VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted by

Aparna Sankar(1BM23CS047)

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 **Aparna Sankar (1BM23CS047),** 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.

Sheetal V A Assistant Professor Department of CSE, BMSCE	Dr. Jyothi S Nayak Professor & HOD Department of CSE, BMSCE

Index

Github Link:
https://github.com/1BM23CS047/java-lab

APARNA SANKAR

1BM23CS047

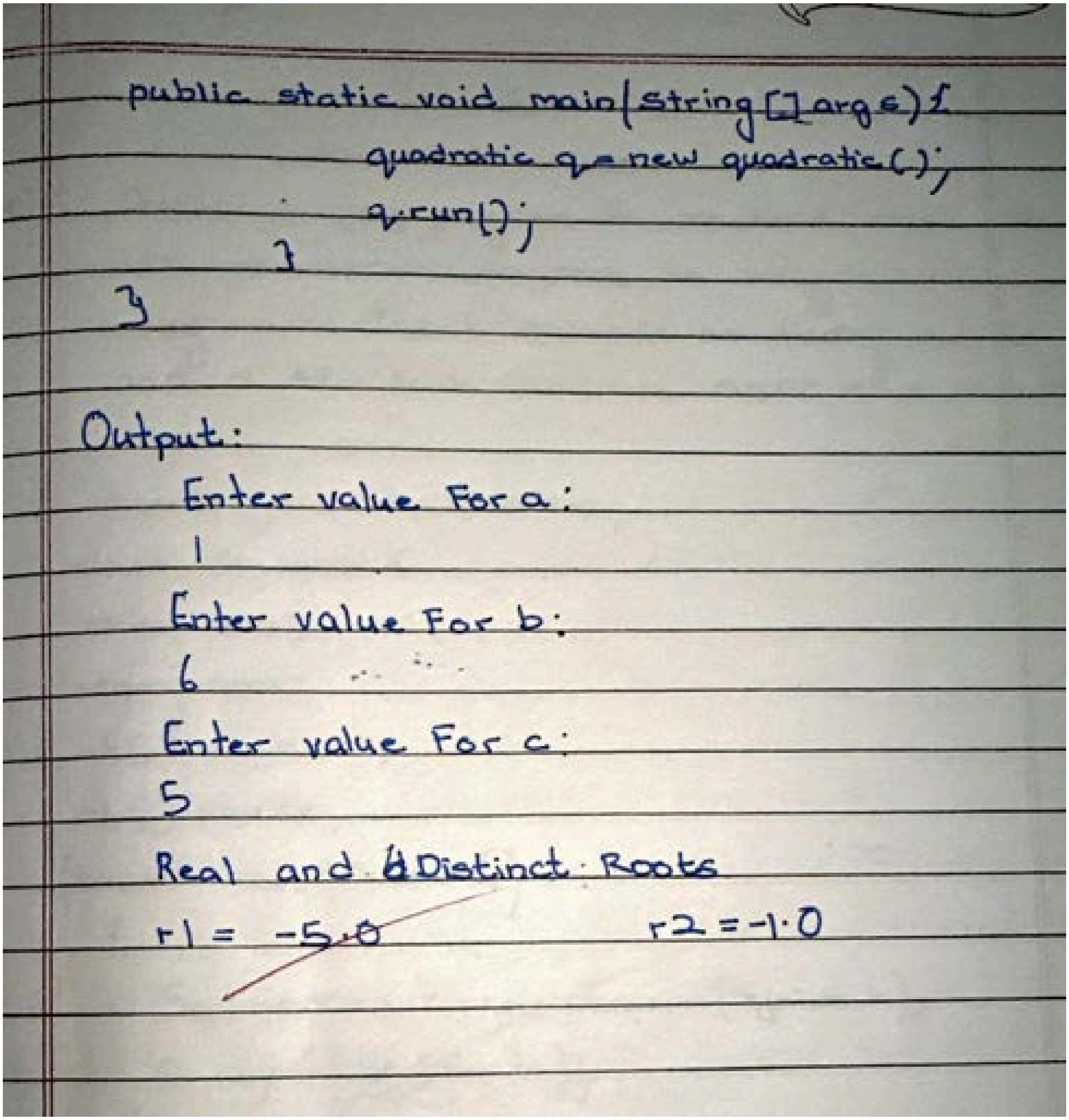
Program 1
Implement Quadratic Equation

Algorithm:

LAB-1 Program

Q) Program to develop a java program that prints all real solutions to the quadratic equation ax2+bx+c=0 Read in a,b,c and use all quadratic Formula

```
→ import java.util.Scanner;
import java.long.Math;

class quadratic
{
   int a,b,c;
  double b;
  double r1,r2;
                          System.out.println("Enter value for b:");
                          b=s.nextInt();
void run()
{
 Scanner s=new Scanner(system.in);
 System.out.println("Enter value for a: ");
 a=s.nextInt();
 System.out.println("Enter value for c: ");
 c=s.nextInt();
 if(a==0){
        System.out.println(" Not a quadratic equation");
   }
 else {
      d= b*b- 4*a*c ;
     if (d==0) {
        r1 = (-b) /(2*a);
       System.out.println(" Roots are real and equal \n
                                  Root : " +r1 + "\n");
      }
 else if(d>0){
      r1 = ((-b) +Math.sqrt(d))/(2*a);
      r2 =((-b)} -Math.sqrt(d))/2*a);
      System.out.println(" Roots are real and distinct \n
                              Roots: r1 =" +r1+"\t r2="
                              +r2 + "\n");
     }
 else {
      r1 =(-b)/(2*a);
      r2 =Math.sqrt (-d)/(2*a);
      System.out.println("Roots are imaginary \n Roots:"
                          +r1+ " j+ " +r2 + "\n");
     }
   }
  }
}
class week 1 {
```



```
public static void main(String[] args){
            quadratic q = new quadratic();
            q.run();
      }
}
```

Output:

```
  Enter value For a:
  1
 Enter value For b:
 6
 Enter value For c:
 5
 Real and ~~d~~Distinct Roots
 r1 =  -5.0                r2 = -1.0
```

code:

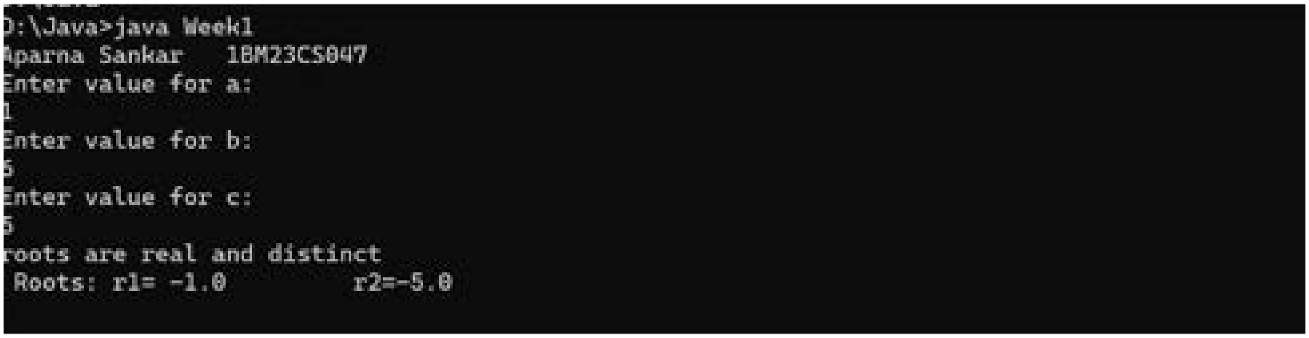
```
import java.util.Scanner;

import java.lang.Math;
```

```
class Quadratic
{
   int a,c,b;
   double d;
   double r1,r2;

void run ()
{
  System.out.println("Aparna Sankar   1BM23CS047");
  Scanner S=new Scanner(System.in);
  System.out.println("Enter value for a: ");
  a=S.nextInt();
  System.out.println("Enter value for b: ");
  b=S.nextInt();
  System.out.println("Enter value for c: ");
  c=S.nextInt();

  if(a==0){
    System.out.println("not a quadratic equation");
  }
```

```
 else {
    d=b*b-4*a*c;
    if (d==0){
    r1=(-b)/(2*a);
     System.out.println("roots are real and equal \n Root: " +r1+
"\n");
    }
    else if(d>0){
    r1=((-b) + Math.sqrt(d))/ (2*a);
    r2=((-b) - Math.sqrt(d))/ (2*a);
    System.out.println("roots are real and distinct \n Roots: r1= "
+r1+ "\t r2=" +r2+ "\n");
    }
else {
    r1=(-b)/(2*a);
   r2=Math.sqrt(-d)/(2*a);
     System.out.println("roots are imaginary \n Roots:" +r1+ "i+"
+r2+ "\n");
    }
    }
 }
```

```
}
class Week1 {
        public static void main(String[] args) {
        Quadratic q = new Quadratic();
        q.run();
        }
}
```

Output:



```
D:\Java>java Week1
Aparna Sankar   1BM23CS047
Enter value for a:
1
Enter value for b:
6
Enter value for c:
5
roots are real and distinct
 Roots: r1= -1.0         r2=-5.0
```

Program 2
sgpa calculator

Algorithm:

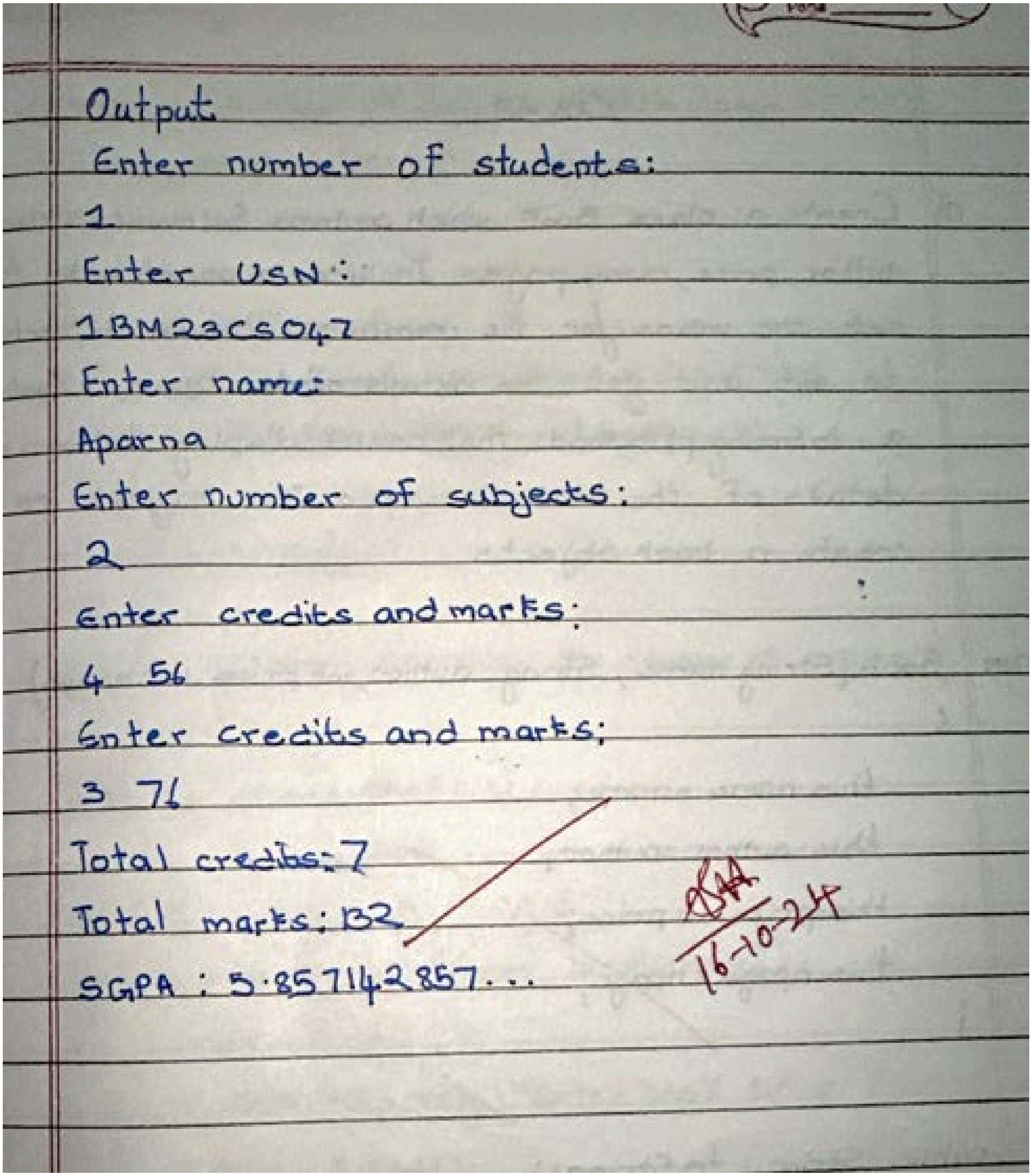
Lab - 2

Q) Develop a java program to create a class Student with members USN, name, an array credits and an array marks. Include methods to display details and a method to calculate SGPA of a student.

```
import java.util.Scanner;
class Student {
string usn;
string name;
int[] credits;
int[] marks;
void acceptDetails()
 {
    Scanner mark =new Scanner(System.in);
    ~~Scanner(System.in);~~
    ~~System.out~~
    System.out.println("Enter USN:");
    usn = mark.nextLine();
    System.out.println("Enter Name:");
    name = mark.nextLine();
    System.out.println("Enter number of subjects:");
    int n = mark.nextInt();
    credits = new int[n];
```

```
        marks= new int[n];
        for (int i =0 ; i<credits.length; i++)
      {
          System.out.println("Enter credits and marks");
          credits[i] =mark.nextInt();
          marks[i] =mark.nextInt();
       }
   }

   void calculateSgpa()
   {
       int c=0;
       int m=0;
      for(int i=0; i<credits.length; i++)
   {
         c += credits[i];
         m += marks[i];
       }
      System.out.println(" Total credits;" +c);
      System.out.println(" Total marks:" +m);
      double s =0;
      for (int i =0; i <credits.length;i++){
          s += (marks[i]/10 *credits[i]);
      }
      double sgpa=s/c;
      System.out.println("SGPA:" +sgpa);
```

```
}
void display(){
  System.out.println("USN:" +usn);
  System.out.println("Name:" +name);
  For( int i =0; i< credits.length; i++){
     System.out.println("Credits for subject "+(i+1)
                        + "is:" +credits[i]);
     S.O.P ("Marks for subject" +(i+1)+
                     "is:"+ marks [i] + "\n")
  }
}

public static voidmain ( String[] Args)
{
   SOP(" Name: Aparna Sankar  USN: 1BM23CS047");
   int a, i ,
   Scanner object=new Scanner( system.in);
   System.out.println(" Enter no. of students;");
   a =Object.nextInt();
   For(i=0; i<a; i++){
      student obj =new student();
      obj.acceptdetails();
      obj.calculate Sgpa();
      obj.display();
   }
}
```



Output

```
Enter number of students:
1
Enter USN:
1BM23CS047
Enter name:
Aparna
Enter number of subjects:
2
Enter credits and marks:
4 56
Enter credits and marks;
3 76
Total credits:7
Total marks:132
SGPA : 5.857142857...
```

16-10-24

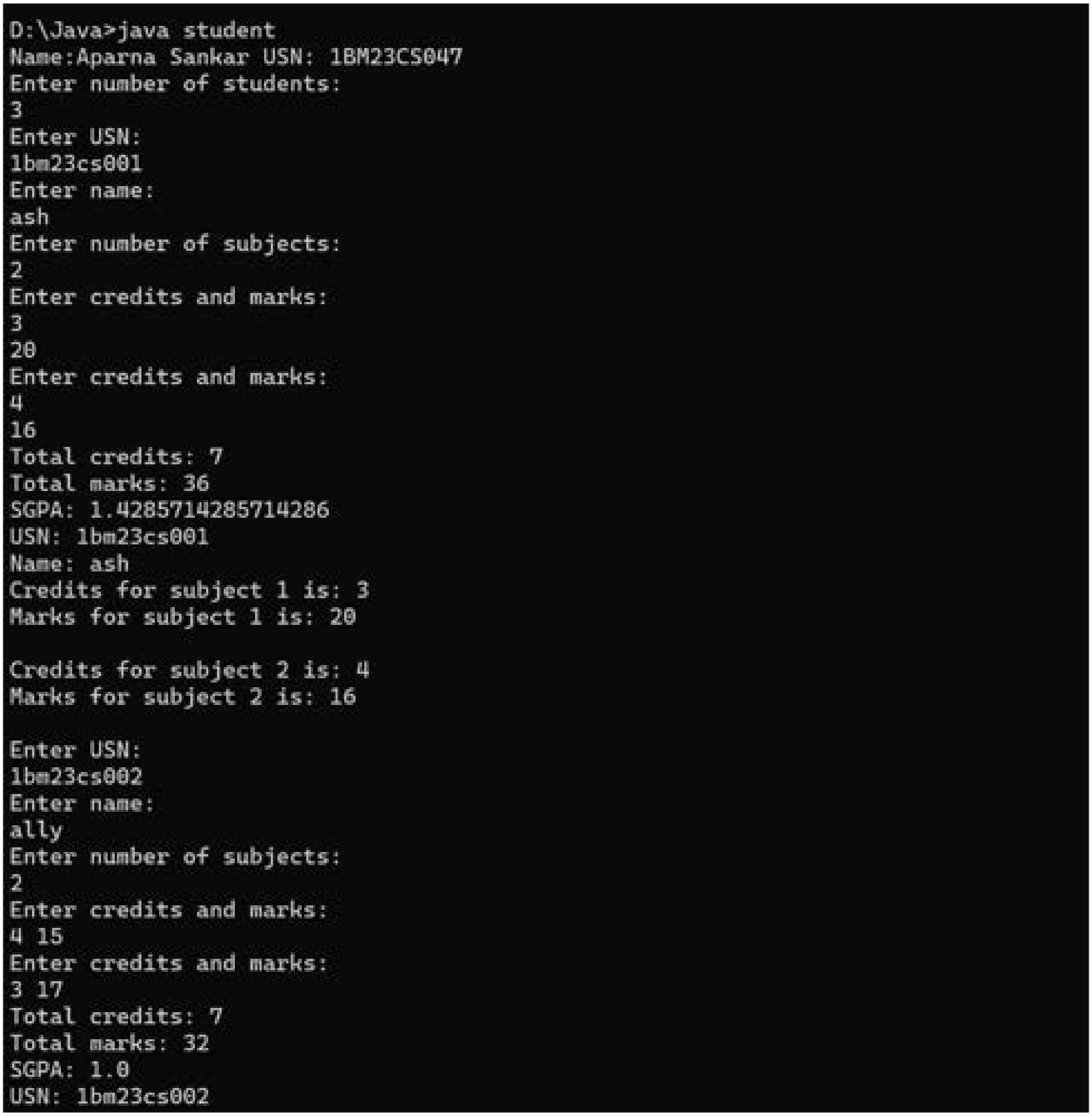
Program:

```
import java.util.Scanner;

class student
{
String usn;
String name;
int[] credits;
int[] marks;

void acceptdetails()
{
        Scanner mark = new Scanner(System.in);
        System.out.println("Enter USN: ");
        usn = mark.nextLine();
        System.out.println("Enter name: ");
        name = mark.nextLine();
        System.out.println("Enter number of subjects: ");
        int n = mark.nextInt();
        credits = new int[n];
        marks = new int[n];
        for (int i = 0; i < credits.length; i++)
{
        System.out.println("Enter credits and marks: ");
```

