VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming

(23CS3PCOOJ)

Submitted by

Afreen Anz (1BM23CS016)

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 **Afreen Anz (1BM23CS016),** 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.

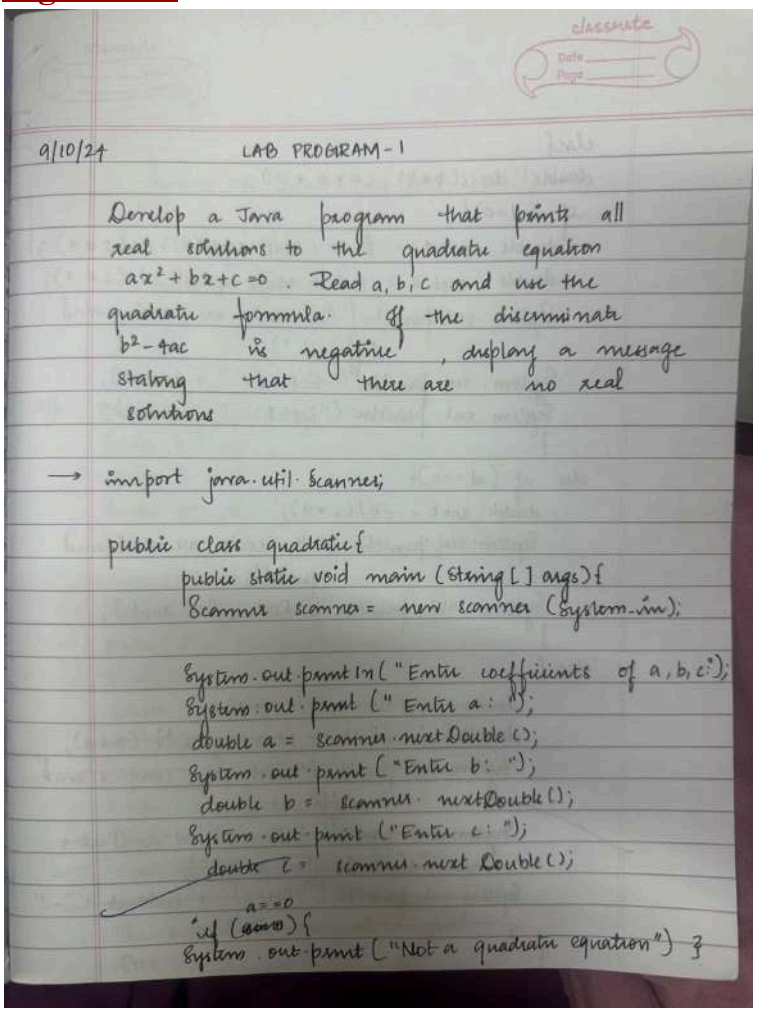
Lab faculty Incharge Name Assistant Professor Department of CSE, BMSCE	Dr. Jyothi S Nayak Professor & HOD Department of CSE, BMSCE

Index

Github Link:
https://github.com/1BM23CS016/Java-lab

Program 1
Implement Quadratic Equation

Algorithm:



9/10/24 LAB PROGRAM-1

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

```
import java.util.Scanner;

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

System.out.println("Entre coefficients of a,b,c:");
System.out.print ("Entre a: ");
 double a = scanner.nextDouble();
System.out.print("Entre b: ");
 double b = scanner.nextDouble();
System.out.print("Entre c: ");
 double c = scanner.nextDouble();
 a==0
if (a==0){
System.out.print("Not a quadratic equation") }
```

```
else{
double d= (b*b)-(4*a*c);
if (d>0){
double root1 = (-b + Math.sqrt(d)) / (2*a);
double root2 = (-b + Math.sqrt(d)) / (2*a);
System.out.println("The roots are real and
                    different ");
System.out.println(" Root 1: " + root1);
System.out.println ("Root 2: " + root2);
}
else if (d==0){
double root = -b/(2*a);
System.out.println ("The roots are real and
                    equal");
System.out.println ("Root : " + root);
}
else {
double realPart = -b/(2*a);
double imagPart = Math.sqrt(d)/(2*a);
System.out.println ("Roots are complex and
                    different");
System.out.println ("Root 1 : " + realPart +
    "+" + imagPart + "i");
System.out.println ("Root 2 :" + realPart + "-"
    + imagPart + "i");
}
}
System.out.print ("no real roots");
}
}
scanner.close();
}
}
```

o/p: Enter coefficients of a,b,c:
Enter a: 0
Enter b: 4
Enter c: 6
Not a quadratic equation

Enter coefficients of a,b,c:
Enter a: 5
Enter b: 7
Enter c: 9
No real roots

Enter coefficients of a,b,c:
Enter a: 2
Enter b: 4
Enter c: 2
The roots are real and equal
Root : -1.0

Enter coefficients a,b,c:
Enter a: 2
Enter b: 8
Enter c: 2
The roots are real and different
Root 1: -0.2679491
Root 2: -3.7320508

o/p seen
Gl
9/10/24

Code:

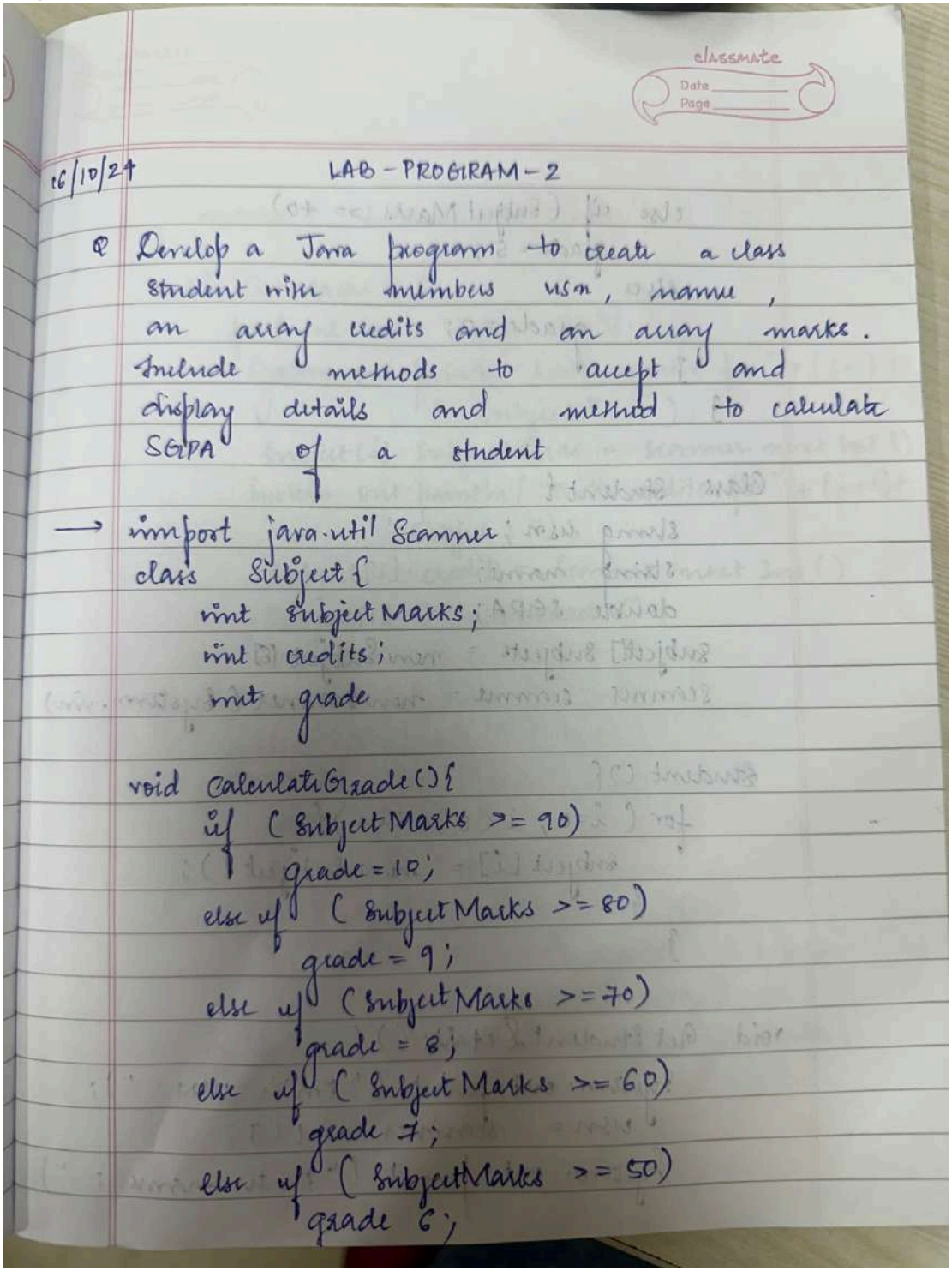
```
import java.util.Scanner;

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

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

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

         System.out.println("aaryan");

      if (d > 0) {
         double root1 = (-b + Math.sqrt(d)) / (2 * a);
         double root2 = (-b - Math.sqrt(d)) / (2 * a);

         System.out.println("Real Roots");
         System.out.println("Root 1: " + root1);
         System.out.println("Root 2: " + root2);
      }
         else if (d == 0) {
         double root = -b / (2 * a);
         System.out.println("Roots are real and equal");
         System.out.println("Root: " + root);
      }
         else {
         System.out.println("Roots are complex");
      }

      scanner.close();
   }
}
```

```
D:\1BM23CS016>java quadratic.java
Enter the coefficients of a,b,c:
Enter a: 5
Enter b: 7
Enter c: 9
no real roots
NAME: AFREEN ANZ
USN: 1BM23CS016

D:\1BM23CS016>java quadratic.java
Enter the coefficients of a,b,c:
Enter a: 2
Enter b: 4
Enter c: 2
The roots are real and equal.
Root: -1.0

NAME: AFREEN ANZ
USN: 1BM23CS016

D:\1BM23CS016>java quadratic.java
Enter the coefficients of a,b,c:
Enter a: 2
Enter b: 8
Enter c: 2
The roots are real and different.
Root 1: -0.2679491924311228
Root 2: -3.732050807568877

NAME: AFREEN ANZ
USN: 1BM23CS016

D:\1BM23CS016>java quadratic.java
Enter the coefficients of a,b,c:
Enter a: 0
Enter b: 4
Enter c: 6
Not a quadratic quation
NAME: AFREEN ANZ
USN: 1BM23CS016
```

Program 2

Calculate SGPA (Student Class)

Algorithm:



16/10/24 LAB - PROGRAM - 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 accept and display details and method to calculate SGPA of a student

```
import java.util.Scanner
class Subject {
    int subjectMarks;
    int credits;
    int grade

void CalculateGrade(){
    if (subjectMarks >= 90)
        grade = 10;
    else if (subjectMarks >= 80)
        grade = 9;
    else if (subjectMarks >= 70)
        grade = 8;
    else if (subjectMarks >= 60)
        grade 7;
    else if (subjectMarks >= 50)
        grade 6;
```

```
        else if (SubjectMarks >= 40)
            grade = 5;
        else
            grade = 0;
    }
}

Class Student {
    String usn;
    String name;
    double SGPA;
    Subject[] subjects = new Subject[8]
    Scanner scanner = new Scanner(System.in)

    Student(){
        for(i=0; i<8; i++){
            subject[i] = new Subject();
        }
    }

void GetStudentDetails(){
    System.out.print("Enter USN: ");
    usn = scanner.next();
    System.out.print("Enter name: ");
    name = scanner.next();
}
void GetMarks(){
    for(int i=0; i<8; i++){
        System.out.println("Enter marks for" + (i+1) +
                           " Subject #: ");
        subject[i].SubjectMarks = scanner.nextInt();
        System.out.println("Enter credits for" + (i+1)+
                           "Subject: ");
        subject[i].credits = scanner.nextInt()
        subject[i].CalculateGrade();
    }
}

void SGPA(){
    double score = 0;
    int total_credits = 0;
    for(int i=0; i<8; i++){
        score = score + (subject[i].credit *
                         subject[i].grade);
        total_credits += subject[i].credits;
    }
    if(total_credits > 0){
        SGPA = score / total_credits;
    } else{
        SGPA = 0;
    }
```

```
void Display(){
    System.out.print("USN: " + usn);
    System.out.print("Name: " + name);
    System.out.print("SGPA: " + SGPA);
}
}

public class StudDetails{
    public static void main(String[] args){
        Student[] students = new Student[3]

    for(j=0; j<3; j++){
        System.out.println("Enter details for
            student" + (j+1) "#");
        students[j] = new Student()
        ~~students[j].Get~~
        students[j].GetStudentDetails();
        students[j].GetMarks();
        students[j].SGPA();
    }
    for(int i=0; i<3; i++){
        students[j].display();
    }
}
```

