VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted by

Annamsetti Saran Tej (1BM23CS040)

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 **Annamsetti Saran Tej (1BM23CS040),** 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.

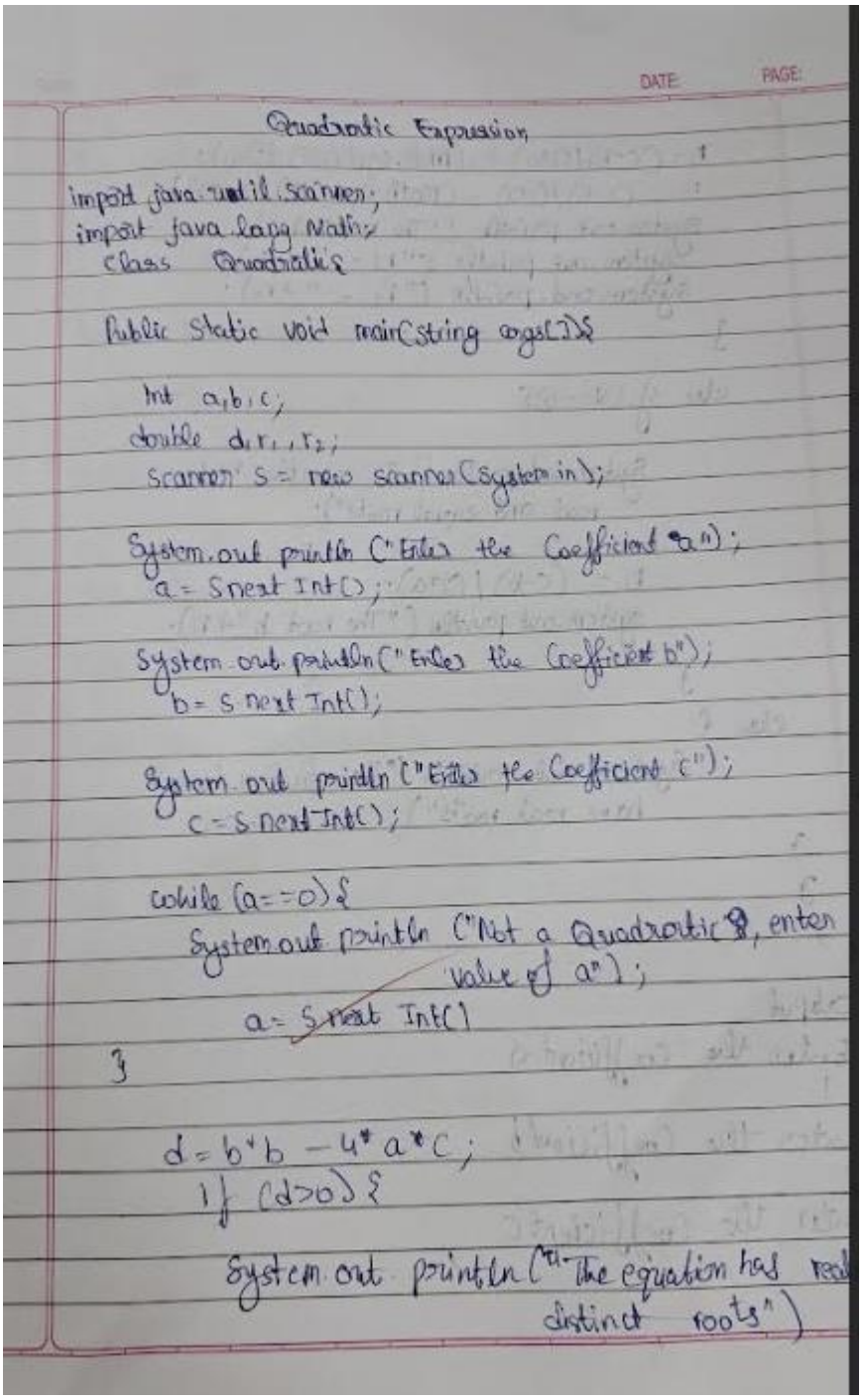
Sheetal V A Assistant Professor Department of CSE, BMSCE	Dr. Jyothi S Nayak Professor & HOD Department of CSE, BMSCE

Index

Github Link:
https://github.com/Sarantej555/java-lab

Program 1
Implement Quadratic Equation

Algorithm



```
r1 = (-b)/(2*a) + (Math.sqrt(d))/(2*a);
r2 = (-b)/(2*a) - (Math.sqrt(d))/(2*a);
System.out.println("The roots are");
System.out.println("r1 = " + r1);
System.out.println("r2 = " + r2);
}
else if(d==0){
System.out.println("The equation has
real and equal roots");
r1 = (-b)/(2*a);
System.out.println("The root b" + r1);
}
else {
System.out.println("The equation doesn't
have real roots");
}
}
}
```

Output

Enter the Coefficient a
1
Enter the Coefficient b
0
Enter the Coefficient c
-4
the equation has real and distinct roots
the roots are
r1 = 2.0
r2 = -2.0

Enter the Coefficient a
1
Enter the Coefficient b
2
Enter the Coefficient c
1
The equation has real and equal roots
the root is -1.0

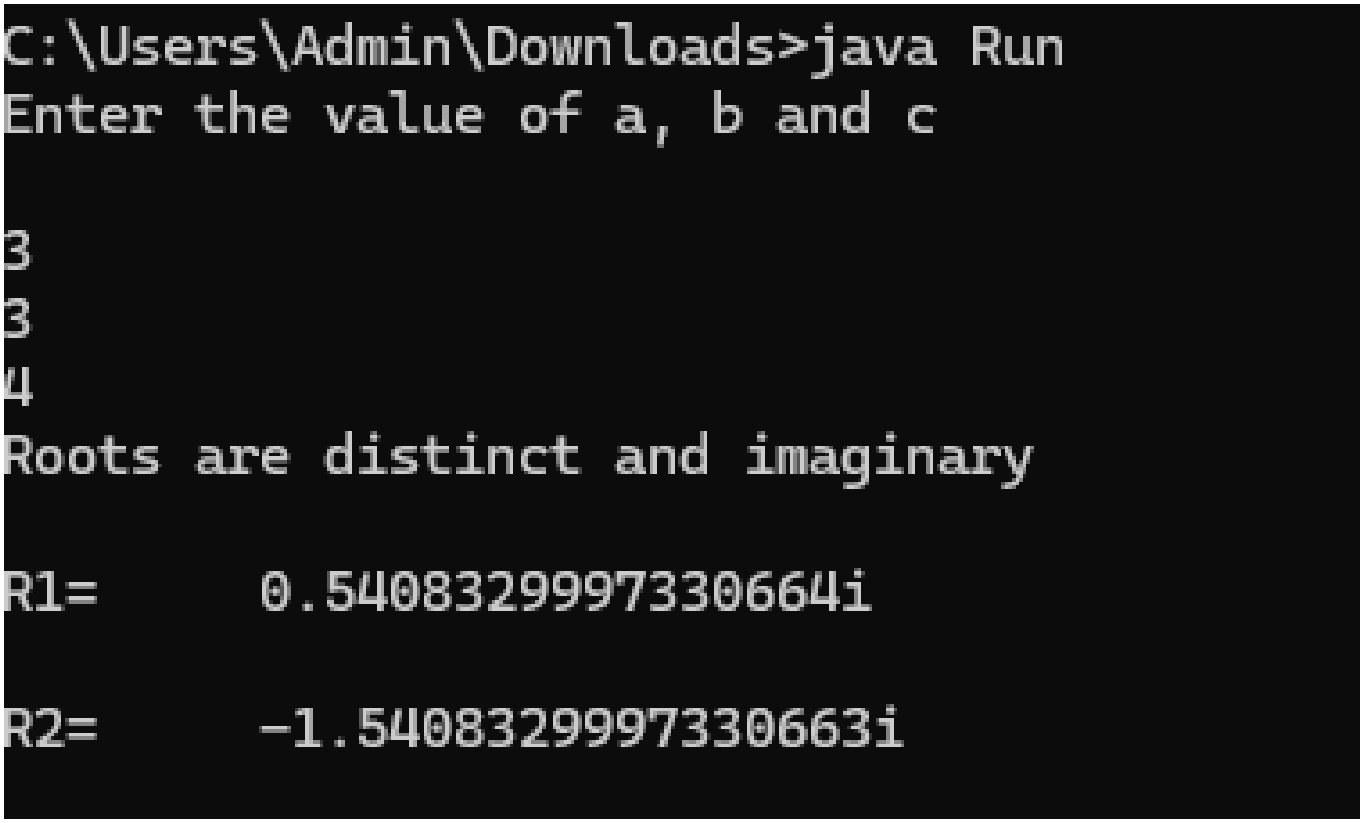
Enter the Coefficient a
1
Enter the Coefficient b
0
Enter the Coefficient c
1
The equation doesn't have real roots

