 . Exit
Enter choice: 5
```

Program 6: Packages

Code:
```
package CIE;

import java.util.Scanner;

public class internals extends student {
   protected int marks[] = new int[5];

   public void inputCIEmarks() {
      Scanner s = new Scanner(System.in);
      System.out.println("Enter the Marks of the Subject");
      for (int std : marks) {
         std = s.nextInt();
      }
   }
}

package CIE;

import java.util.Scanner;

public class student {
   protected String usn = new String();
   protected String name = new String();
   protected int sem;

   public void inputStudentDetails() {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter the USN");
      usn = sc.nextLine();
      System.out.println("Enter the Name");
      name = sc.nextLine();
      System.out.println("Enter the Sem");
      sem = sc.nextInt();
   }

   public void displayStudentDetails() {
      System.out.println("USN : " + usn);
      System.out.println("Name : " + name);
      System.out.println("SEM : " + sem);
   }
}

package SEE;

import CIE.internals;
import java.util.Scanner;


public class Externals extends internals{
   protected int marks[];
   protected int finalMarks[];
   public Externals(){
      marks = new int[5];
      finalMarks = new int[5];
   }

   public void inputSEEmarks(){
      Scanner s = new Scanner(System.in);
      System.out.println("Enter 5 see marks\n");

      for(int i=0;i<5;i++){
         System.out.println("SEE mark sin Subject "+ (i+1)+ ":");
         marks[i] = s.nextInt();
      }
   }
   public void calculateFinalMarks(){
      for(int i=0;i<5;i++){
         finalMarks[i] = super.marks[i] + marks[i];
      }
   }
   public void displayFinalMarks(){
      System.out.println("----Final Marks ----");
      this.displayFinalMarks();
      for(int var1 =0;var1 <5;++var1){
         System.out.println("Final Marks in Subject " + (var1 +1) + this.finalMarks[var1]);
      }
   }
}

import SEE.Externals;

public class main {
   public static void main(String[] args) {
      Externals obj = new Externals();
      obj.inputStudentDetails();
      obj.inputCIEmarks();
      obj.inputSEEmarks();
      obj.calculateFinalMarks();
      obj.displayFinalMarks();
   }
}
```

Output:



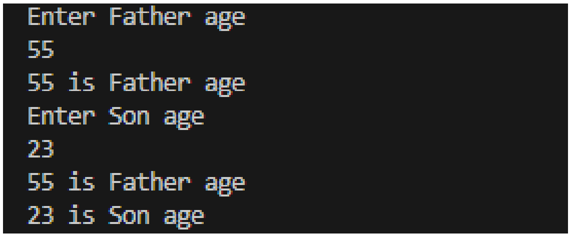
```
in.java } ; if ($?) { java main }
Enter USN:
1BF24CS256
Enter Name:
Rohit Kumawat
Enter Semester:
3

Enter CIE Marks (out of 50):
Subject 1: 47
Subject 2: 48
Subject 3: 49
Subject 4: 50
Subject 5: 46

Enter SEE Marks (out of 100):
Subject 1: 98
Subject 2: 99
Subject 3: 97
Subject 4: 96
Subject 5: 95

USN: 1BF24CS256
Name: Rohit Kumawat
Semester: 3

---- Final Marks ----
Subject 1: 96
Subject 2: 97
Subject 3: 97
Subject 4: 98
Subject 5: 93
PS C:\Users\Rohit Kumawat\OneDrive\Des
```

Program 7: Errors

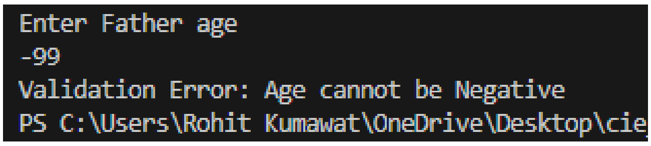
Code:

```
import java.util.Scanner;


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

class Father {
    int Fatherage;
    Father(int Fatherage) throws WrongAgeException{
        this.Fatherage = Fatherage;
        if(Fatherage < 0){
            throw new WrongAgeException("Age cannot be Negative");
        }
        System.out.printf("%d is Father age\n",Fatherage);
    }

}

class Son extends  Father{
    int Sonage;

    Son(int Sonage,int Fatherage) throws WrongAgeException{
        super(Fatherage);
        this.Sonage = Sonage;

        if(Fatherage < Sonage){
            throw new WrongAgeException("Father age Cannot be less than Sonage");
        }
        else if(Sonage < 0){
            throw new WrongAgeException("Age cannot be Negative");
        }
        System.out.printf("%d is Son age\n",Sonage);

    }
}

public  class AgeValidator{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int fage =0;
        int sage = 0;
        try {
        System.out.println("Enter Father age\n");
        fage = sc.nextInt();
        Father fobj = new Father(fage);
        System.out.println("Enter Son age\n");
        sage = sc.nextInt();
        Son sobj = new Son(sage,fage);
        } catch (WrongAgeException e) {

