

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.



LAB REPORT
on

Object Oriented Java Programming **(23CS3PCOOJ)**

Submitted by

K Raghavendra S Adiga (1BM23CS133)

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 **K Raghavendra S Adiga(1BM23CS133)**, 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

Sl. No.	Date	Experiment Title	Page No.
1		Quadratic Equation Program - Lab 1	4-6
2		Student CGPA - Lab 2	7-12
3		Book Class - Lab 3	13-17
4		Abstraction - Lab 4	18-20
5		Inheritance - Lab 5	21-27
6		Packages - Lab 6	28-32
7		Exception Handling - Lab 7	33-36
8		Multi-Threading - Lab 8	37-39
9		Graphical User Interface - Lab 9	40-43
10		Inter Process Communication and Deadlock - Lab 10	44-51

Github Link:

<https://github.com/kraghavendra-adiga/OOJ-Lab-cs133>

Program 1

Implement Quadratic Equation

Algorithm:

```

20/09/2021 Lab programs : 3
② Quadratic using Java.
Develop a java program that prints real & equal
solutions to quadratic equation  $ax^2 + bx + c = 0$ .
Read in a, b, c & use Quadratic formula. If
the discriminant  $b^2 - 4ac$  is >ve, display
a message saying there are no real
solutions.

import java.util.Scanner;

class ComputeQuadratic {
    public static void main (double delta,
        double a, double b, double c) {
        if (delta < 0) System.out.println("No real roots");
        else if (delta == 0) { System.out.println("Real root
            equal roots");
            double root1 = -b / (2*a);
            System.out.println("Roots are: root = " +
                root1);
        }
        else {
            System.out.println("Roots are real &
                distinct");
            double root1 = (-b + Math.sqrt(delta)) / (2*a);
            double root2 = (-b - Math.sqrt(delta)) / (2*a);
            System.out.println("Root1 = " + root1 + " Root2 = "
                + root2);
        }
    }
}

public class Quadratic133 {
    public static void main (String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a,b,c here:");
        double a = sc.nextDouble();
        double b = sc.nextDouble();
        double c = sc.nextDouble();
        double d = (b*b) - (4*a*c);

        if (a == 0) {
            System.out.println("Invalid Equation");
        }
        else {
            ComputeQuadratic.compute(d, a, b, c);
        }

        System.out.println("Name: K Raghavendra S Adiga");
        System.out.println("USN: 18BMS0333");
    }
}

o/p
Enter a,b,c here:
+
+
+
1
Roots of the quadratic equation are real and equal
Roots are: root1 = root2 = -0.5

o/p
Enter a,b,c here:
0
+
+
1
Invalid Equation
  
```

```

Invalid Equation
o/p
Enter a,b,c here:
10
9
1
No real roots.

o/p
Enter a,b,c here:
3
3
1
Roots are real.
root1 = -0.131+3.2
root2 = -2.53518
  
```

Code:

```

import java.util.Scanner;

class ComputeQuadratic {
    public static void compute(double delta, double a, double b, double c) {
        if (delta < 0)
            System.out.println("Given quadratic equation has no real roots. ");

        else if(delta==0){

            double root1 = -b / (2*a);
            System.out.println("Roots of the quadratic equation are real and equal");
            System.out.println("Roots are: root1=root2= " + root1);

        } else {

            double root1 = (-b + Math.sqrt(delta))/(2*a);
            double root2 = (-b - Math.sqrt(delta))/(2*a);

            System.out.println("Roots of the quadratic equation are real and distinct ");
            System.out.println("Root1= "+root1 + "\nRoot2=" + root2);

        }

    }

}

public class quadratic133 {
    public static void main(String[] args) {
        Scanner sc= new Scanner(System.in);

        System.out.println("Enter a,b and c of the quadratic equation here please: ");

        double a =sc.nextDouble();
        double b= sc.nextDouble();
        double c = sc.nextDouble();

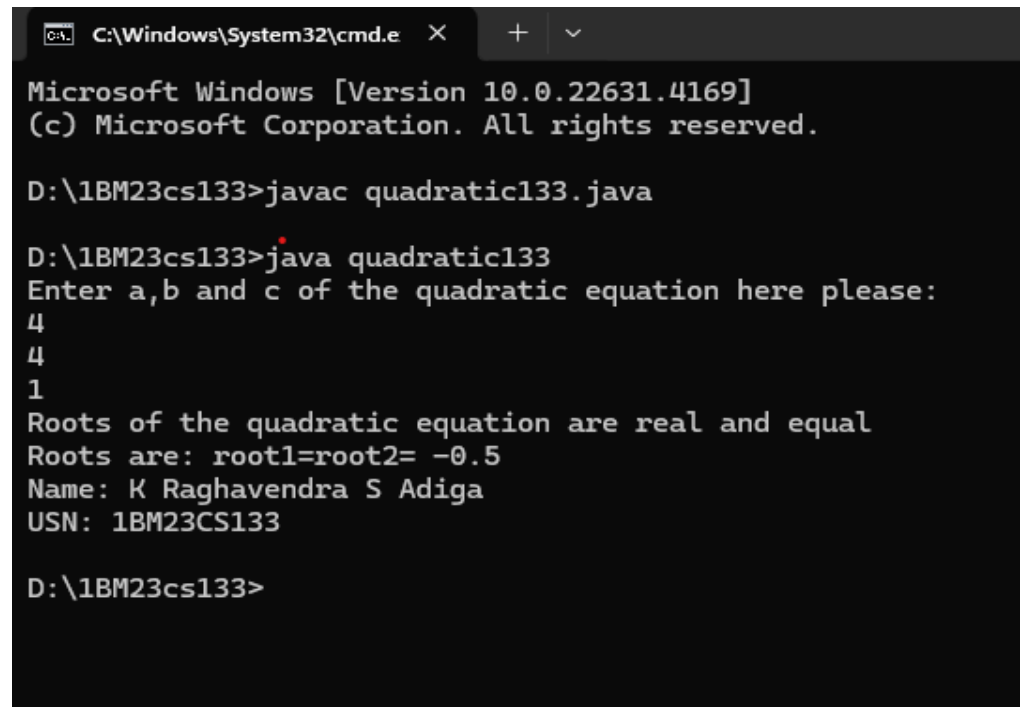
        double delta = (b*b) - (4*a*c);

        if(a==0){
            System.out.println("The given equation is not a QUADRATIC EQUATION: ");
        }
        else{
            ComputeQuadratic.compute(delta,a,b,c);
        }
    }
}

```

```
        System.out.println("Name: K Raghavendra S Adiga");  
        System.out.println("USN: 1BM23CS133");  
    }  
}
```

Output:



```
C:\Windows\System32\cmd.e  X  +  v  
Microsoft Windows [Version 10.0.22631.4169]  
(c) Microsoft Corporation. All rights reserved.  
  
D:\1BM23cs133>javac quadratic133.java  
  
D:\1BM23cs133>java quadratic133  
Enter a,b and c of the quadratic equation here please:  
4  
4  
1  
Roots of the quadratic equation are real and equal  
Roots are: root1=root2= -0.5  
Name: K Raghavendra S Adiga  
USN: 1BM23CS133  
  
D:\1BM23cs133>
```

Program 2**SGPA Calculator for Student Class**

Algorithm:

07/10/2024

② Develop a Java program to create a class Student with members usn, name, an array, credits & an array marks. Include methods to accept & display details and a method to calculate SGPA of a student.

```

import java.util.Scanner;
class Student {
    private String usn;
    private String name;
    private int[] credits;
    private int[] marks;
    private int numSubjects;

    public Student(int numSubjects) {
        this.numSubjects = numSubjects;
        credits = new int[numSubjects];
        marks = new int[numSubjects];
    }

    public void getStudentDetails() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the name");
        name = sc.nextLine();
        System.out.println("Enter the Usn");
        usn = sc.nextLine();

        for(int i=0; i < numSubjects; i++) {
            System.out.println("Enter credits of " + i);
            credits[i] = sc.nextInt();

            System.out.println("Enter marks of " + i);
            marks[i] = sc.nextInt();
        }
    }
}

```



```

public void getmarks() {
    System.out.println("Name of the student" +
                        name);
    System.out.println("USN of the student" +
                        usn);
}

```

```

for (int i = 0; i < numSubjects; i++)
{
    System.out.println("Marks of Subject" +
                        + (i+1) + " is: " + marks[i]);
}