Seen
execute

```
Enter details for student 1
Enter the USN: 1BM23CS016
Enter Name: Afreen
Enter marks for subject 1: 97
Enter credits for subject 1: 4
Enter marks for subject 2: 87
Enter credits for subject 2: 3
Enter marks for subject 3: 95
Enter credits for subject 3: 4
Enter marks for subject 4: 86
Enter credits for subject 4: 3
Enter marks for subject 5: 89
Enter credits for subject 5: 3
Enter marks for subject 6: 95
Enter credits for subject 6: 2
Enter marks for subject 7: 96
Enter credits for subject 7: 1
Enter marks for subject 8: 93
Enter credits for subject 8: 1
```

o/p seen
16/10/24

```
USN: 1BM23CS016
Name: Afreen
SGPA: 9.571428571
```

Code:

```
import java.util.Scanner;
class Subject {
    int subjectMarks;
    int credits;
    int grade;


    void calculateGrade() {
        if (subjectMarks >= 90) {
            grade = 10;
        } else if (subjectMarks >= 80) {
            grade = 9;
        } else if (subjectMarks >= 70) {
            grade = 8;
        } else if (subjectMarks >= 60) {
            grade = 7;
        } else if (subjectMarks >= 50) {
            grade = 6;
        } else if (subjectMarks >= 40) {
            grade = 5;
        } else {
            grade = 0; // Fail
        }
    }
}

class Student {
    String usn;
    String name;
    double SGPA;
    Subject[] subjects = new Subject[8];
    Scanner scanner = new Scanner(System.in);


    Student() {
        for (int i = 0; i < 8; i++) {
            subjects[i] = new Subject();
        }
    }


    void getStudentDetails() {
        System.out.print("Enter the USN: ");
        usn = scanner.next();
        System.out.print("Enter the Name: ");
```

```
        name = scanner.next();
    }
    void getMarks() {
        for (int i = 0; i < 8; i++) {
            System.out.print("Enter marks for subject " + (i + 1) + ": ");
            subjects[i].subjectMarks = scanner.nextInt();
            System.out.print("Enter credits for subject " + (i + 1) + ": ");
            subjects[i].credits = scanner.nextInt();
            subjects[i].calculateGrade();
        }
    }


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

        for (int i = 0; i < 8; i++) {
            effectiveScore += (subjects[i].grade * subjects[i].credits);
            totalCredits += subjects[i].credits;
        }
        SGPA = (totalCredits > 0) ? (effectiveScore / totalCredits) : 0;
    }


    void display() {
        System.out.println("USN: " + usn);
        System.out.println("Name: " + name);
        System.out.println("SGPA: " + SGPA);
    }
}

public class StudDetails {
    public static void main(String[]args) {
        Student[] students = new Student[3];

        for (int j = 0; j < 3; j++) {
            System.out.println("Enter the details for student " + (j + 1) + ":");
            students[j] = new Student();
            students[j].getStudentDetails();
            students[j].getMarks();
            students[j].computeSGPA();
        }

        for (Student student : students) {
            student.display();
        }
```

```
    }
}
```

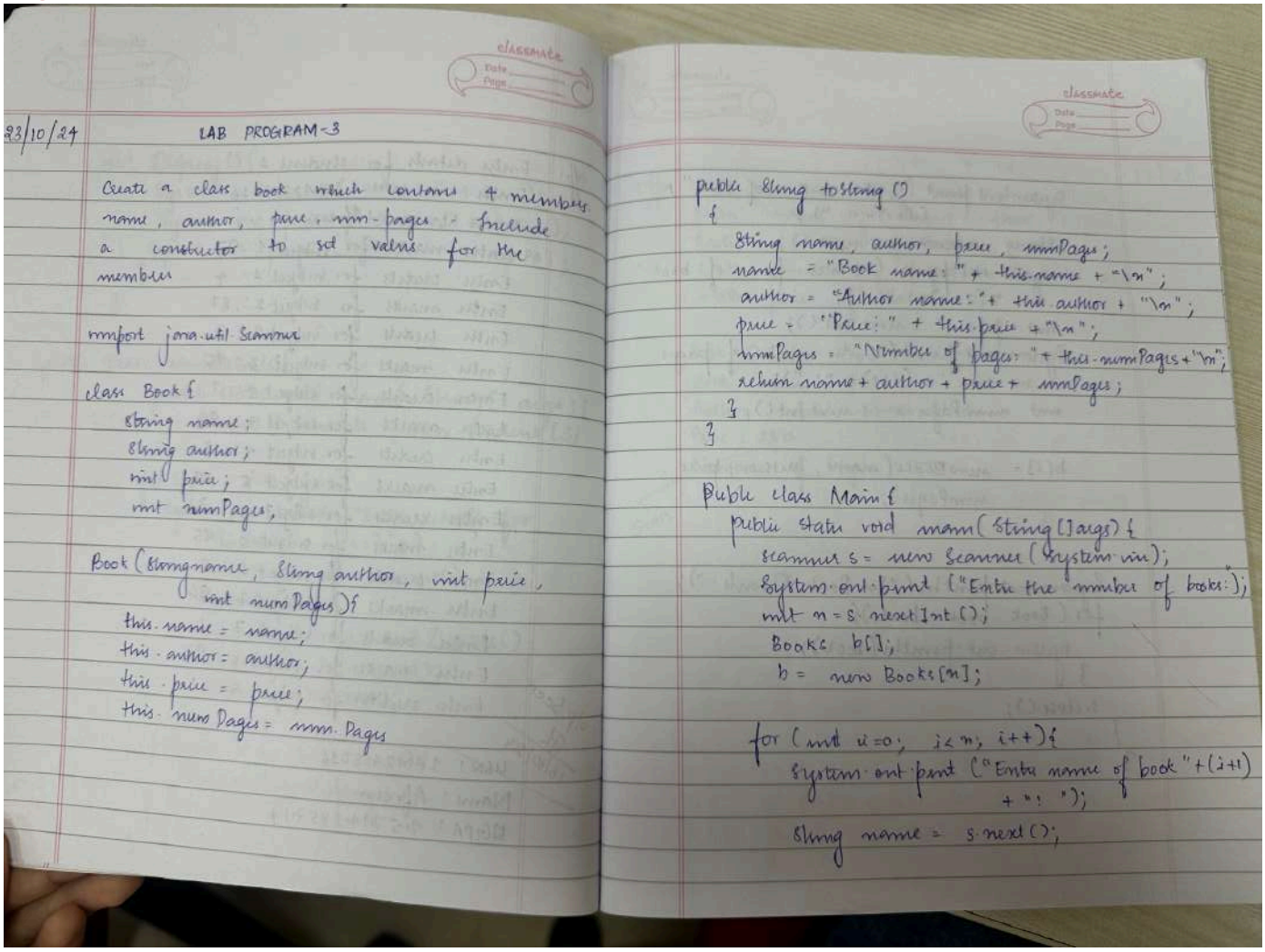
```
D:\usn16>java StudDetails
Enter the details for student 1:
Enter the USN: 1BM23CS016
Enter the Name: Afreen
Enter marks for subject 1: 97
Enter credits for subject 1: 4
Enter marks for subject 2: 87
Enter credits for subject 2: 3
Enter marks for subject 3: 95
Enter credits for subject 3: 4
Enter marks for subject 4: 86
Enter credits for subject 4: 3
Enter marks for subject 5: 89
Enter credits for subject 5: 3
Enter marks for subject 6: 95
Enter credits for subject 6: 2
Enter marks for subject 7: 96
Enter credits for subject 7: 1
Enter marks for subject 8: 93
Enter credits for subject 8: 1
Enter the details for student 2:
Enter the USN: 1BM23CS017
Enter the Name: Aisha
Enter marks for subject 1: 93
Enter credits for subject 1: 4
Enter marks for subject 2: 92
Enter credits for subject 2: 4
Enter marks for subject 3: 81
Enter credits for subject 3: 3
Enter marks for subject 4: 83
Enter credits for subject 4: 3
Enter marks for subject 5: 99
Enter credits for subject 5: 3
Enter marks for subject 6: 98
Enter credits for subject 6: 2
Enter marks for subject 7: 91
Enter credits for subject 7: 1
Enter marks for subject 8: 83
Enter credits for subject 8: 1
```

```
Enter the details for student 3:
Enter the USN: 1BM23CS047
Enter the Name: Aparna
Enter marks for subject 1: 92
Enter credits for subject 1: 4
Enter marks for subject 2: 93
Enter credits for subject 2: 4
Enter marks for subject 3: 85
Enter credits for subject 3: 3
Enter marks for subject 4: 81
Enter credits for subject 4: 3
Enter marks for subject 5: 89
Enter credits for subject 5: 3
Enter marks for subject 6: 94
Enter credits for subject 6: 2
Enter marks for subject 7: 99
Enter credits for subject 7: 1
Enter marks for subject 8: 85
Enter credits for subject 8: 1
USN: 1BM23CS016
Name: Afreen
SGPA: 9.571428571428571
USN: 1BM23CS017
Name: Aisha
SGPA: 9.666666666666666
USN: 1BM23CS047
Name: Aparna
SGPA: 9.523809523809524

D:\usn16>
```

Program 3

Constructors and methods (Book Class)

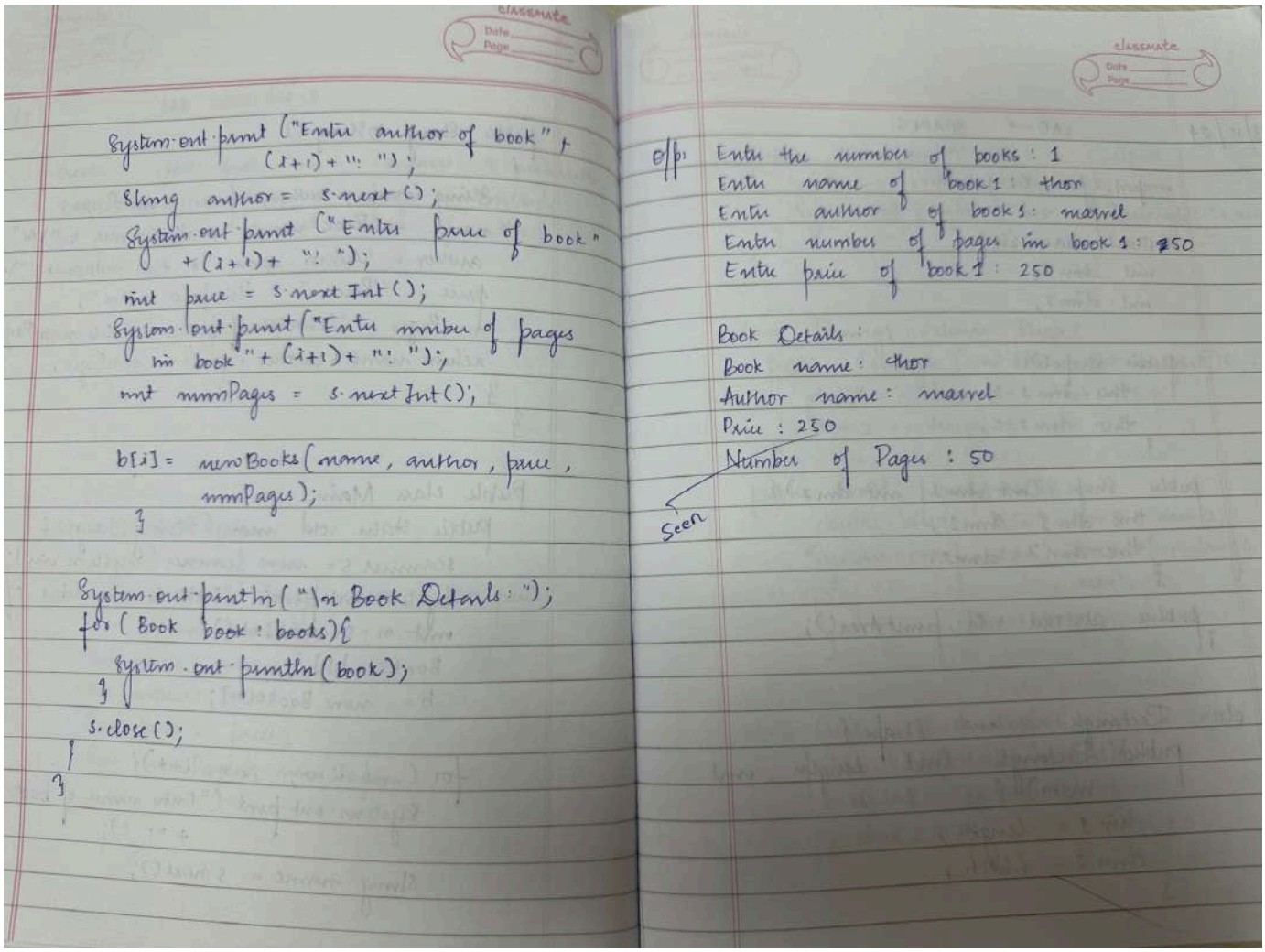
Algorithm:



23/10/24 LAB PROGRAM-3

Create a class book which contains 4 members name, author, price, num_pages. Include a constructor to set values for the members

```
import java.util.Scanner;

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

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

Public class Main{
    public static void main(String[] args){
        Scanner s = new Scanner(System.in);
        System.out.print("Enter the number of books: ");
        int n = s.nextInt();
        Book b[];
        b = new Book[n];

        for(int i=0; i<n; i++){
            System.out.print("Enter name of book " + (i+1)
                    + ": ");
            String name = s.next();
```



Code:

```
import java.util.Scanner;


class Book {
    // Class members
    String name;
    String author;
    int price;
    int numPages;


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

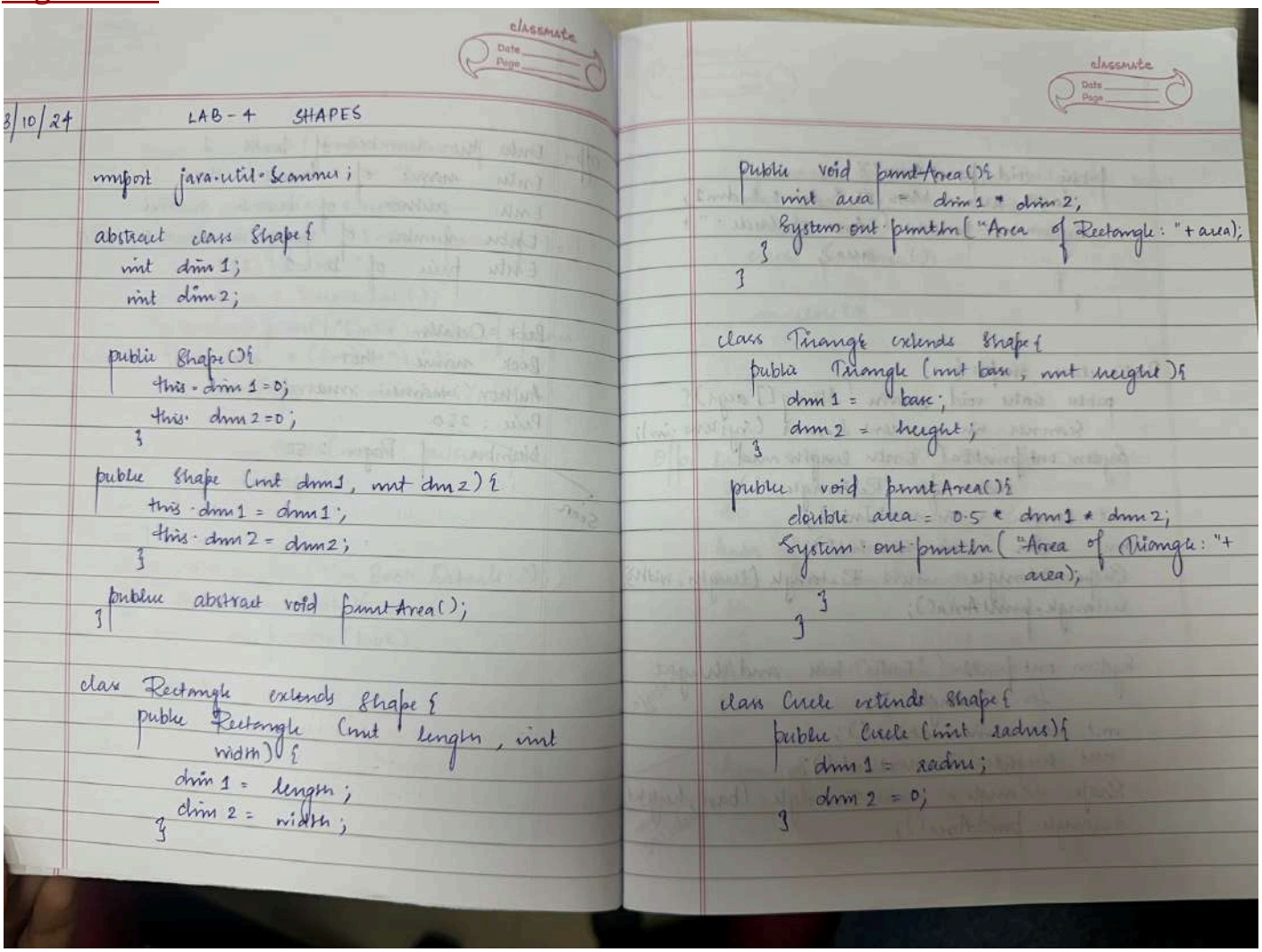
```
    public String toString() {
        String bookDetails = "Book name: " + this.name + "\n" +
                             "Author name: " + this.author + "\n" +
                             "Price: " + this.price + "\n" +
                             "Number of pages: " + this.numPages + "\n";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

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

        Scanner s = new Scanner(System.in);

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

        Book[] books = new Book[n];

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

            System.out.print("Enter name of book " + (i + 1) + ": ");
            String name = s.next();
            System.out.print("Enter author of book " + (i + 1) + ": ");
            String author = s.next();
            System.out.print("Enter price of book " + (i + 1) + ": ");
            int price = s.nextInt();
            System.out.print("Enter number of pages in book " + (i + 1) + ": ");
            int numPages = s.nextInt();


            books[i] = new Book(name, author, price, numPages);
        }


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

        s.close();
System.out.println("Name : Afreen Anz\n1BM23CS016");
    }

}
```

```
D:\cs3a>java Main
Enter the number of books: 3
Enter name of book 1: thor
Enter author of book 1: afreen
Enter price of book 1: 250
Enter number of pages in book 1: 40
Enter name of book 2: iron
Enter author of book 2: afeefah
Enter price of book 2: 200
Enter number of pages in book 2: 50
Enter name of book 3: amal
Enter author of book 3: captain
Enter price of book 3: 300
Enter number of pages in book 3: 60

Book Details:
Book name: thor
Author name: afreen
Price: 250
Number of pages: 40

Book name: iron
Author name: afeefah
Price: 200
Number of pages: 50

Book name: amal
Author name: captain
Price: 300
Number of pages: 60

Name : Afreen Anz
1BM23CS016

D:\cs3a>
```

Program 4

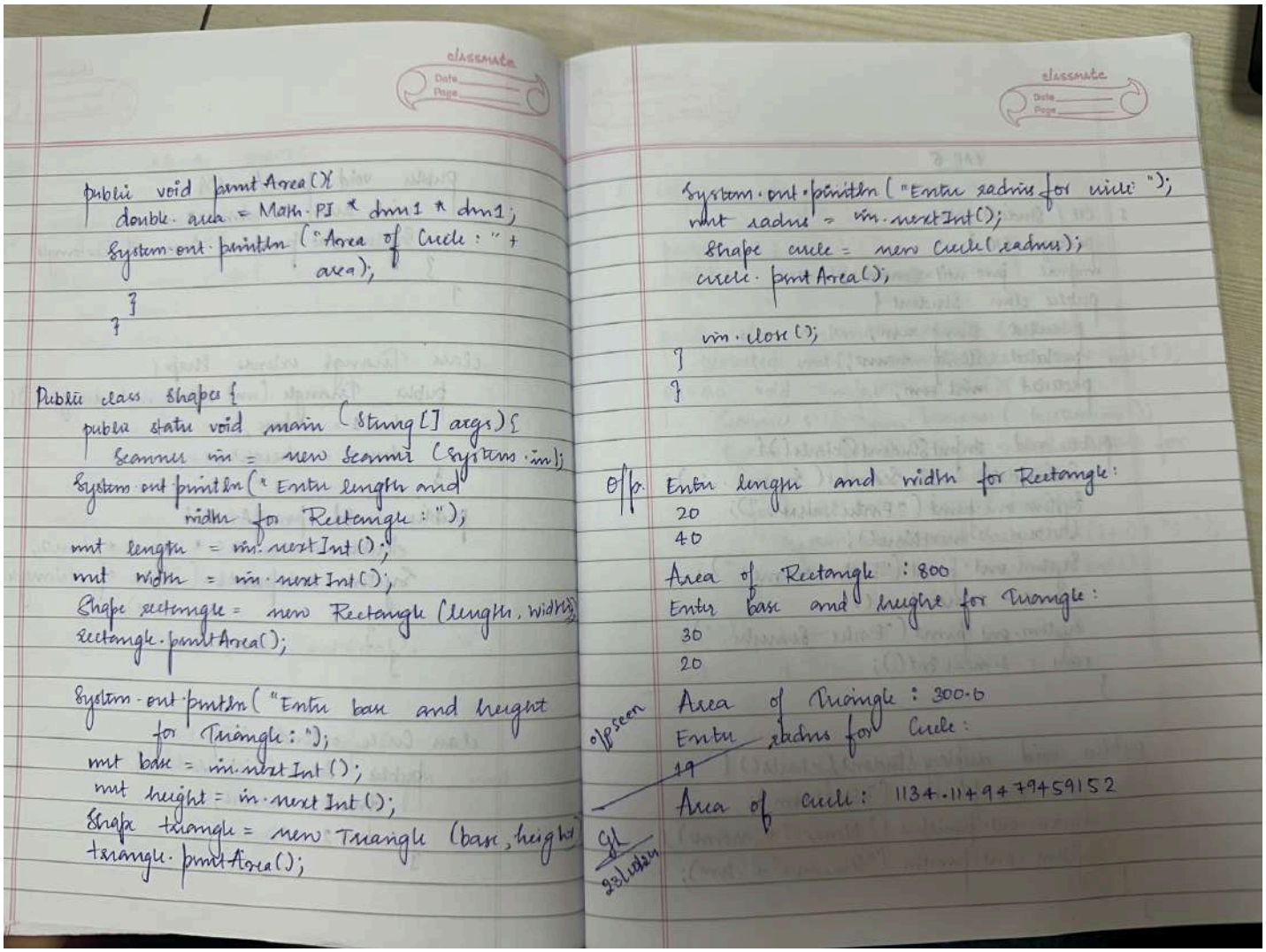
Abstract Class (Shape Class)

Algorithm:



3/10/24 LAB - 4 SHAPES

```
import java.util.Scanner;
abstract class Shape{
    int dim1;
    int dim2;

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

class Rectangle extends Shape{
    public Rectangle(int length, int
            width){
        dim1 = length;
        dim2 = width;
    }
    public void printArea(){
        int area = dim1 * dim2;
        System.out.println("Area of Rectangle: "+area);
    }
}

class Triangle extends Shape{
    public Triangle(int base, int height){
        dim1 = base;
        dim2 = height;
    }
    public void printArea(){
        double area = 0.5 * dim1 * dim2;
        System.out.println("Area of Triangle: "+
                area);
    }
}

class Circle extends Shape{
    public Circle(int radius){
        dim1 = radius;
        dim2 = 0;
    }
```



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

Public class Shapes {
    public static void main(String[] args){
    Scanner in = new Scanner(System.in);
    System.out.println("Enter length and
                       width for Rectangle:");
    int length = in.nextInt();
    int width = in.nextInt();
    Shape rectangle = new Rectangle(length, width);
    rectangle.printArea();

    System.out.println("Enter base and height
                       for Triangle:");
    int base = in.nextInt();
    int height = in.nextInt();
    Shape triangle = new Triangle(base, height);
    triangle.printArea();

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

    in.close();
  }
}
```

```
O/p: Enter length and width for Rectangle:
20
40
Area of Rectangle: 800
Enter base and height for Triangle:
30
20
Area of Triangle: 300.0
Enter radius for Circle:
19
Area of Circle: 1134.1149479459152
```

Code:

```
import java.util.Scanner;

abstract class Shape {
   int dim1;
   int dim2;


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

   }

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

```
    }

    public abstract void printArea();