Code:

```
import java.lang.Math;
import java.util.Scanner;
class Quadratic
{
    int a,b,c;
    double r1,r2,d;

    void calculate()
    {
        if(a==0)
        {
            System.out.println("Not a quadratic equation \n");
        }
        else{
            d=(b*b)-(4*a*c);
        }
        if(d>0)
        {
            System.out.println("Roots are real and distinct \n");
            r1=((-b) + (Math.sqrt(d)))/(double)(2*a);
            r2=((-b) - (Math.sqrt(d)))/(double)(2*a);
            System.out.println("R1= \t" + r1 +"\n");
            System.out.println("R2= \t" + r2 +"\n");
        }
        if(d==0)
        {
            System.out.println("Roots are real and equal \n");
            r1=((-b)/(double)(2*a));
            r2=r1;
            System.out.println("R1= \t" + r1 +"\n");
            System.out.println("R2= \t" + r2 +"\n");
        }
        if(d<0)
        {
            System.out.println("Roots are distinct and imaginary \n");
            r1=((-b) + (Math.sqrt(-d)))/(double)(2*a);
            r2=((-b) - (Math.sqrt(-d)))/(double)(2*a);
            System.out.println("R1= \t" + r1 +"i\n");
            System.out.println("R2= \t" + r2 +"i\n");
        }

    }


}


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

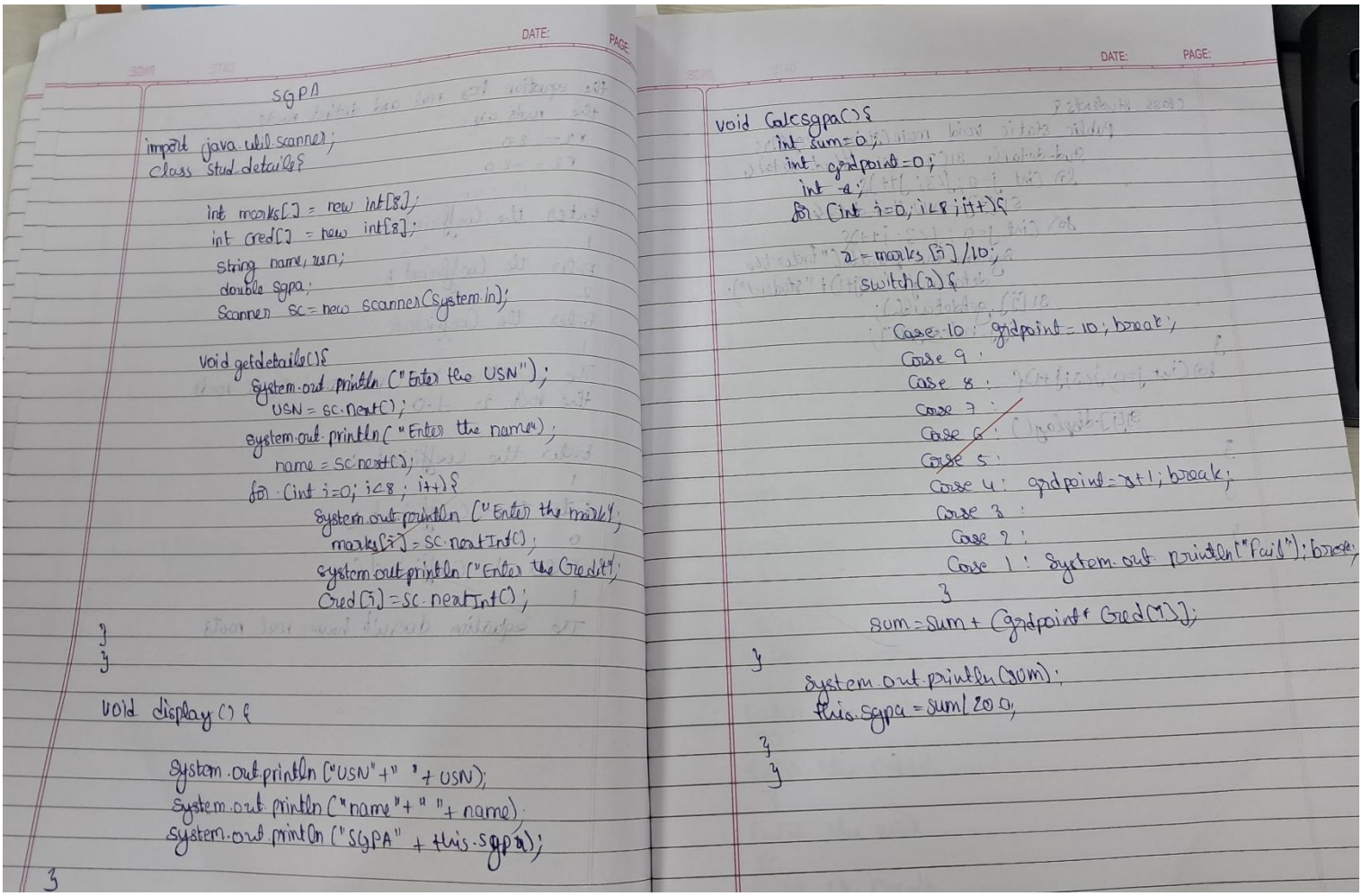
```
        Quadratic Q= new Quadratic();
        System.out.println("Enter the value of a, b and c \n");
        Scanner sc = new Scanner(System.in);
        Q.a=sc.nextInt();
        Q.b=sc.nextInt();
        Q.c=sc.nextInt();
        Q.calculate();
    }
}
```

Output:



Program 2
SGPA Calculator

Algorithm:



```
SGPA

import java.util.scanner;
class Stud_details{

    int marks[] = new int[8];
    int cred[] = new int[8];
    String name, usn;
    double sgpa;
    Scanner sc = new scanner(System.in);

Void getdetails(){
    System.out.println("Enter the USN");
      USN = sc.next();
    system.out.println("Enter the name");
      name = sc.next();
    for(int i=0; i<8 ; i++){
        System.out.println("Enter the marks");
        marks[i] = sc.nextInt();
        system.out.println("Enter the Credit");
        Cred[i] = sc.nextInt();
}
}

void display(){

    System.out.println("USN"+" "+USN);
    System.out.println("name"+" "+name);
    system.out.println("SGPA" + this.sgpa);
}
```

```
void Calcsgpa(){
    int sum=0;
    int gradpoint=0;
    int x;
    for(int i=0; i<8; i++){
        x = marks[i]/10;
        switch(x){
            Case 10: gradpoint=10; break;
            Case 9:
            Case 8:
            Case 7:
            Case 6:
            Case 5:
            Case 4: gradpoint=x+1; break;
            Case 3:
            Case 2:
            Case 1: System.out.println("Fail"); break;
        }
    sum=sum+(gradpoint*Cred[i]);
}
    System.out.println(sum);
    this.sgpa=sum/20.0;
}
}
```

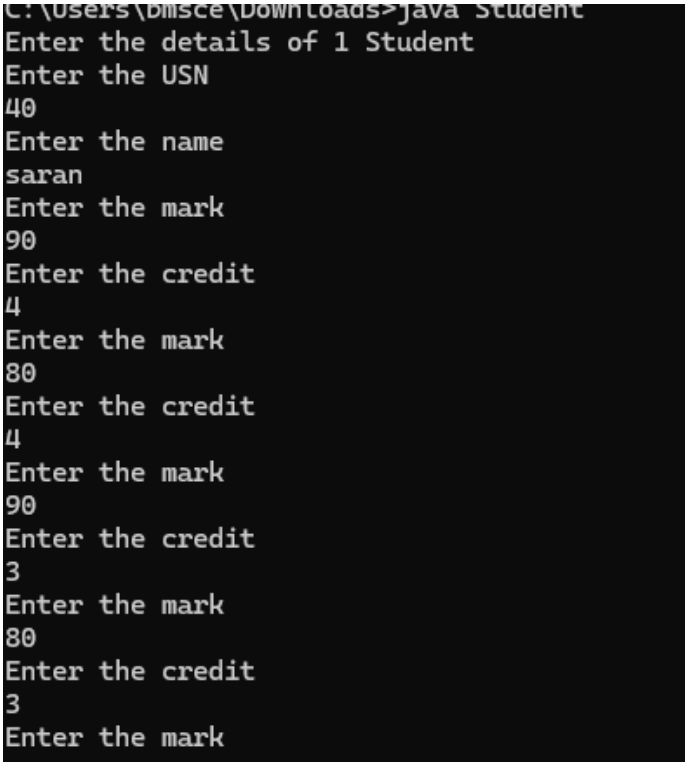
DATE: PAGE:

```
class students {
    public static void main(String args[]){
        Stud_details s1[] = new Stud_details
        for(int j=0;j<3; j++){
            s1[j] = new Stud_details();
        for(int j=0 ; j<3; j++){
            System.out.println("Enter the
                details of " +(j+1)+ "student");
            s1[j].getdetails();
            s1[j].calcSGPA();
    }
    for(int j=0;j<3;j++){
        s1[j].display()
    }
    }
}
```

Code:

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

	int marks[]=new int[8];
	int cred[]= new int[8];
	String name,usn;
	double sgpa;
	Scanner sc =new Scanner(System.in);


	void getdetails(){
		System.out.println(" Enter the USN ");
		usn=sc.next();
		System.out.println(" Enter the name ");
		name=sc.next();
		for(int i=0;i<8;i++){
			System.out.println(" Enter the mark ");
			marks[i]=sc.nextInt();
			System.out.println(" Enter the credit ");
			cred[i]=sc.nextInt();
}
}

void display(){



	System.out.println("usn"+" "+usn);
	System.out.println("name"+" "+name);


	System.out.println("SGPA is"+this.sgpa);


}

void calcSGPA(){
	int sum=0;
	int grdpoint=0;

	int x;
	for(int i=0;i<8;i++){
```

```
                x=marks[i]/10;
                switch(x){

                        case 10: grdpoint=10;break;
                        case 9:
                        case 8:
                        case 7:
                        case 6:
                        case 5:
                        case 4: grdpoint=x+1;break;
                        case 3:
                        case 2:
                        case 1:System.out.println("Fail");break;
                        }
                sum=sum+ (grdpoint*cred[i]);
}
        System.out.println(sum);
        this.sgpa=sum/20.0;

}
}


class Student{
        public static void main(String args[]){
                Stud_details s1[]=new Stud_details[3];
                for(int j=0;j<3;j++){
                        s1[j]=new Stud_details();       }
                for(int j=0;j<3;j++){
                        System.out.println("Enter the details of "+(j+1)+" Student");
                        s1[j].getdetails();
                        s1[j].calcSGPA();
}
for(int j=0;j<3;j++){
        s1[j].display();
}

}

}
```


C:\Users\bmsce\Downloads>java Student
Enter the details of 1 Student
Enter the USN
40
Enter the name
saran
Enter the mark
90
Enter the credit
4
Enter the mark
80
Enter the credit
4
Enter the mark
90
Enter the credit
3
Enter the mark
80
Enter the credit
3
Enter the mark

Gfy7ghu8hj90oklo

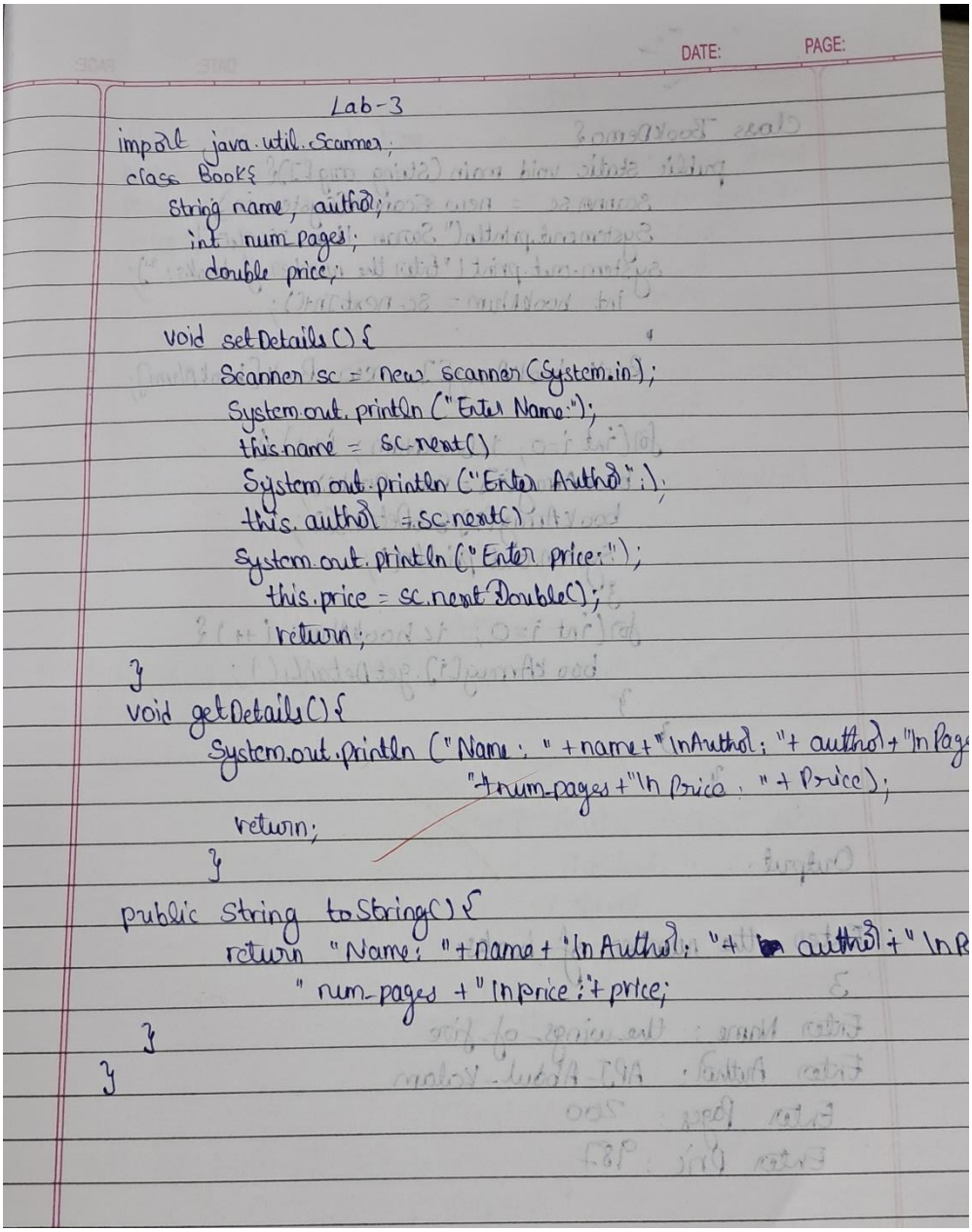
```
Enter the mark
80
Enter the credit
4
Enter the mark
90
Enter the credit
3
Enter the mark
80
Enter the credit
3
Enter the mark
90
Enter the credit
3
Enter the mark
80
Enter the credit
1
Enter the mark
90
Enter the credit
1
Enter the mark
80
Enter the credit
1
191
```

```
Enter the credit
3
Enter the mark
78
Enter the credit
1
Enter the mark
67
Enter the credit
1
Enter the mark
90
Enter the credit
1
169
usn 40
name saran
SGPA is9.55
usn 42
name anoop
SGPA is5.95
usn 14
name abhi
SGPA is8.45
```