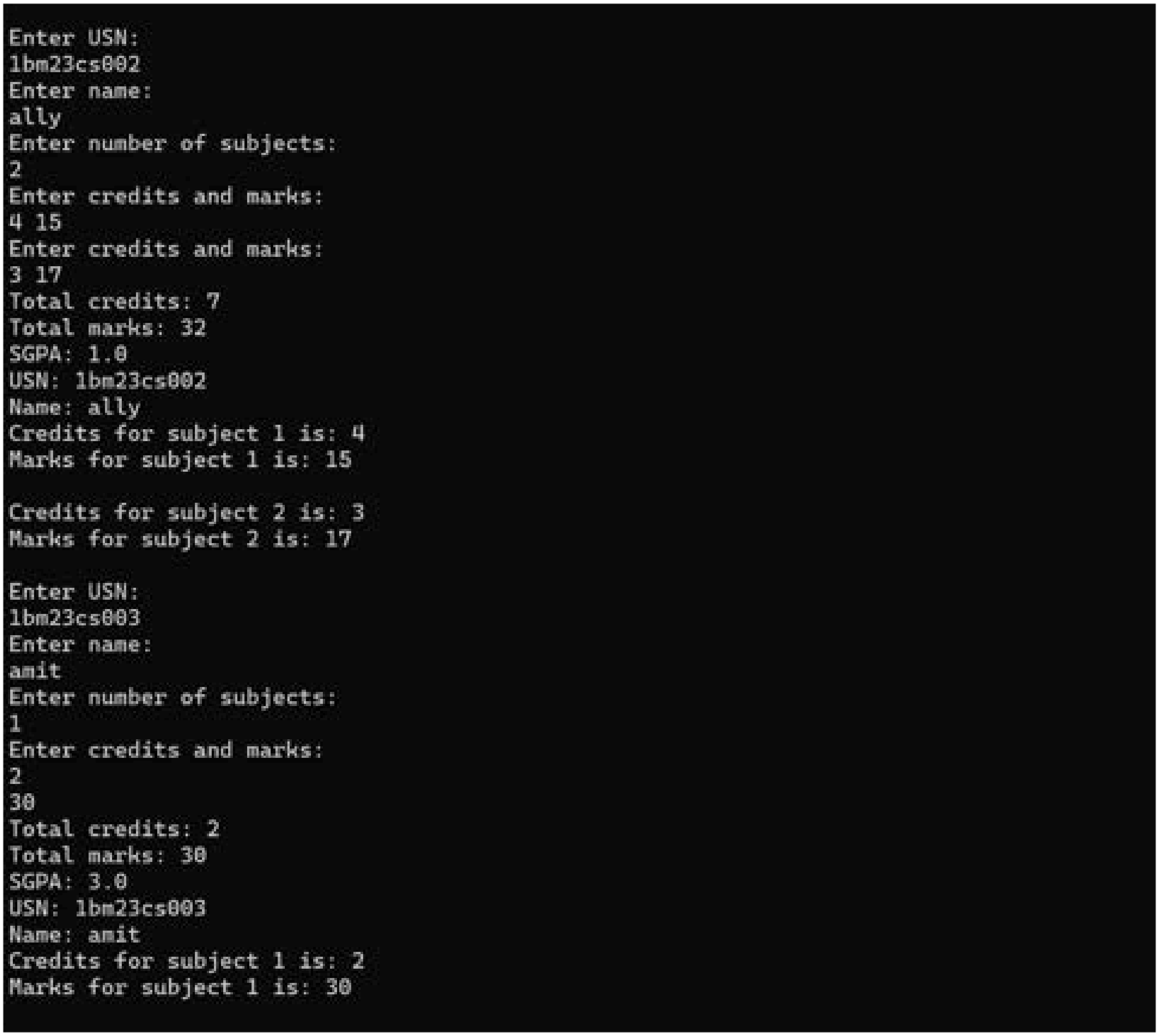
```
    credits[i] = mark.nextInt();
    marks[i] = mark.nextInt();
    }
}

    void calculateSgpa()
{
    int c = 0;
    int m = 0;
    for (int i = 0; i < credits.length; i++)
{
    c += credits[i];
    m += marks[i];
    }
    System.out.println("Total credits: " + c);
    System.out.println("Total marks: " + m);
    double s = 0;
    for (int i = 0; i < credits.length; i++) {
    s += (marks[i] / 10 * credits[i]);
    }
    double sgpa = s / c;
    System.out.println("SGPA: " + sgpa);
    }

    void display() {
    System.out.println("USN: " + usn);
  System.out.println("Name: " + name);
    for (int i = 0; i < credits.length; i++) {
    System.out.println("Credits for subject " + (i + 1) + " is: " + credits[i]);
    System.out.println("Marks for subject " + (i + 1) + " is: " + marks[i] + "\n");
    }
    }

    public static void main(String[] Args)
    {
    System.out.println("Name:Aparna Sankar USN: 1BM23CS047");
    int a, i;
    Scanner object = new Scanner(System.in);
    System.out.println("Enter number of students: ");
    a = object.nextInt();
    for (i = 0; i < a; i++) {
    student obj = new student();
    obj.acceptdetails();
    obj.calculateSgpa();
    obj.display();
    }
    }
}import java.util.Scanner;

class student
{
String usn;
String name;
int[] credits;
int[] marks;

void acceptdetails()
{
    Scanner mark = new Scanner(System.in);
    System.out.println("Enter USN: ");
    usn = mark.nextLine();
    System.out.println("Enter name: ");
    name = mark.nextLine();
    System.out.println("Enter number of subjects: ");
    int n = mark.nextInt();
    credits = new int[n];
    marks = new int[n];
    for (int i = 0; i < credits.length; i++)
{
       System.out.println("Enter credits and marks: ");
    credits[i] = mark.nextInt();
    marks[i] = mark.nextInt();
    }
}

    void calculateSgpa()
{
    int c = 0;
    int m = 0;
    for (int i = 0; i < credits.length; i++)
{
    c += credits[i];
    m += marks[i];
    }
    System.out.println("Total credits: " + c);
    System.out.println("Total marks: " + m);
    double s = 0;
    for (int i = 0; i < credits.length; i++) {
    s += (marks[i] / 10 * credits[i]);
    }
    double sgpa = s / c;
    System.out.println("SGPA: " + sgpa);
    }

    void display() {
    System.out.println("USN: " + usn);
    System.out.println("Name: " + name);
    for (int i = 0; i < credits.length; i++) {
    System.out.println("Credits for subject " + (i + 1) + " is: " + credits[i]);
    System.out.println("Marks for subject " + (i + 1) + " is: " + marks[i] + "\n");
    }
    }

    public static void main(String[] Args)
```

```
        {
        System.out.println("Name:Aparna Sankar USN: 1BM23CS047");
        int a, i;
        Scanner object = new Scanner(System.in);
        System.out.println("Enter number of students: ");
        a = object.nextInt();
        for (i = 0; i < a; i++) {
        student obj = new student();
        obj.acceptdetails();
        obj.calculateSgpa();
        obj.display();
        }
        }
}
```

Output:



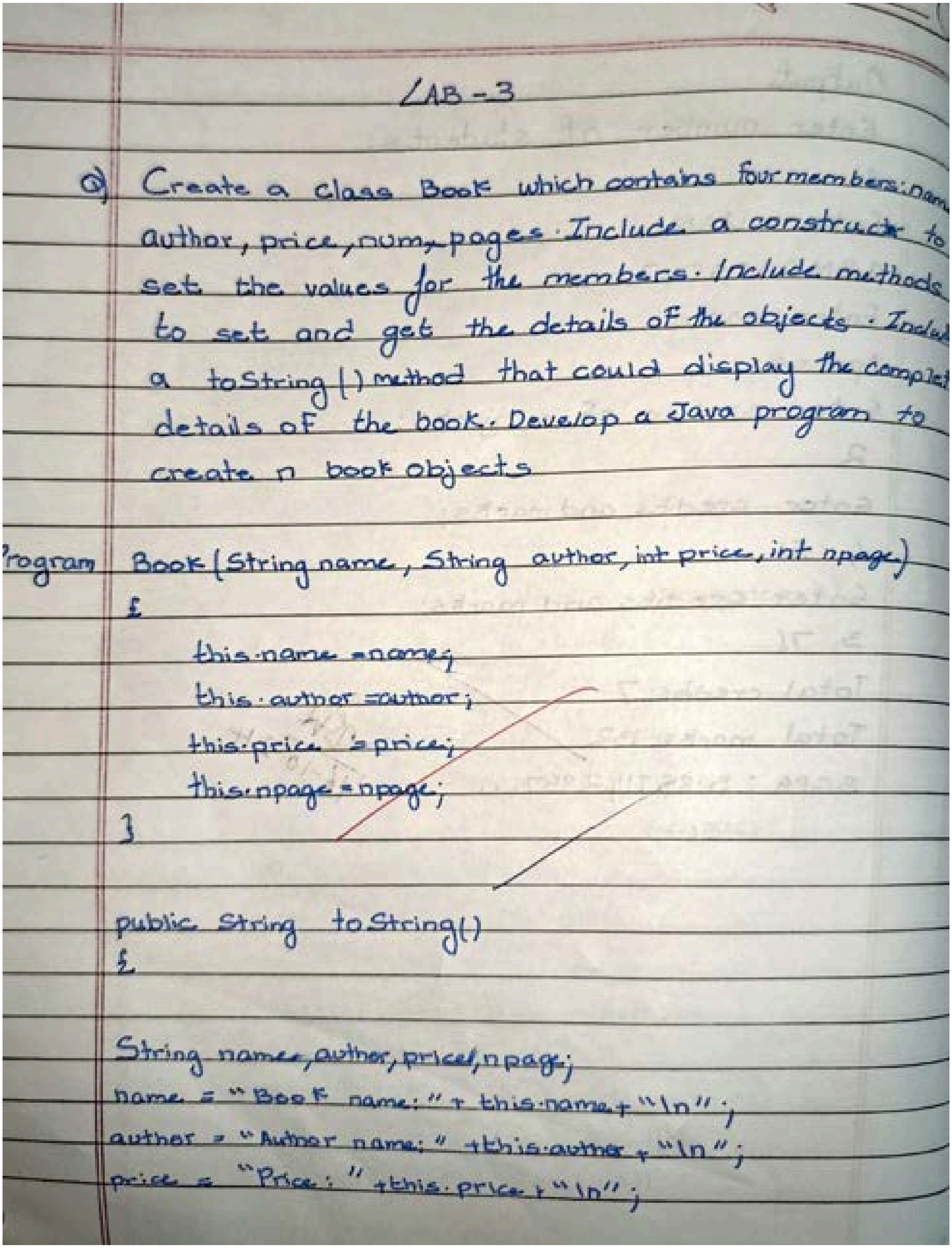
```
D:\Java>java student
Name:Aparna Sankar USN: 1BM23CS047
Enter number of students:
3
Enter USN:
1bm23cs001
Enter name:
ash
Enter number of subjects:
2
Enter credits and marks:
3
20
Enter credits and marks:
4
16
Total credits: 7
Total marks: 36
SGPA: 1.4285714285714286
USN: 1bm23cs001
Name: ash
Credits for subject 1 is: 3
Marks for subject 1 is: 20

Credits for subject 2 is: 4
Marks for subject 2 is: 16

Enter USN:
1bm23cs002
Enter name:
ally
Enter number of subjects:
2
Enter credits and marks:
4 15
Enter credits and marks:
3 17
Total credits: 7
Total marks: 32
SGPA: 1.0
USN: 1bm23cs002
```



```
Enter USN:
1bm23cs002
Enter name:
ally
Enter number of subjects:
2
Enter credits and marks:
4 15
Enter credits and marks:
3 17
Total credits: 7
Total marks: 32
SGPA: 1.0
USN: 1bm23cs002
Name: ally
Credits for subject 1 is: 4
Marks for subject 1 is: 15

Credits for subject 2 is: 3
Marks for subject 2 is: 17

Enter USN:
1bm23cs003
Enter name:
amit
Enter number of subjects:
1
Enter credits and marks:
2
30
Total credits: 2
Total marks: 30
SGPA: 3.0
USN: 1bm23cs003
Name: amit
Credits for subject 1 is: 2
Marks for subject 1 is: 30
```

Program 3
book class

Algorithm:



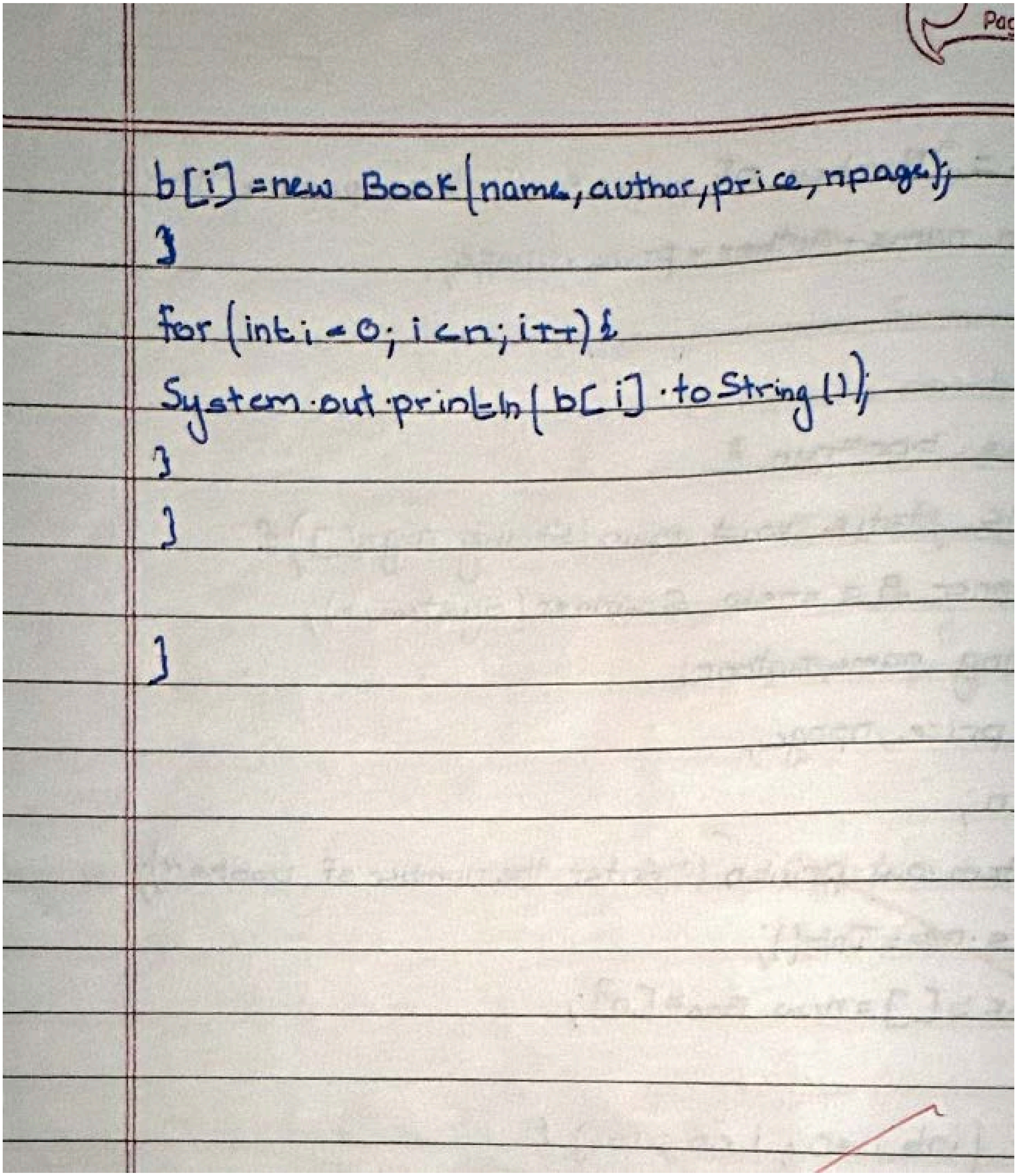
LAB - 3

Q) Create a class Book which contains four members: nam[e] author, price, num-pages. Include a construct[or] to set the values for the members. Include methods to set and get the details of the objects. Includ[e] a toString() method that could display the complet[e] details of the book. Develop a Java program to create n book objects