}

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

    public void printArea() {

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

    }
}

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

    public void printArea() {

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

    }
}

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

        dim1 = radius;
        dim2 = 0;
    }

    public void printArea() {

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

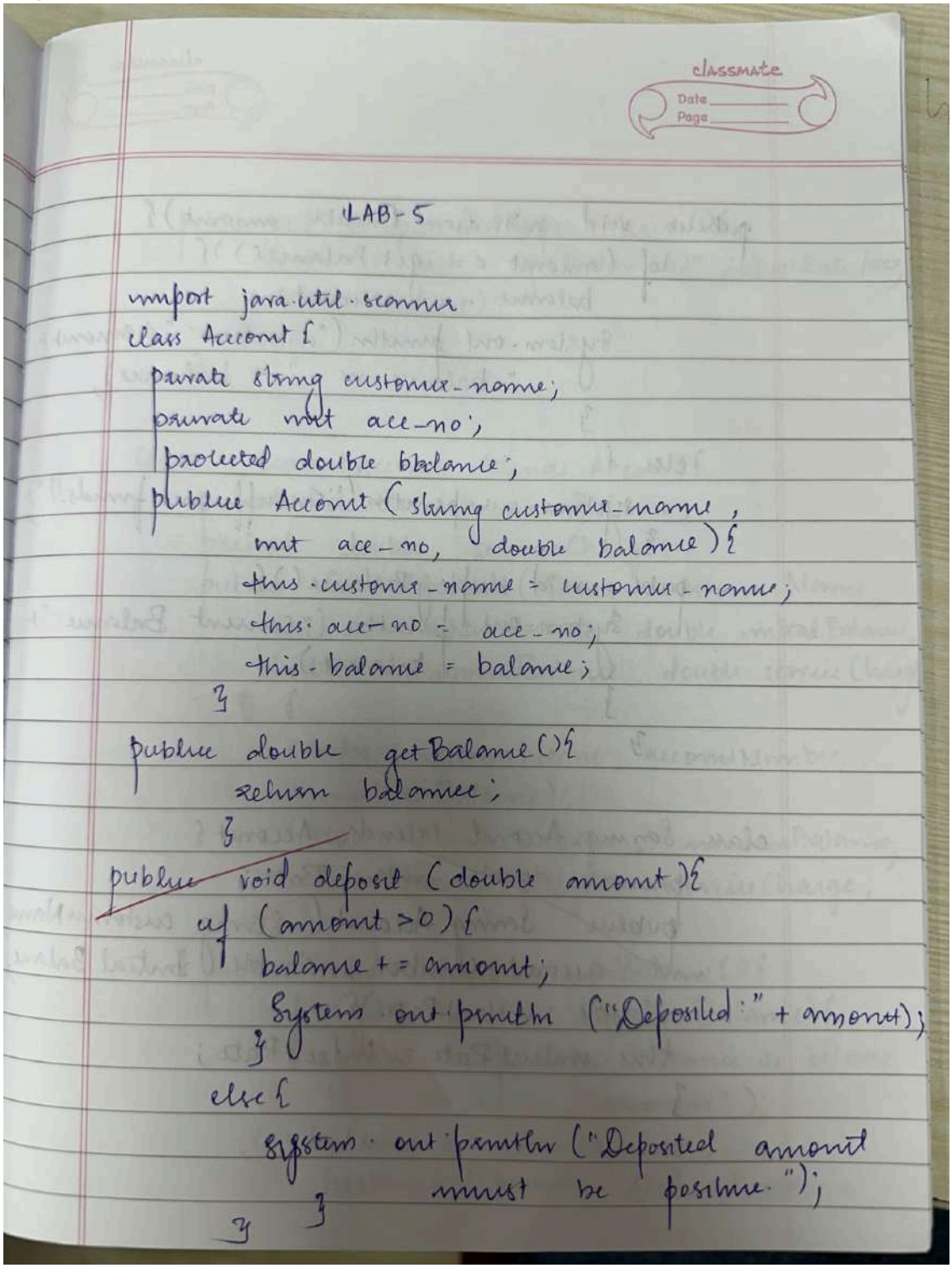
```
public class shapes {

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

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

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

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

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

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

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

        in.close();
System.out.println("Name : Afreen Anz\n1BM23CS016");
    }
}
```

```
D:\cs3a>javac shapes.java

D:\cs3a>java shapes
Enter length and width for Rectangle:
20
40
Area of Rectangle: 800
Enter base and height for Triangle:
30
20
Area of Triangle: 300.0
Enter radius for Circle:
19
Area of Circle: 1134.1149479459152
Name : Afreen Anz
1BM23CS016

D:\cs3a>
```

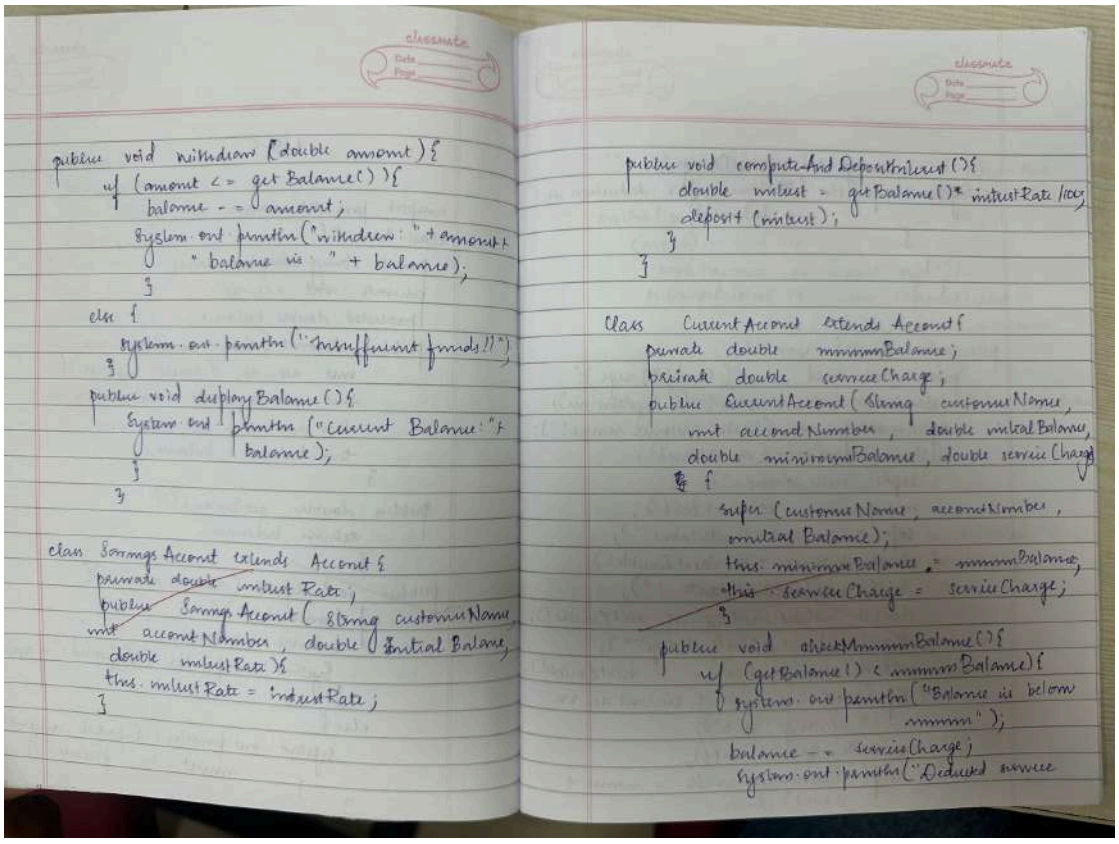
Program 5

Inheritance (Bank Class)

Algorithm:

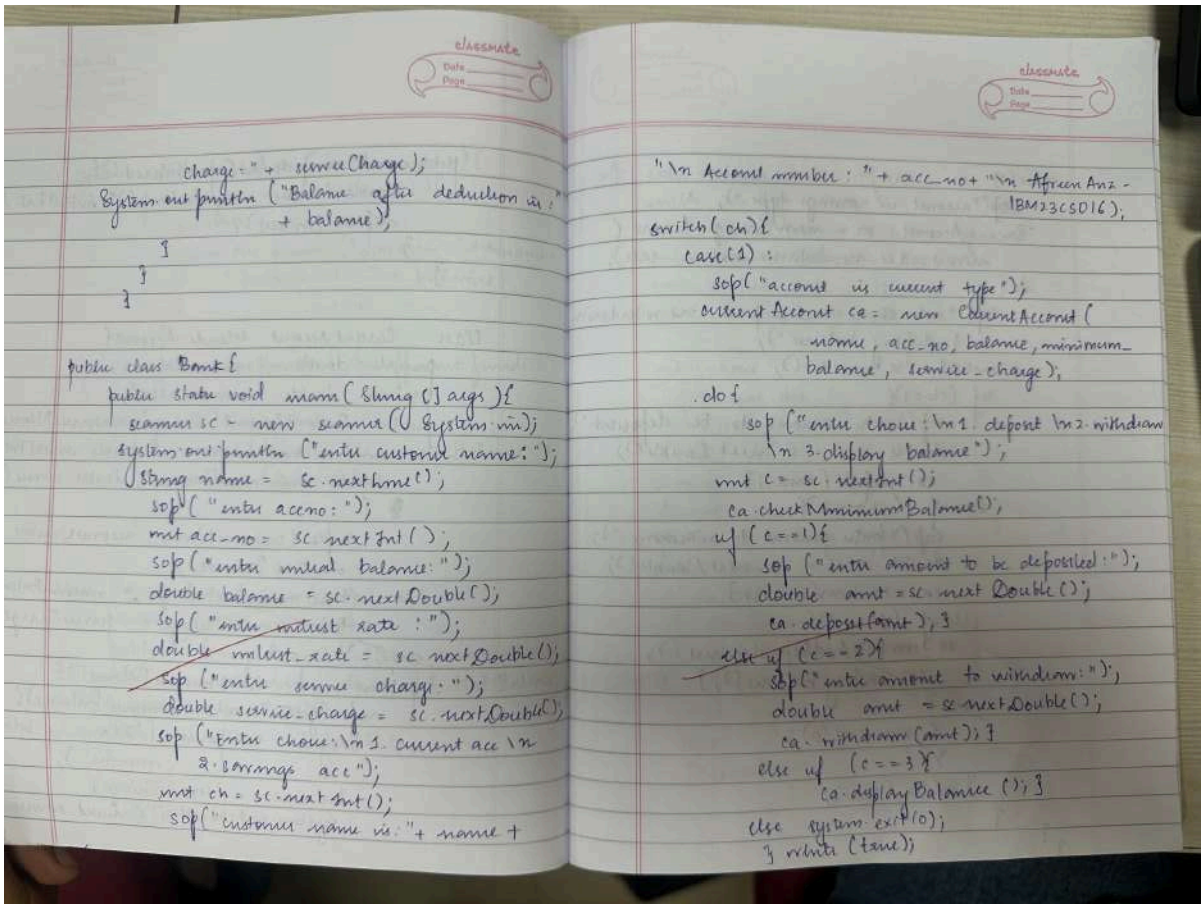


```
LAB-5

import java.util.scanner
class Account {
    private string customer_name;
    private int acc_no;
    protected double bbalance;
    public Account (string customer_name,
            int acc_no, double balance) {
        this.customer_name = customer_name;
        this.acc_no = acc_no;
        this.balance = balance;
    }

public double getBalance(){
        return balance;
    }

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



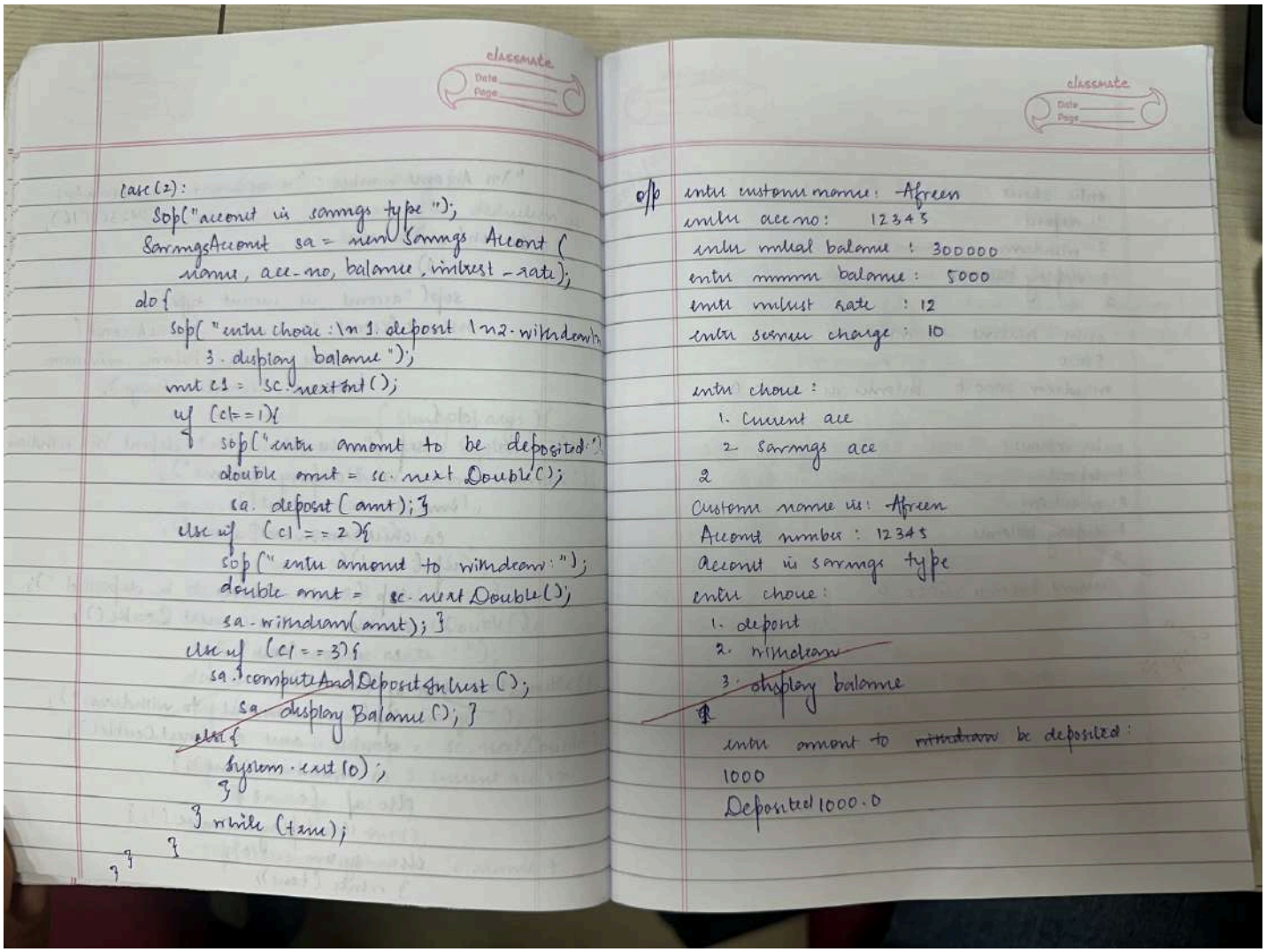
```
public void withdraw (double amount){
    if (amount <= getBalance()){
        balance -= amount;
        System.out.println("withdrew: " + amount +
            " balance is: " + balance);
    }
    else {
        System.out.println("Insufficient funds!!");
    }
    public void displayBalance(){
        System.out.println("Current Balance: " +
            balance);
    }
}

class SavingsAccount extends Account {
    private double interestRate;
    public SavingsAccount(String customerName,
    int accountNumber, double initialBalance,
    double interestRate){
    this.interestRate = interestRate;
}
```