Program 3

Book Program

Algorithm:



```
Lab-3
import java.util.Scanner;
class Books
    String name, author;
    int num_pages;
    double price;

    void setDetails(){
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Name:");
        this.name = sc.next()
        System.out.println("Enter Author:");
        this.author = sc.next();
        System.out.println("Enter price:");
        this.price = sc.nextDouble();
        return;
    }
    void getDetails(){
        System.out.println("Name: "+name+"\nAuthor: "+author+"\nPag
                "+num_pages+"\nPrice: "+Price);
        return;
    }
public String toString(){
    return "Name: "+name+"\nAuthor: "+[redacted]author+"\nP
        "num_pages+"\nprice:"+price;
    }
}
```

```
class BookDemo{
    public static void main(String arg[]){
        Scanner sc = new Scanner(System.in);
        System.out.println("Sowan 1bm23cs040");
        System.out.print("Enter the number of books: ");
        int bookNum = sc.nextInt();

        Book bookArray[] = new Book[bookNum];

        for(int i=0; i<bookNum; i++){
            bookArray[i] = new Book();
            bookArray[i].setDetails();
            System.out.println();
        }
        for(int i=0; i<bookNum; i++){
            bookArray[i].getDetails();
        }
    }
}
```

Output:

```
Enter the number of books:
3
Enter Name: the_wings_of_fire
Enter Author: APJ_Abdul_Kalam
Enter Pages: 200
Enter Price: 987
```

Code:

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

    void setDetails(){
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Name:");
        this.name = sc.next();
        System.out.print("Enter Author:");
        this.author = sc.next();
        System.out.print("Enter Pages:");
        this.num_pages = sc.nextInt();
        System.out.print("Enter Price:");
        this.price = sc.nextDouble();
        return;
    }

    void getDetails(){
        System.out.println("Name: "+name+"\nAuthor: "+author+"\nPages: "+num_pages+"\nPrice: "+price);
        return;
    }

    public String toString(){
        return "Name: "+name+"\nAuthor: "+author+"\nPages: "+num_pages+"\nPrice: "+price;
    }
}

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

        Scanner sc = new Scanner(System.in);
        System.out.print("Saran 1bm23cs040");
        System.out.println("Enter the number of books: ");
        int bookNum = sc.nextInt();
```

```
        Book bookArray[] = new Book[bookNum];

        for(int i = 0; i<bookNum; i++){
        bookArray[i] = new Book();
        bookArray[i].setDetails();
        System.out.println();
        }

        for (int i = 0; i<bookNum; i++){
        bookArray[i].getDetails();
        }
    }
}
```

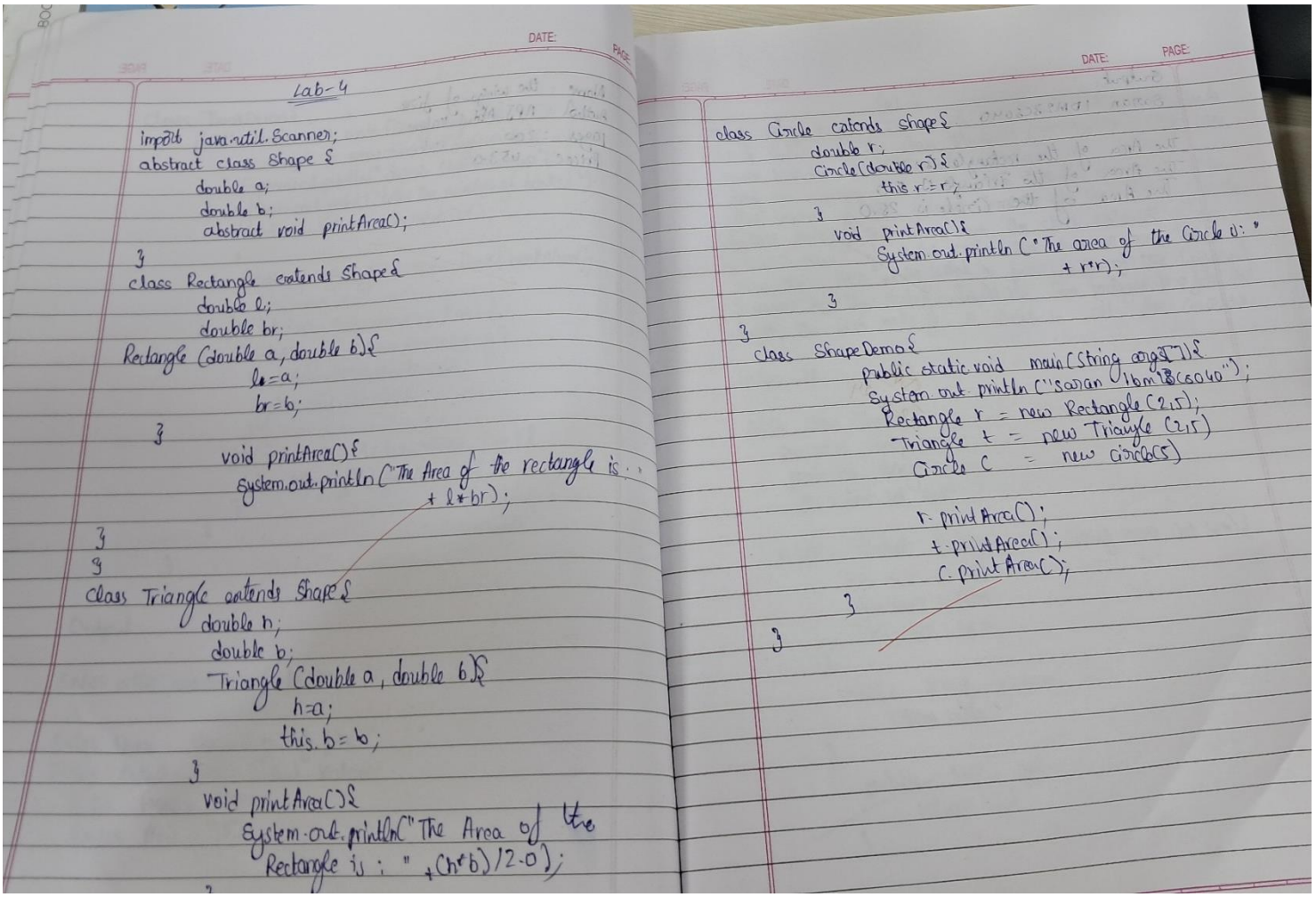
```
Saran 1bm23cs040Enter the number of books:
3
Enter Name:the_wings_of_fire
Enter Author:APJ_Abdul_kalam
Enter Pages:200
Enter Price:453

Enter Name:Geronimo_Stilton
Enter Author:dami
Enter Pages:345
Enter Price:543

Enter Name:the_lord_of_rings
Enter Author:john_ronald
Enter Pages:345
Enter Price:987

Name: the_wings_of_fire
Author: APJ_Abdul_kalam
Pages: 200
Price: 453.0
Name: Geronimo_Stilton
Author: dami
Pages: 345
Price: 543.0
```

```
Name: the_wings_of_fire
Author: APJ_Abdul_kalam
Pages: 200
Price: 453.0
Name: Geronimo_Stilton
Author: dami
Pages: 345
Price: 543.0
Name: the_lord_of_rings
Author: john_ronald
Pages: 345
Price: 987.0
```

Program 4

Abstract Class shape program

Algorithm



```
Lab-4

import java.util.Scanner;
abstract class Shape {
    double a;
    double b;
    abstract void printArea();
}
class Rectangle extends Shape{
    double l;
    double br;
Rectangle (double a, double b){
        l=a;
        br=b;
    }
    void printArea(){
    System.out.println("The Area of the rectangle is: "
                + l*br);
}
}
class Triangle extends Shape{
    double h;
    double b;
    Triangle (double a, double b){
        h=a;
        this.b=b;
    }
    void printArea(){
        System.out.println("The Area of the
        Rectangle is: "+(h*b)/2.0);
```