Program

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

public String toString()
{

String names,auther,price,npage;
name = "Book name:" + this.name+"\n";
auther = "Author name:" +this.auther + "\n";
price = "Price:" +this.price + "\n";
```

```
npage = "Number of pages :"+this.npage +"\n";
return name+author + price +npage;
}
}
class bookrun {
public static void main(String args[]) {
Scanner ~~A~~ s =new Scanner(System.in);
String name,author;
int price,npage;
int n;
System.out.println("Enter the number of books");
n = s.nextInt();
Book b[] =new Book[n];

for (int i=0; i<n; i++) {
System.out.println("Enter book name");
    name = s.next();
    System.out.println("Enter book Author");
    author = s.next();
    System.out.println("Enter book Price");
    price = s.nextInt();
    System.out.println("Enter the number of Pages in the book
   npage =s.nextInt();
```



```
b[i]=new Book(name,author,price,npage);
}
for(int i=0; i<n; i++){
System.out.println(b[i].toString());
}
}

}
```

Output.

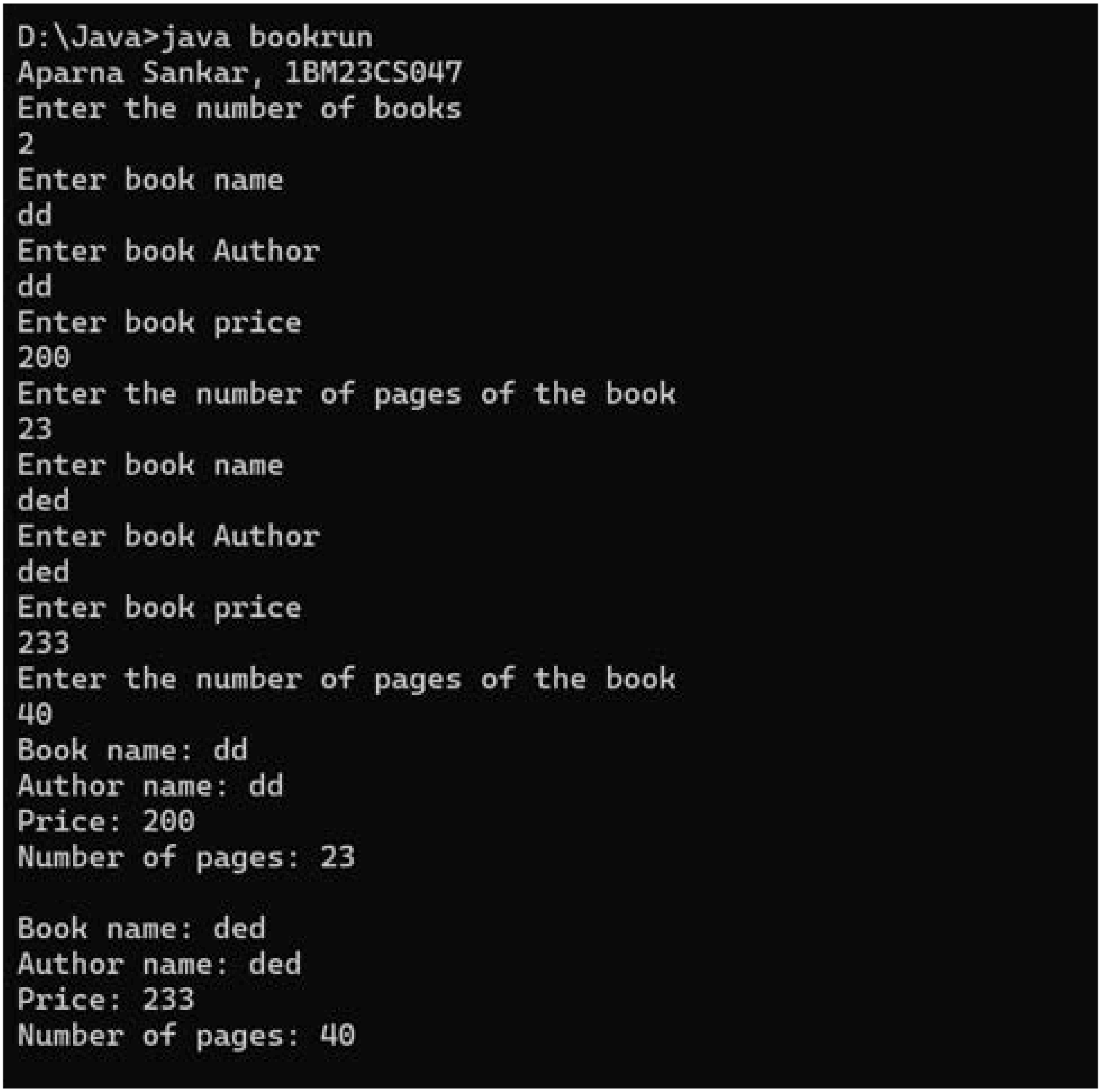
```
Enter the number of books:
1
Enter book name.
abc
Enter book: Author
xyz
~~Enter the number of pages of the book~~
Enter book price
350
Enter the number of pages of the book
100

Book name: abc
Author name: xyz.
Price: 350
Number of pages: 100
```

Program:

```
import java.util.Scanner;


class Book {
        String name, author;
        int price, npage;


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


        public String toString() {
        String name, author, price, npage;


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

```
class bookrun {
      public static void main(String args[]) {
      // Display name and ID at the start
      System.out.println("Aparna Sankar, 1BM23CS047");

      Scanner s = new Scanner(System.in);
      String name, author;
      int price, npage;
      int n;

      System.out.println("Enter the number of books");
      n = s.nextInt();
      Book b[] = new Book[n];

    for (int i = 0; i < n; i++) {
      System.out.println("Enter book name");
      name = s.next();
      System.out.println("Enter book Author");
      author = s.next();
      System.out.println("Enter book price");
          price = s.nextInt();
      System.out.println("Enter the number of pages of the book");
      npage = s.nextInt();

      b[i] = new Book(name, author, price, npage);
      }
```

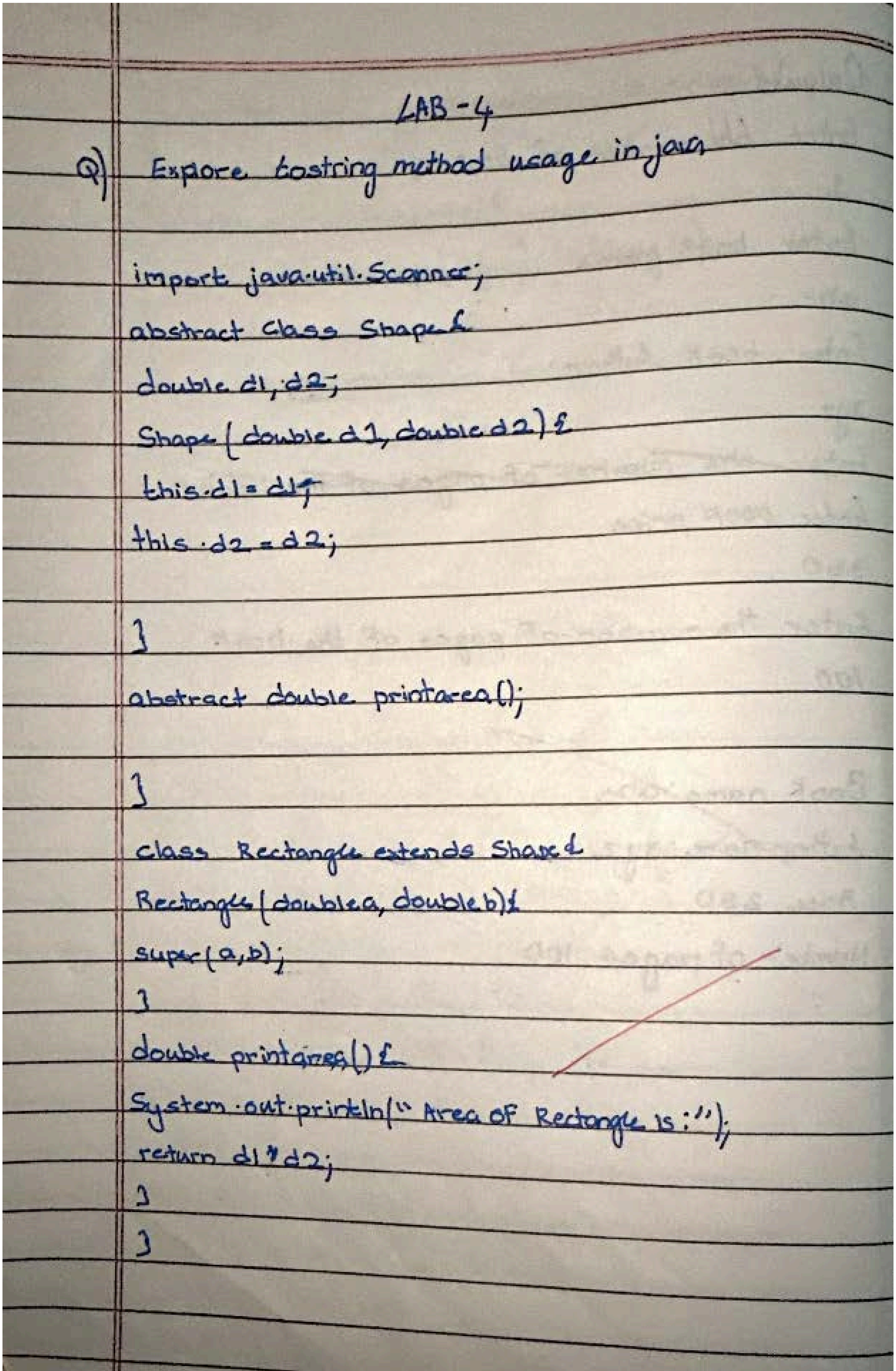
```
        for (int i = 0; i < n; i++) {
        System.out.println(b[i].toString());
        }


        s.close();
        }
}
```



```
D:\Java>java bookrun
Aparna Sankar, 1BM23CS047
Enter the number of books
2
Enter book name
dd
Enter book Author
dd
Enter book price
200
Enter the number of pages of the book
23
Enter book name
ded
Enter book Author
ded
Enter book price
233
Enter the number of pages of the book
40
Book name: dd
Author name: dd
Price: 200
Number of pages: 23

Book name: ded
Author name: ded
Price: 233
Number of pages: 40
```

Program 4
shape area

Algorithm:

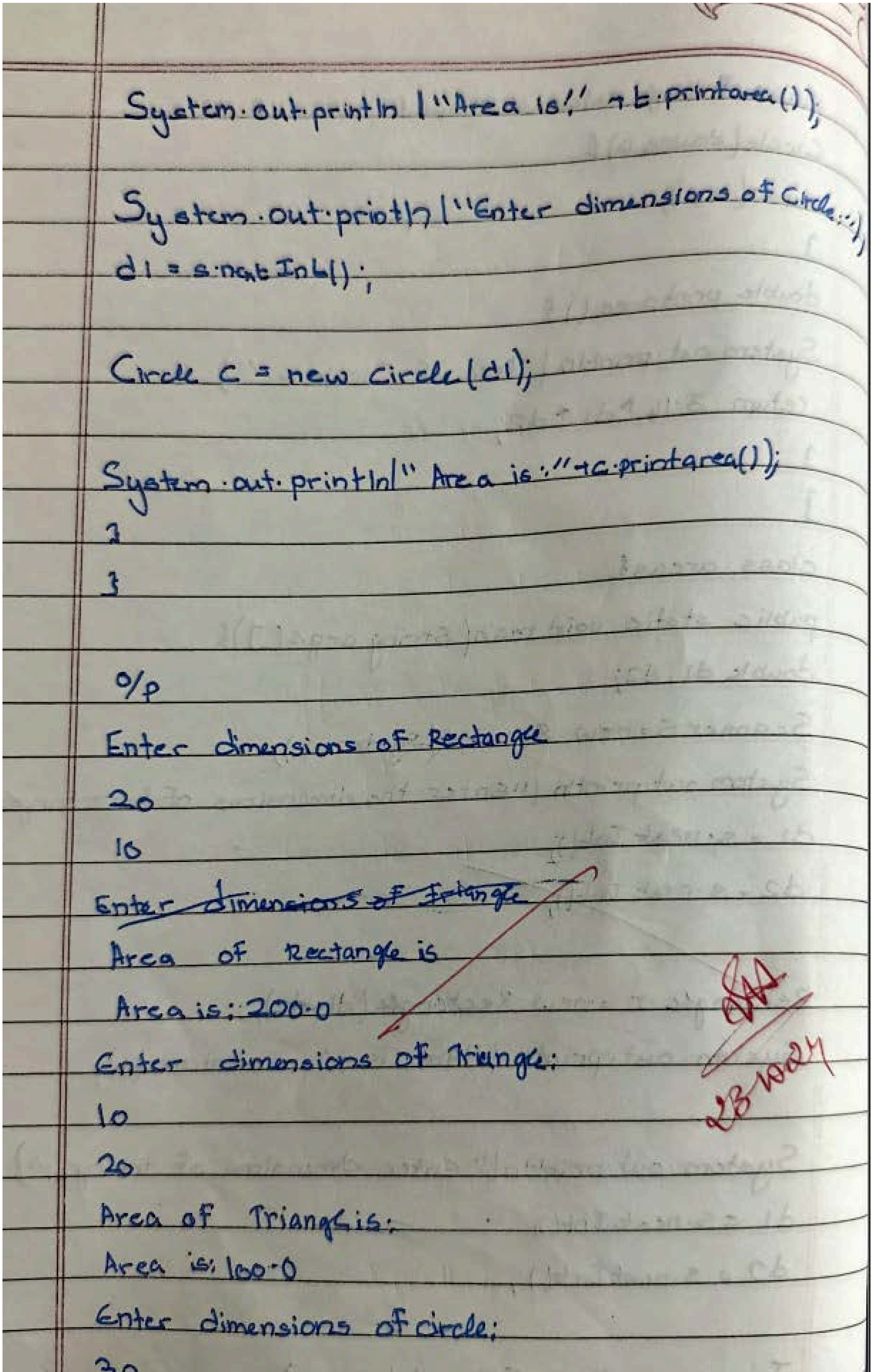


LAB - 4

Q) Explore tostring method usage in java

```
import java.util.Scanner;
abstract Class Shape{
double d1,d2;
Shape(double d1, double d2){
this.d1=d1;
this.d2=d2;

}
abstract double printarea();

}
class Rectangle extends Shape{
Rectangle(double a, double b){
super(a,b);
}
double printarea(){
System.out.println(" Area of Rectangle is:");
return d1*d2;
}
}
```

```
class Triangle extends shape {
Circle( double a) {
super( a ,1);
}
double printarea() {
System.out.println (''Area of Circle is:'');
return 3.14 * d1 * d2;
}
}
class areas {
public static void main( String args[]) {
double d1, d2;
Scanner S=new Scanner (system.in);
System.out.println (''Enter the dimensions of the rectangle:'')
d1 = s.nextInt();
d2 = s.nextInt();

Rectangle r =new Rectangle(d1,d2);
System.out.println (''Area is ;''+r.printarea());

System.out.println('' Enter dimensions of triangle:'')
d1 =s.nextInt();
d2 = s.nextInt();

Triangle t=new Triangle(d1, d2);
```



```
System.out.println ("Area is!' +t.printarea());

System.out.println("Enter dimensions of Circle:");
d1 = s.nextInt();

Circle c = new circle(d1);

System.out.println(" Area is:"+c.printarea());
}
}
```

O/p

```
Enter dimensions of Rectangle
20
10
Enter ~~dimensions of Triangle~~
Area of Rectangle is
Area is: 200.0
Enter dimensions of Triangle:
10
20
Area of Triangleis:
Area is: 100.0
Enter dimensions of circle:
20
```