```
public void computeAndDepositInterest(){
    double interest = getBalance()* interestRate/100;
    deposit(interest);
}
}

class CurrentAccount extends Account{
    private double minimumBalance;
    private double serviceCharge;
    public CurrentAccount(String customerName,
        int accountNumber, double initialBalance,
        double minimumBalance, double serviceCharge)
    {
        super(customerName, accountNumber,
        initialBalance);
        this.minimumBalance = minimumBalance;
        this.serviceCharge = serviceCharge;
    }
    public void checkMinimumBalance(){
        if (getBalance() < minimumBalance){
        System.out.println("Balance is below
                minimum");
        balance -= serviceCharge;
        System.out.println("Deducted service
```



```
        charge:" + serviceCharge);
    System.out.println("Balance after deduction is:"
                + balance);
        }
    }
}

public class Bank{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        System.out.println("enter customer name:");
        String name = sc.nextLine();
        sop("enter accno:");
        int acc_no = sc.nextInt();
        sop("enter initial balance:");
        double balance = sc.nextDouble();
        sop("enter interest rate:");
        double interest_rate = sc.nextDouble();
        sop("enter service charge:");
        double service_charge = sc.nextDouble();
        sop("Enter choice:\n1. current acc\n
            2. Savings acc");
        int ch = sc.nextInt();
        sop("customer name is:" + name +
```

```
"\n Account number:" + acc_no + "\n African Anz -
                IBM23CSD16");
switch(ch){
    case(1):
        sop("account is current type");
        CurrentAccount ca = new CurrentAccount(
                name, acc_no, balance, minimum_
                balance, service_charge);
        .do{
            sop("enter choice:\n1. deposit\n2. withdraw
                \n3. display balance");
        int c = sc.nextInt();
        ca.checkMinimumBalance();
        if(c==1){
            sop("enter amount to be deposited:");
            double amt = sc.nextDouble();
            ca.deposit(amt);}
        else if(c==2){
            sop("enter amount to withdraw:");
            double amt = sc.nextDouble();
            ca.withdraw(amt);}
        else if(c==3){
            ca.displayBalance();}
        else system.exit(0);
        }while(true);
```



Code:

```
import java.util.Scanner;

class Account {
    private String customer_name;
    private int acc_no;
    protected double balance;

    public Account(String customer_name, int acc_no, double balance) {
        this.customer_name = customer_name;
        this.acc_no = acc_no;
        this.balance = balance;
    }

    public double getBalance() {
        return balance;
```

```
    }

    public void deposit(double amount) {
        if (amount > 0) {
            balance += amount;
            System.out.println("Deposited: " + amount);
        } else {
            System.out.println("Deposit amount must be positive.");
        }
    }
   public void withdraw(double amount)
     {
      if(amount<=getBalance()){
         balance-=amount;
         System.out.println("withdrew:"+amount + " balance is:"+ balance);
         }
       else
        System.out.println("Insufficient funds!!");
     }
    public void displayBalance(){
        System.out.println("Current Balance: " + balance);
    }
}

class SavingsAccount extends Account {
    private double interestRate;

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

    public void computeAndDepositInterest() {
        double interest = getBalance() * interestRate / 100;
        deposit(interest);
    }
}
class CurrentAccount extends Account {
    private double minimumBalance;
    private double serviceCharge;

    public CurrentAccount(String customerName, int accountNumber, double initialBalance, double
minimumBalance, double serviceCharge) {
        super(customerName, accountNumber, initialBalance);
        this.minimumBalance = minimumBalance;
        this.serviceCharge = serviceCharge;
```

```
    }
    public void checkMinimumBalance() {
        if (getBalance() < minimumBalance) {
            System.out.println("Balance is below minimum");
            balance-=serviceCharge;
            System.out.println("Deducted service charge:" +serviceCharge);
            System.out.println("Balance after deduction is:"+balance);
        }
    }
}
public class Bank {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("enter customer name:");
        String name=sc.nextLine();
        System.out.println("enter accno:");
        int acc_no=sc.nextInt();
        System.out.println("enter initial balance:");
        double balance=sc.nextDouble();
        System.out.println("enter minimum balance:");
        double minimum_balance=sc.nextDouble();
        System.out.println("enter interest rate:");
        double interest_rate=sc.nextDouble();
        System.out.println("enter service charge:");
        double service_charge=sc.nextDouble();
        System.out.println("Enter choice:\n 1.Current acc\n 2.Savings acc");
        int ch=sc.nextInt();
        System.out.println("Customer name is:"+ name+"\nAccount number:"+acc_no);
        switch(ch){
            case(1):
                System.out.println("account is current type");
                CurrentAccount ca = new
CurrentAccount(name,acc_no,balance,minimum_balance,service_charge);
               do{ System.out.println("enter choice:\n 1.deposit\n 2.withdraw\n 3.display balance");
                int c=sc.nextInt();
                ca.checkMinimumBalance();
                if(c==1){
                   System.out.println("enter amount to be deposited:");
                   double amt=sc.nextDouble();
                     ca.deposit(amt);}
                else if(c==2){
                   System.out.println("enter amount to withdraw:");
                   double amt=sc.nextDouble();
                   ca.withdraw(amt);}
                else if(c==3){
                   ca.displayBalance();}
                else
```

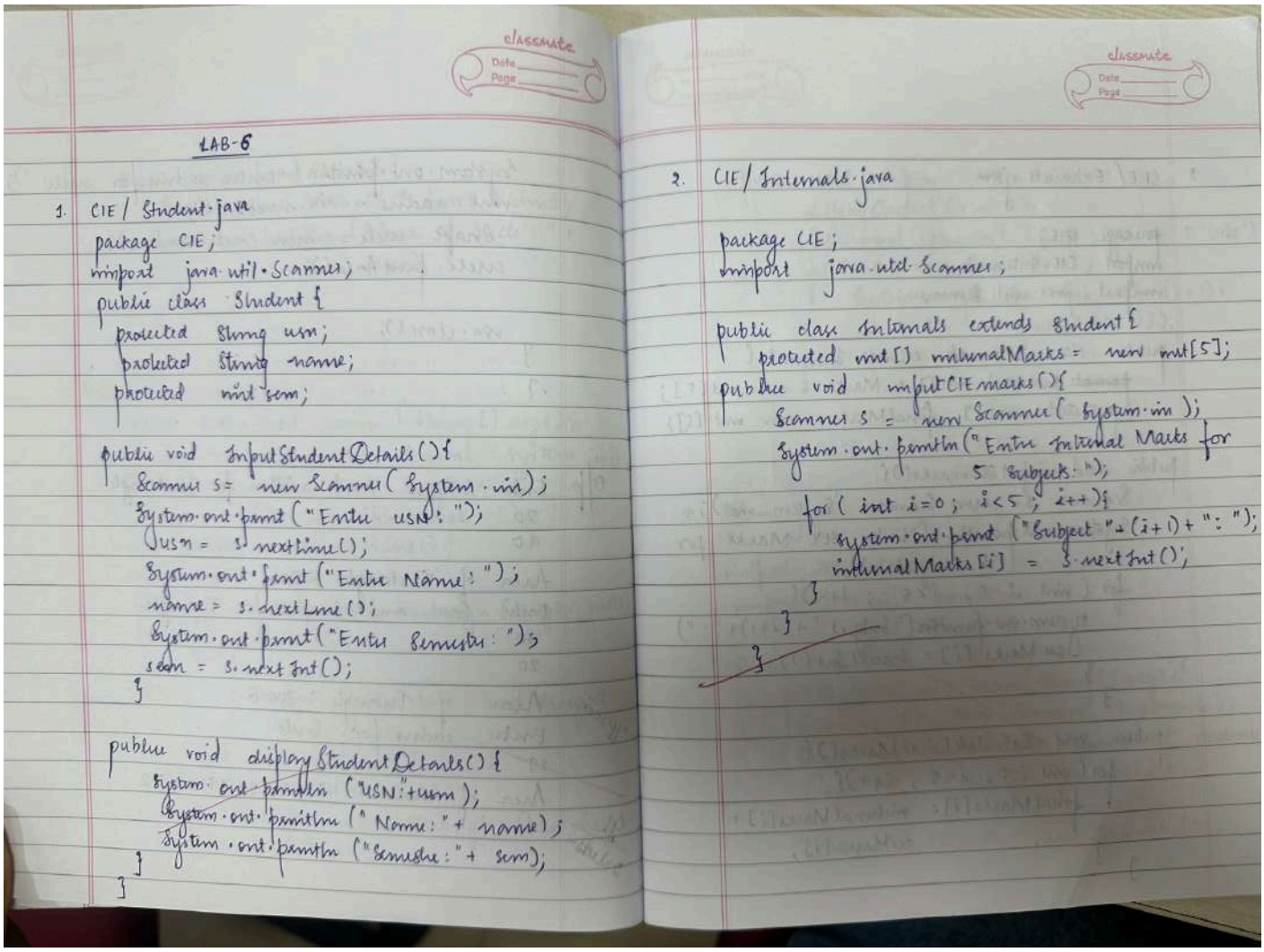
```
            System.exit(0);
           }while(true);

        case(2):
            System.out.println("account is savings type");
           SavingsAccount sa=new SavingsAccount(name,acc_no,balance,interest_rate);
           do{ System.out.println("enter choice:\n 1.deposit\n 2.withdraw\n 3.display balance");
           int c1=sc.nextInt();
           if(c1==1){
              System.out.println("enter amount to be deposited:");
              double amt=sc.nextDouble();
                sa.deposit(amt);}
           else if(c1==2){
              System.out.println("enter amount to withdraw:");
              double amt=sc.nextDouble();
              sa.withdraw(amt);}
           else if(c1==3){
            sa.computeAndDepositInterest();
              sa.displayBalance();}
           else{
             System.exit(0);
                }
           }while(true);
    }
}
}
```

```
D:\16>java Bank
enter customer name:
Afreen
enter accno:
242424
enter initial balance:
10000
enter minimum balance:
2000
enter interest rate:
12
enter service charge:
10
Enter choice:
 1.Current acc
 2.Savings acc
1
Customer name is:Afreen
Account number:242424
account is current type
enter choice:
 1.deposit
 2.withdraw
 3.display balance
1
enter amount to be deposited:
1000
Deposited: 1000.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
3
Current Balance: 11000.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
2
enter amount to withdraw:
5000
withdrew:5000.0 balance is:6000.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
3
Current Balance: 6000.0
```

```
enter choice:
 1.deposit
 2.withdraw
 3.display balance
3
Current Balance: 6000.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
2
enter amount to withdraw:
5000
withdrew:5000.0 balance is:1000.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
3
Balance is below minimum
Deducted service charge:10.0
Balance after deduction is:990.0
Current Balance: 990.0
enter choice:
 1.deposit
 2.withdraw
 3.display balance
5
Balance is below minimum
Deducted service charge:10.0
Balance after deduction is:980.0
```

Program 6
Packages (Student Marks)

Algorithm:



LAB-6

1. CIE / Student.java

```
package CIE;
import java.util.Scanner;
public class Student {
    protected String usn;
    protected String name;
    protected int sem;

public void inputStudentDetails(){
    Scanner s= new Scanner( System.in);
    System.out.print("Enter USN: ");
    usn = s.nextLine();
    System.out.print("Enter Name: ");
    name = s.nextLine();
    System.out.print("Enter Semester: ");
    sem = s.nextInt();
}