```

```

public double computeGpa() {
    int grade;
    double total = 0;
    int totalcred = 0;
    for (int i = 0; i < numSubjects; i++)
    {

```

```

        grade = marks[i] / 10;
        if (grade > 10) grade = 10;
        if (grade < 4) grade = 0;

```

```

        double temp = grade * marks[i];
        total += temp;
        totalcred += credits[i];
    }

```

```

    return total / totalcred;
}

```



```

public class StudentMethod {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter total no. of students");
        int totalStudents = sc.nextInt();
        Student[] arrayOfObj = new Student[totalStudents];
        for (int i = 0; i < totalStudents; i++) {
            arrayOfObj[i] = new Student(i);
            arrayOfObj[i].getStudentDetails();
            arrayOfObj[i].getMarks();
            double sgp = arrayOfObj[i].getcomputeSGP();
            System.out.println("SGPA of the Student is: " + sgp);
        }
    }
}

```

O/P

O/P
Enter the total no. of Students here: 3
Enter the name: Karthik
Enter the USN: IBM23CS131
Enter the credits for ^{Sub}sgpa here: 5
Enter the marks for sub1 here: 87
Enter the credits for sub2 here: 7
Enter the marks for sub2 here: 98
Name: Karthik
USN: IBM23CS131
Marks of sub1: 87
Marks of sub2: 98
SGPA of the student is: 8.422445

Code:

```

import java.util.Scanner;
class Student {
    private String usn;
    private String name;
    private int[] credits;
    private int[] marks;
    private int numSubjects;

    public Student(int numSubjects) {
        this.numSubjects = numSubjects;
        credits = new int[numSubjects];
        marks = new int[numSubjects];
    }

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

        System.out.println("Enter the name: ");
        name=sc.nextLine();

        System.out.println("Enter the USN: ");
        usn=sc.nextLine();

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

            System.out.println("Enter the credits for subject " + (i+1) + " here: ");
            credits[i]=sc.nextInt();

            System.out.println("Enter the marks for subject " + (i+1) + " here: ");
            marks[i]=sc.nextInt();

        }

    } //getstudentdetails method

    public void getmarks(){
        System.out.println("Name of the student: " + name);
        System.out.println("USN of the student: " + usn);
        for (int i = 0; i < numSubjects; i++) {
            System.out.println("marks of subject " + (i+1) + " is:" + marks[i]);

        }

    }

    public double computesgpa(){

```

```

    int grade;
    double total=0;
    int totalcred=0;
    for (int i = 0; i < numSubjects; i++) {
        grade= marks[i]/10;
        if(grade>10) grade=10;

        if(grade<4) grade=0;

        double temp= grade*credits[i];
        total+=temp;
        totalcred+=credits[i];

    }

    return total/totalcred;
}

}

public class intellijStudent {
    public static void main(String[] args) {
        Scanner st = new Scanner(System.in);
        System.out.println("Enter the total number of students here: ");
        int totalstudents= st.nextInt();
        Student [] arrayOfObj = new Student[totalstudents];

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

            arrayOfObj[i] = new Student(2);
            arrayOfObj[i].getStudentDetails();
            arrayOfObj[i].getmarks();
            double sgpaofs1=arrayOfObj[i].computesgpa();
            System.out.println("SGPA of the student is: " + sgpaofs1);

        }

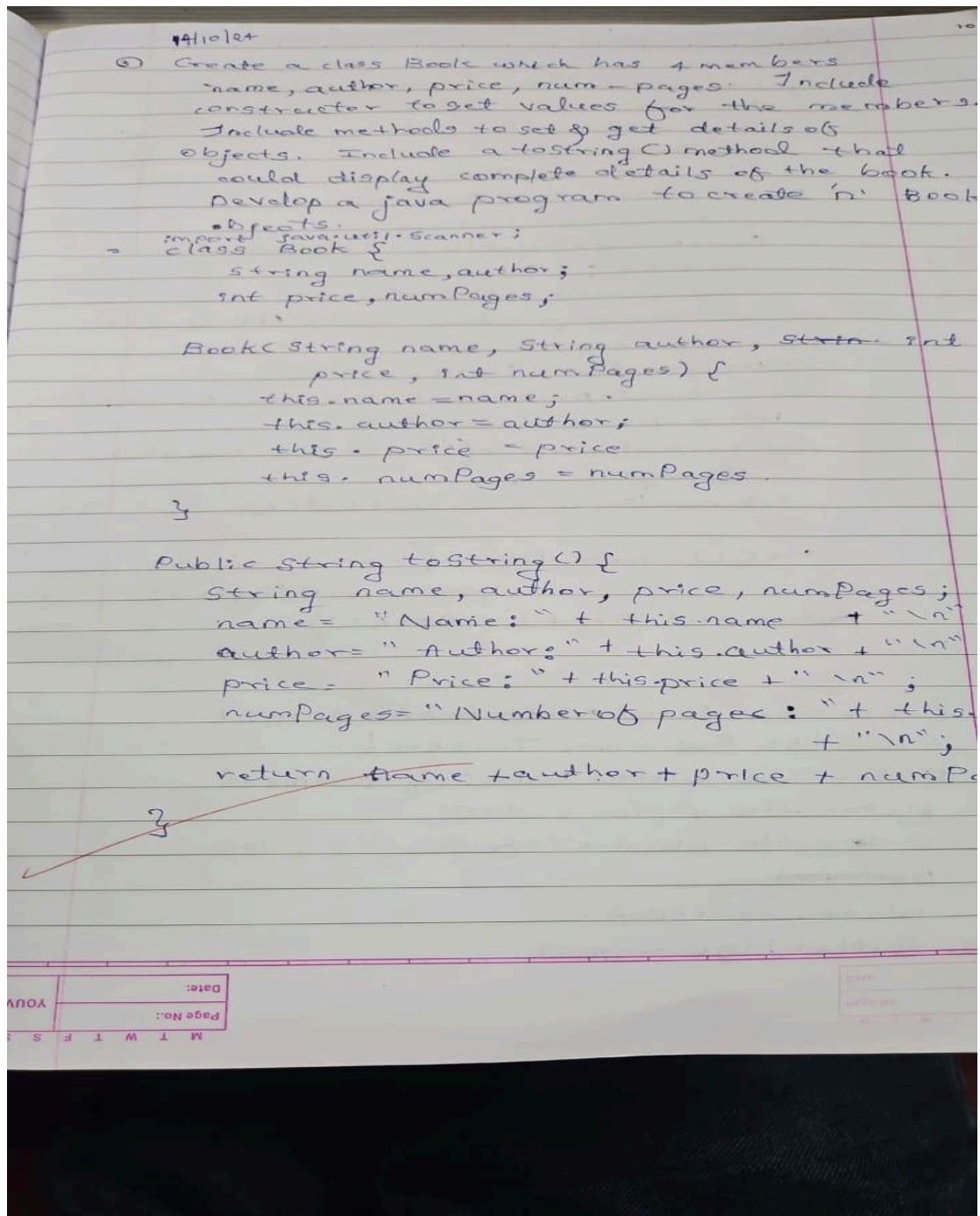
    }
}

```


Program 3

Implement Book Class and Demonstrate toString() Method

Algorithm:




```

Public class labthree {
    public static void main (String[] args) {
        Scanner sc = new Scanner (System.in);
        System.out.println (" Enter total no. of
        Books here: ");
        int n = sc.nextInt();
        Book[] arr = new Book[n];
        for (int i=0; i<n; i++) {
            System.out.println (" Enter the name
            of book: ");
            String name = sc.nextLine();
            System.out.println (" Enter the Author
            name: ");
            String author = sc.nextLine();
            System.out.println (" Enter the Price: ");
            int price = sc.nextInt();
            System.out.println (" Enter the Number
            of pages: ");
            int numPages = sc.nextInt();

            arr[i] = new Book (name, author,
                                price, numPages);
            System.out.println (arr[i]);
        }
    }
}