23/10/24

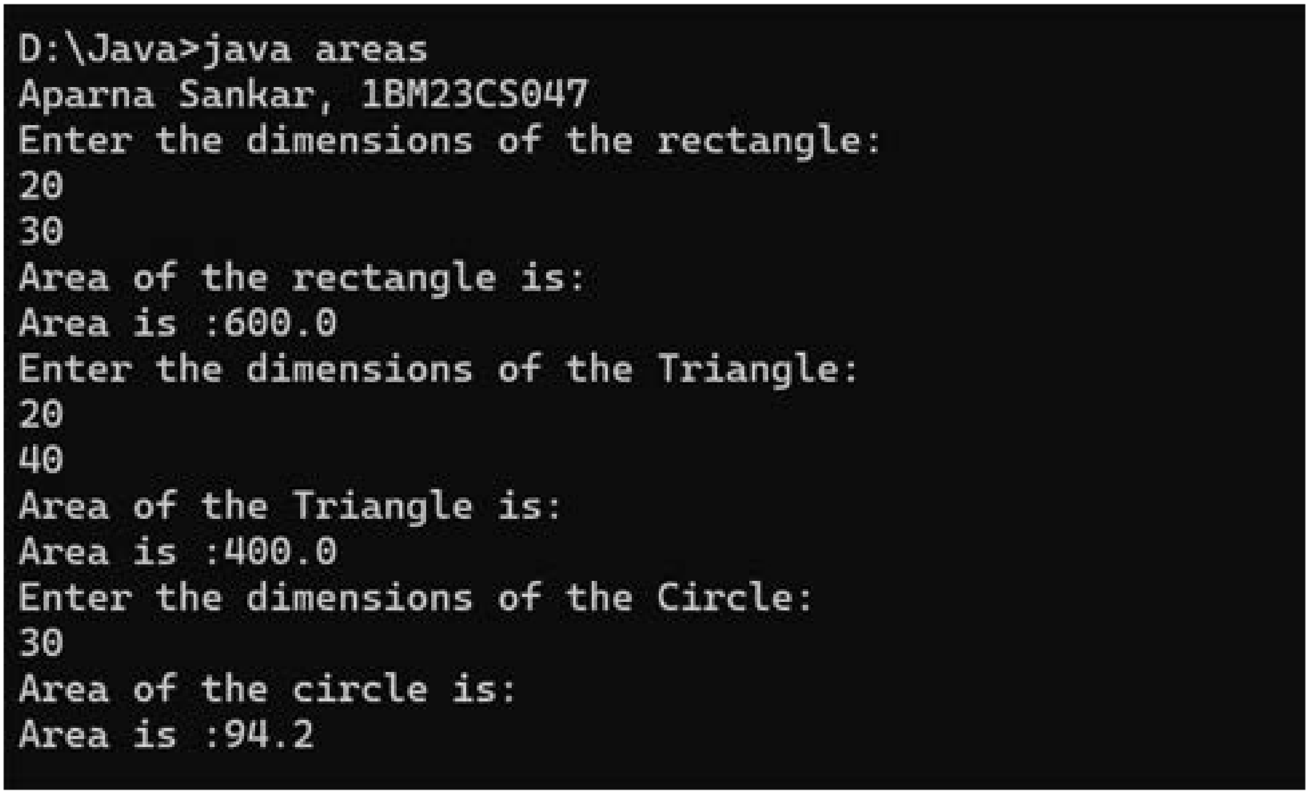
Program:

```
import java.util.Scanner;
abstract class Shape {
double d1,d2;
Shape(double d1,double d2) {
this.d1=d1;
this.d2=d2;


}
abstract double printarea();


}
class Rectangle extends Shape {
Rectangle(double a,double b) {
super(a,b);
}
double printarea() {
System.out.println("Area of the rectangle is: ");
return d1*d2;
}
}

class Triangle extends Shape {
Triangle(double a,double b) {
super(a,b);
```

```
}
double printarea() {
System.out.println("Area of the Triangle is: ");
return d1*d2/2;
}
}

class Circle extends Shape {
Circle(double a) {
super(a,1);
}
double printarea() {
System.out.println("Area of the circle is: ");
return 3.14*d1*d2;
}
}
class areas {
public static void main(String args[]) {
System.out.println("Aparna Sankar, 1BM23CS047");
double d1,d2;
Scanner s=new Scanner(System.in);
System.out.println("Enter the dimensions of the rectangle:");
d1=s.nextInt();
d2=s.nextInt();

Rectangle r = new Rectangle(d1,d2);
```

```
System.out.println("Area is :"+r.printarea());


System.out.println("Enter the dimensions of the Triangle:");
d1=s.nextInt();
d2=s.nextInt();


Triangle t = new Triangle(d1,d2);
System.out.println("Area is :"+t.printarea());


System.out.println("Enter the dimensions of the Circle:");
d1=s.nextInt();


Circle c = new Circle(d1);


System.out.println("Area is :"+c.printarea());
}
}
```

Output:



```
D:\Java>java areas
Aparna Sankar, 1BM23CS047
Enter the dimensions of the rectangle:
20
30
Area of the rectangle is:
Area is :600.0
Enter the dimensions of the Triangle:
20
40
Area of the Triangle is:
Area is :400.0
Enter the dimensions of the Circle:
30
Area of the circle is:
Area is :94.2
```

Program 5
bank account

Algorithm:

LAB-5

Q3) ~~import java.util.Scanner;~~ 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 ~~account~~ current accoun The savings account provides compound interest and withdrawal Facilities but nocheque book facility, The current account provides cheque book facility but no interest. Current account holders shou 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 drive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the Following ta

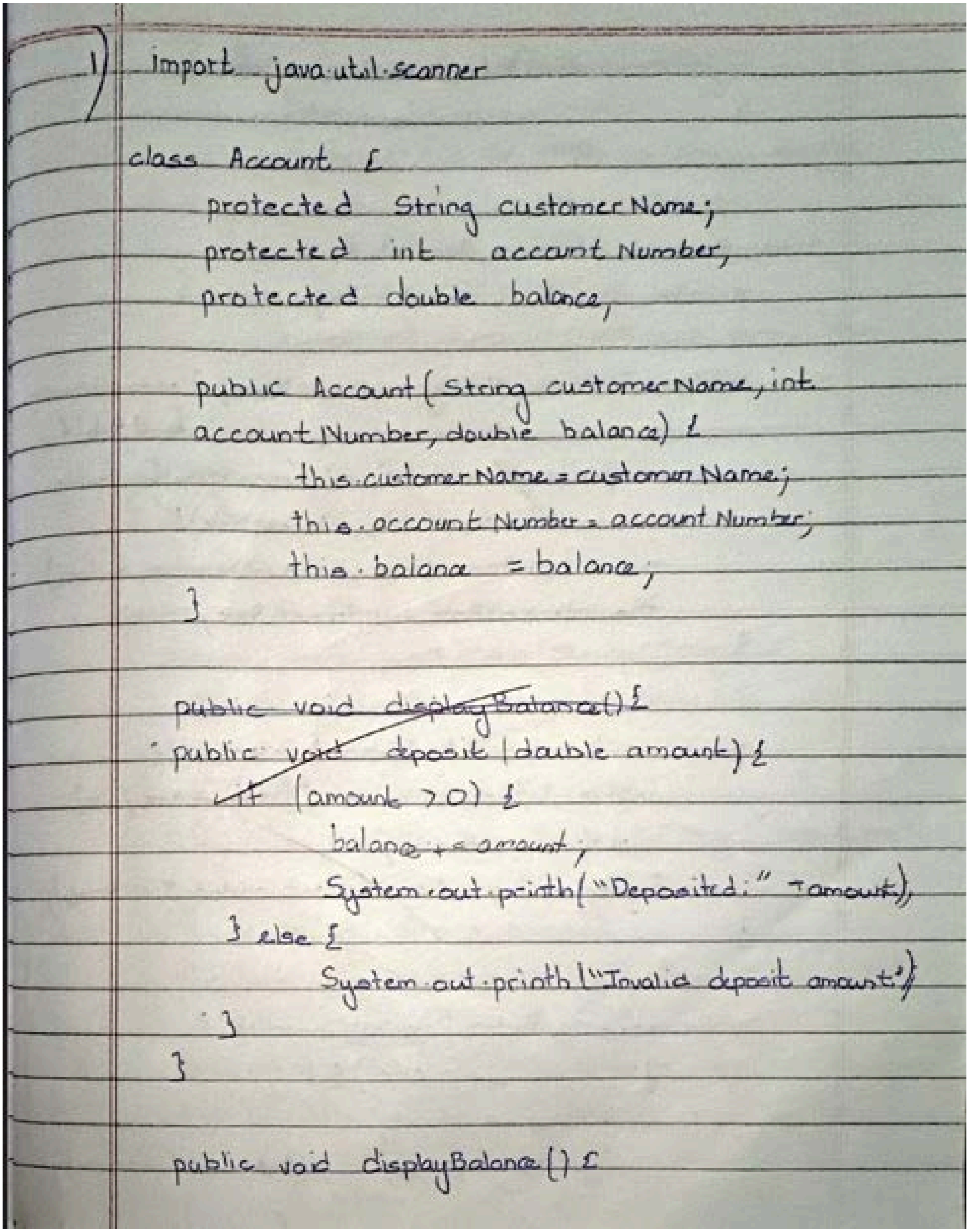
a) Accept Deposit from customer and update the balance

b) Display the balance.

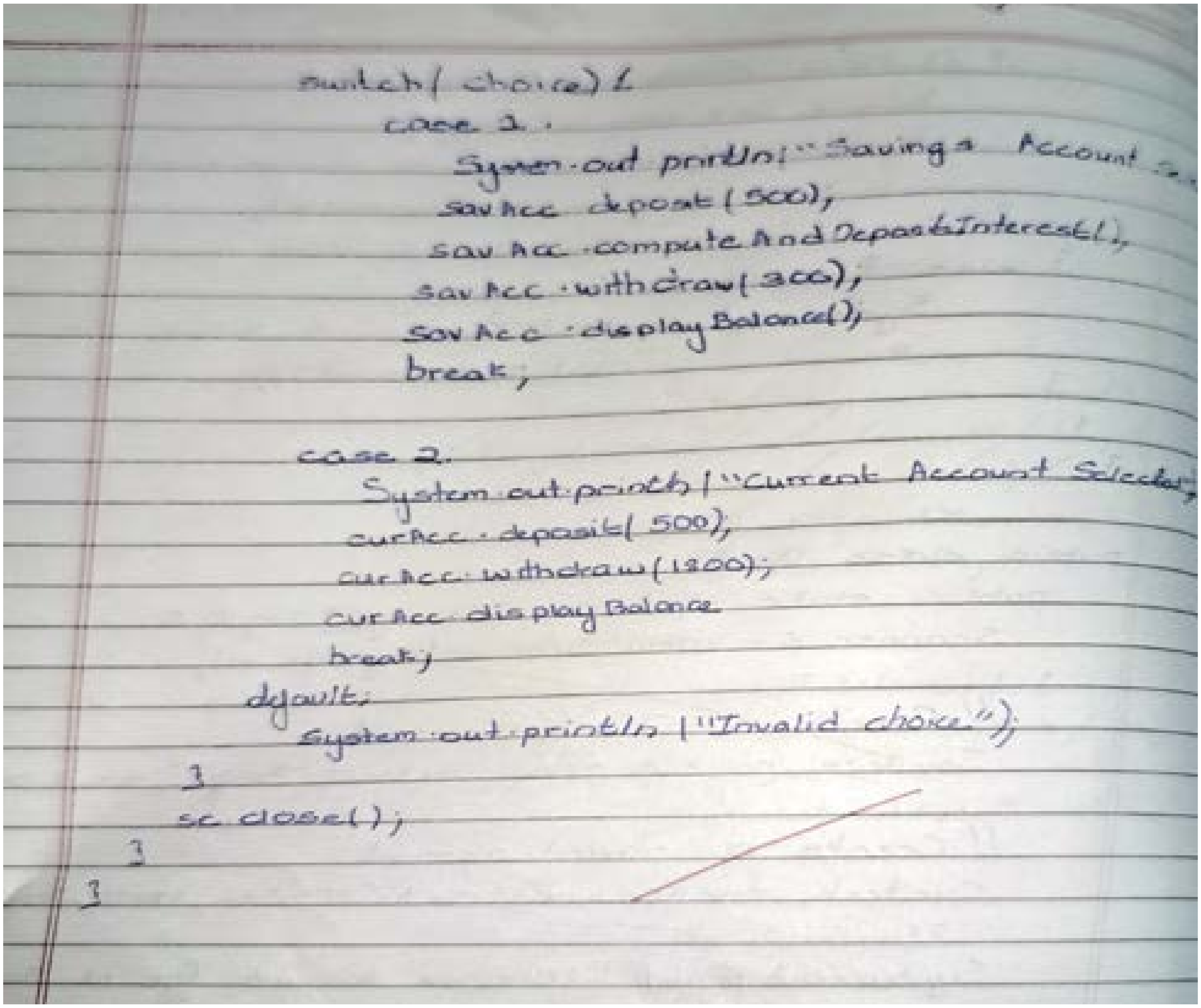
c) Compute and deposit interest

d) Permit withdrawal and update the balance

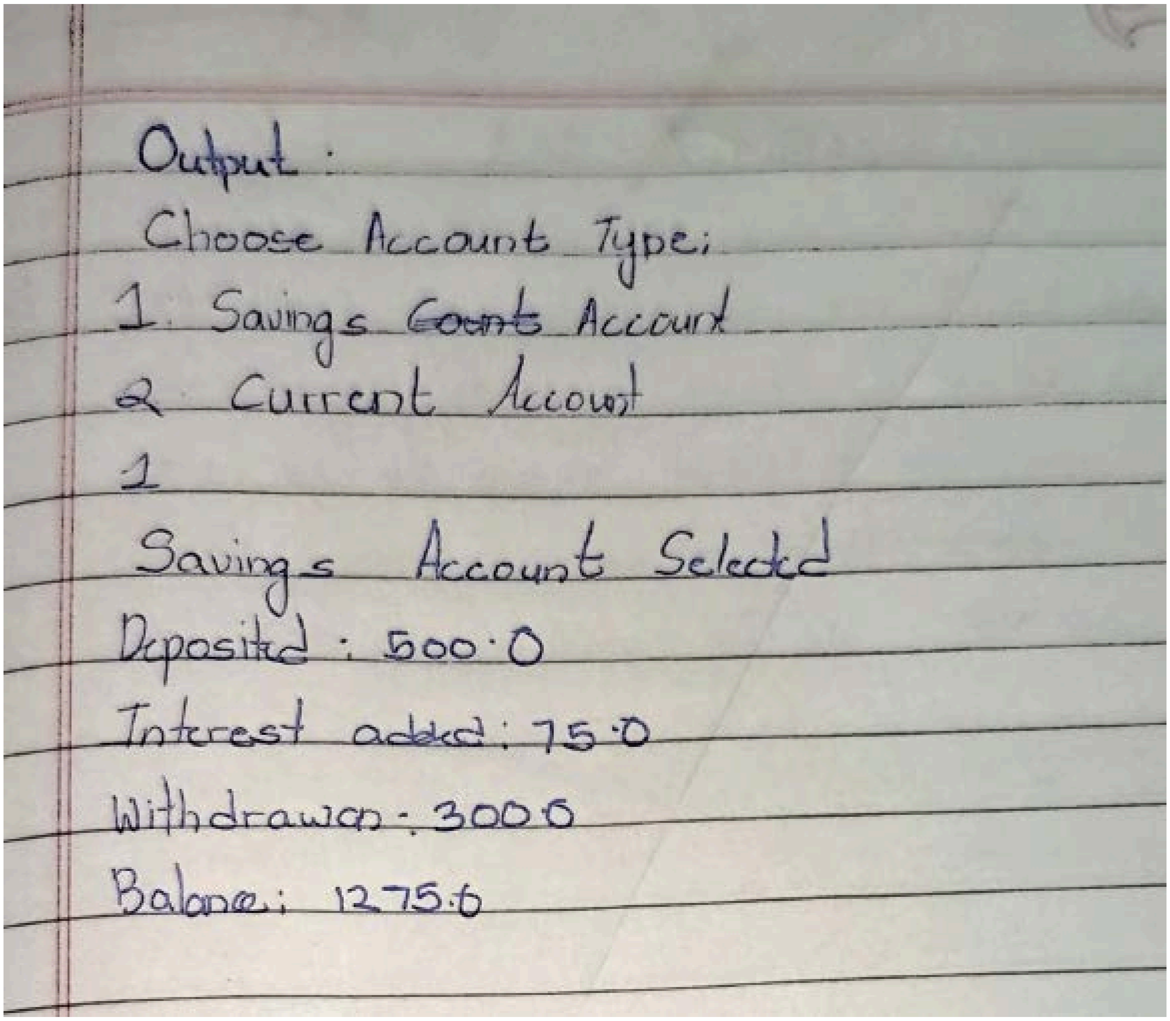
e) Check for the minimum balance, impose penalty if necessary and update the balance



```
1) import java.util.scanner

class Account {
    protected String customerName;
    protected int accaunt Number,
    protected double balance,

    public Account(String customerName, int
    accountNumber, double balance) {
        this.customerName = customerName;
        this.accountNumber = accountNumber;
        this.balana = balance;
    }

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

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

class SavAcct extends Account {
    private double interestRate;

    public SavAcct(String customerName, int accountNum
                        double balance, double interestRate){
            ~~super(customerName, int AccountNumber, double
                  balance, double interestRate){~~
            super(customerName, accountNumber, balance)
            this.interestRate = interestRate;
    }

    public void computeAndDepositInterest(){
          double interest = balance * (interestRate / 100);
          balance += interest;
          System.out.println("Interest added:" +interest);
    }

   public void withdraw(double amount){
         if (amount <= balance){
             balance -= amount;
             System.out.println("Withdrawn:" +amount)
          iF (balance < minimumBalance) {
             balance -= serviceCharge;
             System.out.println("Service charge imposed."
                                    + serviceCharge);
          }
     } else {
          System.out.println("Insufficent balana for
                                    Withdrawal");
     }
   }
}

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

     // Create a Savings accout
     SavAcct savAcc = new SavAcct("Amy ", 12345, 1000, 5)

   // Create a current account
   CurAcct curAcc = new CurAcct("John", 67890, 2000, 500, 5)

System.out.println("Choose Account Type: n1. Savigs
Account \n 2. current Account");
int choice = sc.nextInt();
```



```
    switch( choice) {
        case 1 .
            Symen.out.println("Savings Account S
          savAcc deposit(500),
          savAcc.computeAndDepositInterest(),
         savAcc.withdraw(300),
         savAcc.displayBalance(),
         break,

    case 2.
      System.out.println("Current Account Selected");
     curAcc.deposit( 500),
     curAcc.withdraw(1200);
     curAcc.displayBalance
     break;
  default:
    System.out.println ("Invalid choice");
 }
 sc.close();
}
}
```



```
Output :
Choose Account Type;
1. Savings ~~Count~~ Account
2 Current Account
1
Savings Account Selected
Deposited : 500.0
Interest added: 75.0
Withdrawn : 300.0
Balance: 1275.0
```