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

2. CIE / Internals.java

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

public class Internals extends Student {
    protected int[] internalMarks = new int[5];
public void inputCIEmarks(){
    Scanner s = new Scanner( System.in );
    System.out.println("Enter Internal Marks for
                        5 subjects:");
    for( int i=0; i<5; i++){
        System.out.print("Subject "+(i+1)+": ");
        internalMarks[i] = s.nextInt();
    }
}
}
```

3. SEE/ Externals.java

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

public class Externals extends Internals {
    protected int[] seeMarks = new int[5];
    protected int[] finalMarks = new int[5];

public void inputSEEmarks(){
    Scanner s = new Scanner(System.in);
    System.out.println("Enter SEE Marks for
                5 subjects:");
    for (int i=0; i<5 ; i++){
        System.out.println("Subject "+(i+1)+": ")
        seeMarks[i] = s.nextInt();
    }
}

public void calculateFinalMarks(){
    for(int i=0; i<5 ; i++){
        finalMarks[i] = internalMarks[i] +
                        seeMarks[i];
    }
}
```

```
public void displayFinalMarks(){
    displayStudentDetails();
    System.out.println("Final Marks for 5 subs");
    for(int i=0 ; i<5 ; i++){
        System.out.println("Subject "+(i+1)+
                ": "+finalMarks[i]);
    }
}
}
```

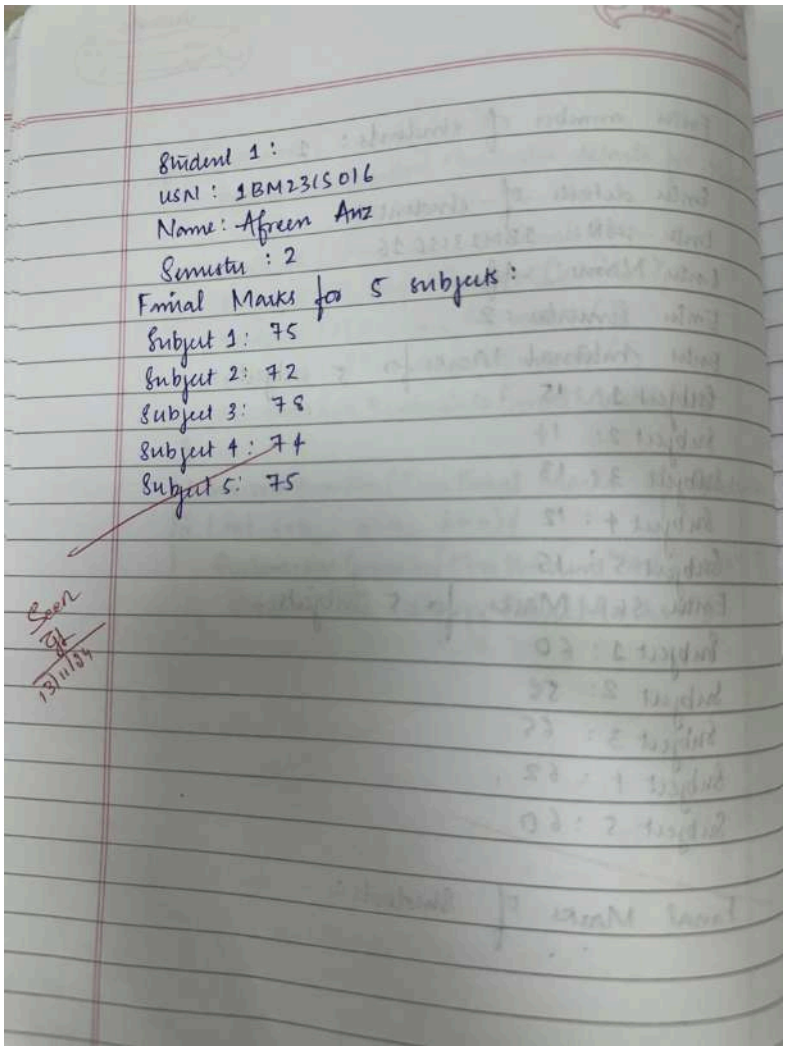
4. Main.java

```
import SEE.Externals;
import java.util.Scanner;

Class Main {
    public static void main(String[] args){
    Scanner s = new Scanner(System.in);
    System.out.print("Enter number of students:");
    int n = s.nextInt();

    Externals[] students = new Externals[n];
```

```
for (int i=0 ; i<n; i++){
    System.out.println("\nEnter details for student
            +(i+1)+": ");
    students[i] = new Externals();
    students[i].inputStudentDetails();
    students[i].inputCIEmarks();
    students[i].inputSEEmarks();
    students[i].calculateFinalMarks();
}
System.out.println("\n Final Marks of Students:");
for(int i=0; i<n; i++){
    System.out.println("\n Student"+(i+1)+":");
    students[i].displayFinalMarks();
}
}
}
```

O/P:

```
Enter number of students: 1

Enter details of student 1:
Enter USN: 1BM23CSD16
Enter Name: Afreen
Enter Semester: 2
Enter Internal Marks for 5 subjects:
Subject 1: 15
Subject 2: 14
Subject 3: 13
Subject 4: 12
Subject 5: 15
Enter SEE Marks for 5 subjects:
Subject 1: 60
Subject 2: 58
Subject 3: 65
Subject 4: 62
Subject 5: 60

Final Marks of Students:
```



Code:

Main

```
import SEE.Externals;
import java.util.Scanner;

class Main {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter number of students: ");
        int n = s.nextInt();

        Externals[] students = new Externals[n];

        for (int i = 0; i < n; i++) {
            System.out.println("\nEnter details for student " + (i + 1) + ":");
            students[i] = new Externals();
            students[i].inputStudentDetails();
            students[i].inputCIEmarks();
```

```
            students[i].inputSEEmarks();
            students[i].calculateFinalMarks();
        }

        System.out.println("\nFinal Marks of Students:");
        for (int i = 0; i < n; i++) {
            System.out.println("\nStudent " + (i + 1) + ":");
            students[i].displayFinalMarks();
        }
System.out.println("\n");
System.out.println("\nName: Afreen Anz\nUSN: 1BM23CS016");
    }
}
```

Externals

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

public class Externals extends Internals {
    private int[] seeMarks = new int[5];
    private int[] finalMarks = new int[5];

    public void inputSEEmarks() {
        Scanner s = new Scanner(System.in);
        System.out.println("Enter SEE Marks for 5 subjects:");
        for (int i = 0; i < 5; i++) {
            System.out.print("Subject " + (i + 1) + ": ");
            seeMarks[i] = s.nextInt();
        }
    }

    public void calculateFinalMarks() {
        for (int i = 0; i < 5; i++) {
            finalMarks[i] = internalMarks[i] + seeMarks[i];
        }
    }

    public void displayFinalMarks() {
        displayStudentDetails();
        System.out.println("Final Marks for 5 subjects:");
        for (int i = 0; i < 5; i++) {
            System.out.println("Subject " + (i + 1) + ": " + finalMarks[i]);
        }
    }
}
```

Internals

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

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

   public void inputCIEmarks() {
      Scanner s = new Scanner(System.in);
      System.out.println("Enter Internal Marks for 5 subjects:");
      for (int i = 0; i < 5; i++) {
         System.out.print("Subject " + (i + 1) + ": ");
         internalMarks[i] = s.nextInt();
      }
   }
}
```

Student

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

public class Student {
   protected String usn;
   protected String name;
   protected int sem;

   public void inputStudentDetails() {
      Scanner s = new Scanner(System.in);
      System.out.print("Enter USN: ");
      usn = s.nextLine();
      System.out.print("Enter Name: ");
      name = s.nextLine();
      System.out.print("Enter Semester: ");
      sem = s.nextInt();
   }

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

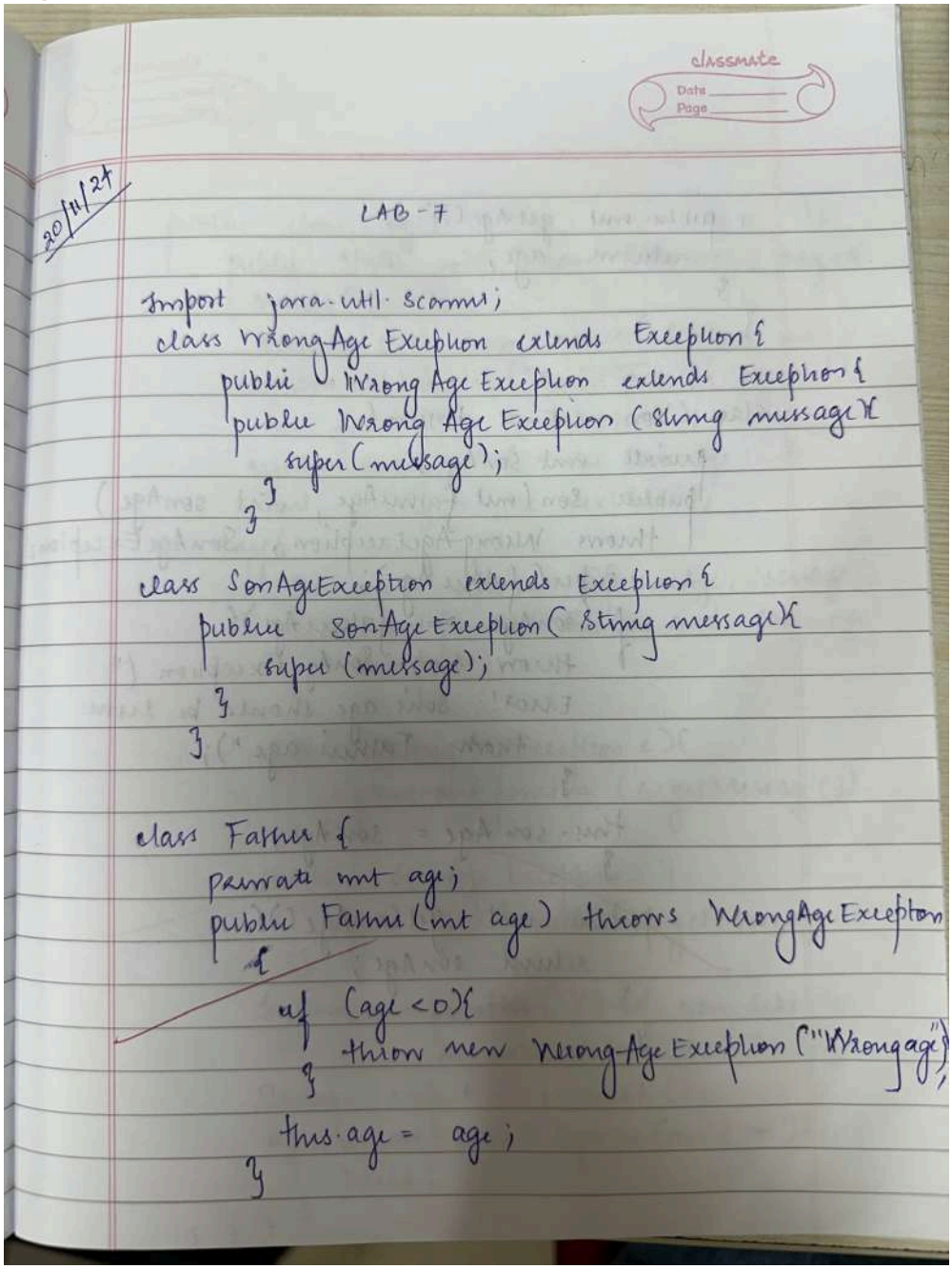
```
C:\Users\Admin\Desktop\1BM23CS016>java Main.java
Enter number of students: 3

Enter details for student 1:
Enter USN: 1BM23CS001
Enter Name: Afeefah
Enter Semester: 2
Enter Internal Marks for 5 subjects:
Subject 1: 12
Subject 2: 13
Subject 3: 15
Subject 4: 12
Subject 5: 15
Enter SEE Marks for 5 subjects:
Subject 1: 67
Subject 2: 68
Subject 3: 69
Subject 4: 70
Subject 5: 56

Enter details for student 2:
Enter USN: 1BM23CS002
Enter Name: Afreen
Enter Semester: 2
Enter Internal Marks for 5 subjects:
Subject 1: 15
Subject 2: 13
Subject 3: 13
Subject 4: 14
Subject 5: 15
Enter SEE Marks for 5 subjects:
Subject 1: 70
Subject 2: 69
Subject 3: 73
Subject 4: 65
Subject 5: 68

Enter details for student 3:
Enter USN: 1BM23CS003
Enter Name: Aaryan
Enter Semester: 2
Enter Internal Marks for 5 subjects:
Subject 1: 12
Subject 2: 12
Subject 3: 13
Subject 4: 13
Subject 5: 13
Enter SEE Marks for 5 subjects:
Subject 1: 67
Subject 2: 68
Subject 3: 68
Subject 4: 69
Subject 5: 70
```

```
Final Marks of Students:

Student 1:
USN: 1BM23CS001
Name: Afeefah
Semester: 2
Final Marks for 5 subjects:
Subject 1: 79
Subject 2: 81
Subject 3: 84
Subject 4: 82
Subject 5: 71

Student 2:
USN: 1BM23CS002
Name: Afreen
Semester: 2
Final Marks for 5 subjects:
Subject 1: 85
Subject 2: 82
Subject 3: 86
Subject 4: 79
Subject 5: 83

Student 3:
USN: 1BM23CS003
Name: Aaryan
Semester: 2
Final Marks for 5 subjects:
Subject 1: 79
Subject 2: 80
Subject 3: 81
Subject 4: 82
Subject 5: 83



Name: Afreen Anz
USN: 1BM23CS016

C:\Users\Admin\Desktop\1BM23CS016>
```