```

o/p

Enter the total no. of books here: 1

Enter the Book name: Textbook

Enter the author name: Government

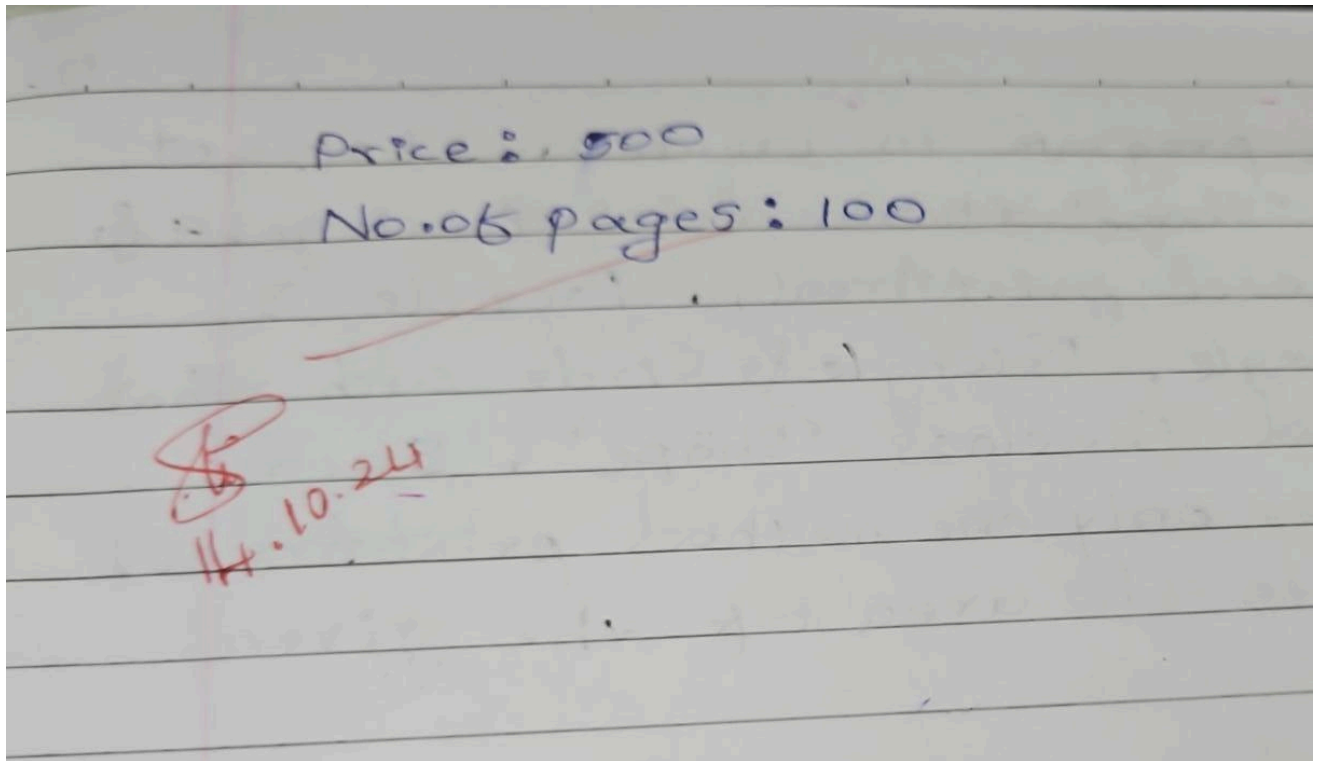
Enter the price: 500

Enter the number of pages: 100

~~Book~~

Name: Textbook

Author: Government



Code:

```
package CS133programs;
```

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

```
class Book{
```

```
    String name;
```

```
    String author;
```

```
    int price;
```

```
    int numPages;
```

```
//Constructor:
```

```
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 = "Name: " + this.name + "\n";
```

```
    author = "Author: " + this.author + "\n";
```

```
    price = "Price: " + this.price + "\n";
```

```
    numPages = "Number of Pages: " + this.numPages + "\n";
```

```
    String total = "";
```

```
    total+=name+author+price+numPages;
```

```

        return total;
    }
}

public class labthree {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the total number of Books here: ");
        int n = sc.nextInt();
        //Array of objects:
        Book[] arr = new Book[n];

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

            String name,author;
            int price,numPages;

            System.out.println("Enter the name of " + (i+1) + " book: ");
            name = sc.next();

            System.out.println("Enter the Price of the " + (i+1) + " Book: ");
            price = sc.nextInt();

            System.out.println("Enter the Author of the " + (i+1) + " Book: ");
            author = sc.next();

            System.out.println("Enter the Total Number of Pages of the " + (i+1) + " Book: ");
            numPages = sc.nextInt();

            arr[i] = new Book(name,author,price,numPages);
            System.out.println(arr[i]);

        }

        System.out.println("Name: K Raghavendra S Adiga ");
        System.out.println("USN: 1BM23CS133 ");
    }
}

```

Output:

```
"C:\Users\STUDENT\AppData\Local\Programs\Eclipse Adoptium\jdk-21.0.4-hotspot\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.3\lib\
Enter the total number of Books here:
2
Enter the name of 1 book:
AtomicHabits
Enter the Price of the 1 Book:
500
Enter the Author of the 1 Book:
JamesClerk
Enter the Total Number of Pages of the 1 Book:
250
Name: AtomicHabits
Author: JamesClerk
Price: 500
Number of Pages: 250

Enter the name of 2 book:
HarryPotter
Enter the Price of the 2 Book:
1000
Enter the Author of the 2 Book:
James
Enter the Total Number of Pages of the 2 Book:
200
Name: HarryPotter
Author: James
Price: 1000
Number of Pages: 200

Name: K Raghavendra S Adiga
USN: 18M23CS133
```

Program 4

Demonstrating Abstract Classes and Abstract Methods

Algorithm:

```

11/10/2024
② Develop a java program to create an abstract
class named 'Shape' that contains 2 integers &
a method named printArea(). Provide 3
classes Rectangle, Triangle & Circle such that
they extend the class 'Shape'. Each
class contains only one method printArea()
that prints the area of the given
Shape.
abstract class Shape {
    int length;
    int height;
    int radius;
    abstract void printArea();
}

class Rectangle extends Shape {
    Scanner sc = new Scanner(System.in);
    public void accept() {
        System.out.print("Enter length & width: ");
        length = sc.nextInt();
        height = sc.nextInt();
    }

    void printArea() {
        System.out.print("Area of rectangle is: "
            + length * height);
    }
}

class Triangle extends Shape {
    Scanner sc = new Scanner(System.in);
    public void accept() {
        System.out.print("Enter Base & height: ");
        length = sc.nextInt();
        height = sc.nextInt();
    }

    void printArea() {
        System.out.print("Area of Triangle: "
            + (length * height) / 2);
    }
}

class Circle extends Shape {
    Scanner sc = new Scanner(System.in);
    public void accept() {
        System.out.print("Enter the Radius: ");
        radius = sc.nextInt();
    }

    void printArea() {
        System.out.print("Area of Circle: "
            + 3.1428 * radius * radius);
    }
}

public class ShapeClass {
    public static void main(String[] args) {
        Rectangle obj1 = new Rectangle();
        Triangle obj2 = new Triangle();
        Circle obj3 = new Circle();

        obj1.accept();
        obj1.printArea();

        obj2.accept();
        obj2.printArea();

        obj3.accept();
        obj3.printArea();
    }
}

```

O/P

Enter the length & width: 5 10
Area of Rectangle : 50

Enter the Base & height: 5 10
Area of Triangle : 25

Enter the Radius: 10
Area of Circle: 314.28

21/10

Code:

```

package CS133programs;

import java.util.Scanner;

abstract class Shape{
    int length,height,radius;

    abstract void printArea();
}

class Rectangle extends Shape{
    Scanner sc=new Scanner(System.in);
    public void accept(){
        System.out.println("Enter the length of the Rectangle: ");
        length=sc.nextInt();
        System.out.println("Enter the width of the Rectangle: ");
        height=sc.nextInt();
    }
    void printArea(){
        System.out.println("Area of the rectangle is: " + length*height);
    }
}

class Triangle extends Shape{
    Scanner sc=new Scanner(System.in);
    public void accept(){
        System.out.println("Enter the Base of the Triangle: ");
        length=sc.nextInt();
        System.out.println("Enter the Height of the Triangle: ");
        height=sc.nextInt();
    }
    void printArea(){
        float temp= (float) (length*height)/2;
        System.out.println("Area of the Triangle is: " + temp);
    }
}

class Circle extends Shape{
    Scanner sc=new Scanner(System.in);
    public void accept(){
        System.out.println("Enter the Radius of the Circle: ");
        radius=sc.nextInt();
    }
    void printArea(){
        float temp= (float) (radius*radius*3.1428);
        System.out.println("Area of the Triangle is: " + temp);
    }
}