```
class Circle extends Shape{
    double r;
    Circle(double r){
        this.r=r;
    }
    void printArea(){
        System.out.println("The area of the Circle is: "
                + r*r);
    }
}
class ShapeDemo{
    public static void main(String args[]){
    System.out.println("Saran Tom B(6046)");
    Rectangle r = new Rectangle(2,5);
    Triangle t = new Triangle(2,5)
    Circle c = new Circle(5)

    r.printArea();
    t.printArea();
    c.printArea();
    }
}
```

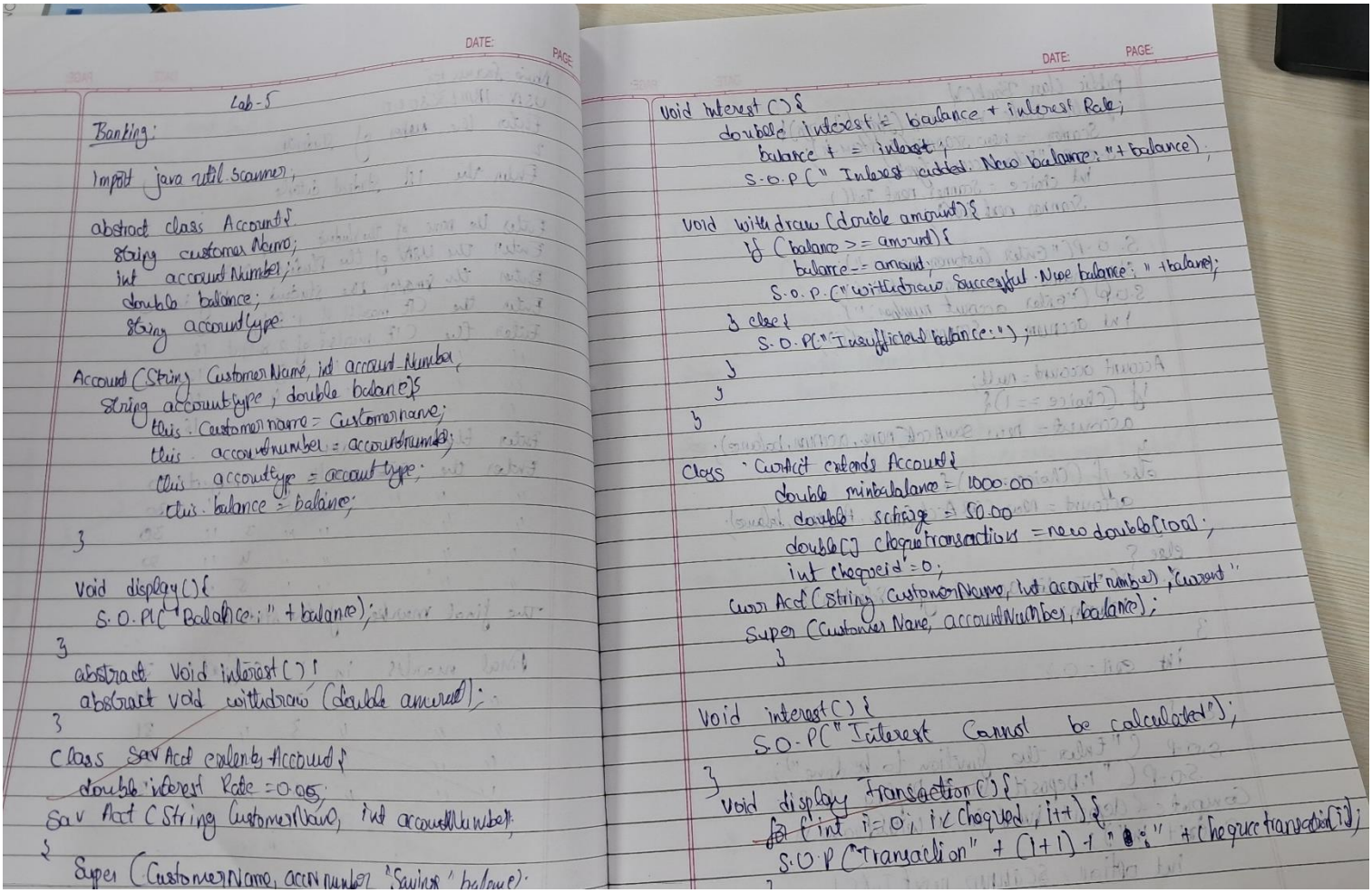
```
import java.util.Scanner;
abstract class Shape  {
	double a;
	double b;
	abstract void printArea();
}
class Rectangle extends Shape{
	double l;
	double br;
  Rectangle(double a, double b){
	l=a;
	br=b;
}

	void printArea(){
		System.out.println("The Area of the rectangle is: "+l*br);
}
}
class Triangle extends Shape{
	double h;
	double b;
	Triangle(double a, double b){
		h=a;
		this.b=b;
	}
	void printArea(){
		System.out.println("The Area of the Triangle is: "+(h*b)/2.0);
	}
}
class Circle extends Shape{
	double r;
	Circle(double r){
		this.r=r;
	}
	void printArea(){
		System.out.println("The area of the Circle is: "+ r*r);

	}
}
class ShapeDemo{
	public static void main(String args[]){
	   System.out.println("saran 1bm23cs040");
		Rectangle r = new Rectangle(2,5);
		Triangle t = new Triangle(2,5);
		Circle c = new Circle(5);

		r.printArea();
		t.printArea();
		c.printArea();

	}
}
```

```
C:\Users\bmsce\Downloads\cs040>java ShapeDemo
The Area of the rectangle is: 10.0
The Area of the Rectangle is: 5.0
The area of the Circle is: 25.0
```

Program 5

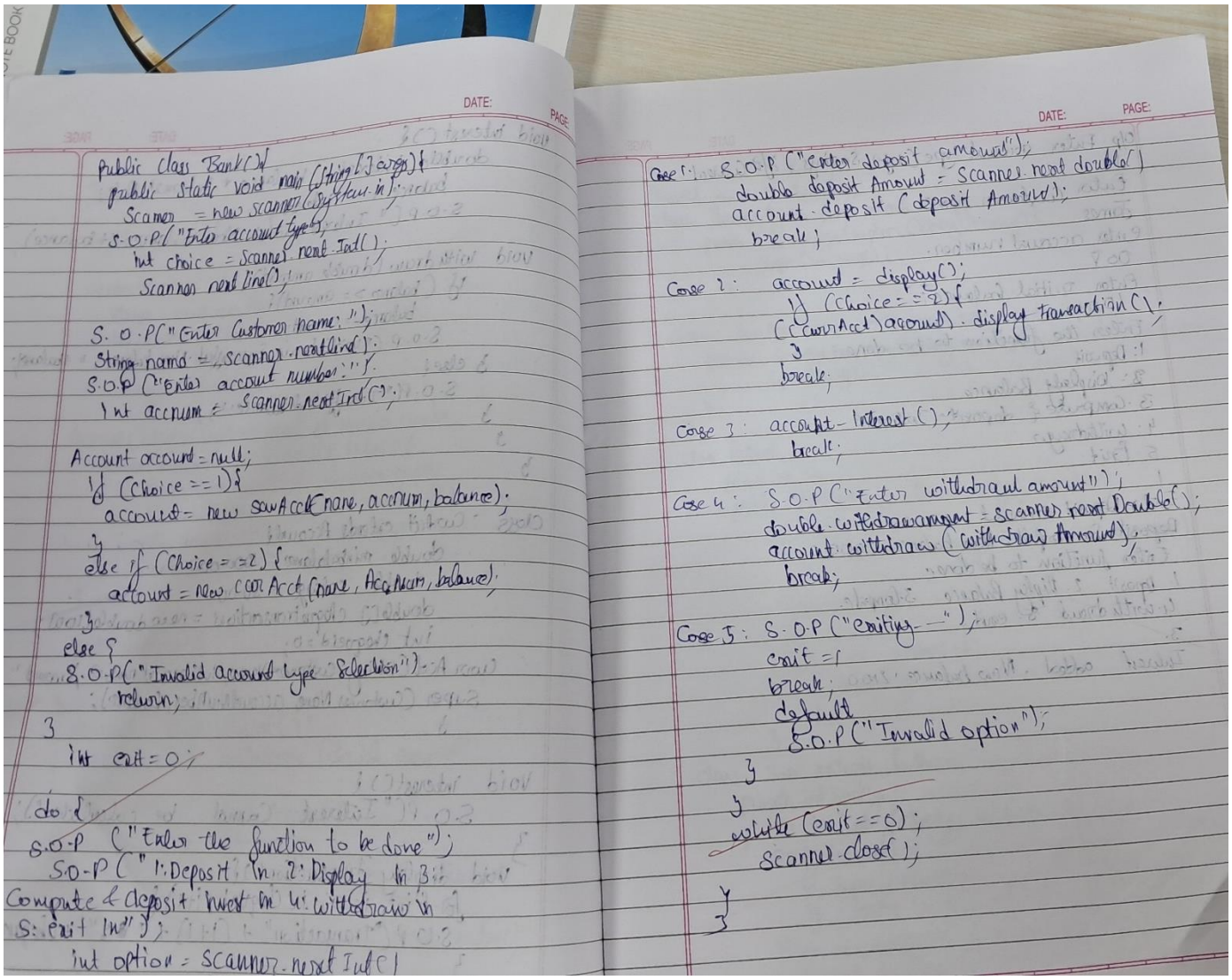
Bank program

Algorithm



```
Lab-5
Banking:
Import java.util.Scanner;

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

Account(String CustomerName, int accountNumber,
    String accountType, double balance){
        this.customername = Customername;
        this.accountnumber = accountnumber;
        this.accounttype = accounttype;
        this.balance = balance;
}

void display(){
    S.O.P("Balance:" + balance);
}
abstract void interest();
abstract void withdraw(double amount);
}
Class SavAcct extends Account{
    double interestRate = 0.05;
SavAct(String CustomerName, int accountNumber){
    Super(CustomerName, accNumber, "Saving", balance);
```

```
void interest(){
    double interest = balance * interestRate;
        balance += interest;
    S.O.P(" Interest added. New balance:" + balance);

void withdraw(double amount){
    if(balance >= amount){
        balance -= amount;
    S.O.P("withdraw successful. New balance:" + balance);
} else{
    S.O.P("Insufficient balance:");
}
}
}
Class CurrAcct extends Account{
    double minbalance = 1000.00
    double service = 50.00
    double[] chequetransactions = new double[100];
    int chequeid = 0;
CurrAcct(String CustomerName, int acountnumber, Current"
    Super(CustomerName, accountNumber, balance);
    }

void interest(){
    S.O.P("Interest Cannot be calculated");
}
void displayTransaction(){
    for(int i=0; i<chequeid; i++){
    S.O.P("Transaction" + (i+1) + ":" + chequetransaction[i]);
```



```
public class Bank{
  public static void main(String[] args){
    Scanner = new scanner(System.in);
    S.O.P("Enter account type");
     int choice = Scanner.nextInt();
      Scanner.nextLine();

S.O.P("Enter Customer name:");
String name = scanner.nextLine();
S.O.P("Enter account number:");
 Int accnum = scanner.nextInt();

Account account = null;
 if (choice == 1){
   account = new SavAcct(name, accnum, balance);
}
else if (choice == 2){
   account = new CurrAcct(name, Acc.num, balance);
}
else {
  S.O.P("Invalid account type Selection")
    return;
}
  int exit = 0;

do {
S.O.P ("Enter the function to be done");
  S.O.P("1: Deposit In 2: Display In 3:
Compute & deposit Invest In 4: withdraw In
5: Exit In");
  int option = scanner.nextInt()
```

```
Case 1: S.O.P("Enter deposit amount");
    double depositAmount = Scanner.nextDouble()
    account.deposit(depositAmount);
    break;

Case 2: account = display();
    if (choice == 2){
    ((CurrAcct)account).displayTransaction();
    }
    break;

Case 3: account.Interest();
    break;

Case 4: S.O.P("Enter withdraw amount");
    double withdrawamount = scanner.nextDouble();
    account.withdraw(withdrawAmount);
    break;

Case 5: S.O.P("exiting...");
    exit = 1
    break;
    default
     S.O.P("Invalid option");
  }
}
while(exit == 0);
  Scanner.close();

}
}
```

Code:

```
import java.util.Scanner;

class Account {
    private String custName;
    private String accNo;
    private double balance;

    public Account(String custName, String accNo, double balance) {
        this.custName = custName;
        this.accNo = accNo;
        this.balance = balance;
    }

    public double getBalance() {
        return this.balance;
    }

    public void deposit(double amount) {
        if (amount > 0) {
            this.balance += amount;
            System.out.println("The current balance is " + this.balance);
        } else {
            System.out.println("Amount should not be negative");
        }
    }

    public void withdraw(double amount) {
        if (amount > 0 && (balance - amount) >= 0) {
            this.balance -= amount;
            System.out.println("Withdraw successful. Current balance: " + this.balance);
        } else {
            System.out.println("Withdraw is not possible");
        }
    }
}

class SavingsAccount {
    private double interestRate;
    private Account account;

    public SavingsAccount(String custName, String accNo, double balance, double interestRate) {
        this.interestRate = interestRate;
        this.account = new Account(custName, accNo, balance);
    }

    public void addInterest() {
        double interest = account.getBalance() * this.interestRate;
        account.deposit(interest);
    }
```

```
    public Account getAccount() {
        return account;
    }
}

class CurrentAccount {
    private double minBalance;
    private Account account;

    public CurrentAccount(String custName, String accNo, double balance, double minBalance) {
        this.minBalance = minBalance;
        this.account = new Account(custName, accNo, balance);
    }

    public void withdraw(double amt) {
        if (amt > 0 && (account.getBalance() - amt) >= minBalance) {
            account.withdraw(amt);
        } else {
            System.out.println("Withdraw is not possible");
        }
    }

    public Account getAccount() {
        return account;
    }
}

public class Bank {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the name:");
        String name = sc.nextLine();
        System.out.println("Enter the account number:");
        String accnt = sc.nextLine();

        while (true) {
            System.out.println("Enter your choice:");
            System.out.println("1. Savings Account");
            System.out.println("2. Current Account");
            System.out.println("3. Exit");
            int choice = sc.nextInt();

            switch (choice) {
                case 1:
                    System.out.println("Enter initial balance:");
                    double savingsBalance = sc.nextDouble();
                    System.out.println("Enter the interest rate:");
                    double interestRate = sc.nextDouble();
                    SavingsAccount savingsAccount = new SavingsAccount(name, accnt, savingsBalance, interestRate);
                    savingsAccount.addInterest();
                    break;
```

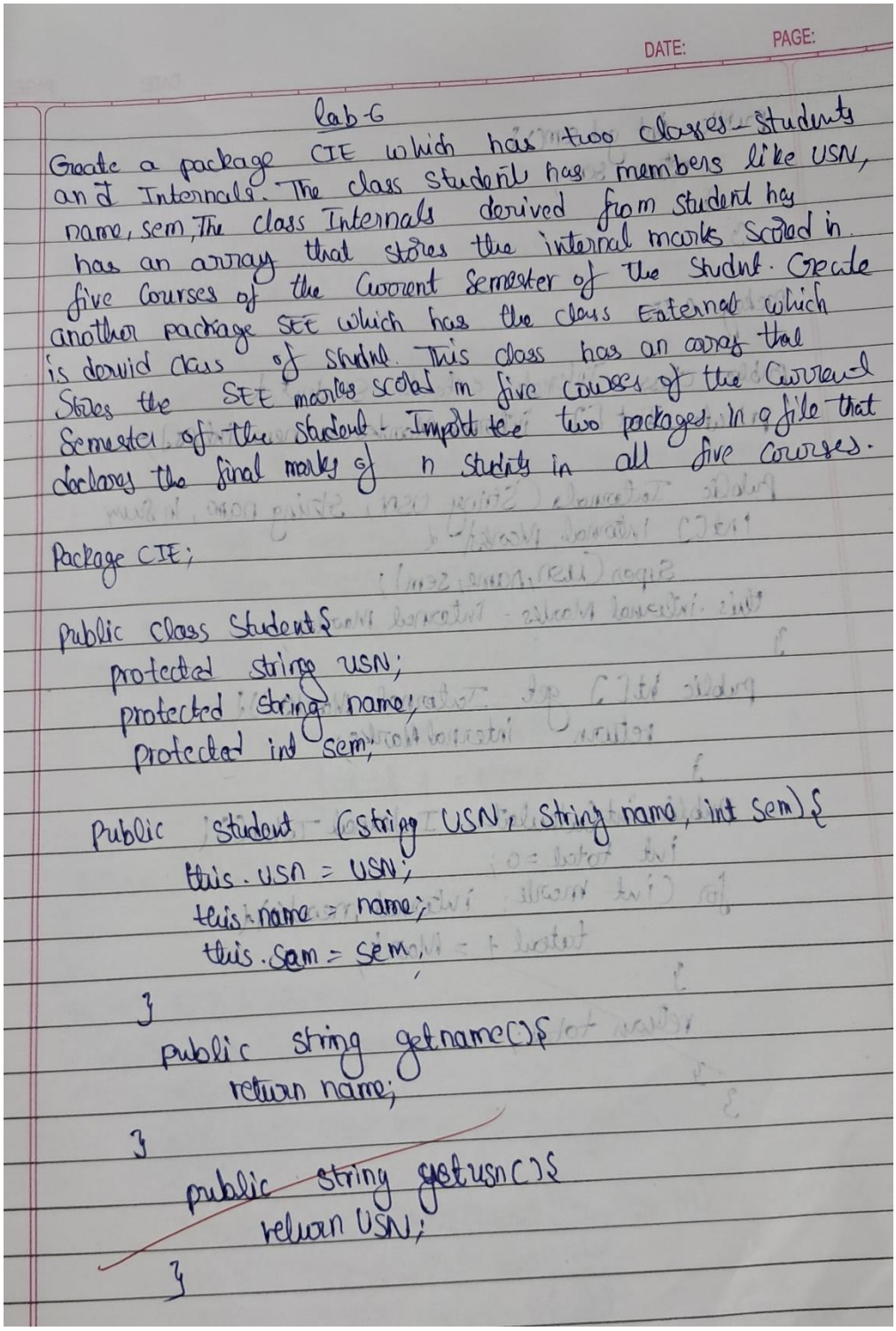
```
            case 2:
                System.out.println("Enter initial balance:");
                double currentBalance = sc.nextDouble();
                System.out.println("Enter minimum balance:");
                double minBalance = sc.nextDouble();
                CurrentAccount currentAccount = new CurrentAccount(name, accnt, currentBalance, minBalance);

                System.out.println("entr the amount to be withdraw");
                double q = sc.nextInt();
                currentAccount.withdraw(q);
                System.out.println("Account created. Current balance: " + currentAccount.getAccount().getBalance());
                break;

            case 3:
                System.out.println("Exiting...");
                sc.close();
                return;

            default:
                System.out.println("Invalid choice. Please try again.");
        }
    }
}
}
```

Output:

```
Enter the account number:
987653
Enter your choice:
1. Savings Account
2. Current Account
3. Exit
1
Enter initial balance:
456
Enter the interest rate:
7
The current balance is 3648.0
Enter your choice:
1. Savings Account
2. Current Account
3. Exit
2
Enter initial balance:
456
Enter minimum balance:
5
entr the amount to be withdraw
654
Withdraw is not possible
Account created. Current balance: 456.0
Enter your choice:
1. Savings Account
2. Current Account
3. Exit
```

Program 6
Packages

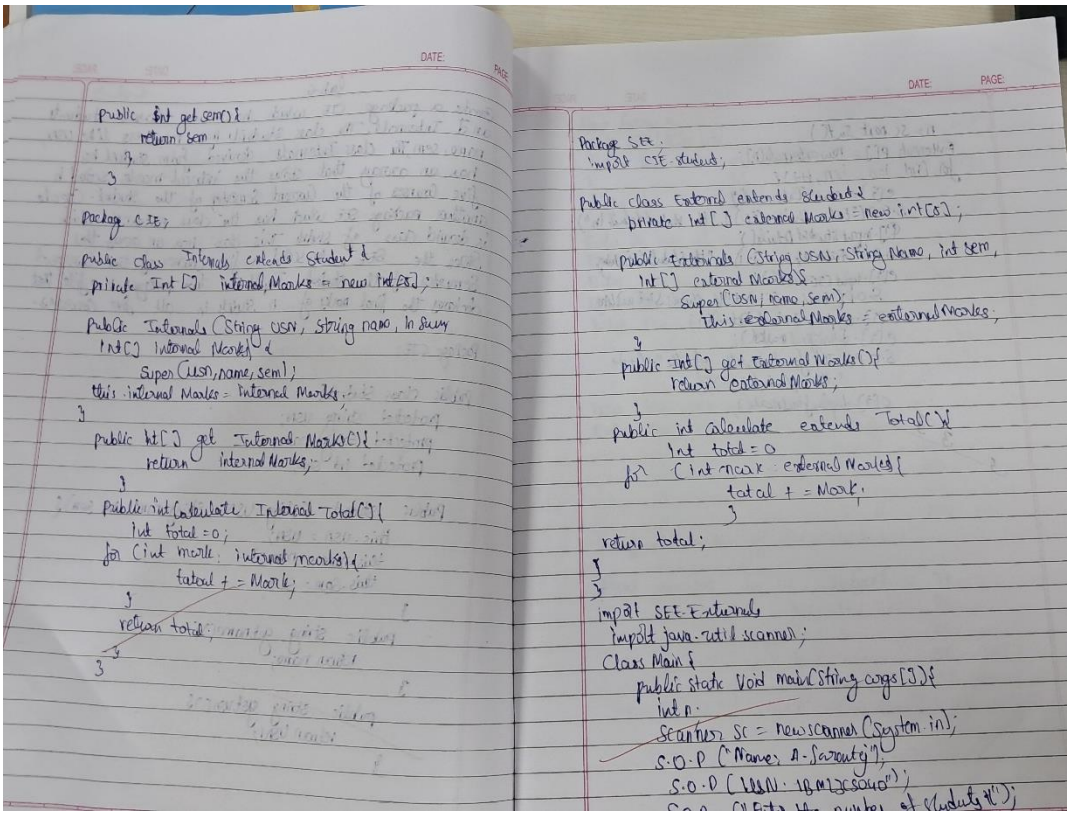
Algorithm


DATE: PAGE:

lab-6

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

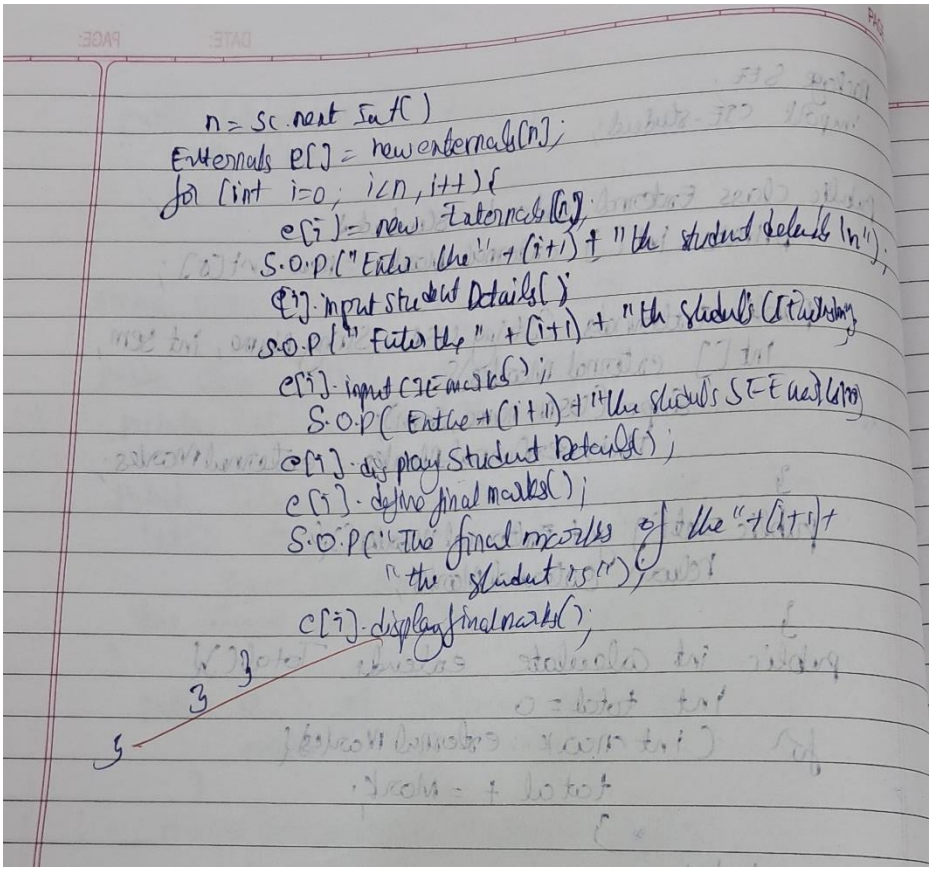
```
Package CIE;