Program:

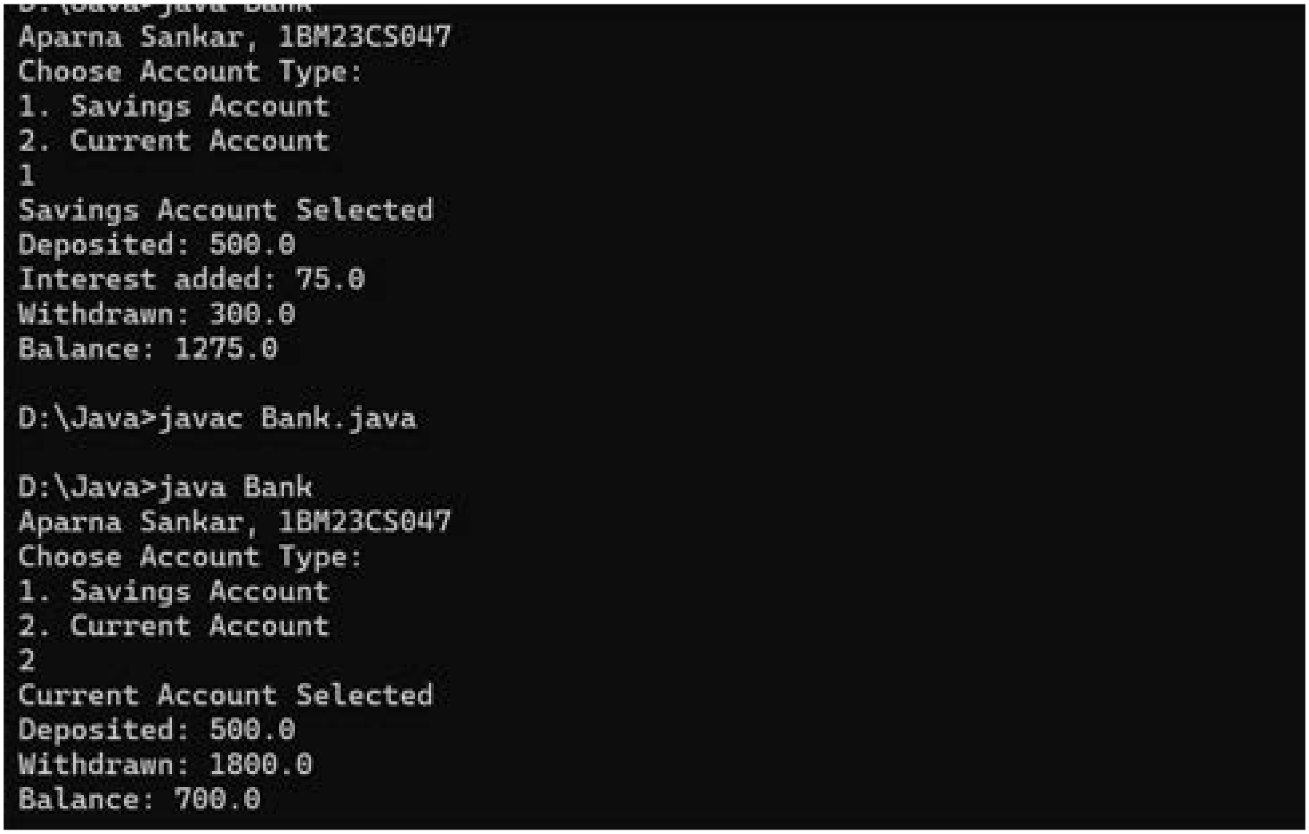
```
import java.util.Scanner;

class Account {
        protected String customerName;
        protected int accountNumber;
        protected double balance;
```

```
    public Account(String customerName, int accountNumber, double balance) {
    this.customerName = customerName;
    this.accountNumber = accountNumber;
    this.balance = balance;
    }

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

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

```
class SavAcct extends Account {
    private double interestRate;

    public SavAcct(String customerName, int accountNumber, double balance, double interestRate) {
    super(customerName, accountNumber, balance);
    this.interestRate = interestRate;
    }

    public void computeAndDepositInterest() {
    double interest = balance * (interestRate / 100);
    balance += interest;
    System.out.println("Interest added: " + interest);
    }

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

class CurAcct extends Account {
    private double minimumBalance;
    private double serviceCharge;

    public CurAcct(String customerName, int accountNumber, double balance, double minimumBalance, double serviceCharge) {
    super(customerName, accountNumber, balance);
    this.minimumBalance = minimumBalance;
    this.serviceCharge = serviceCharge;
    }

    public void withdraw(double amount) {
    if (amount <= balance) {
    balance -= amount;
    System.out.println("Withdrawn: " + amount);
```

```
        if (balance < minimumBalance) {
                balance -= serviceCharge;
               System.out.println("Service charge imposed: " +
serviceCharge);
        }
        } else {
           System.out.println("Insufficient balance for withdrawal");
        }
        }
}

public class Bank {
        public static void main(String[] args) {
        System.out.println("Aparna Sankar, 1BM23CS047");
        Scanner sc = new Scanner(System.in);

        // Create a savings account
        SavAcct savAcc = new SavAcct("Alice", 12345, 1000, 5);

        // Create a current account
        CurAcct curAcc = new CurAcct("Bob", 67890, 2000, 500, 50);
           System.out.println("Choose Account Type:\n1. Savings Account\n2. Current Account");
        int choice = sc.nextInt();


        switch (choice) {
        case 1:
               System.out.println("Savings Account Selected");
                savAcc.deposit(500);
               savAcc.computeAndDepositInterest();
                savAcc.withdraw(300);
             savAcc.displayBalance();
                break;


        case 2:
               System.out.println("Current Account Selected");
                curAcc.deposit(500);
                curAcc.withdraw(1800);
                curAcc.displayBalance();
                break;
        default:
            System.out.println("Invalid choice");
        }


        sc.close();
        }
}
```

Output:



```
D:\Java>java Bank
Aparna Sankar, 1BM23CS047
Choose Account Type:
1. Savings Account
2. Current Account
1
Savings Account Selected
Deposited: 500.0
Interest added: 75.0
Withdrawn: 300.0
Balance: 1275.0

D:\Java>javac Bank.java

D:\Java>java Bank
Aparna Sankar, 1BM23CS047
Choose Account Type:
1. Savings Account
2. Current Account
2
Current Account Selected
Deposited: 500.0
Withdrawn: 1800.0
Balance: 700.0
```

Program 6
SEE marks

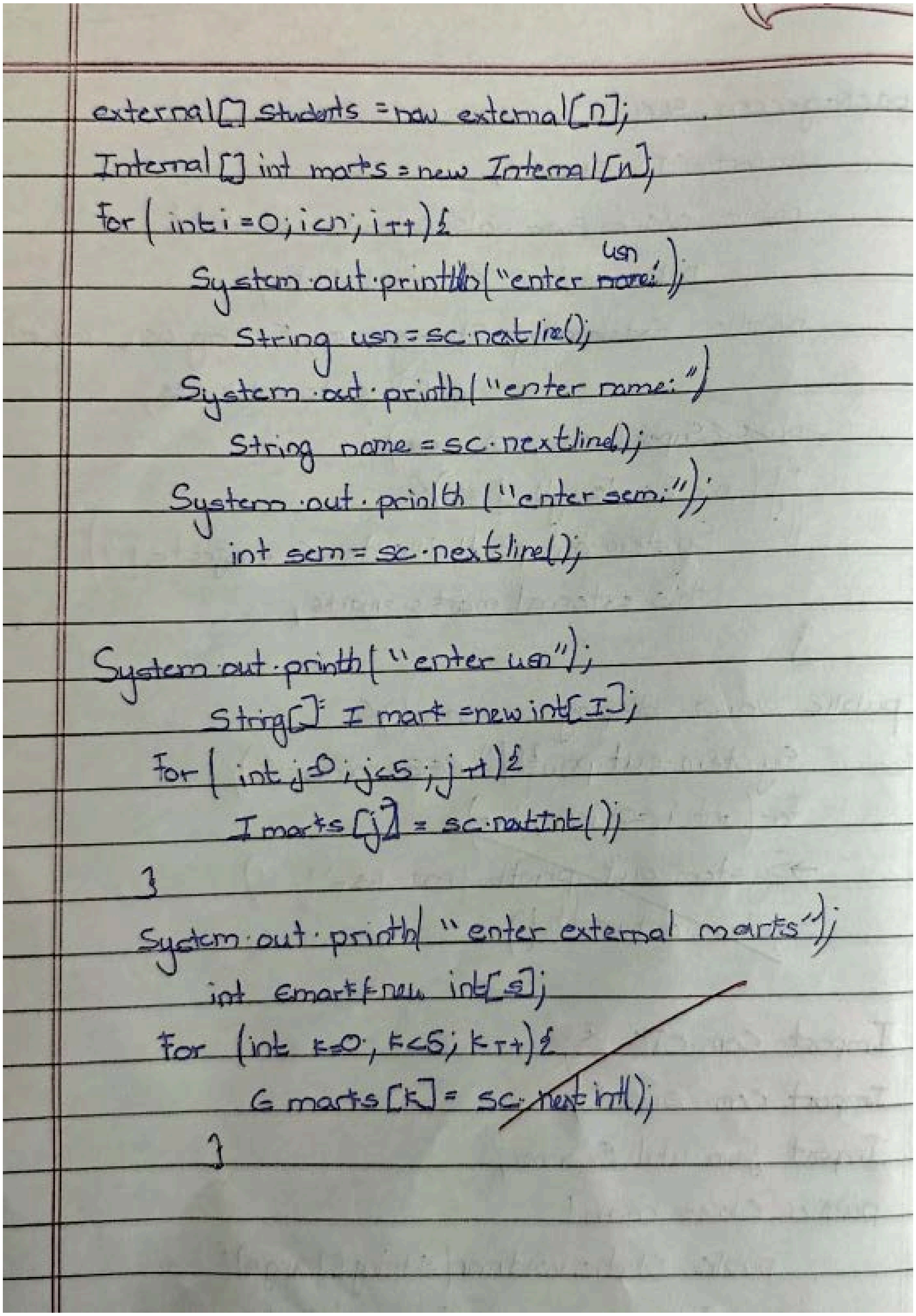
Algorithm:

LAB-6

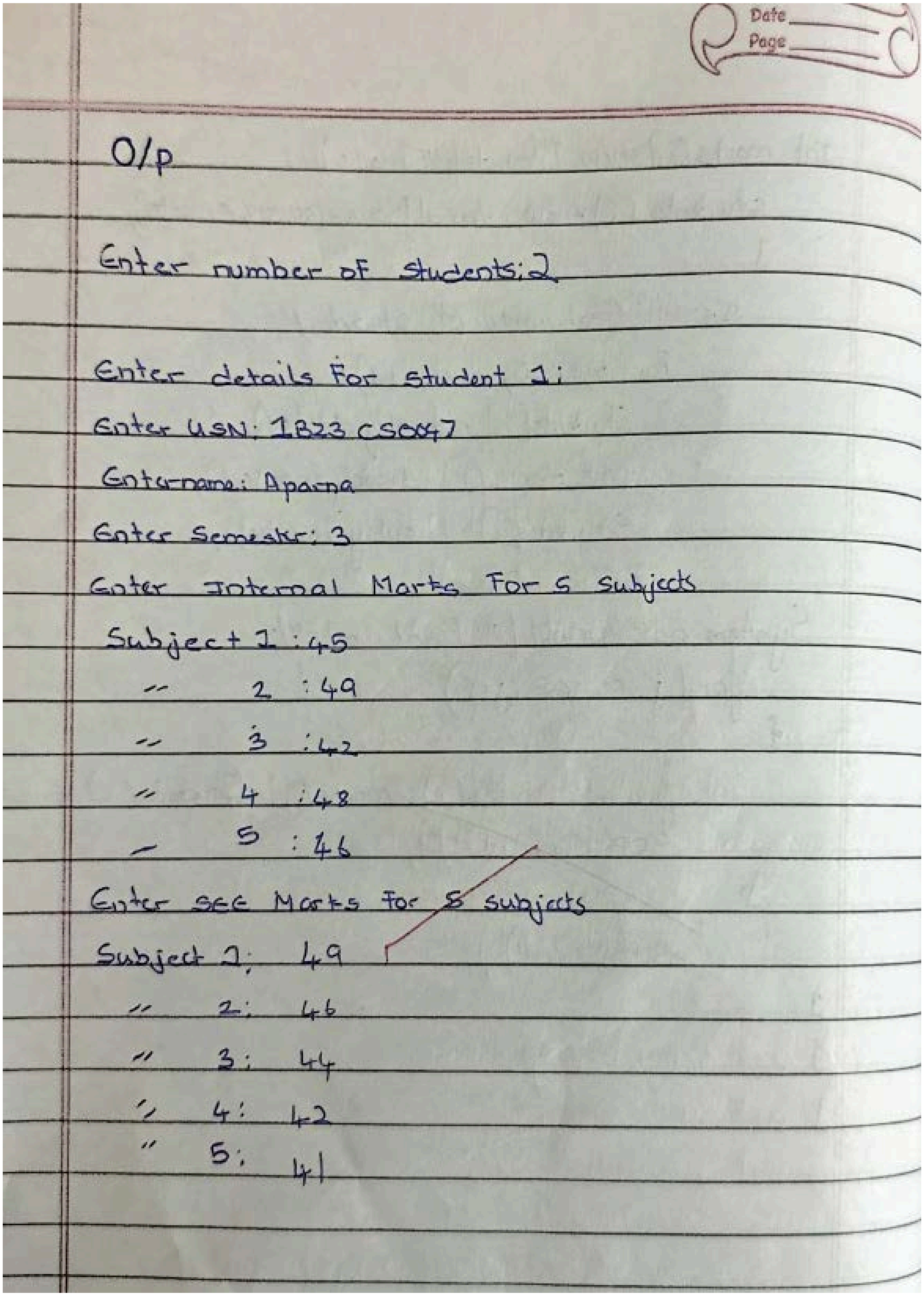
Q) Create a package CIE which has two classes - Student and Internals. The class student has members like usn, name, sem. The class internals derived from Student has an array that stores the internal marks scored in Five courses of the current semester of the student. Create another package sec which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

```
package com 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;
~~package.com.CIE~~
            this.sem = sem
        }

    public void displayDetails(){
            System.out.println("USN:" +usn),
            System.out.println("Name:" +name);
            System.out.println("Sem:" +sem);
    }

    public class Internals {
            public int[] internalMarks;
            public Internals(int[]marks){
                if(marks.length != 5){
                    System.out.println("Enter 5 marks")
                }
                this.internalMarks=marks,
            }
    public void display I Marks(){
          System.out.println("Internal marks:")
           for(int i=0; i<5; i++)
           {
                System.out.println(mark+" ")
           }
           System.out.println();
          }
    }
```

```
packagecom. SEE;
        import CIE. Student;
        public class Externals extends student{
                public int[] External Marks;
        public Externals( String name, String usn, int sem,
                                int[] External Marks;
        ~~public~~Super( name, usn, sem);
        If( marks. length != 5)<
                System.out.println("enter 5 subjects!")
                this. external marks =marks;
        }
public void display E marks ()<
        System.out.println(" SEE marks: ");
        For( int i =0; i<5 ; i++)
        →System.out.println (marks+" ")
        System.out.println()

Import com.CIE. *;
Import com . SEE *;
Import java.util. Scanner;
public Class main{
        public static void main( string[] args){
                Scanner sc = new Scanner( System.in);
System.out.println(" enter no of Students:")
        int n=Scanner Int();
```



```
external[] students =new external[n];
Internal[] int marks =new Internal[n];
for( int i=0;i<n;i++){
        System.out.println("enter ~~name:~~ usn");
          String usn=sc.nextline();
       System.out.println("enter name:")
          String name=sc.nextline();
       System.out.prinlth("enter sem:");
          int sem=sc.nextline();

System.out.println("enter usn");
        String[] I mark =new int[I];
    for( int j=0;j<5;j++){
          Imarks[j] = sc.nextInt();
  }
    System.out.println("enter external marks");
        int Emark=new int[5];
    for (int k=0, k<5;k++){
          E marks[k]= sc.nextint();
        }
Int marks[i]=new Internals( I marks);
      students [i]=new external (name, usn, sem, Emarks];
      }

        s.o.p( "final marks of student )
           for( int i =0; i<n; i++) {
             Students[i] . display details();
             int marks[i] .display marts ();
             Student [i] . display marks();