```

```

    }
}

public class shapeclass {
    public static void main(String[] args) {
        Rectangle obj1= new Rectangle();
        Triangle obj2= new Triangle();
        Circle obj3= new Circle();
        System.out.println("Shape: Rectangle: ");
        System.out.println();
        obj1.accept();
        obj1.printArea();

        System.out.println("Shape: Triangle: ");
        System.out.println();
        obj2.accept();
        obj2.printArea();

        System.out.println("Shape: Circle: ");
        System.out.println();
        obj3.accept();
        obj3.printArea();

    }
}

```

Output:

```

:
"C:\Users\STUDENT\AppData\Local\Programs\Eclipse Adoptium\jdk-21.0.4-hotspot\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.3\lib\ic
Shape: Rectangle:

Enter the length of the Rectangle:
10
Enter the width of the Rectangle:
5
Area of the rectangle is: 50
Shape: Triangle:

Enter the Base of the Triangle:
10
Enter the Height of the Triangle:
5
Area of the Triangle is: 25.0
Shape: Circle:

Enter the Radius of the Circle:
10
Area of the Triangle is: 314.28
Name: K Raghavendra S Adiga
USN: 18M23CS133

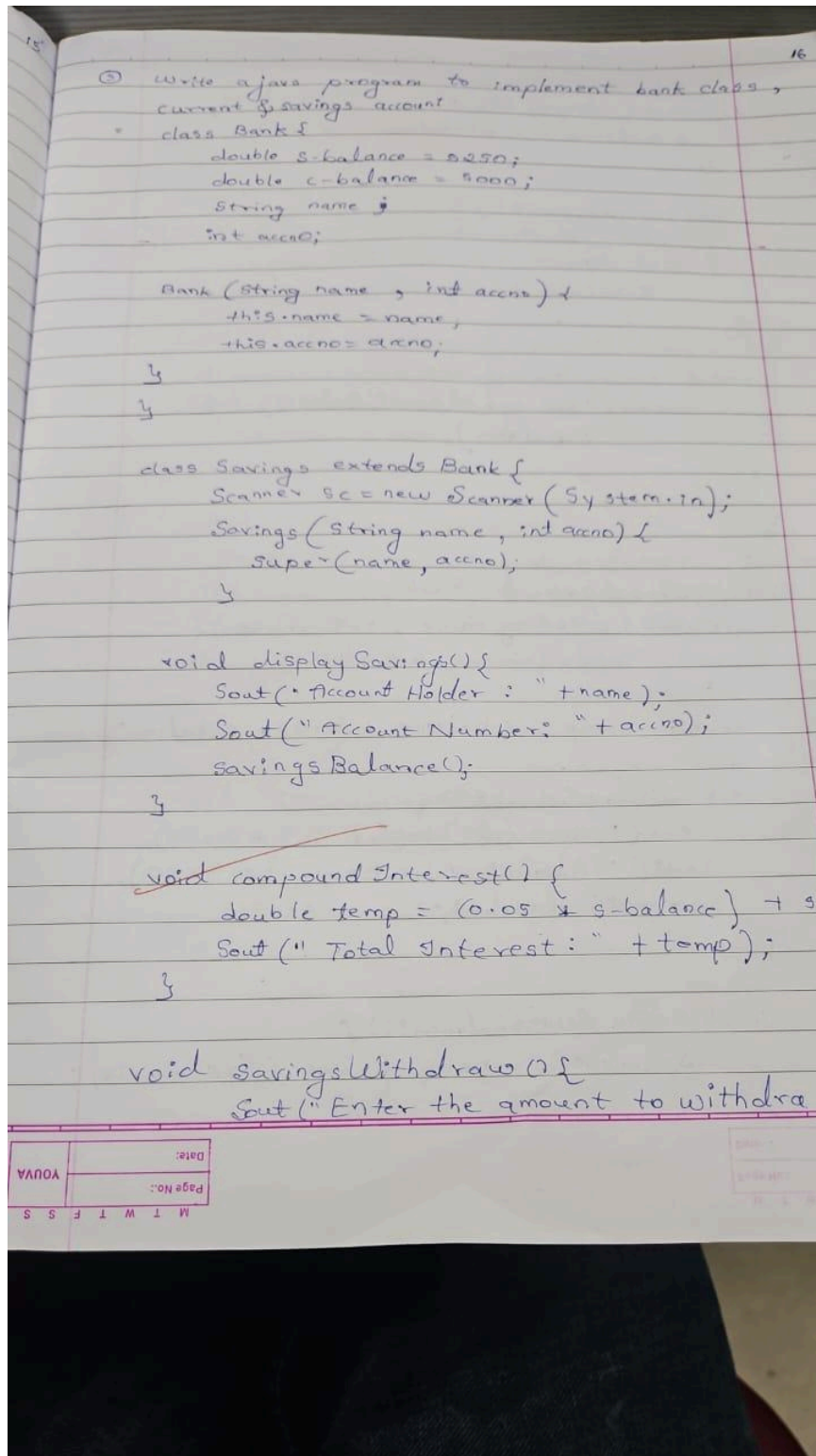
Process finished with exit code 0

```

Program 5

Demonstrating Inheritance using Savings Class and Current Class

Algorithm:



```

int wm = sc.nextInt();
Sout("Money has been withdrawn");
c.balance -= withdraw wm;
}

void savingsDeposit() {
    Sout("Enter amount to Deposit:");
    int deposit = sc.nextInt();
    c.balance += deposit;
}

void savingsBalance() {
    Sout("Savings Account Balance: " +
        c.balance);
}

class Current extends Bank {
    Current(String name, int accno) {
        super(name, accno);
    }
    double temp = c.balance + (gross * c.balance);

    void displayCurrent() {
        Sout("Account Holder: " + name);
        Sout("Account Number: " + accno);
        cur.getBalance();
    }

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

```

```

        if (c.balance < 5000) {
            Sout("You have a penalty of ₹ 100");
            return;
        }
        else if (c.balance < 4000) {
            Sout("Insufficient Balance");
            Sout("You have penalty of ₹ 500");
            return;
        }
        Sout("Enter the amount to withdraw:");
        int wm = sc.nextInt();
        Sout("Money has been withdrawn");
        c.balance -= wm;
    }

    void currentBalance() {
        Sout("Current Account Balance: " + c.balance);
        if (c.balance < 5000 & c.balance > 4000) {
            Sout("You have penalty of ₹ 100");
        }
        else if (c.balance < 4000) {
            Sout("You have penalty of ₹ 500");
        }
    }

    void currentDeposit() {
        Sout("Enter the amount to Deposit");
        int dp = sc.nextInt();
        c.balance = c.balance + deposit;
    }
}

```

```

        else if (choice == 6) {
            obj1.displaySavings();
        }

        else if (choice == 7) {
            obj2.displayCurrent();
        }

        else break;
    }

    }

    }

    }

    o/p

```

```

public class Bank {
    public static void main (String[] args) {
        Scanner sc = new Scanner (System.in);
        Savings obj1 = new Savings ("John", 1000);
        Current obj2 = new Current ("John", 1000);

        System.out.println("Select a choice :");
        int choice;
        while (true) {
            System.out.println("choice 1 -> Deposit to Savings");
            System.out.println("choice 2 -> Withdraw from Savings");
            System.out.println("choice 3 -> Deposit to Current");
            System.out.println("choice 4 -> Withdraw from Current");
            System.out.println("choice 5 -> Compound Interest");
            System.out.println("choice 6 -> Show Savings");
            System.out.println("choice 7 -> Show Current");
            System.out.println("choice 8 -> Exit");

            if (choice == 1) {
                obj1.savingsDeposit();
            }
            else if (choice == 2) {
                obj1.savingsWithdraw();
            }
            else if (choice == 3) {
                obj2.currentDeposit();
            }
            else if (choice == 4) {
                obj2.currentWithdraw();
            }
            else if (choice == 5) {
                obj1.compoundInterest();
            }
        }
    }
}

```

choice 1 -> Deposit to Savings
 choice 2 -> Withdraw from Savings
 choice 3 -> Deposit to Current
 choice 4 -> Withdraw from Current
 choice 5 -> Compound Interest
 choice 6 -> Show Savings
 choice 7 -> Show Current
 choice 8 -> Exit