Program 7

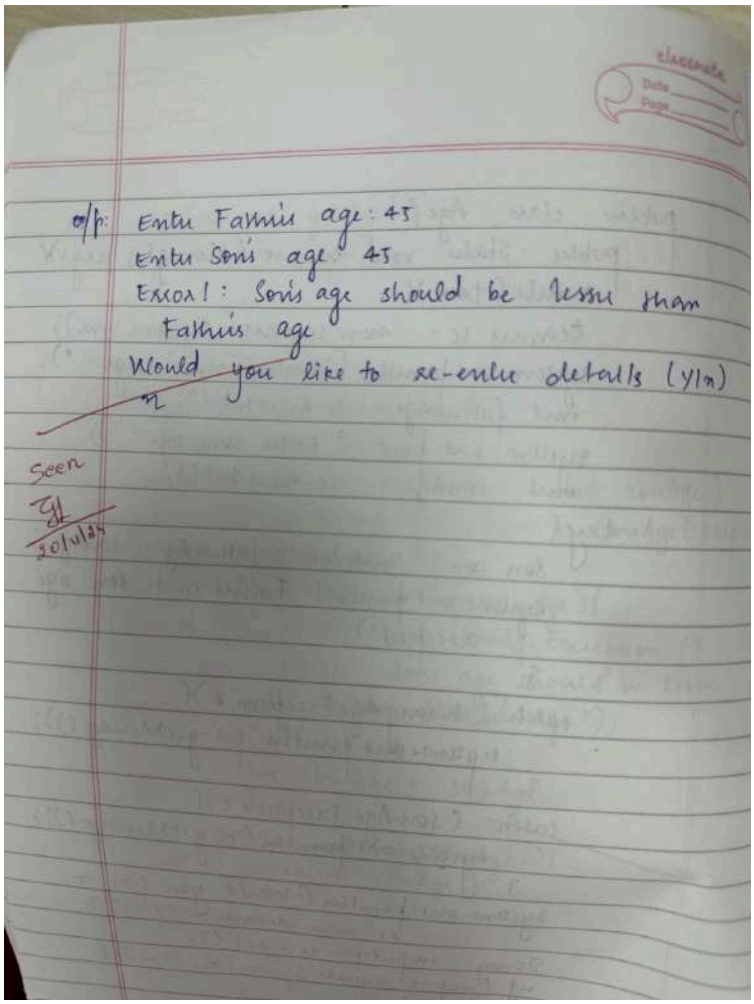
Exception Handling (Father-Son)

Algorithm::



```
    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("
            Error!: Son's age should be less
                than Father's age");
        }
        this.sonAge = sonAge;
    }
    public int getSonAge(){
        return sonAge;
    }
}
```

```
public class Age{
    public static void main(String[] args){
        while(true){
            Scanner sc = new Scanner(System.in);
            System.out.println("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("Father and son age
                accepted");
        }
        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;
    }}}}
```



Code:

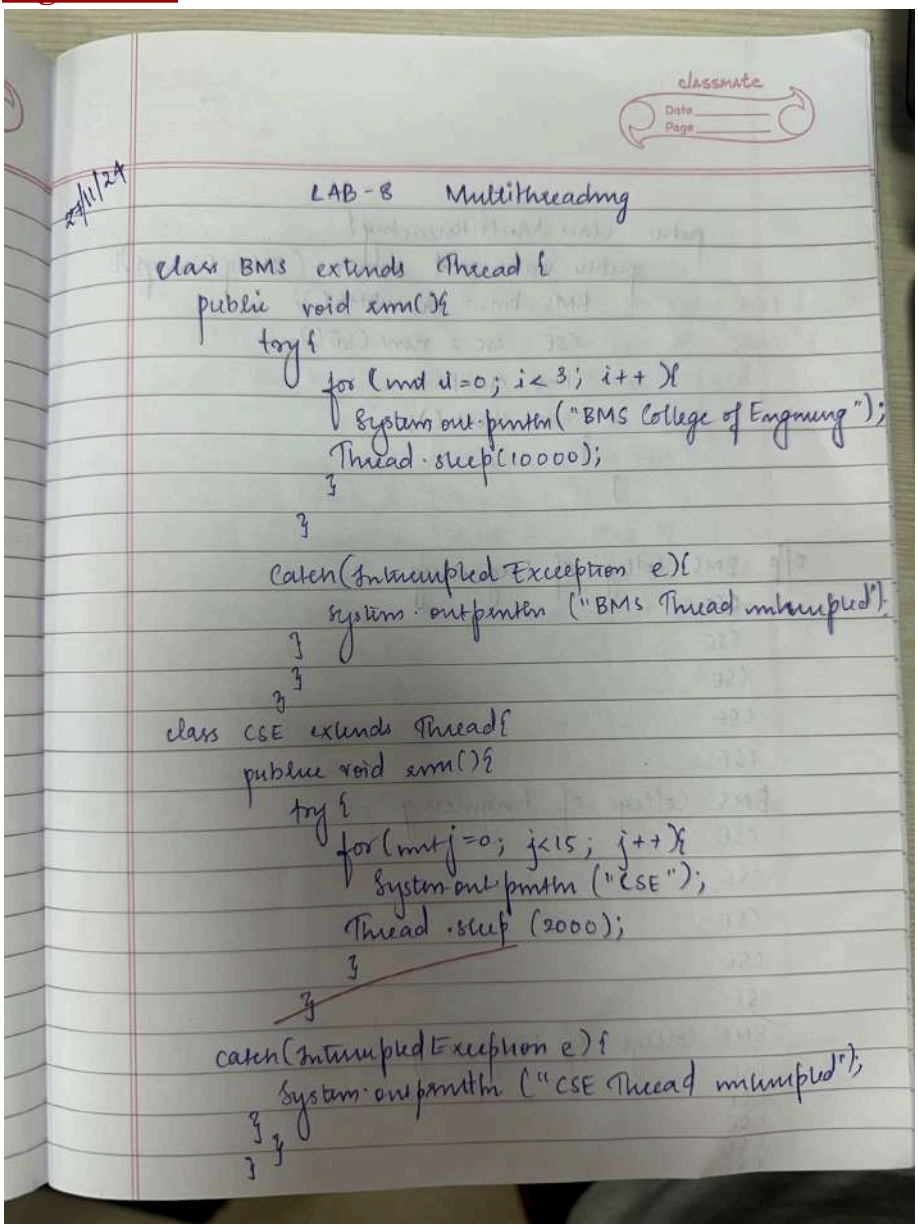
```
import java.util.Scanner;
class WrongAgeException extends Exception {
    public WrongAgeException(String message) {
        super(message);
    }
}

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

class Father {
    private int age;
    public Father(int age) throws WrongAgeException {
        if (age < 0) {
            throw new WrongAgeException("Wrong age");
```

```
        }
        this.age = age;
    }
    public int getAge() {
        return age;
    }
}

class Son extends Father {
    private int sonAge;
    public Son(int fatherAge, int sonAge) throws WrongAgeException, SonAgeException {
        super(fatherAge);
        if (sonAge >= fatherAge) {
            throw new SonAgeException("Error!: Son's age should be lesser than Father's age");
        }
        this.sonAge = sonAge;
    }
    public int getSonAge() {
        return sonAge;
    }
}
public class Age{
    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("Father and Son age accepted");
            }
            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;
            }
        }
    }
}
```

```
D:\16>cd Age

D:\16\Age>javac Age.java

D:\16\Age>java Age
Enter Father's Age: 45
Enter Son's Age: 19
Father and Son age accepted
Would you like to re-enter details (Y/n)
y
Enter Father's Age: 45
Enter Son's Age: 45
Error!: Son's age should be lesser than Father's age
Would you like to re-enter details (Y/n)
y
Enter Father's Age: 19
Enter Son's Age: 45
Error!: Son's age should be lesser than Father's age
Would you like to re-enter details (Y/n)
y
Enter Father's Age: 0
Enter Son's Age: 12
Error!: Son's age should be lesser than Father's age
Would you like to re-enter details (Y/n)
y
Enter Father's Age: 0
Enter Son's Age: 0
Error!: Son's age should be lesser than Father's age
Would you like to re-enter details (Y/n)
y
Enter Father's Age: -23
Enter Son's Age: -12
Wrong age
Would you like to re-enter details (Y/n)
n

D:\16\Age>
```

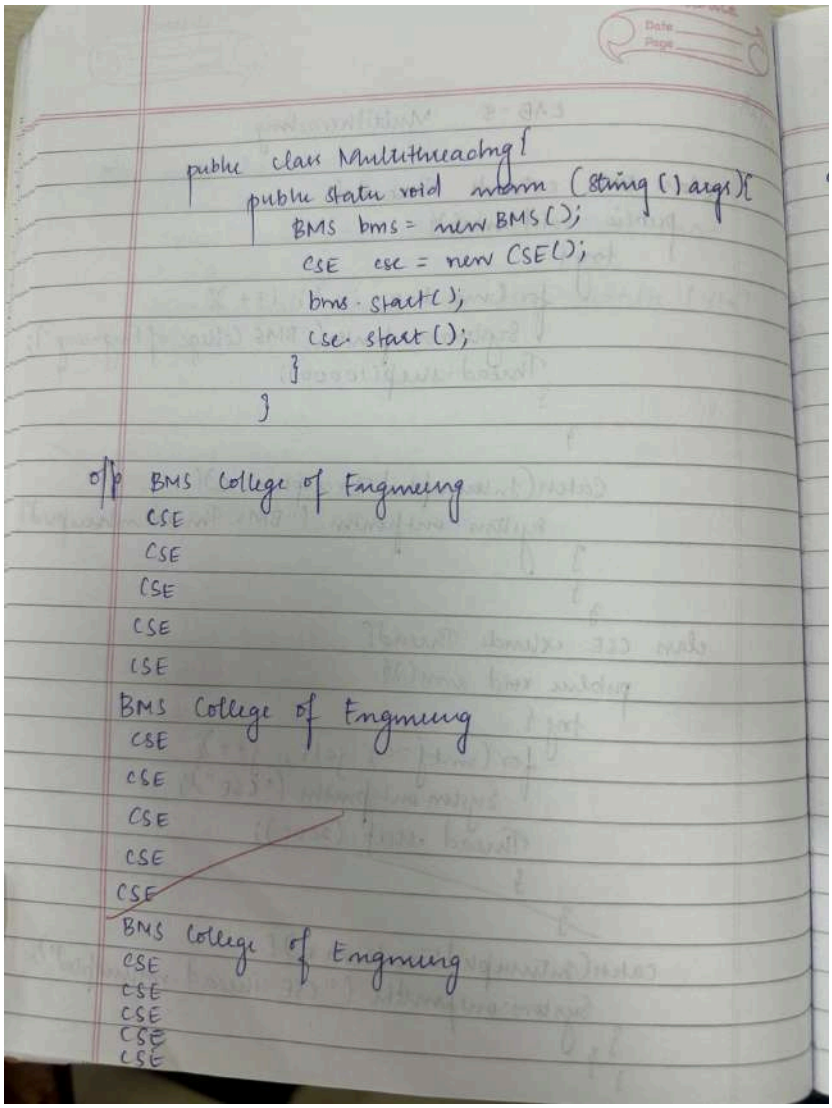
Program 8
Multithreading

Algorithm:



LAB - 8 Multithreading

```
class BMS extends Thread {
    public void run(){
        try{
            for (int i=0; i<3; i++){
                System.out.println("BMS College of Engineering");
                Thread.sleep(10000);
            }
        }
        catch(InterruptedException e){
            System.out.println("BMS Thread interrupted");
        }
    }
class CSE extends Thread{
    public void run(){
        try{
            for(int j=0; j<15; j++){
                System.out.println("CSE");
                Thread.sleep(2000);
            }
        }
        catch(InterruptedException e){
            System.out.println("CSE Thread interrupted");
        }
    }
}
```



Code:

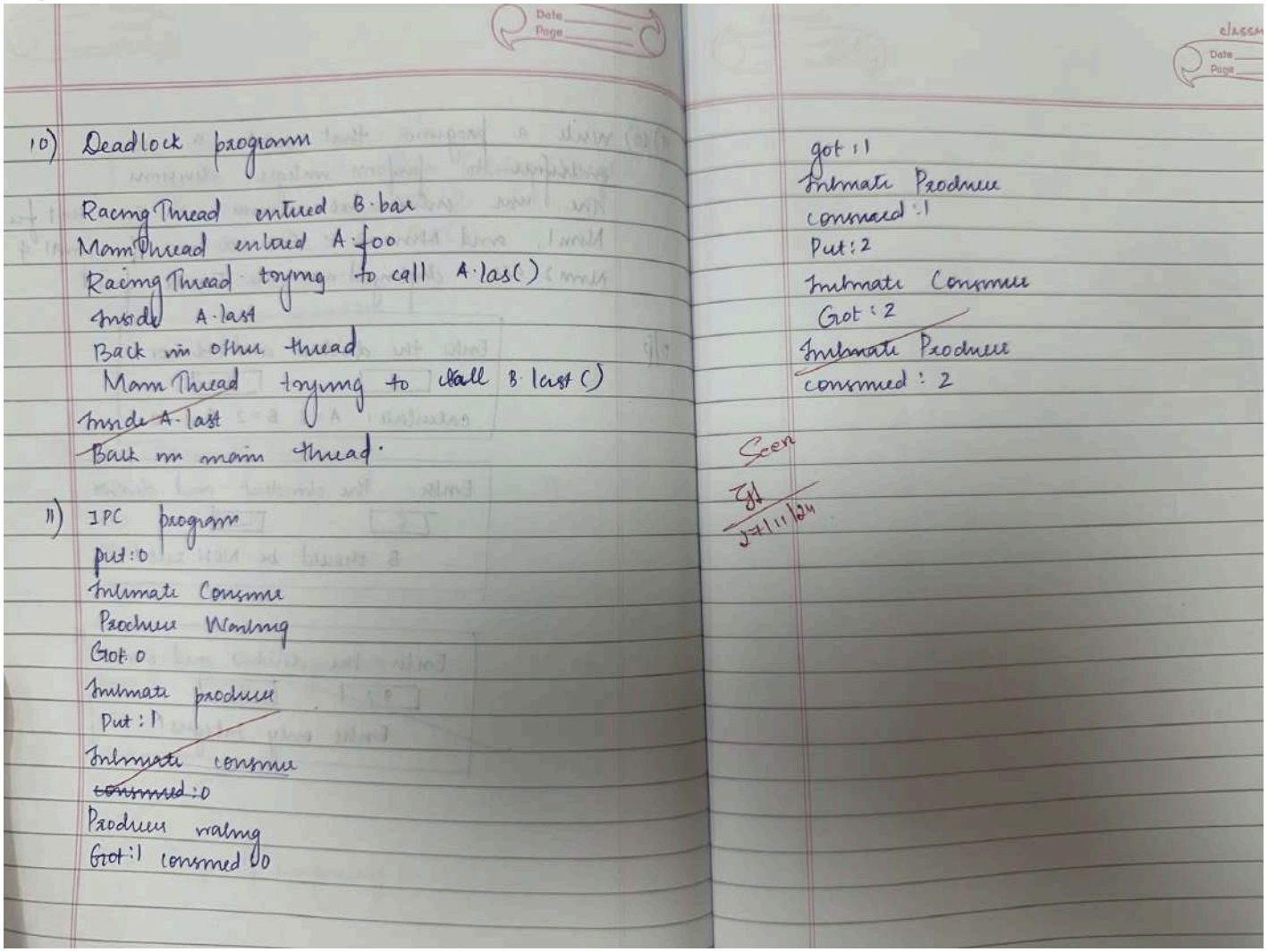
```
class BMS extends Thread {
   public void run() {
      try {
         for(int i=0;i<3;i++) {
            System.out.println("BMS College of Engineering");
            Thread.sleep(10000); // Sleep for 10 seconds
         }
      }
catch (InterruptedException e) {
  System.out.println("BMS Thread interrupted");
}
   }
}
class CSE extends Thread {
   public void run() {
```

```
        try {
            for(int j=0;j<15;j++) {
                System.out.println("CSE");
                Thread.sleep(2000); // Sleep for 2 seconds
            }
        }
catch (InterruptedException e) {
  System.out.println("CSE Thread interrupted");
}
    }
}

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

```
F:\Year 2\Semester 3\Object Oriented Programming in Java\Lab Class\Week 8>javac ThreadExample.java

F:\Year 2\Semester 3\Object Oriented Programming in Java\Lab Class\Week 8>java ThreadExample
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
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE

F:\Year 2\Semester 3\Object Oriented Programming in Java\Lab Class\Week 8>
```

Program 9
Deadlock And IPC Program

Algorithm:



Code:

DeadLock

```
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) {
```

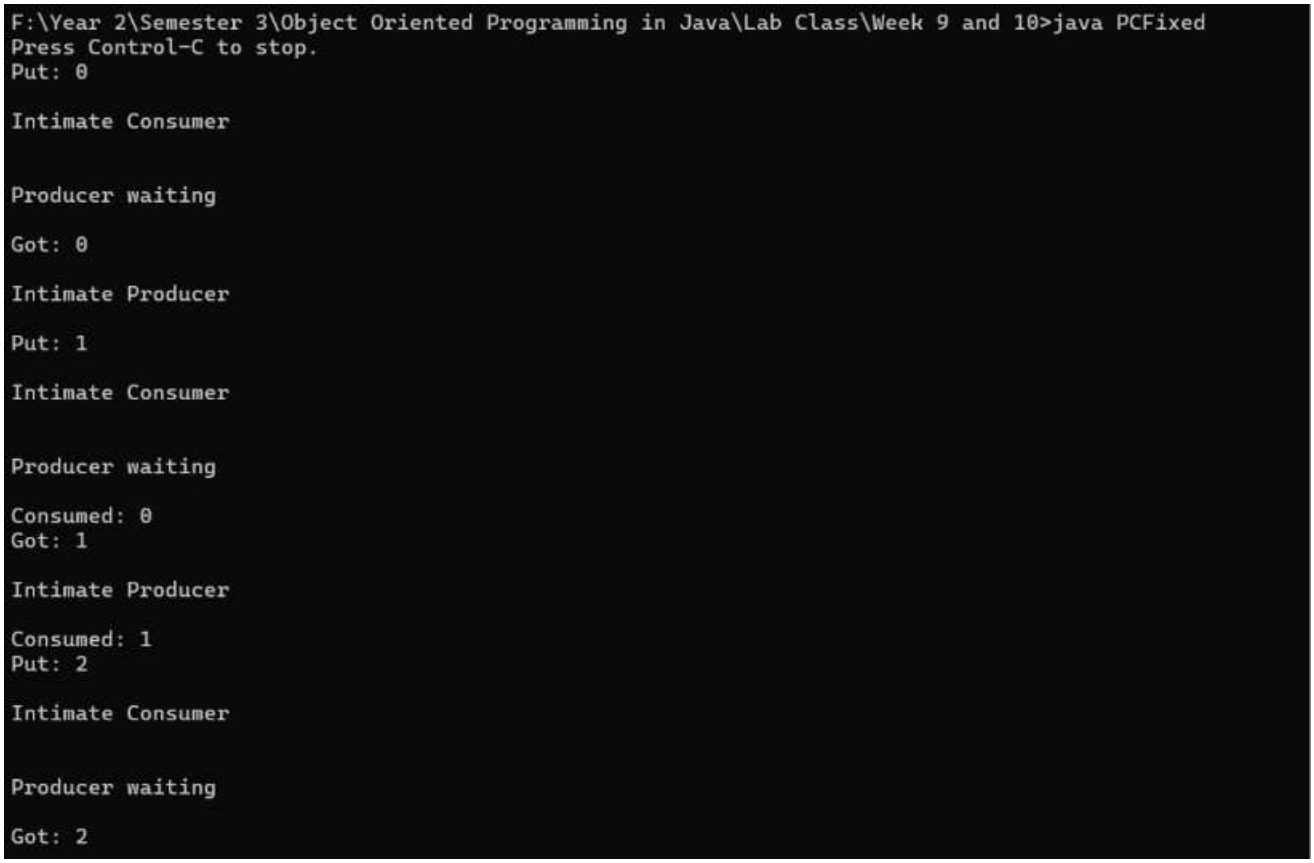
```
            System.out.println("A Interrupted");
        }

        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();

        a.foo(b);
        System.out.println("Back in main thread");
    }

    public void run() {
        b.bar(a);
```

```
        System.out.println("Back in other thread");
    }

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

PCFixed

```
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++;
        }
    }
}

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

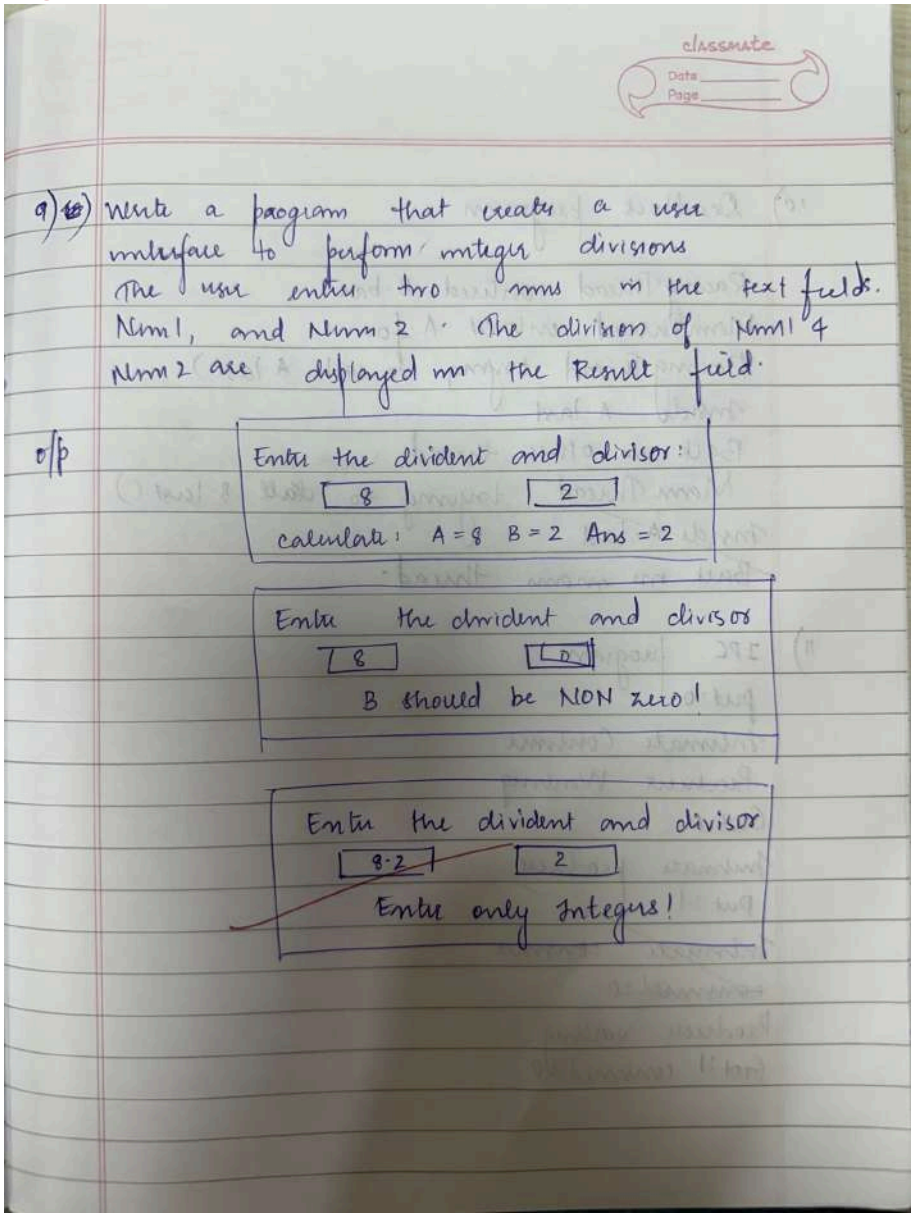

F:\Year 2\Semester 3\Object Oriented Programming in Java\Lab Class\Week 9 and 10>java PCFixed
Press Control-C to stop.
Put: 0
Intimate Consumer
Producer waiting
Got: 0
Intimate Producer
Put: 1
Intimate Consumer
Producer waiting
Consumed: 0
Got: 1
Intimate Producer
Consumed: 1
Put: 2
Intimate Consumer
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

Put: 5

Intimate Consumer


Producer waiting

Consumed: 4
Got: 5
```

```
Intimate Producer

Consumed: 5
Put: 6

Intimate Consumer


Producer waiting

Got: 6

Intimate Producer

Consumed: 6
Put: 7

Intimate Consumer


Producer waiting

Got: 7

Intimate Producer

Consumed: 7
Put: 8

Intimate Consumer


Producer waiting

Got: 8
```

Program 10
Swing Demo (Calculator)

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());
// Terminate on close
jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

// Create components
JLabel jlab = new JLabel("Enter the divider 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();

// Add components in order
jfrm.add(err); // To display errors
jfrm.add(jlab);
jfrm.add(ajtf);
jfrm.add(bjtf);
jfrm.add(button);
jfrm.add(alab);
jfrm.add(blab);
jfrm.add(anslab);

// Add ActionListeners
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(""); // Clear error message
```

```
            } 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 the 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();
      }
    });
  }
}
```