System.out.printh( "final marks");
     for ( j =0; j<5; j++)
  {
     int  final= int marks[i].Imarks[j] +[Students[i].Emarks[j]];
          SOP (Final +" ");
   }
          SOP ( "/n");
}
}
}
```

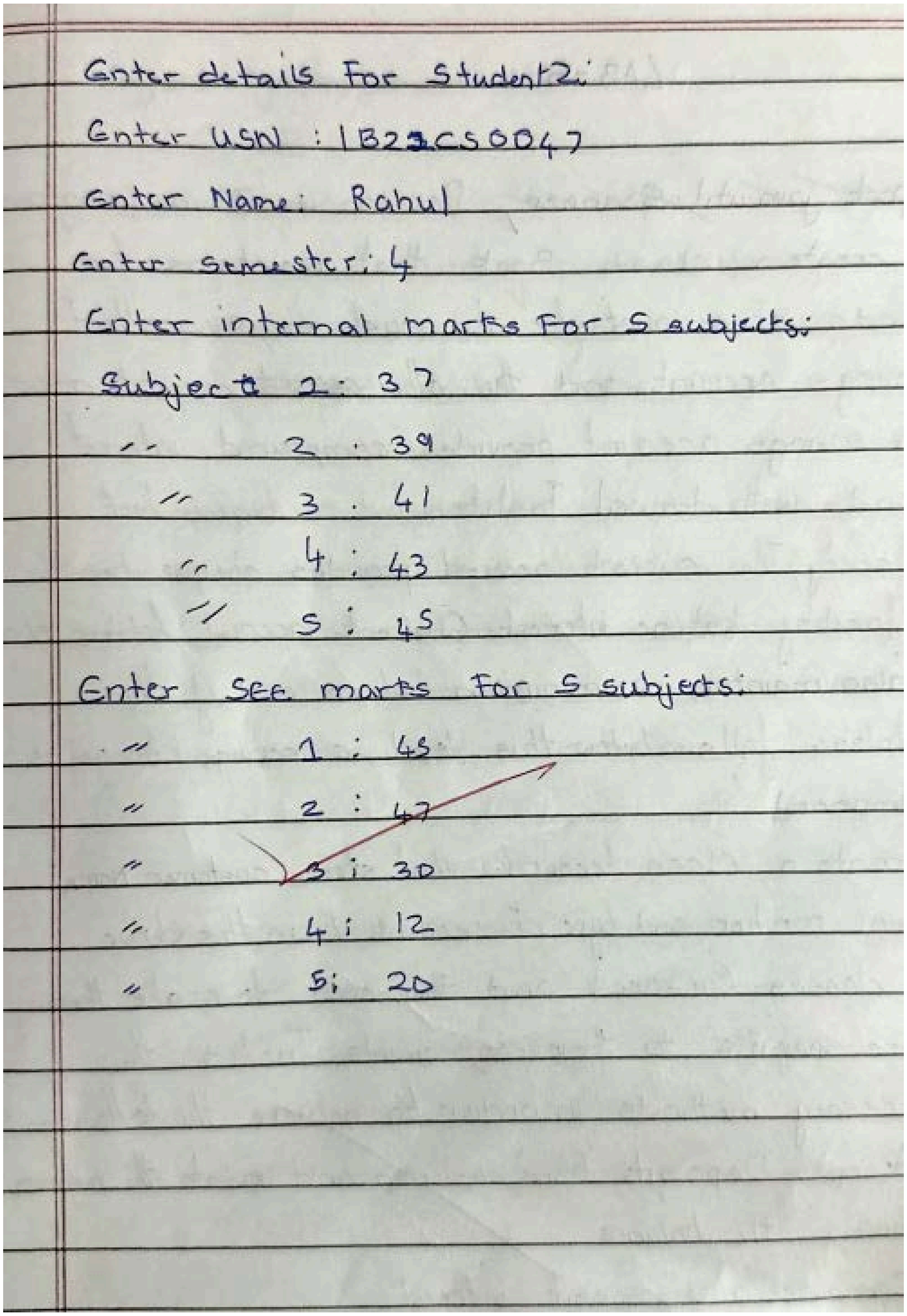


Date

Page

O/p

```
Enter number of students:2

Enter details for student 1:
Enter USN: 1BZ3 CS047
Enter name: Aparna
Enter Semester: 3
Enter Internal Marks for 5 subjects
Subject 1 :45
   "      2 :49
   "      3 :42
   "      4 :48
   "      5 :46
Enter see Marks for 5 subjects
Subject 1:  49
   "    2:  46
   "    3:  44
   "    4:  42
   "    5:  41
```



```
Enter details for student2:
Enter USN : 1B23CS0047
Enter Name:  Rahul
Enter semester: 4
Enter internal marks for 5 subjects:
 Subject 2: 37
   〃      2 : 39
   〃      3 : 41
   〃      4 : 43
   〃      5 : 45
Enter  SEE  marks  for 5 subjects:
  〃       1 : 45.
  〃       2 : 47
  〃       3 : 30
  〃       4 : 12
  〃       5: 20
```

Program:

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

```
class Student {
      public String usn;
      public String name;
      public String sem;

      public Student(String usn, String name, String sem) {
      this.usn = usn;
      this.name = name;
      this.sem = sem;
      }

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

class Internals {
      public int[] internalMarks;
```

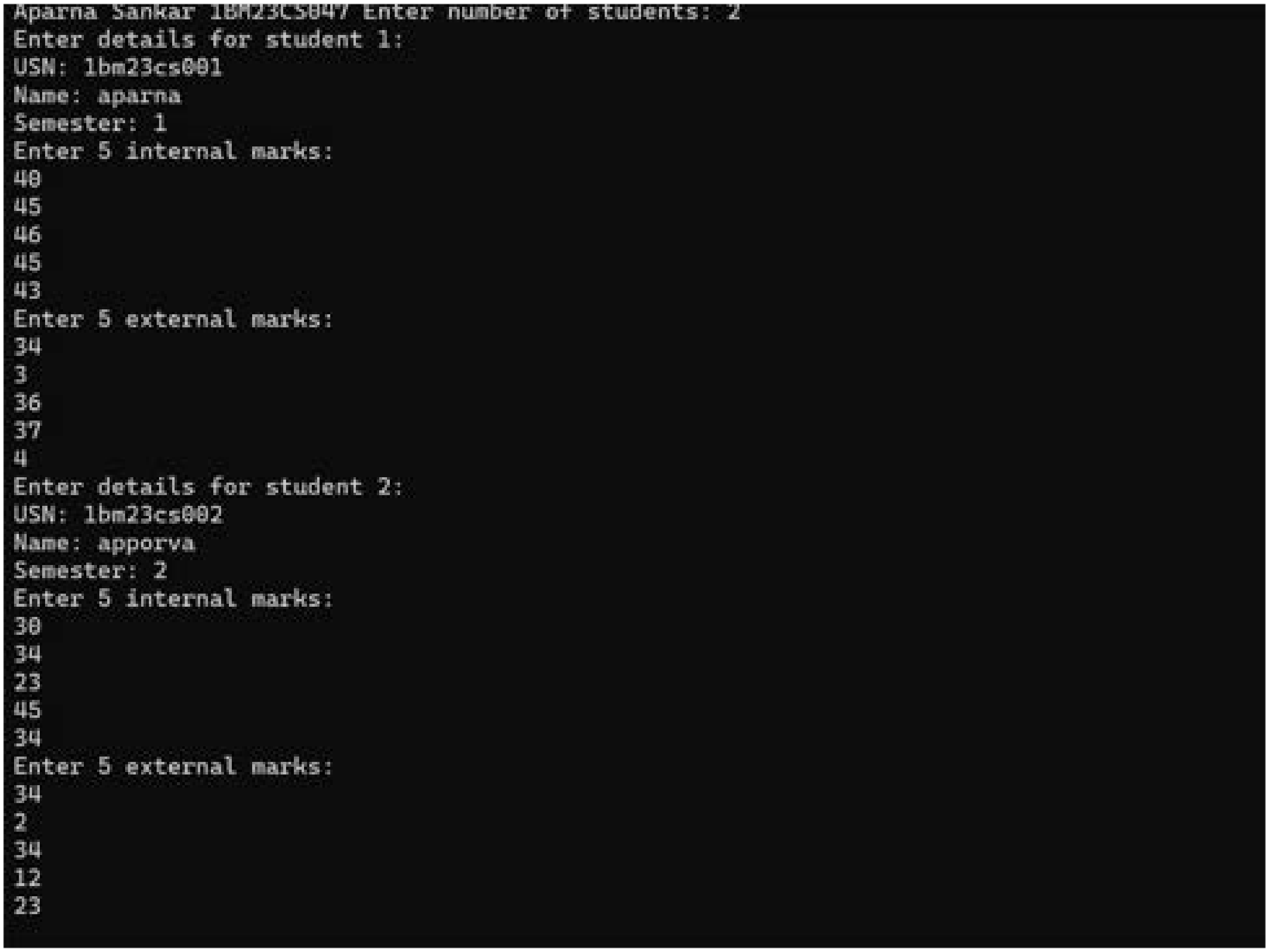
```
        public Internals(int[] marks) {
        if (marks.length != 5) {
          System.out.println("Enter marks for 5 subjects.");
        return;
        }
       this.internalMarks = marks;
        }

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

class Externals extends Student {
        public int[] externalMarks;

        public Externals(String usn, String name, String sem, int[] marks) {
        super(usn, name, sem);
```

```
     if (marks.length != 5) {
       System.out.println("Enter marks for 5 subjects.");
     return;
     }
    this.externalMarks = marks;
     }

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

public class Main {
     public static void main(String[] args) {
    System.out.print("Aparna Sankar 1BM23CS047 ");
     Scanner sc = new Scanner(System.in);

    System.out.print("Enter number of students: ");
```

```
int n = sc.nextInt();
sc.nextLine(); // Consume the newline character

Externals[] students = new Externals[n];
Internals[] internalMarks = new Internals[n];

for (int i = 0; i < n; i++) {
  System.out.println("Enter details for student " + (i + 1) + ":");
System.out.print("USN: ");
String usn = sc.nextLine();

  System.out.print("Name: ");
String name = sc.nextLine();

  System.out.print("Semester: ");
String sem = sc.nextLine();

System.out.println("Enter 5 internal marks:");
int[] iMarks = new int[5];
for (int j = 0; j < 5; j++) {
     iMarks[j] = sc.nextInt();
}
```

```
          System.out.println("Enter 5 external marks:");
        int[] eMarks = new int[5];
        for (int j = 0; j < 5; j++) {
              eMarks[j] = sc.nextInt();
        }
          sc.nextLine(); // Consume the newline character

          internalMarks[i] = new Internals(iMarks);
        students[i] = new Externals(usn, name, sem, eMarks);
        }

       System.out.println("\nFinal Marks of Students:");
        for (int i = 0; i < n; i++) {
          students[i].displayDetails();
          internalMarks[i].displayMarks();
        students[i].displayMarks();

          System.out.print("Final Marks: ");
        for (int j = 0; j < 5; j++) {
                 int finalMarks = internalMarks[i].internalMarks[j] +
students[i].externalMarks[j];
              System.out.print(finalMarks + " ");
        }
          System.out.println("\n");
        }


        sc.close();
        }
}
```

Output:



```
Aparna Sankar 1BM23CS047 Enter number of students: 2
Enter details for student 1:
USN: 1bm23cs001
Name: aparna
Semester: 1
Enter 5 internal marks:
40
45
46
45
43
Enter 5 external marks:
34
3
36
37
4
Enter details for student 2:
USN: 1bm23cs002
Name: apporva
Semester: 2
Enter 5 internal marks:
30
34
23
45
34
Enter 5 external marks:
34
2
34
12
23
```

Program 7
father-son age

Algorithm:

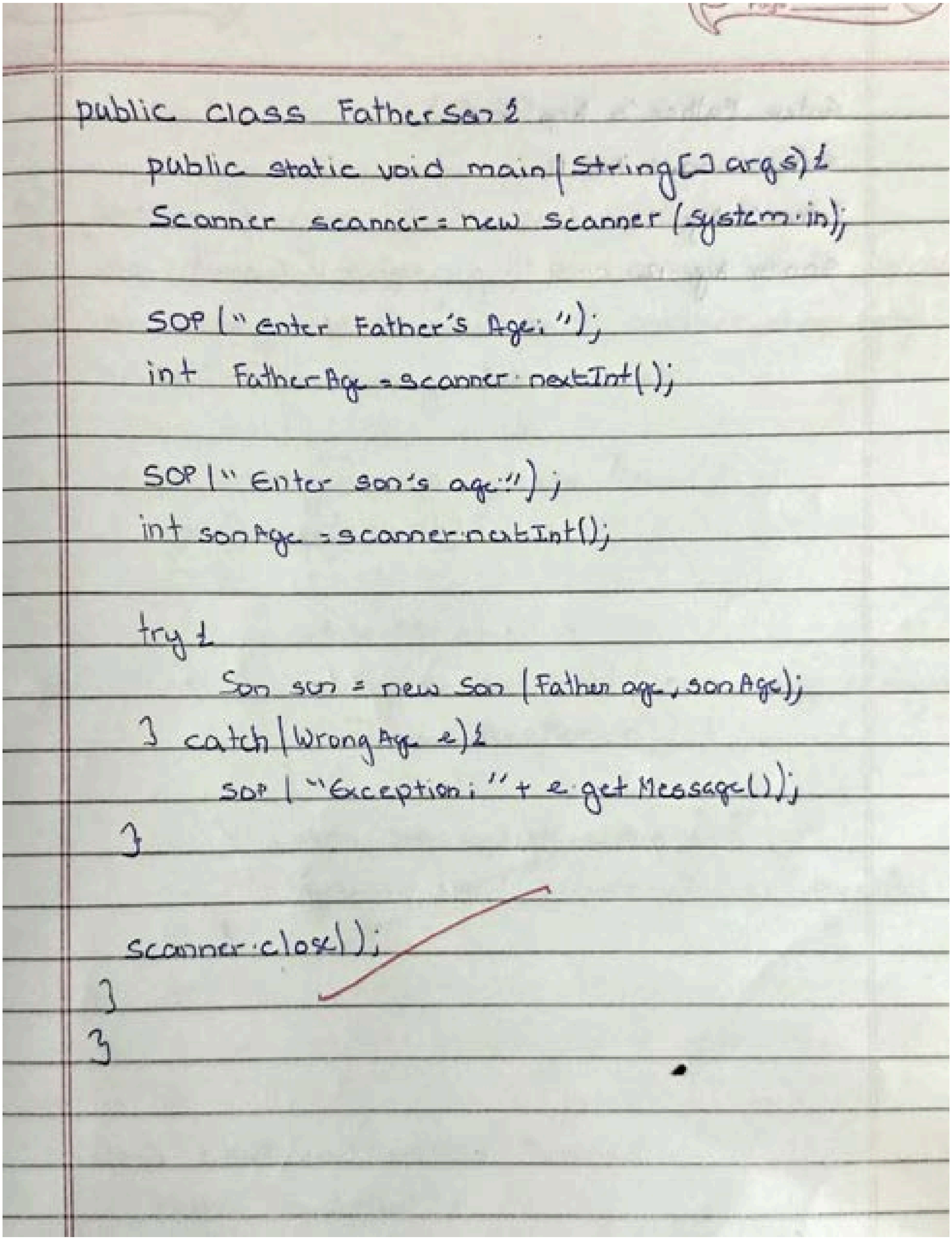
LAB - 7

Q) WAP that demonstrates exception handling in inheritance stream. Create a base class called "Fathe" and derive class called as "son" which extent the base class. in Father class implement a constructor which

→ import java.util.Scanner

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

class Father {
    int age;'

    public Father(int age) throws WrongAge {
      if (age<0) {
          throw new WrongAge("Age cannot be negative");
      }
```

```
        this.age = age;
        SOP(" Father's Age: "+ this.age);
    }
}


class Son extends Father {
    int sonAge;


    public son(int FatherAge, int sonAge throws WrorngAge
        super(FatherAge);
        iF(sonAec <0){
            throw new WrongAge("son's age cannot be
                                        negative");
        }
        if (sonAge >= FatherAge) {
            thrownew WrongAge(" Son's Age Cannot
                                        be Greater or Equal to
                                        Father's age");
        }
        this.sonAge = sonAge;
        SOP("son's age:" + this.sonAge))
    }
}
```



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

    SOP("enter Father's Age: ");
    int FatherAge = scanner.nextInt();