Account Holder Name : John
 Account Number : 1234
 Account Balance : 5000.0

choice 1 -> Deposit to Savings
 choice 2 -> Withdraw from Savings
 choice 3 -> Deposit to Current
 choice 4 -> Withdraw from Current
 choice 5 -> Compound Interest
 choice 6 -> Show Savings
 choice 7 -> Show Current
 choice 8 -> Exit

8
 28.10

Code:
package CS133programs;


```

import java.util.Scanner;

class Bank{
    double s_balance=5250;
    double c_balance=5000;
    String name;
    int accno;

    Bank(String name, int accno){
        this.name=name;
        this.accno=accno;
    }
}

class Savings extends Bank{
    Scanner sc= new Scanner(System.in);

    Savings(String name, int accno) {
        super(name, accno);
    }

    void displaySavings(){
        System.out.println("Account Holder Name: " + name);
        System.out.println("Account Number: " + accno);
        savingsBalance();
    }

    void compoundInterest(){
        double temp=(0.05*s_balance) + s_balance;
        System.out.println("Total Interest: " + temp);
    }

    void savingsWithdrawl(){
        System.out.println("Enter the amount to Withdraw: ");
        int withdrawnMoney = sc.nextInt();
        System.out.println("Money has been Withdrawn successfully: ");
        s_balance-=withdrawnMoney;
    }

    void savingsDeposit(){
        System.out.println("Enter the amount to Deposit: ");
        int deposit=sc.nextInt();
        s_balance = deposit + s_balance;
    }
}

```

```

void savingsBalance(){
    System.out.println("Savings Account Balance: " + s_balance);
}

}

class Current extends Bank{
    double temp=(0.05*c_balance) + c_balance;
    Current(String name, int accno) {
        super(name, accno);
    }

    void displayCurrent(){
        System.out.println("Account Holder Name: " + name);
        System.out.println("Account Number: " + accno);
        currentBalance();
    }

    void currentWithdrawl(){
        Scanner sc= new Scanner(System.in);
        if(c_balance<5000 && c_balance>=4000){
            System.out.println("You have received a penalty of ₹100 ");
            System.out.println("Bank Balance Low: Cannot Withdraw");
            return;
        } else if (c_balance<4000 && c_balance>=3000) {
            System.out.println("You have received a penalty of ₹500 ");
            System.out.println("Bank Balance Low: Cannot Withdraw");
            return;
        }
        System.out.println("Enter the amount to Withdraw: ");
        int withdrawnMoney = sc.nextInt();
        System.out.println("Money has been Withdrawn successfully: ");
        c_balance-=withdrawnMoney;
    }

    void currentBalance(){
        System.out.println("Current Account Balance: " + c_balance);
        if(c_balance<5000 && c_balance>=4000){
            System.out.println("You have received a penalty of ₹100 ");

        } else if (c_balance<4000 && c_balance>=3000) {
            System.out.println("You have received a penalty of ₹500 ");
        }
    }

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

```

```

        System.out.println("Enter the amount to Deposit: ");
        int deposit=sc.nextInt();
        c_balance = deposit + c_balance;
    }

}

public class bankProgram {
    public static void main(String[] args) {
        Scanner st= new Scanner(System.in);
        Savings obj1savings = new Savings("John Kumar",123456);
        Current obj1current = new Current("John Kumar",123456);

//        Savings obj2savings = new Savings("John Kumar",123456);
//        Current obj2current = new Current("John Kumar",123456);
        System.out.println("SELECT A CHOICE: ");
        int choice;
        while(true){
            System.out.println("Choice 1 → Deposit to Savings \n Choice 2 → Withdrawl From Savings
\n Choice 3→ Deposit to Current \n Choice 4→Withdrawl From Current \n Choice 5 → Compound
Interest \n Choice 6→Show Savings Account Information \n choice 7→ Show Current Account
Information \n Choice 8→ Exit");
            choice = st.nextInt();

            if(choice==1){
                obj1savings.savingsDeposit();
            }

            else if(choice==2){
                obj1savings.savingsWithdrawl();
            }

            else if(choice==3){
                obj1current.currentDeposit();
            }

            else if(choice==4){
                obj1current.currentWithdrawl();
            }

            else if(choice==5){
                obj1savings.compoundIntest();
            }er

            else if(choice==6){
                obj1savings.displaySavings();
            }

```

```

        else if(choice==7){
            obj1.current.displayCurrent();
        }
        else break;
    }
}
}
}

```

Output:

```

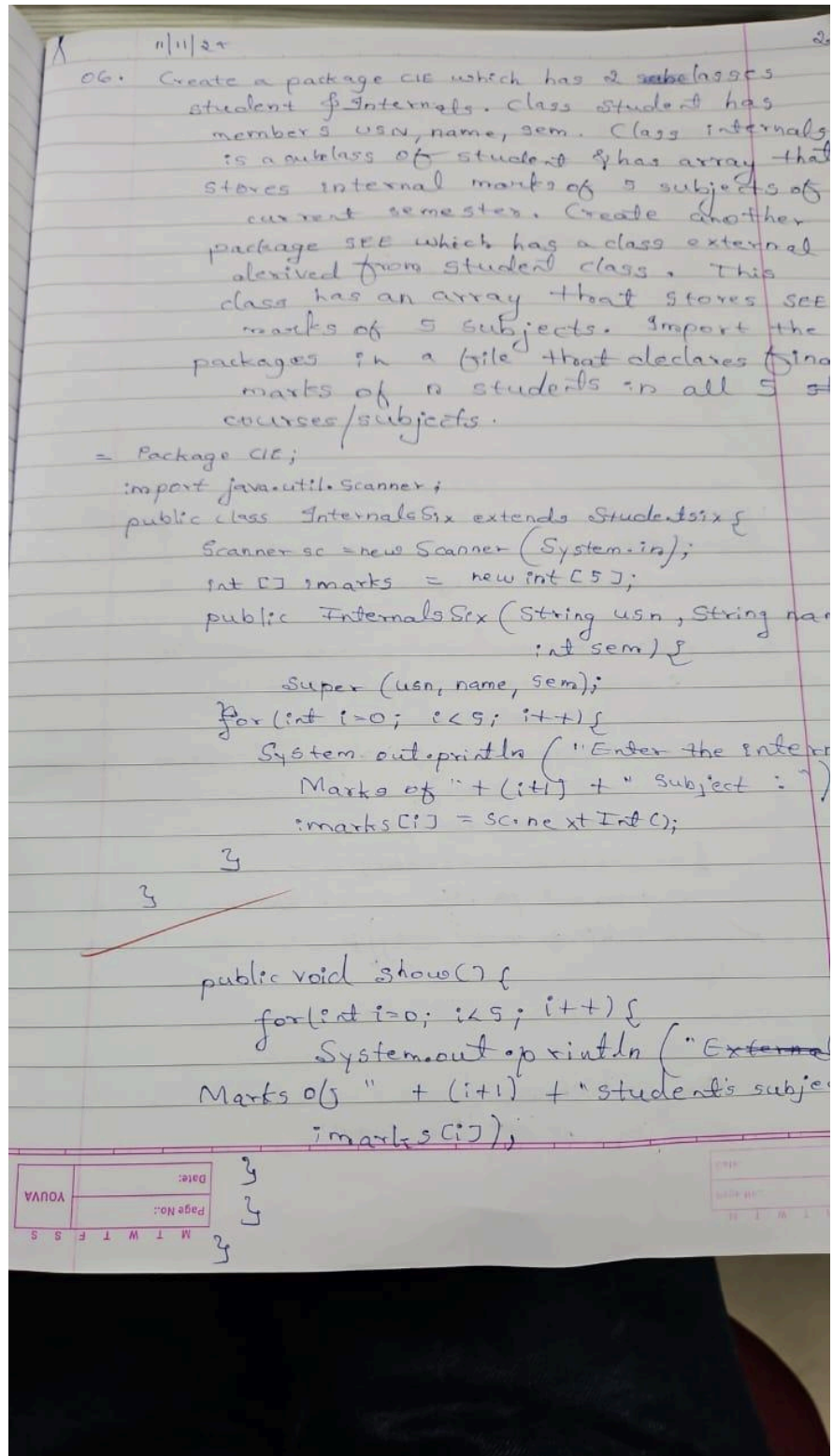
C:\Users\STUDENT\AppData\Local\Programs\Eclipse Adoptium\jdk-21.0.4.7-hotspot\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.3\lib\j
SELECT A CHOICE:
Name: K Raghavendra S Adiga
USN: 1BM23CS133
Choice 1 → Deposit to Savings
Choice 2 → Withdrawl From Savings
Choice 3→ Deposit to Current
Choice 4→Withdrawl From Current
Choice 5 → Compound Interest
Choice 6→Show Savings Account Information
choice 7→ Show Current Account Information
Choice 8→ Exit
$
Enter the amount to Deposit:
1000
Choice 1 → Deposit to Savings
Choice 2 → Withdrawl From Savings
Choice 3→ Deposit to Current
Choice 4→Withdrawl From Current
Choice 5 → Compound Interest
Choice 6→Show Savings Account Information
choice 7→ Show Current Account Information
Choice 8→ Exit
$
Account Holder Name: John Kumar
Account Number: 123456