            System.err.println("Validation Error: " + e.getMessage());
        } catch (java.util.InputMismatchException e) {

            System.err.println("Input Error: Please enter a valid integer for the age.");
        } finally {
            sc.close();
        }

    }
}
```

Output:



```
Enter Father age
55
55 is Father age
Enter Son age
23
55 is Father age
23 is Son age
```



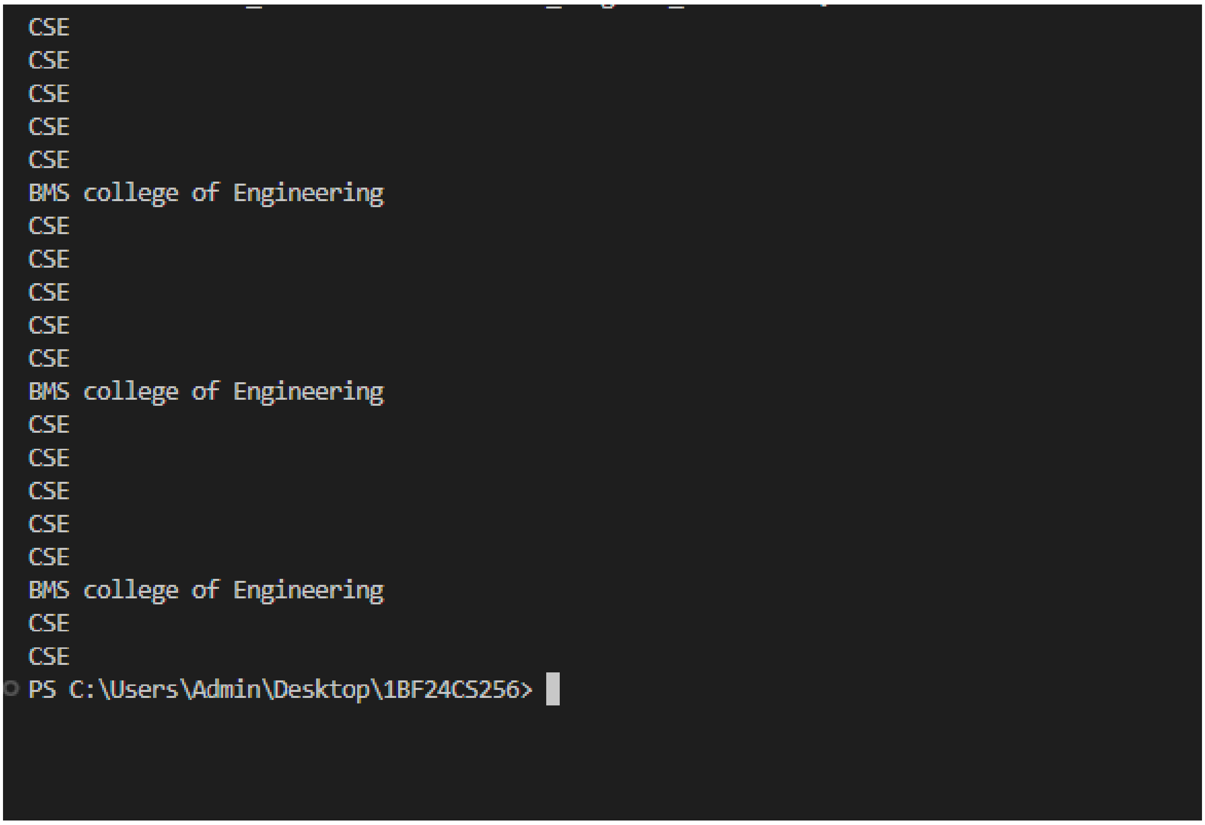
```
Enter Father age
-99
Validation Error: Age cannot be Negative
PS C:\Users\Rohit Kumawat\OneDrive\Desktop\cie
```

Program 8: Multi Threading

Code:

```
package lab_Programn_8;

class college extends Thread {
   public void run()
   {
      try {
         while (true) {
            System.out.println("BMS college of Engineering");
            Thread.sleep(10000);
         }
      } catch (InterruptedException e) {
         System.out.println(e);
      }
   }
}

class Cse extends Thread{
   public void run(){
      try{
         while (true) {
            System.out.println("CSE");
            Thread.sleep(2000);
         }
      }
      catch(InterruptedException e){
         System.out.println(e);
      }
   }
}

public class thread {
   public static void main(String[] args) {
      college t1 = new college();
      Cse t2 = new Cse();