Public class Student{
    protected string USN;
    protected string name;
    protected int sem;

Public Student (String USN, String name, int sem){
        this.usn = USN;
        this.name = name;
        this.sem = sem;
    }
    public string getname(){
        return name;
    }
    public string getusn(){
        return USN;
    }
```



```
public int getSem(){
    return Sem;
}
}
```

```
package CIE;
public class Internals extends Student{
private int[] internalMarks = new int[5];
public Internals(String USN, String name, int sem,
int[] internalMarks){
    super(USN, name, sem);
this.internalMarks = internalMarks;
}
public int[] getInternalMarks(){
    return internalMarks;
}
public int calculateInternalTotal(){
    int total = 0;
for(int mark: internalMarks){
    total += Mark;
}
return total;
}
```

```
package SEE;
import CIE.Student;
public class External extends Student{
    private int[] externalMarks = new int[5];
    public Externals(String USN, String Name, int sem,
    int[] externalMarks){
        super(USN, name, sem);
        this.externalMarks = externalMarks;
    }
public int[] getExternalMarks(){
    return externalMarks;
}
public int calculate extends Total{
    int total = 0
for(int mark: externalMarks){
        total += Mark;
    }
return total;
}
}
import SEE.Externals
import java.util.scanner;
Class Main{
    public static void main(String args[]){
    int n.
    Scanner sc = new Scanner(System.in);
    S.O.P("Name & Seventy");
    S.O.P("USN: 1BM23CS040");
```



```
n = sc.nextInt()
External e[] = new external[n];
for (int i=0; i<n, i++){
    e[i] = new External();
S.O.P("Enter the" + (i+1) + "th student details \n");
e[i].inputStudentDetails();
S.O.P("Enter the" + (i+1) + "th student's CIE marks");
e[i].inputCIEmarks();
S.O.P("Enter the" + (i+1) + "th student's SEE marks");
e[i].displayStudentDetails();
e[i].defineFinalmarks();
S.O.P("The final marks of the" + (i+1) +
"th student is");
e[i].displayfinalmarks();
}
```

Code:

```
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 name of the student \n");
			this.name=sc.nextLine();
			System.out.println("Enter the USN of the student \n");
			this.usn=sc.nextLine();
			System.out.println("Enter the semester the student is studying in \n");
			this.sem=sc.nextInt();
		}

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

package CIE;
import java.util.Scanner;

public class Internals extends Student {
		protected double ciemarks[] = new double[5];
		Scanner sc= new Scanner(System.in);
		public void inputCIEmarks(){
			for(int i=0; i<5; i++){
				System.out.println("Enter the CIE marks of"+ (i+1)+"th subject");
				this.ciemarks[i]= sc.nextDouble();
			}
		}
}


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

public class Externals extends Internals{
				protected double seemarks[] = new double[5];
				protected double finalMarks[] = new double[5] ;

				public void inputSEEmarks() {
					Scanner sc = new Scanner(System.in);
					for(int i=0; i<5; i++){
						System.out.println("Enter the SEE marks of"+ (i+1)+"th subject");
						this.seemarks[i]= sc.nextDouble();
					}
				}

				public void definefinalmarks(){
					for(int i=0; i<5; i++){
						this.finalMarks[i] = ciemarks[i] + (seemarks[i]/2.0);
					}

				}

				public void displayfinalmarks(){
					System.out.println("The final marks of the student is \n");
					for(int i=0; i<5; i++){
						System.out.println("The marks of the" + (i+1)+"th subject is \t");
						System.out.println(this.finalMarks[i]);
					}
```

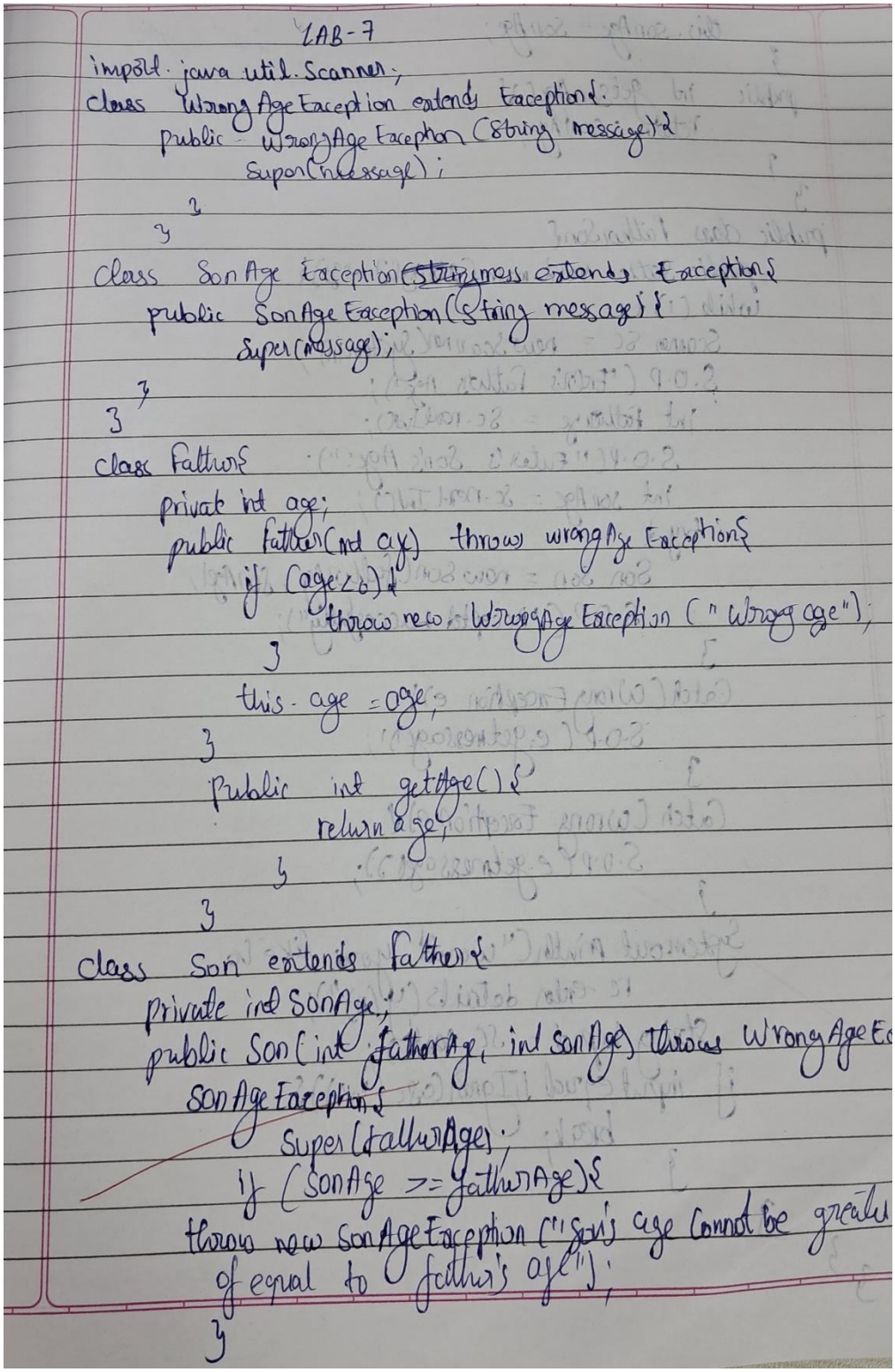
```
                                        }
}


import SEE.Externals;
import java.util.Scanner;
class Main{
                public static void main(String args[]){
                        int n;
                        Scanner sc = new Scanner(System.in);
                        System.out.println("Name: saran tej");
                        System.out.println("USN: 1BM23CS040");
                        System.out.println("Enter the number of students \t");
                        n= sc.nextInt();
                        Externals e[] = new Externals[n];
                        for(int i=0; i<n; i++){
                                e[i] = new Externals();
                                System.out.println("Enter the" +(i+1)+"th student details \n");
                                e[i].inputStudentDetails();
                                System.out.println("Enter the" +(i+1)+"th student's CIE marks\n");
                                e[i].inputCIEmarks();
                                System.out.println("Enter the" +(i+1)+"th student's SEE marks\n");
                                e[i].inputSEEmarks();
                                System.out.println("The details of the" +(i+1)+"th student is");
                                e[i].displayStudentDetails();
                                e[i].definefinalmarks();
                                System.out.println("The final marks of the" +(i+1)+"th student is");
                                e[i].displayfinalmarks();
                        }

                }
}
```

```
Name: saran tej
USN: 1BM23CS040
Enter the number of students
2
Enter the1th student details

Enter the name of the student

anoopa
Enter the USN of the student

042
Enter the semester the student is studying in

3
Enter the1th student's CIE marks

Enter the CIE marks of1th subject
16
Enter the CIE marks of2th subject
16
Enter the CIE marks of3th subject
16
Enter the CIE marks of4th subject
16
Enter the CIE marks of5th subject
16
Enter the1th student's SEE marks

Enter the SEE marks of1th subject
30
Enter the SEE marks of2th subject
30
Enter the SEE marks of3th subject
30
Enter the SEE marks of4th subject
30
Enter the SEE marks of5th subject
30
The details of the1th student is
Name: anoopa
USN: 042
Semester: 3
The final marks of the1th student is
The final marks of the student is

The marks of the1th subject is
31.0
The marks of the2th subject is
31.0
The marks of the3th subject is
31.0
The marks of the4th subject is
31.0
The marks of the5th subject is
31.0
Enter the2th student details