```

Program 6

Demonstrating Packages

Algorithm:




```

package cie;

public class StudentStaff {
    String use, name;
    int sem;

    public StudentStaff(String use, String name,
        int sem) {
        this.use = use;
        this.name = name;
        this.sem = sem;
    }
}

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

public class ExternalSix extends StudentStaff {
    Scanner sc = new Scanner(System.in);
    int i;
    public ExternalSix(String name, String
        name, int sem) {
        super(use, name, sem);
        for (int i = 0; i < 5; i++) {
            System.out.print("Enter external marks
            of subject " + (i+1));
        }
    }

    public void showCie() {
        for (int i = 0; i < 5; i++) {
            System.out.print("External Marks of subject " + (i+1)
                + " : ");
            for (int j = 0; j < 5; j++) {
                System.out.print("Enter the marks of subject " + (j+1)
                    + " : ");
            }
            System.out.print("\n");
        }
    }
}

package LabSix;
import cie.*;
import SEE.*;
import java.util.*;

public class LabSix {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n;
        System.out.print("Enter no. of students: ");
        n = sc.nextInt();
        InternalSix[] arr1 = new InternalSix[n];
        ExternalSix[] arr2 = new ExternalSix[n];

        for (int i = 0; i < n; i++) {
            System.out.print("Enter the use: ");
            String use = sc.next();
            System.out.print("Enter the name: ");
            String name = sc.next();
            System.out.print("Enter the sem: ");
            int sem = sc.nextInt();
            arr1[i] = new InternalSix(use, name, sem);
            arr2[i] = new ExternalSix(use, name, sem);
        }

        for (int i = 0; i < n; i++) {
            System.out.print("Marks of student " + (i+1)
                + " : ");
            arr1[i].show();
            arr2[i].show();
        }
    }
}

```

Enter the total no. of students here: 1
 Enter the use of student 1: 1812345678
 Enter the name of student 1: Raghav
 Enter the sem of student 1: 3
 Enter the marks of subject 1: 100
 Enter internal marks of subject 1: 100
 Enter external marks of subject 1: 98
 Enter internal marks of subject 2: 99
 Enter external marks of subject 2: 99
 Enter internal marks of subject 3: 100
 Enter external marks of subject 3: 99
 Enter internal marks of subject 4: 99
 Enter external marks of subject 4: 99
 Enter internal marks of subject 5: 100
 Enter external marks of subject 5: 100

Internal marks of subject 1: 100
 Internal marks of subject 2: 99
 Internal marks of subject 3: 98
 Internal marks of subject 4: 99
 Internal marks of subject 5: 99

External marks of subject 1: 100
 External marks of subject 2: 99
 External marks of subject 3: 98
 External marks of subject 4: 100
 External marks of subject 5: 100

Code:

//Package CIE

```

package CIE;

import java.util.Scanner;

public class InternalsSix extends Studentsix {
    Scanner sc= new Scanner(System.in);
    int [] imarks = new int[5];
    public InternalsSix(String usn, String name, int sem){
        super(usn,name,sem);
        for (int i = 0; i < 5; i++) {
            System.out.println("Enter the Internal marks of " + (i+1) +" subject:");
            imarks[i]=sc.nextInt();
        }
    }

    public void show(){
        for (int i = 0; i < 5; i++) {
            System.out.println("Internal Marks of " + (i+1) +" subject is: " + imarks[i]);
        }
    }
}

```

```

// Package SEE
package SEE;

import CIE.*;

import java.util.Scanner;

public class ExternalsSix extends Studentsix {
    Scanner sc = new Scanner(System.in);
    int [] emarks = new int[5];
    public ExternalsSix(String usn, String name, int sem){
        super(usn,name,sem);
        for (int i = 0; i < 5; i++) {
            System.out.println("Enter the External marks of " +" subject:" + (i+1));
            emarks[i]=sc.nextInt();
        }
    }

    public void show(){
        for (int i = 0; i < 5; i++) {
            System.out.println("External Marks of subject" + (i+1) + ":" +emarks[i]);
        }
    }
}

```

```

}

//Main Class
import CIE.InternalsSix;
import SEE.ExternalsSix;

import java.util.Scanner;

public class LabSix {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n;
        System.out.println("Enter the total no.of Students here: ");
        n=sc.nextInt();

        InternalsSix [] arr1 = new InternalsSix[n];
        ExternalsSix [] arr2= new ExternalsSix[n];

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

            System.out.println("Enter the name of the student " + (i+1) + ":");
            String name = sc.next();

            System.out.println("Enter the sem of the student " + (i+1) + ":");
            int sem =sc.nextInt();

            arr1[i] = new InternalsSix(usn,name,sem);
            arr2[i] = new ExternalsSix(usn,name,sem);
        }

        for (int i = 0; i < n; i++) {
            System.out.println("Marks of Student" +(i+1) +":");
            arr1[i].show();
            System.out.println();
            arr2[i].show();
            System.out.println();
        }
    }
}

```

Output:

```

C:\Users\STUDENT\AppData\Local\Programs\Eclipse Adoptium\jdk-21.0.4-hotspot\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.2.
Enter the total no.of Students here:
1
Enter the USN of the student 1:
1bm23cs133
Enter the name of the student 1:
raghav
Enter the sem of the student 1:
5
Enter the Internal marks of 1 subject:
100
Enter the Internal marks of 2 subject:
100
Enter the Internal marks of 3 subject:
100
Enter the Internal marks of 4 subject:
100
Enter the Internal marks of 5 subject:
100
Enter the External marks of subject:1
98
Enter the External marks of subject:2
90
Enter the External marks of subject:3
100
Enter the External marks of subject:4
98
Enter the External marks of subject:5
96

```

```

Marks of Student1:
Internal Marks of 1 subject is: 100
Internal Marks of 2 subject is: 100
Internal Marks of 3 subject is: 100
Internal Marks of 4 subject is: 100
Internal Marks of 5 subject is: 100

External Marks of subject1:98
External Marks of subject2:90
External Marks of subject3:100
External Marks of subject4:98
External Marks of subject5:96

Name: K Raghavendra S Adiga
USN: 1BM23CS133

Process finished with exit code 0

```

Program 7

Exception Handling in Inheritance

Algorithm:

LAB

CLASSMATE
Date _____
Page _____

25/11/24

Q7. WAP that handles exceptions in inheritance tree. Create a base class Father & child class Son. Implement constructor that takes age & throws exception if age < 0 & Father's age <= son's age.

```

class AgeEx extends Exception {
    public AgeEx (String str) {
        super (str);
    }
}

class Father {
    Scanner sc = new Scanner (System.in);
    int fage;
    public Father () {
        try {
            System.out.print ("Enter Father's Age: ");
            fage = sc.nextInt();
            if (fage < 0) throw new AgeEx ("Age is");
        }
        catch (AgeEx e) {
            System.out.print ("Re-Enter Father's Age: ");
            fage = sc.nextInt();
        }
    }
}

class Son extends Father {
    Scanner sc = new Scanner (System.in);
    int sage;
    public Son () {
        super ();
        try {
            System.out.print ("Enter Son's Age: ");
            sage = sc.nextInt();
        }
    }
}

```

```

    if (sage > fage) throw new AgeEx("Son's age  
is greater than Father's");
}

```

```

}

```

```

catch (AgeEx e) {
    sout(e);
    System.out.println("Re-Enter Son's Age: ");
    sage = sc.nextInt();
}