      t1.start();
      t2.start();
   }
}
```

Output:



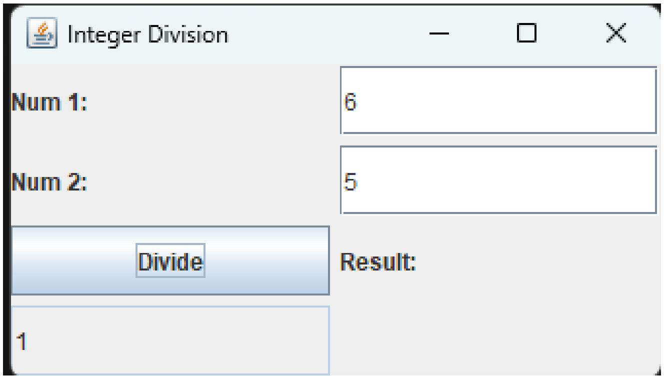
```
CSE
CSE
CSE
CSE
CSE
BMS college of Engineering
CSE
CSE
CSE
CSE
CSE
BMS college of Engineering
CSE
CSE
CSE
CSE
CSE
BMS college of Engineering
CSE
CSE
PS C:\Users\Admin\Desktop\1BF24CS256>
```

Program 9: Open Ended Question 1

Code:

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

public class DivisionUI extends JFrame implements ActionListener {

    JTextField num1Field, num2Field, resultField;
    JButton divideButton;

    DivisionUI() {
        setTitle("Integer Division");
        setSize(350, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new GridLayout(4, 2, 5, 5));


        add(new JLabel("Num 1:"));
        num1Field = new JTextField();
        add(num1Field);

        add(new JLabel("Num 2:"));
        num2Field = new JTextField();
        add(num2Field);

        divideButton = new JButton("Divide");
        add(divideButton);

        divideButton.addActionListener(this);

        add(new JLabel("Result:"));
        resultField = new JTextField();
        resultField.setEditable(false);
        add(resultField);

        setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        try {
            int num1 = Integer.parseInt(num1Field.getText());
            int num2 = Integer.parseInt(num2Field.getText());

            if (num2 == 0) {
                throw new ArithmeticException("Cannot divide by zero");
            }

            int result = num1 / num2;
            resultField.setText(Integer.toString(result));

        } catch (NumberFormatException ex) {
            JOptionPane.showMessageDialog(this,
                    "Please enter valid integers!",
                    "Number Format Error",
                    JOptionPane.ERROR_MESSAGE);

        } catch (ArithmeticException ex) {
            JOptionPane.showMessageDialog(this,
                    ex.getMessage(),
                    "Arithmetic Error",
                    JOptionPane.ERROR_MESSAGE);
        }
    }

    public static void main(String[] args) {
        new DivisionUI();
    }
}
```

Output:

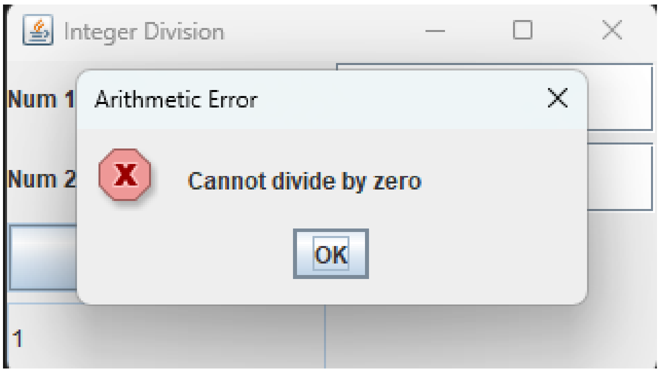


Integer Division

Num 1: 6

Num 2: 5

Result: 1



Integer Division

Arithmetic Error

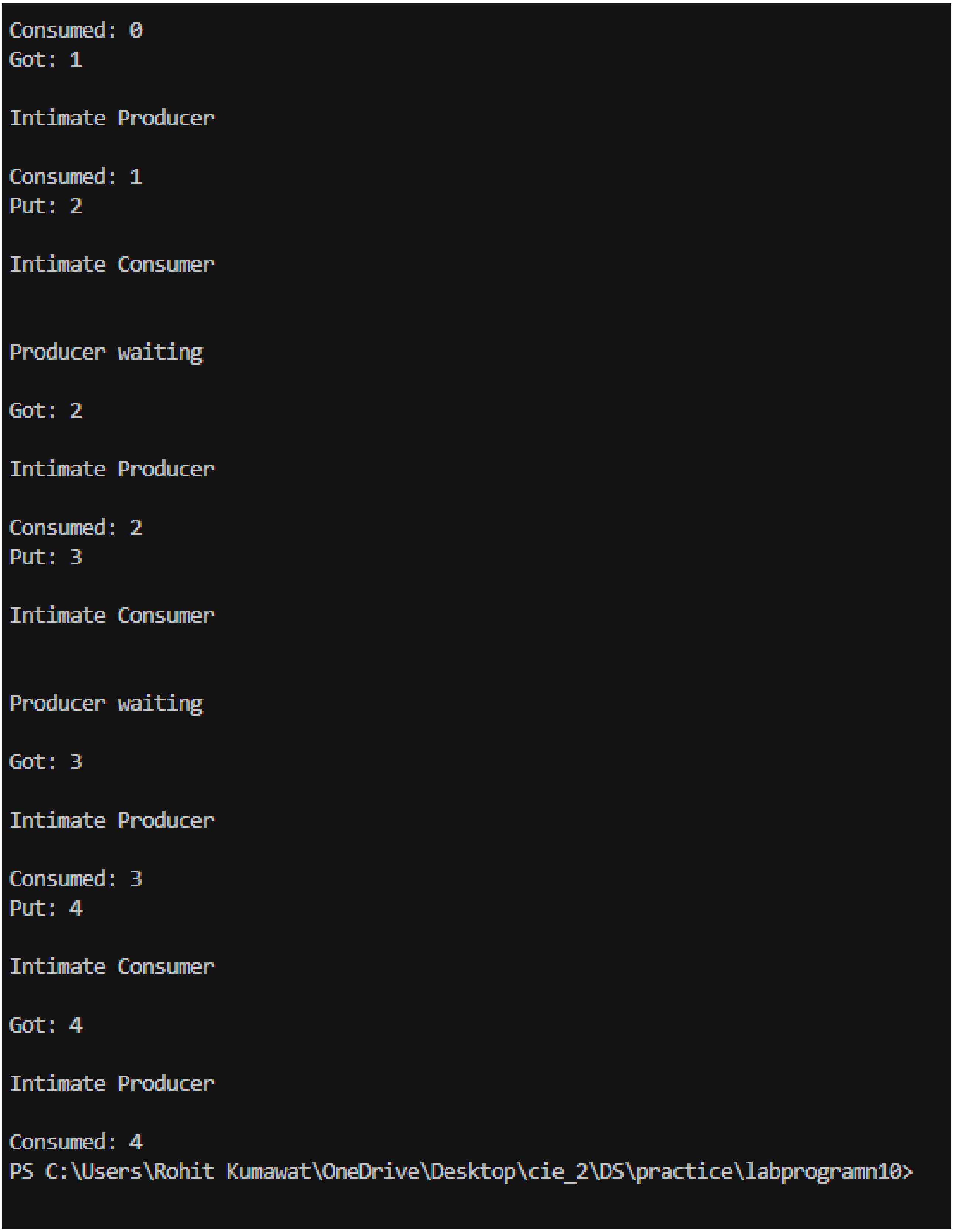
Cannot divide by zero

Program 10: Open Ended Question 2

Code:

```
class Q {

   int n;
   boolean valueSet = false;

   synchronized int get() {
      while (!valueSet) {
         try {
            System.out.println("\nConsumer waiting\n");
            wait();
         } catch (InterruptedException e) {
            System.out.println("InterruptedException caught");
         }
      }

      System.out.println("Got: " + n);
      valueSet = false;

      System.out.println("\nIntimate Producer\n");
      notify();

      return n;
   }

   synchronized void put(int n) {
      while (valueSet) {
         try {
            System.out.println("\nProducer waiting\n");
            wait();
         } catch (InterruptedException e) {
            System.out.println("InterruptedException caught");
         }
      }

      this.n = n;
      valueSet = true;
      System.out.println("Put: " + n);

      System.out.println("\nIntimate Consumer\n");
      notify();
   }
}

class Producer implements Runnable {
    Q q;

    Producer(Q q) {
        this.q = q;
        new Thread(this, "Producer").start();
    }

    public void run() {
        int i = 0;
        while (i < 5) {
            q.put(i++);
        }
    }
}

class Consumer implements Runnable {

    Q q;

    Consumer(Q q) {
        this.q = q;
        new Thread(this, "Consumer").start();
    }

    public void run() {
        int i = 0;
        while (i < 5) {
            int r = q.get();
            System.out.println("Consumed: " + r);
            i++;
        }
    }
}

public class main {
    public static void main(String args[]) {
        Q q = new Q();
        new Producer(q);
        new Consumer(q);
        System.out.println("Press Control-C to stop.");
    }
}
```

Output:



```
Consumed: 0
Got: 1

Intimate Producer

Consumed: 1
Put: 2

Intimate Consumer


Producer waiting

Got: 2

Intimate Producer

Consumed: 2
Put: 3

Intimate Consumer


Producer waiting

Got: 3

Intimate Producer

Consumed: 3
Put: 4

Intimate Consumer

Got: 4

Intimate Producer

Consumed: 4
PS C:\Users\Rohit Kumawat\OneDrive\Desktop\cie_2\DS\practice\labprogramn10>
```