Enter the name of the student

anirudh
Enter the USN of the student

036
Enter the semester the student is studying in

3
Enter the2th student's CIE marks

Enter the CIE marks of1th subject
50
Enter the CIE marks of2th subject
50
Enter the CIE marks of3th subject
50
Enter the CIE marks of4th subject
50
Enter the CIE marks of5th subject
50
Enter the2th student's SEE marks

Enter the SEE marks of1th subject
100
Enter the SEE marks of2th subject
100
Enter the SEE marks of3th subject
100
Enter the SEE marks of4th subject
100
Enter the SEE marks of5th subject
100
The details of the2th student is
Name: anirudh
USN: 036
Semester: 3
The final marks of the2th student is
The final marks of the student is

The marks of the1th subject is
100.0
The marks of the2th subject is
100.0
The marks of the3th subject is
100.0
The marks of the4th subject is
```

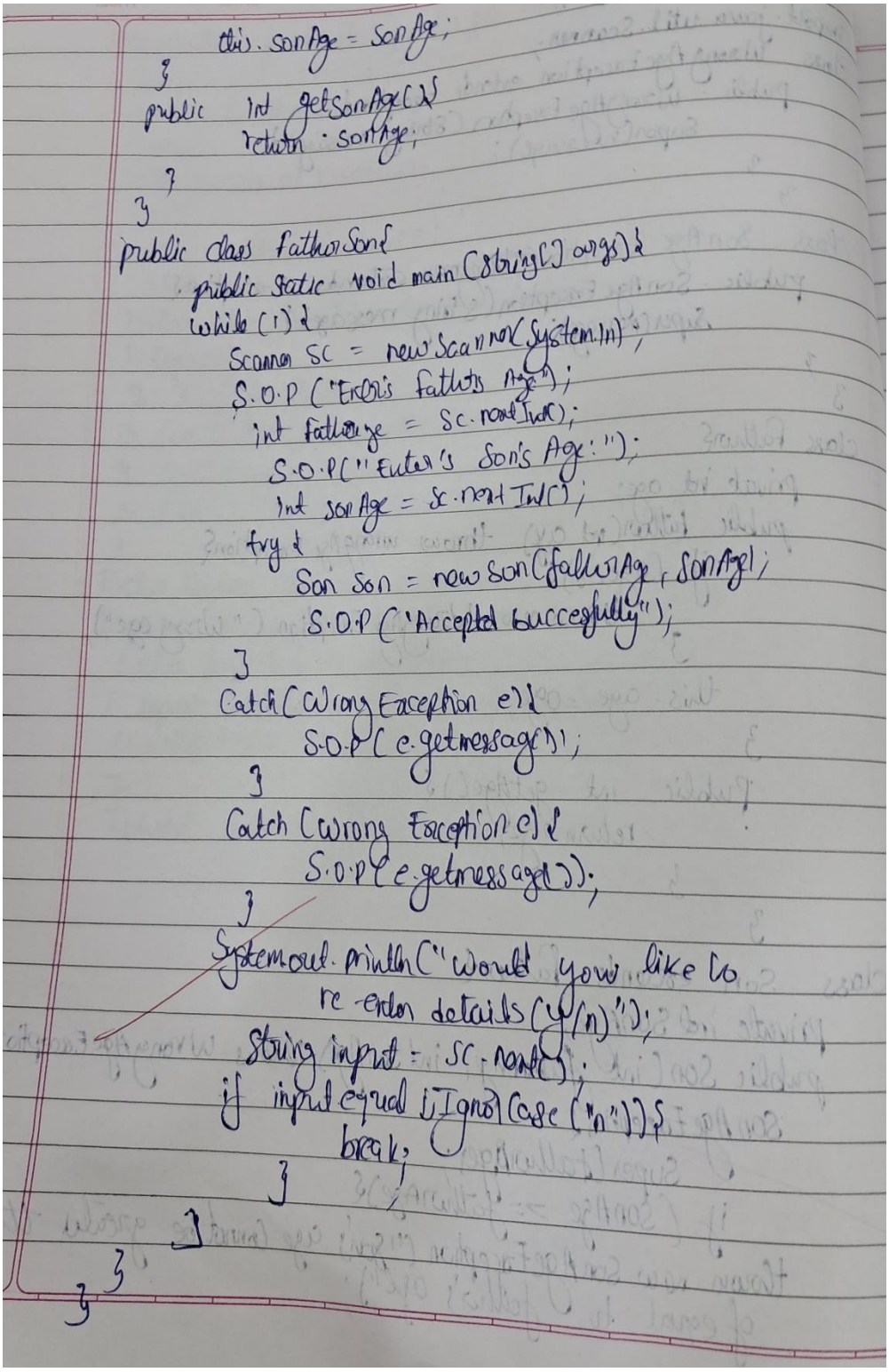
Program 7

Exception handling

Algorithm



```
LAB-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 WrongAgeE
    SonAgeException{
        super(FatherAge);
        if(SonAge>=FatherAge){
    throw new SonAgeException("Son's age cannot be greate
      of equal to father's age");
    }
}
```



```
        this.sonAge = sonAge;
    }
    public int getSonAge(){
        return sonAge;
    }
}
public class FatherSon{
    public static void main(String[] args){
    while(1){
        Scanner sc = new Scanner(System.in);
        S.O.P("Enter's father's Age");
        int fatherAge = sc.nextInt();
        S.O.P("Enter's Son's Age:");
        int sonAge = sc.nextInt();
        try{
            Son son = new Son(fatherAge, sonAge);
            S.O.P("Accepted Successfully");
        }
        Catch(WrongException e){
            S.O.P(e.getMessage());
        }
        Catch(WrongException e){
            S.O.P(e.getMessage());
        }
        System.out.println("Would you like to
            re-enter details(y/n)");
        String input = sc.next();
        if input equalsIgnoreCase("n"){
            break;
        }
    }
}
```

```
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 age");
        }
        this.sonAge = sonAge;
    }
    public int getSonAge() {
        return sonAge;
    }
}
public class FatherSon{
    public static void main(String[] args) {
        while(true){
            Scanner sc = new Scanner(System.in);
            System.out.print("Enter Father's Age: ");
            int fatherAge = sc.nextInt();
            System.out.print("Enter Son's Age: ");
            int sonAge = sc.nextInt();
            try {
                Son son = new Son(fatherAge, sonAge);
```

```
            System.out.println("Accepted Succesfully");
        }
        catch (WrongAgeException e) {
            System.out.println(e.getMessage());
        }
        catch (SonAgeException e) {
            System.out.println(e.getMessage());
        }
        System.out.println("Would you like to re-enter details (Y/n)");
        String input = sc.next();
        if (input.equalsIgnoreCase("n")) {
            break;
        }
    }
  }
}
```

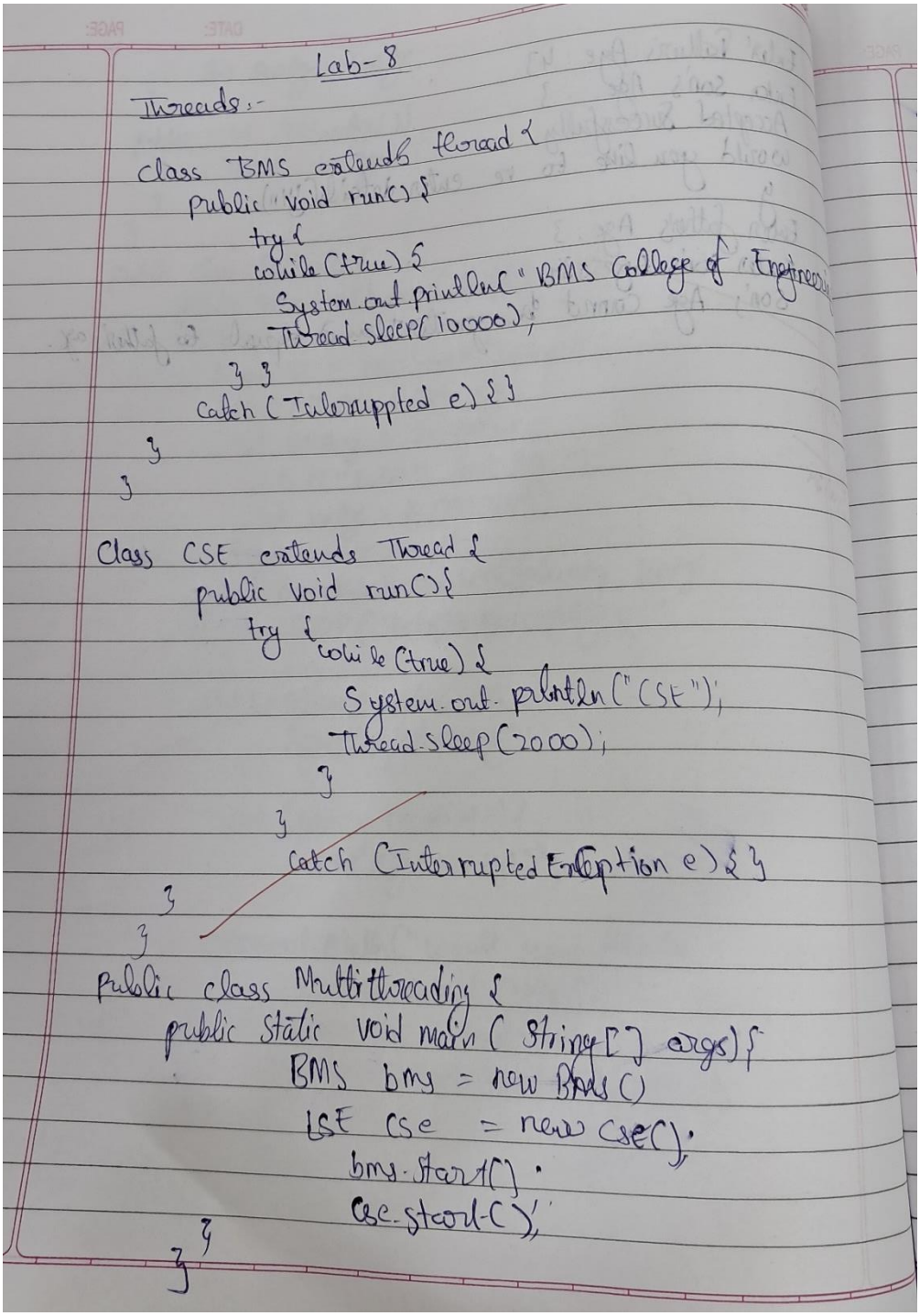
Output

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

C:\Users\Admin\Downloads>javac Fatherson.java

C:\Users\Admin\Downloads>java FatherSon
Enter Father's Age: 43
Enter Son's Age: 3
Accepted Succesfully
Would you like to re-enter details (Y/n)
y
Enter Father's Age: 3
Enter Son's Age: 34
Son's age cannot be greater than or equal to father's age
Would you like to re-enter details (Y/n)
```