    SOP(" Enter son's age:");
    int sonAge = scanner.nextInt();

    try{
        Son son = new Son(Father age, sonAge);
    } catch(WrongAge e){
        SOP("Exception: " + e.getMessage());
    }

    scanner.close();
  }
}
```

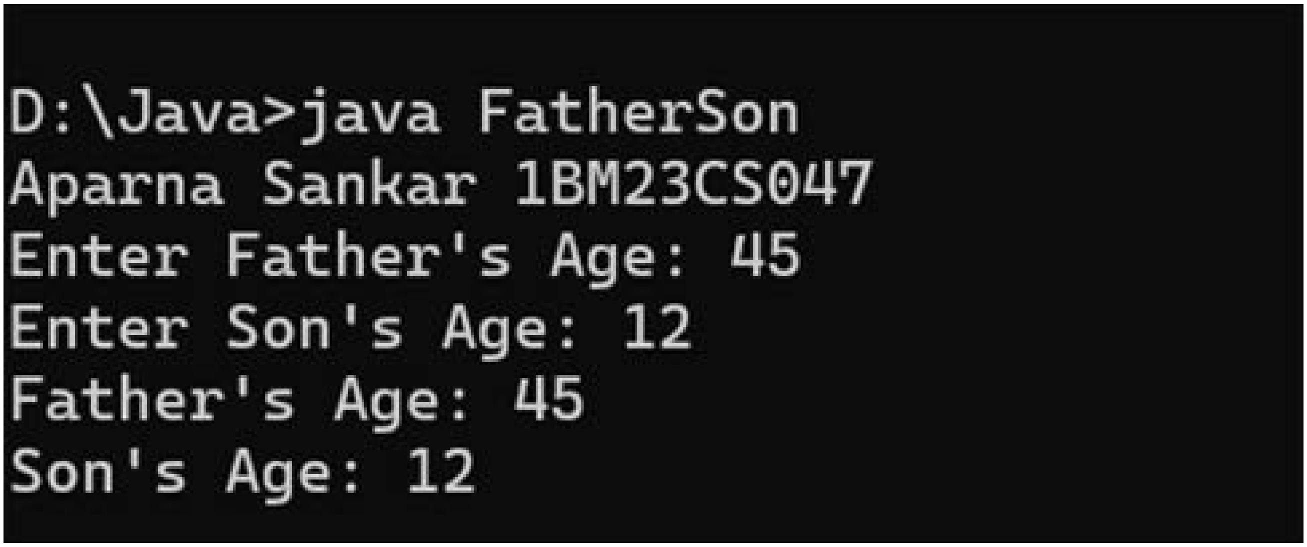
Program:

```
import java.util.Scanner;

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

class Father {
       int age;

       public Father(int age) throws WrongAge {
       if (age < 0) {
       throw new WrongAge("Age Cannot be Negative");
       }
       this.age = age;
     System.out.println("Father's Age: " + this.age);
       }
}

class Son extends Father {
       int sonAge;
```

```
        public Son(int fatherAge, int sonAge) throws WrongAge {
      super(fatherAge);
        if (sonAge < 0) {
        throw new WrongAge("Son's Age Cannot be Negative");
        }
        if (sonAge >= fatherAge) {
        throw new WrongAge("Son's Age Cannot be Greater than or Equal to Father's Age");
        }
        this.sonAge = sonAge;
      System.out.println("Son's Age: " + this.sonAge);
        }
}

public class FatherSon {
        public static void main(String[] args) {
      System.out.println("Aparna Sankar 1BM23CS047");
        Scanner scanner = new Scanner(System.in);

      System.out.print("Enter Father's Age: ");
        int fatherAge = scanner.nextInt();

      System.out.print("Enter Son's Age: ");
        int sonAge = scanner.nextInt();
```

```
            try {
            Son son = new Son(fatherAge, sonAge);
            } catch (WrongAge e){
             System.out.println("Exception: " + e.getMessage());
            }


        scanner.close();
            }
}
```

Output:



```
D:\Java>java FatherSon
Aparna Sankar 1BM23CS047
Enter Father's Age: 45
Enter Son's Age: 12
Father's Age: 45
Son's Age: 12
```

Program 8
Threads

Algorithm:

LAB - 8

Q) Write a program which creates twothreads, one thread displaying "BMS college of Engineering' once every ten seconds and another displaying "CSE" once every two seconds

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


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


public class ThreadExample {
    public static void main(String[] args){
        CollegeThread collegeThread = new CollegeThrea
        CSEThread cseThread = new cseThread();

        collegeThread.start();
        cseThread.start();
    }
}
```

O/p

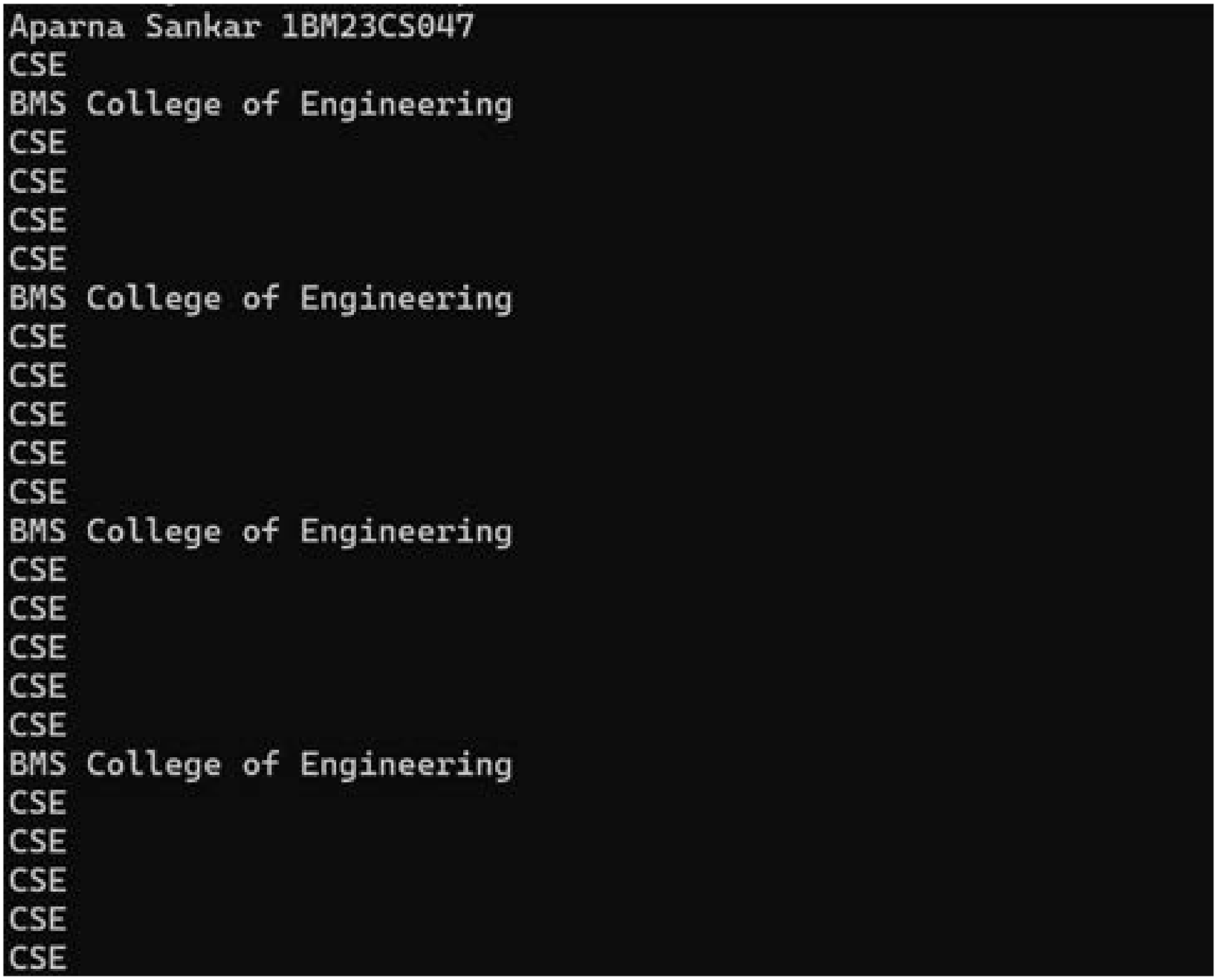
```
BMS college of Engineering
CSE
CSE
```

```
        }
```

program:

```
class CollegeThread extends Thread {
        public void run() {
        try {
        for (int i = 0; i < 5; i++) {
            System.out.println("BMS College of Engineering");
            Thread.sleep(10000);
        }
        } catch (InterruptedException e) {
         System.out.println("CollegeThread interrupted.");
        }
        }
}

class CSEThread extends Thread {
        public void run() {
        try {
        for (int i = 0; i < 25; i++) {
                System.out.println("CSE");
            Thread.sleep(2000);
        }
        } catch (InterruptedException e) {
         System.out.println("CSEThread interrupted.");
        }
        }
}

public class ThreadExample {
        public static void main(String[] args) {
      System.out.println("Aparna Sankar 1BM23CS047");
        CollegeThread collegeThread = new CollegeThread();
        CSEThread cseThread = new CSEThread();

      collegeThread.start();
      cseThread.start();
        }
}
```

Output:

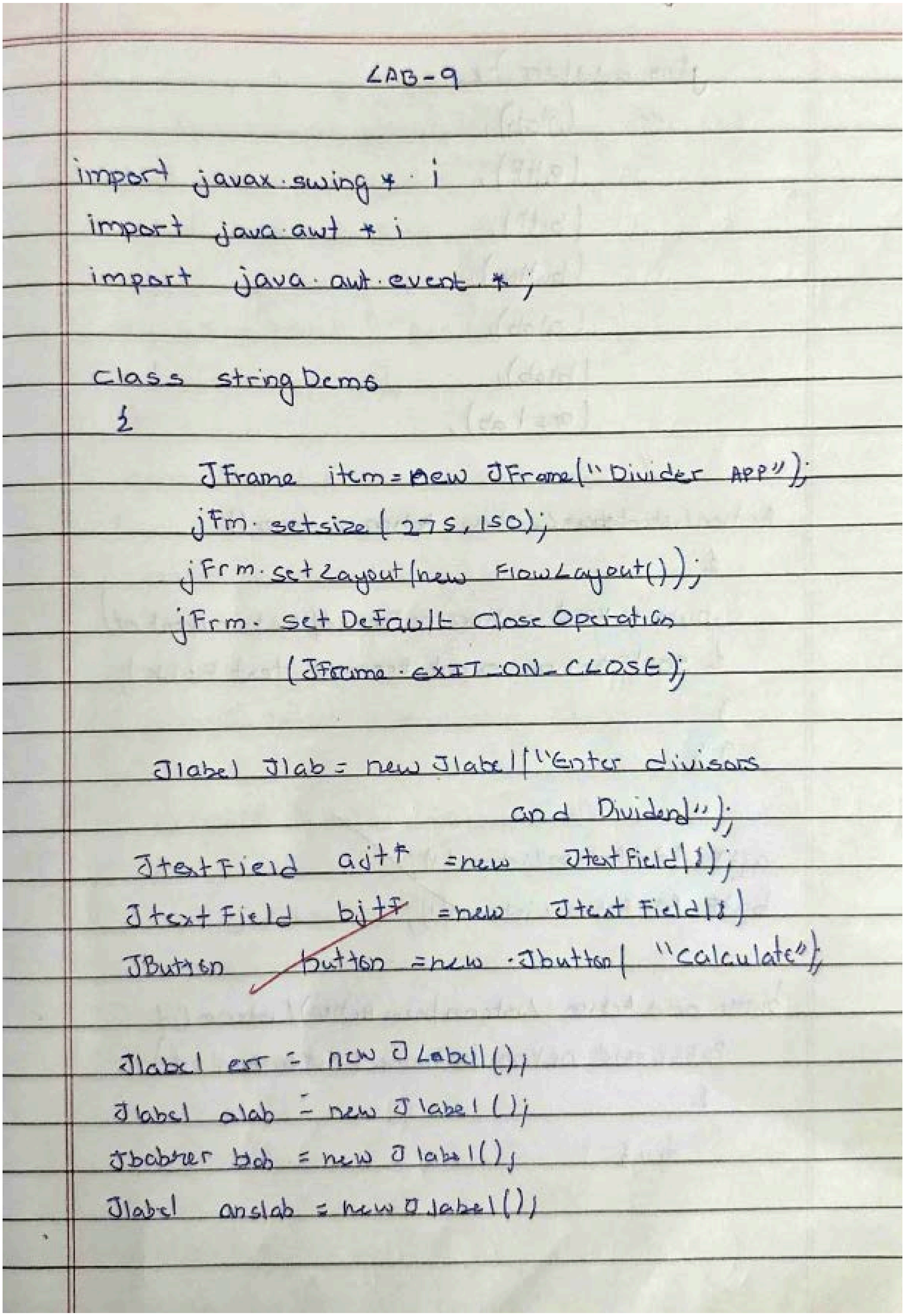


```
Aparna Sankar 1BM23CS047
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
```

Program 9

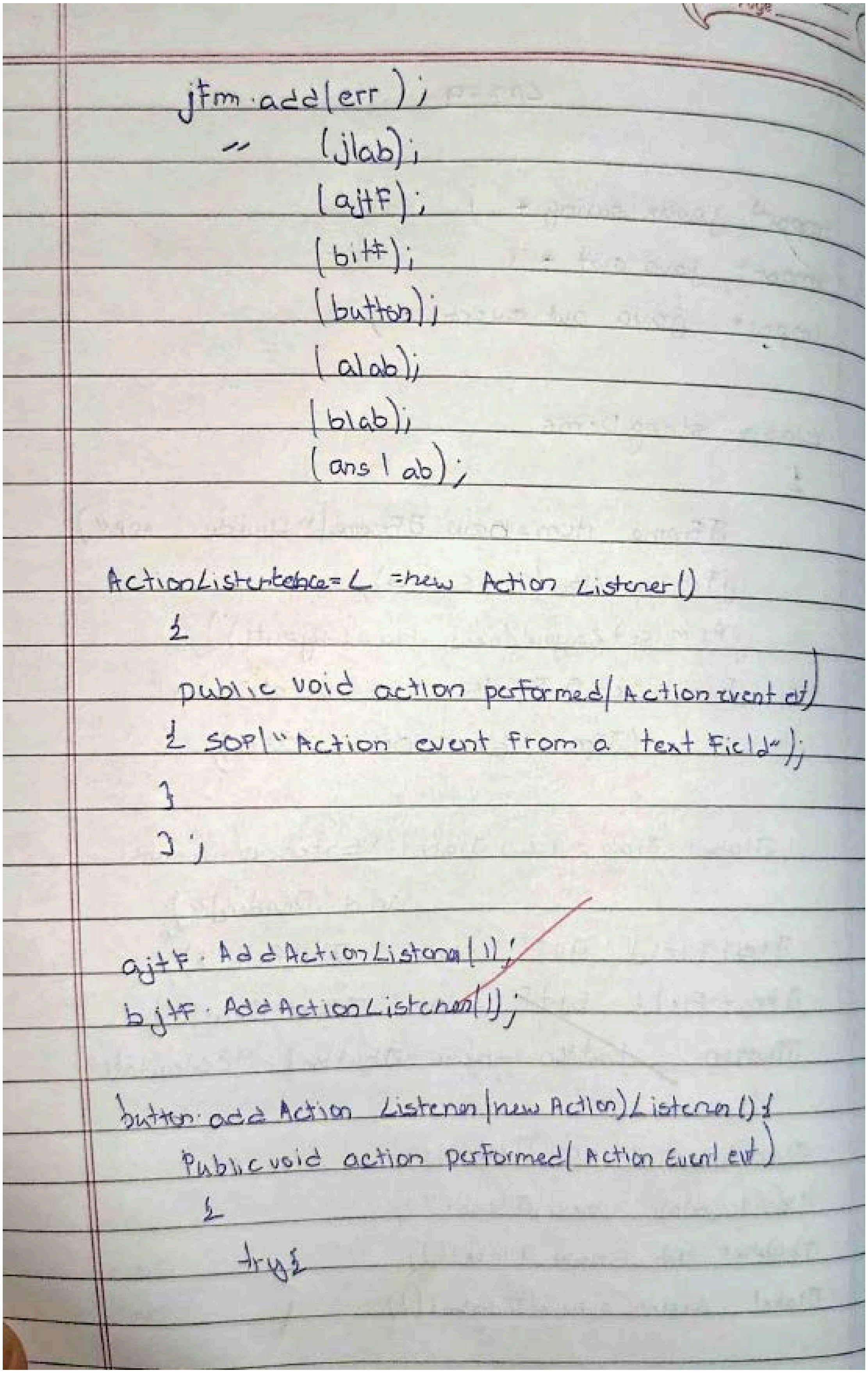
user interface

Algorithm:

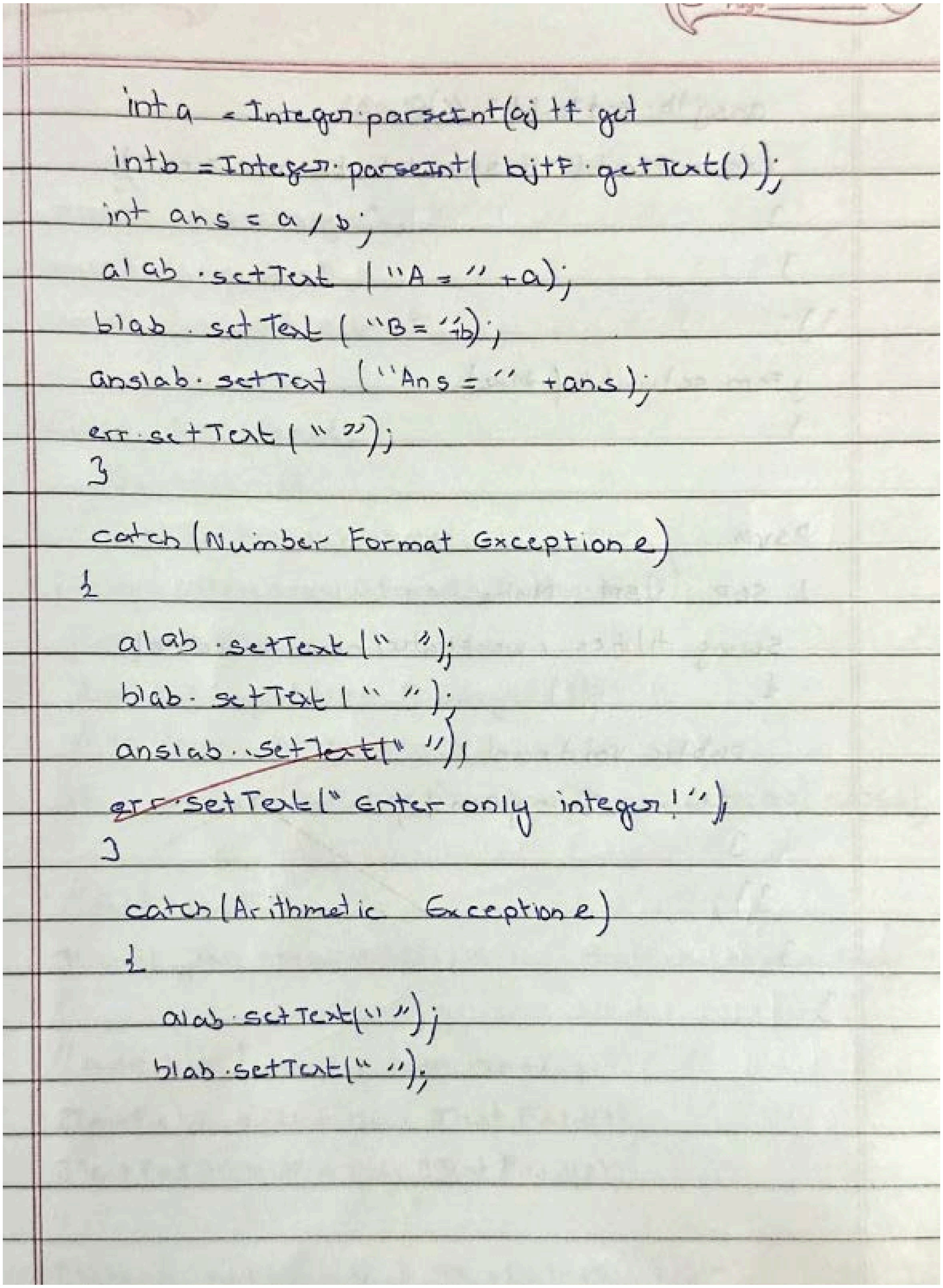


LAB-9

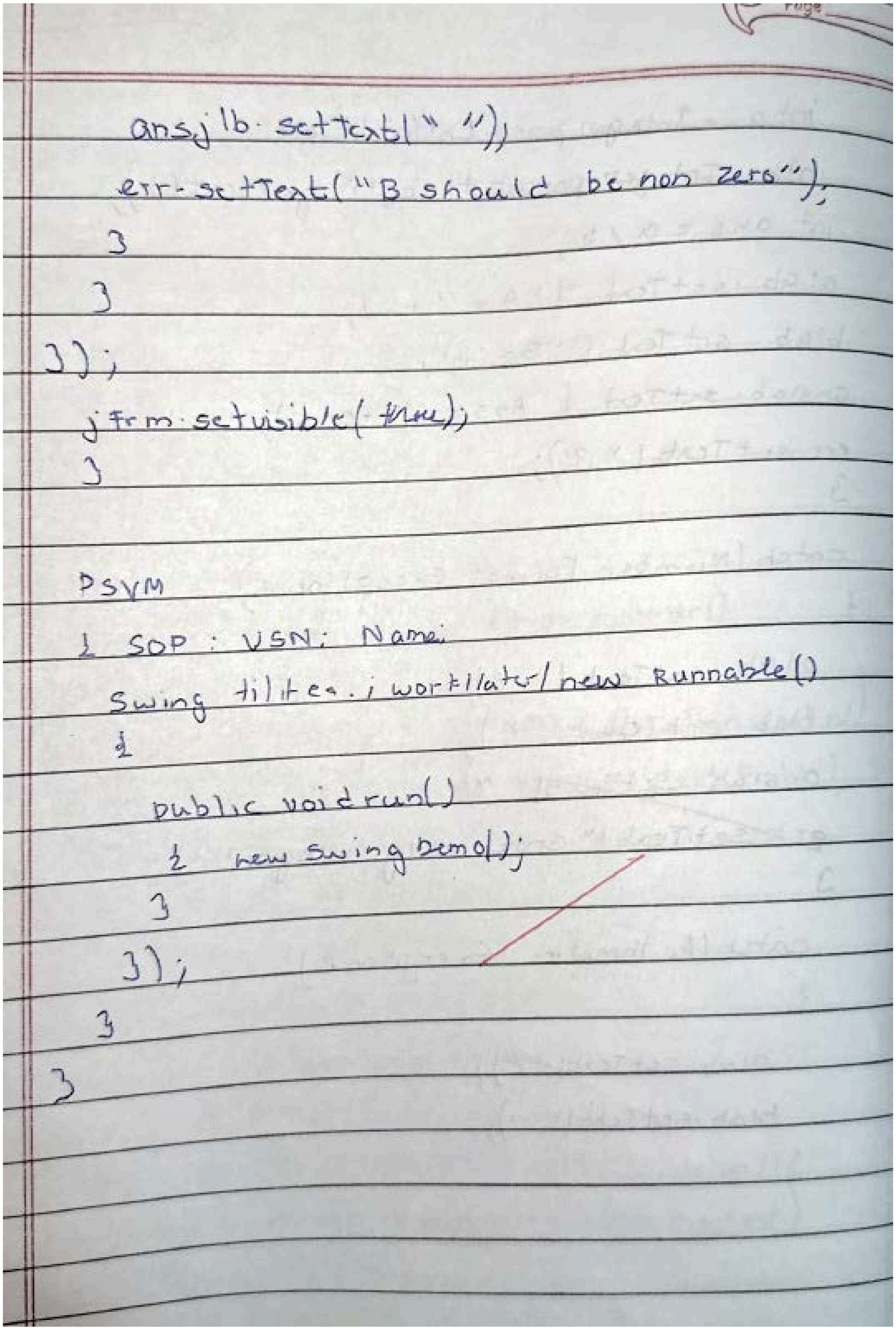
```
import javax.swing *. ;
import java.awt *;
import java.awt.event.*;

class stringDemo
{
        JFrame item=new JFrame("Divider APP");
        jFm.setsize(275,150);
       jFrm.setLayout(new FlowLayout());
      jFrm. setDefault Close Operation
              (JFrame.EXIT_ON_CLOSE);

    Jlabel Jlab = new Jlabel("Enter divisors
                              and Dividend");
   JtextField ajtf =new JtextField(8);
   Jtext Field bjtf =new Jtext Field(8)
   JButton button =new Jbutton( "calculate");

   Jlabel err = new JLabell();
   Jlabel alab = new Jlabel();
   Jbabrer blab = new Jlabel();
   Jlabel anslab = new Jlabel();
```



```
jfm.add(err);
 "       (jlab);
         (ajtF);
         (bjtF);
         (button);
         (alab);
         (blab);
         (ans lab);

ActionListenterface=L =new Action Listener()
    {
    public void action performed(Action event evt)
    { SOP("Action event from a text Field");
    }
    };

ajtF.AddActionListener(l);
bjtF.AddActionListener(l);

button.add Action Listener(new Action)Listener(){
    Public void action performed(Action Event evt)
     {
       try{
```



```
  int a = Integer.parseInt(ajtf.get
  intb = Integer.parseInt( bjtF.getText());
 int ans = a / b;
 alab.setText ("A = " +a);
 blab. setText ("B = "+b);
 anslab.setText ("Ans = " +ans);
 err.setText(" ");
 }
 catch(Number Format Exception e)
{
  alab.setText(" ");
  blab.setText(" ");
  anslab.setText(" ");
  err.setText("Enter only integer!");
 }
  catch(Arithmetic Exception e)
  {
     alab.setText(" ");
    blab.setText(" ");
```



```
    ansjlb.setText(" ");
    err.setText("B should be non zero");
    }
   }
});
 jfrm.setvisible(true);
 }

 PSVM
 { SOP : VSN: Name
  Swing tilites.i workllater/new Runnable()
  {
     public void run()
     {  new SwingDemo();
     }
   });
 }
}
```

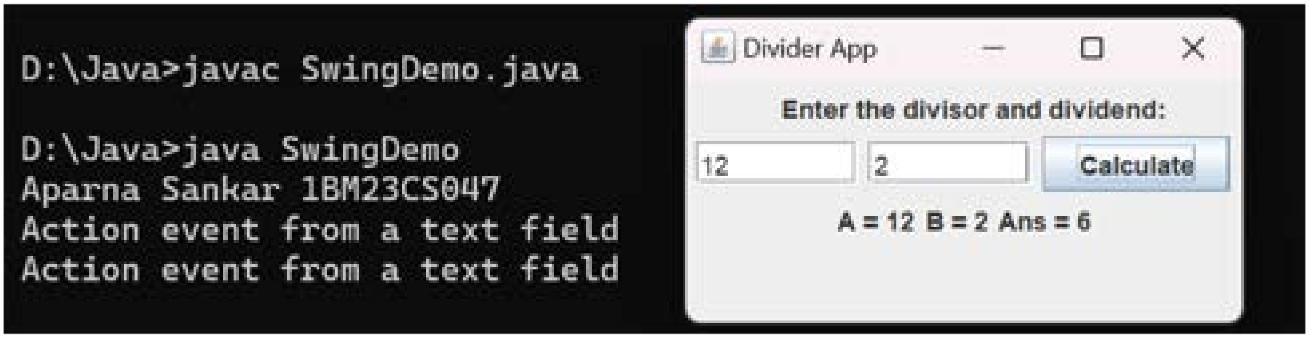
Program:

```
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 divisor and dividend:");


    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("A = " + a);
            blab.setText("B = " + b);
            anslab.setText("Ans = " + ans);
            err.setText("");
            } 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("Aparna Sankar 1BM23CS047");
      SwingUtilities.invokeLater(new Runnable() {
         public void run() {
                 new SwingDemo();
        }
        });
        }
}
```

Output:



```
D:\Java>javac SwingDemo.java

D:\Java>java SwingDemo
Aparna Sankar 1BM23CS047
Action event from a text field
Action event from a text field
```

Enter the divisor and dividend:

12 2

A = 12 B = 2 Ans = 6

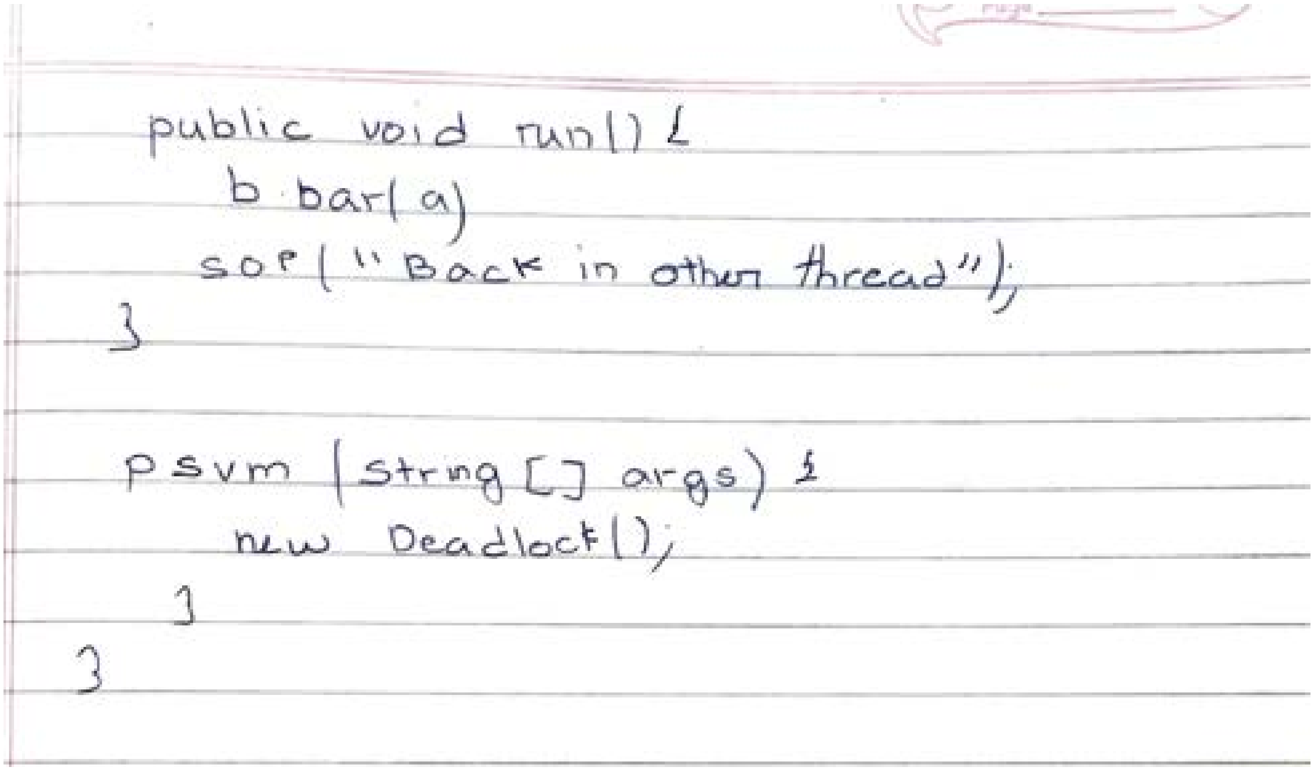
Program 10

interprocess communication and deadlock

Algorithm:

LAB 10

```
class A {
    synchronized void Foo(B b){
        String name = Thread.currentThread().getName()
        SOP(name + " entered A.foo");

        try {
            Thread.sleep(1000);
        } catch(Exception e) {
            SOP("A interrupted");
        }

        SOP(name + " trying to call B.last()");
        b.last();
    }
    synchronized void last(){
        System.out.println("Inside A.last");
    }
}

class B {
    synchronized void bar(A a){
        String name = Thread.currentThread().getName()
        SOP(name + "entered B.bar");
    try {
        Thread.sleep(1000);
      } catch(Exceptione){
          SOP("B interrupted");
    }
    SOP(name + " trying to call A.last()");
    a.last();
  }

  synchronized void last(){
      SOP("Inside B.last");
    }
  }

class Deadlock implements Runnable{
    A a  =new A();
    B b  =new B();

    Deadlock(){
        Thread.currentThread().setName("Main Thread")
        Thread t = new Thread(this, "Racing Thread
        t.start();

        a.Foo(b);
        SOP("Back in main thread"))
```



```
public void run() {
   b.bar(a)
   sop("Back in other thread");
}

psvm (Strng [] args) {
   new Deadlock();
  }
}
```

Program:

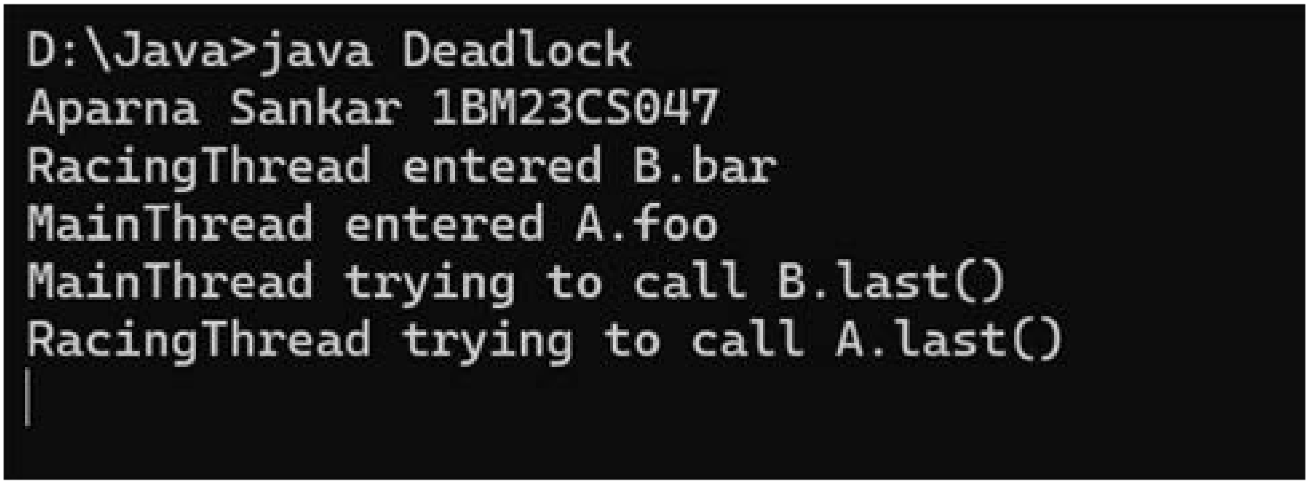
```
class A {
    synchronized void foo(B b) {
    String name = Thread.currentThread().getName();
   System.out.println(name + " entered A.foo");


    try {
     Thread.sleep(1000);
    } catch (Exception e) {
     System.out.println("A Interrupted");
    }


   System.out.println(name + " trying to call B.last()");
```

```
          b.last();
          }

          synchronized 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();
          }

          synchronized void last() {
```

```
       System.out.println("Inside B.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();

         // Get lock on 'a' in this thread
         a.foo(b);
       System.out.println("Back in main thread");
         }

         public void run() {
         // Get lock on 'b' in other thread
         b.bar(a);
       System.out.println("Back in other thread");
         }
```

```
        public static void main(String[] args) {
System.out.println("Aparna Sankar 1BM23CS047");


        new Deadlock();
        }
}
```

Output:



```
D:\Java>java Deadlock
Aparna Sankar 1BM23CS047
RacingThread entered B.bar
MainThread entered A.foo
MainThread trying to call B.last()
RacingThread trying to call A.last()
```