Program 8

Multithreading

Algorithm



Lab-8

Threads:-

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

Class CSE extends Thread {
    public void run() {
        try {
        while (true) {
            System.out.println("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();
    }
}
```

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) {
        System.out.println("saran tej 1BMCS23CS040");
        BMS bms = new BMS();
        CSE cse = new CSE();
        bms.start();
        cse.start();
    }
```

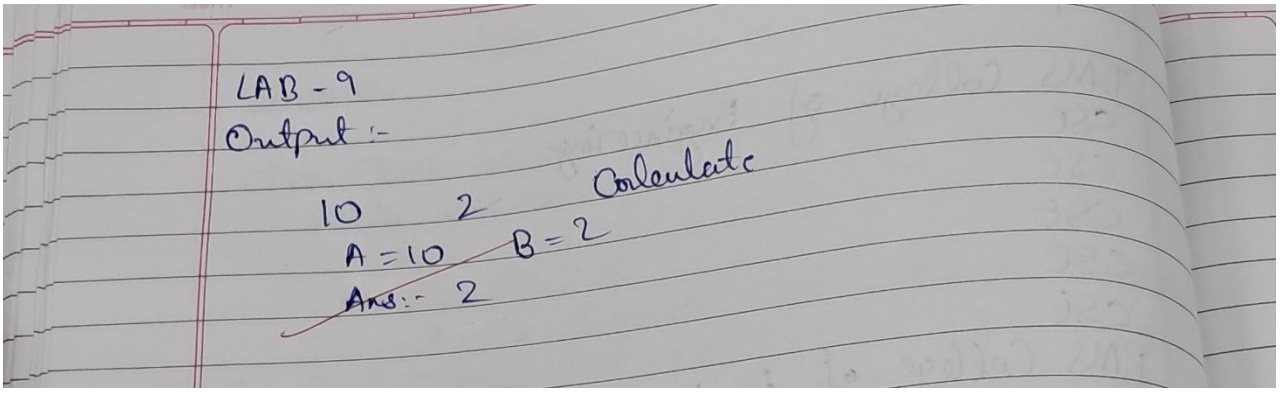
output

```
saran tej 1BMCS23CS040
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
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
BMS College of Engineering
CSE
CSE
CSE
CSE
```

Program 9

Integer division with user interface

Algorithm

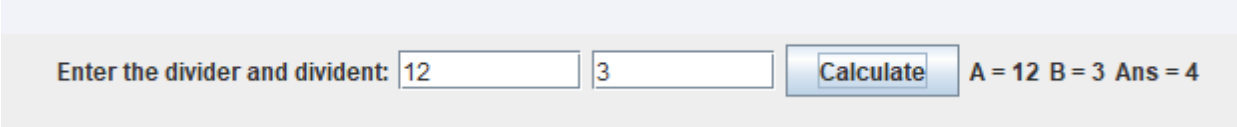


Code:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
class SwingDemo{
SwingDemo(){
// create jframe container
JFrame jfrm = new JFrame("Divider App");
jfrm.setSize(275, 150);
jfrm.setLayout(new FlowLayout());
// to terminate on close
jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
// text label
JLabel jlab = new JLabel("Enter the divider and divident:");
// add text field for both numbers
JTextField ajtf = new JTextField(8);
JTextField bjtf = new JTextField(8);
// calc button
JButton button = new JButton("Calculate");
// labels
JLabel err = new JLabel();
JLabel alab = new JLabel();
JLabel blab = new JLabel();

JLabel anslab = new JLabel();
// add in order :)
jfrm.add(err); // to display error bois
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!");
}
}
});
// display frame
jfrm.setVisible(true);
}
public static void main(String args[]){
// create frame on event dispatching thread
SwingUtilities.invokeLater(new Runnable(){
public void run(){
new SwingDemo();
}
});
}
}
```

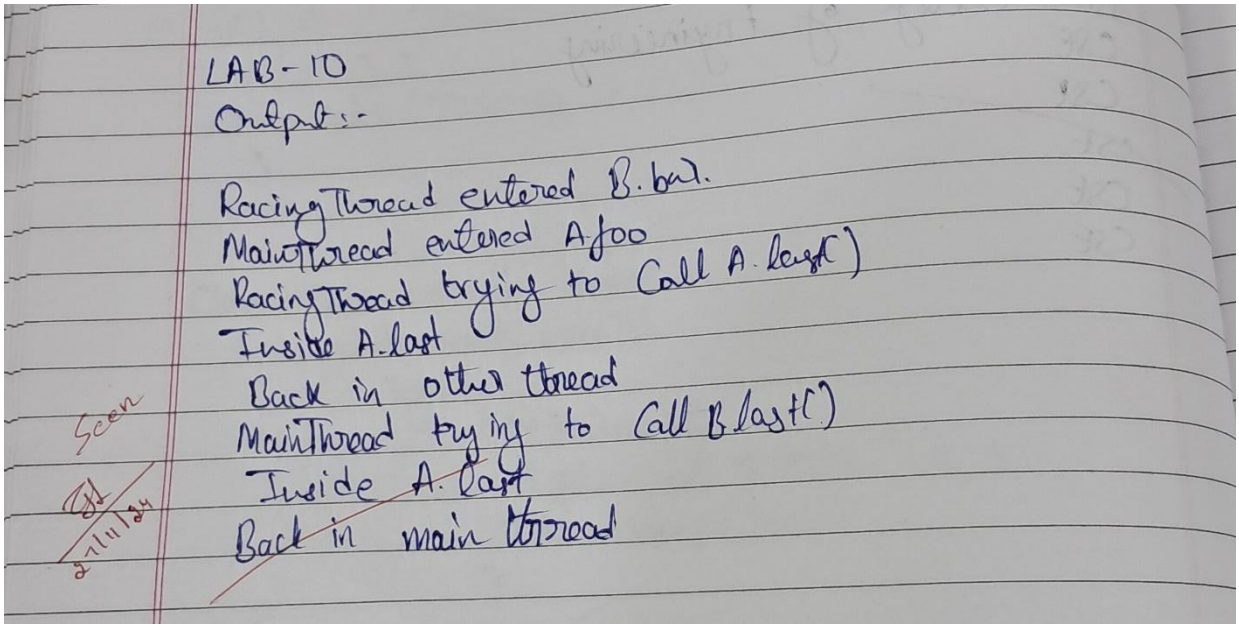
Output

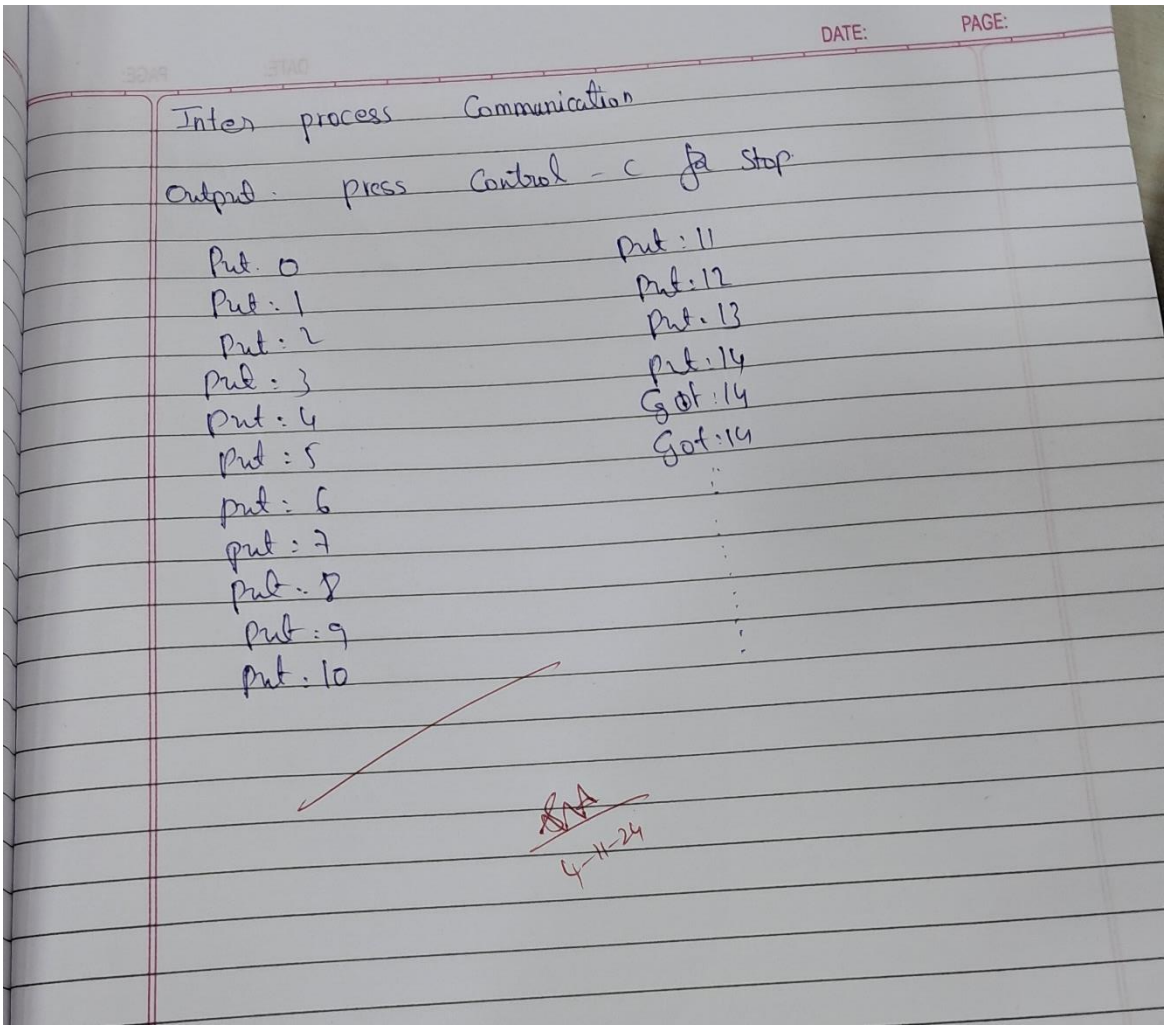


Program 10

Inter process communication and deadlock

Algorithm




DATE:
PAGE:
Inter process Communication
Output: press Control - c for Stop
Put: 0
Put: 1
Put: 2
Put: 3
Put: 4
Put: 5
put: 6
put: 7
Put: 8
Put: 9
Put: 10
Put: 11
Put: 12
Put: 13
Put: 14
Got: 14
Got: 14
4-11-24

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<15) {
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<15) {

int r=q.get();

System.out.println("consumed:"+r);

i++;

}

}

}

class PCFixed{
public static void main(String args[]) {
System.out.println("Name:Anirudh");
System.out.println("USN:1BM23CS036");
Q q = new Q();

new Producer(q);

new Consumer(q);

System.out.println("Press Control-C to stop.");

}

}

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 this thread.

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();

}}
```

Output

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

```
Press Control-C to stop.
Put: 0

Intimate Consumer


Producer waiting

Got: 0

Intimate Producer

Put: 1

Intimate Consumer


Producer waiting

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


Producer waiting

Got: 4

Intimate Producer

consumed:4
Put: 5

Intimate Consumer


Producer waiting

Got: 5

Intimate Producer

consumed:5
Put: 6
```