```

```

}

```

```

public void details() {
    System.out.print("Father's Age: " + fage);
    System.out.print("Son's Age: " + sage);
}

```

```

}

```

```

public class Main {
    public static void main (String[] args) {
        Son ob = new Son();
        ob.details();
    }
}

```

```

}

```

```

}

```

o/p

Enter Father's age: -5

Age is negative

ReEnter Father's Age: 50

Enter Son's Age: 60

Son's age is greater than Father's

Re-Enter Son's Age: 30

Father's Age: 50

Son's Age: 30

Code:

```
package Lab;

import java.util.Scanner;
import java.util.concurrent.ExecutionException;

class AgeException extends java.lang.Exception{
    public AgeException(String str){
        super(str);
    }
}

class Father{
    Scanner sc = new Scanner(System.in);
    int fatherAge;

    public Father(){
        try {
            System.out.println("Enter the age of Father here: ");
            fatherAge = sc.nextInt();
            if (fatherAge < 0) throw new AgeException("Father Age Can't be negative: ");
        }
        catch(AgeException e){
            System.out.println(e);
            System.out.println("Re-Enter Father's age here: ");
            fatherAge = sc.nextInt();
        }
    }

    public void fatherDetails(){
        System.out.println("Father's age is: " + fatherAge);
    }
}

class Son extends Father{
    Scanner sc = new Scanner(System.in);
    int sonAge;
    public Son() {
        super();
        try{
            System.out.println("Enter son age: ");
            sonAge = sc.nextInt();
            if(sonAge>fatherAge) throw new AgeException("Son age can't be greater than Father's age");
        }
    }
}
```

```

        catch (AgeException e){
            System.out.println(e);
            System.out.println("Re-Enter Son's age: ");
            sonAge = sc.nextInt();
        }
    }

    public void sonDetails(){
        System.out.println("Son's age is: " + sonAge);
    }
}

public class Exception {
    public static void main(String[] args) {
        Son sonObj = new Son();

        sonObj.sonDetails();
        sonObj.fatherDetails();
        System.out.println();
        System.out.println("Name: Raghavendra Adiga");
        System.out.println("USN: 1BM23CS133");
    }
}

```

Output:

```

"C:\Program Files\Java\jdk-22\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.1.3\lib\idea_rt.jar=61346:
Enter the age of Father here:
-
Lab.AgeException: Father Age Can't be negative:
Re-Enter Father's age here:
-
Enter son age:
-
Lab.AgeException: Son age can't be greater than Father's age
Re-Enter Son's age:
-
Son's age is: 30
Father's age is: 50

Name: Raghavendra Adiga
USN: 1BM23CS133

Process finished with exit code 0

```

Program 8

Demonstrating Multi-Threading

Algorithm:

25-11-2024

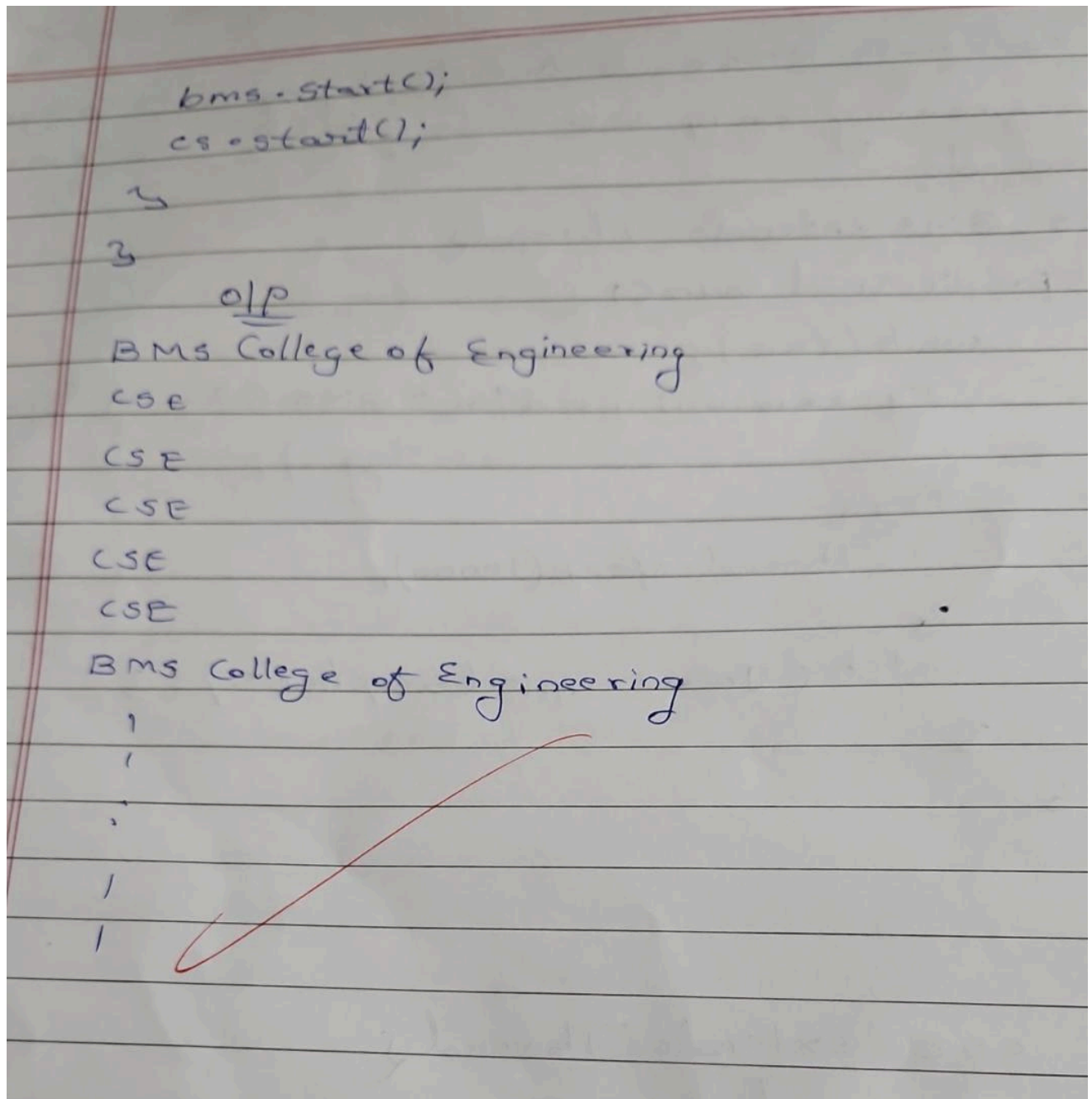
Q8. WAP to create 2 threads one displaying BMS College of Engineering every 10 seconds & other "CSE" every 2 seconds.

```

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

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

public class Main {
    public static void main (String c=
        BMS bms = new BMS();
        CSE cs = new CSE();
    
```



Code:

```
package Lab;

class BMS extends Thread{

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

```

    }
    catch (InterruptedException e){}
  }
}

}

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

```

```

public class ThreadingProgram {
  public static void main(String[] args) {
    BMS bms = new BMS();
    CSE cse = new CSE();
    System.out.println("Name: K Raghavendra S Adiga");
    System.out.println("USN: 1BM23CS133");
    bms.start();
    cse.start();
  }
}

```

Output:

```

"C:\Program Files\Java\jdk-22\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.1.3\lib\idea_rt.jar=61489
Name: K Raghavendra S Adiga
USN: 1BM23CS133
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE

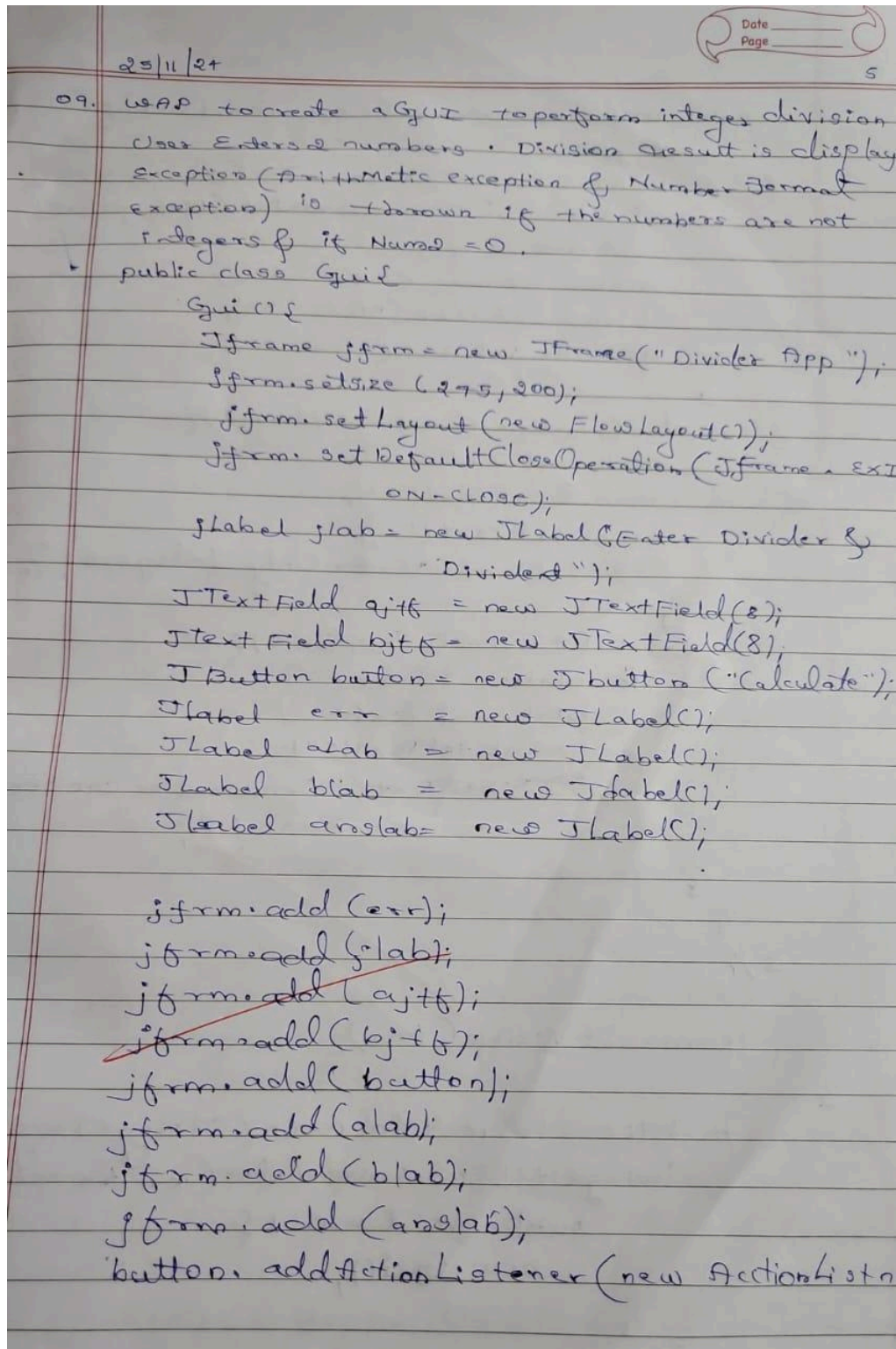
Process finished with exit code 130

```

Program 9

Graphical User Interface Creation Using JAVA

Algorithm:




```

public void actionPerformed(ActionEvent evt) {
    try {
        int a = Integer.parseInt(jTextField1.getText());
        int b = Integer.parseInt(jTextField2.getText());
        int ans = a/b;
        alab.setText("A = " + a);
        blab.setText("B = " + b);
        ansLab.setText("C = " + ans);
        err.setText("");
    }
    catch (NumberFormatException e) {
        alab.setText("");
        blab.setText("");
        ansLab.setText("");
        err.setText("Enter Only Integers!");
    }
    catch (ArithmeticException e) {
        alab.setText("");
        blab.setText("");
        ansLab.setText("");
        err.setText("Denominator can't be  
Zero!");
    }
}
}
}
}
}

```

```

jform.setVisible(true);

```

```

}

```

```

public static void main(String[] args) {
    SwingUtilities.invokeLater(new Runnable() {
        public void run() {
            new Gui();
        }
    });
}
}

```

```

}
}

```

Code:

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

public class SwingDemo {
    SwingDemo() {
        // Create JFrame container
        JFrame jfrm = new JFrame("Divider App");
        jfrm.setSize(275, 200); // Increased size to accommodate the result labels properly
        jfrm.setLayout(new FlowLayout());

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

        // Text label for instructions
        JLabel jlab = new JLabel("Enter the divider and dividend:");

        // Add text fields for both numbers (divider and dividend)
        JTextField ajtf = new JTextField(8);
        JTextField bjtf = new JTextField(8);

        // Button to perform the calculation
        JButton button = new JButton("Calculate");

        // Labels to display error and results
        JLabel err = new JLabel();
        JLabel alab = new JLabel();
        JLabel blab = new JLabel();
        JLabel anslab = new JLabel();

        // Add components to the JFrame in order
        jfrm.add(err); // To display error messages
        jfrm.add(jlab); // Instruction label
        jfrm.add(ajtf); // Text field for divider
        jfrm.add(bjtf); // Text field for dividend
        jfrm.add(button); // Calculate button
        jfrm.add(alab); // Result for A
        jfrm.add(blab); // Result for B
        jfrm.add(anslab); // Result for answer

        // Action listener for the Calculate button
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent evt) {
                try {
                    // Parse input from the text fields
```

```

int a = Integer.parseInt(ajtf.getText());
int b = Integer.parseInt(bjtf.getText());

// Perform division and display results
int ans = a / b;

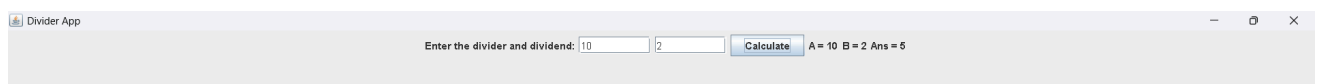
// Set the labels to show the values of A, B, and the result
alab.setText("A = " + a);
blab.setText("B = " + b);
anslab.setText("Ans = " + ans);
err.setText(""); // Clear any previous error message
} catch (NumberFormatException e) {
    // Handle invalid number format input (e.g., non-integer input)
    alab.setText("");
    blab.setText("");
    anslab.setText("");
    err.setText("Enter Only Integers!");
} catch (ArithmeticException e) {
    // Handle division by zero
    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
    System.out.println("Name: K Raghavendra S Adiga");
    System.out.println("USN: 1BM23CS133");
    SwingUtilities.invokeLater(new Runnable() {
        public void run() {
            new SwingDemo();
        }
    });
}
}

```

Output:



Program 10.A

Demonstrate Inter Process Communication

Algorithm:

25-11-2021

10]

A. Demonstrate Inter-process Communication.

```

package Lab2;
class A {
    synchronized void foo (Lab2.B b) {
        String name = Thread.currentThread().
            getName();
        System.out.println ("Name: " + "Entered A.foo");
        try {
            Thread.sleep(1000);
        }
        catch (InterruptedException 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 foo (Lab2.A a) {
        String name = Thread.currentThread().get
            Name();
        System.out.print (name + "Entered B.foo");
        try {
            Thread.sleep(1000);
        }
        catch (InterruptedException e) {
            System.out.println ("B Interrupted");
        }
        System.out.println (name + "Trying to ca
            A.last()");
        a.last();
    }
}

```

```
void last() { System.out.println("Inside  
B.last"); }
```

```
}
```

```
public class IPC implements Runnable {  
    Lab2.A a = new Lab2.A();  
    Lab2.B b = new Lab2.B();
```

```
IPC() {
```

```
    Thread.currentThread().setName("Main  
Thread");
```

```
    Thread t = new Thread(this, "RacingThre  
ad");
```

```
    a.foo(b);
```

```
    System.out.println("Back to main thread  
");
```

```
}
```

```
public void run() {
```

```
    b.bar(a);
```

```
}
```

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

```
}
```

```
}
```

O/P

RacingThread entered B.bar

Main Thread entered A.foo

MainThread trying to call B.last()

Inside B.last

Back to mainThread

RacingThread trying to call A.last()

Inside A.last

Code:

```
package Lab2;

class A {
    synchronized void foo(Lab2.B b) {
        String name = Thread.currentThread().getName();
        System.out.println(name + " entered A.foo");

        try {
            Thread.sleep(1000); // This can throw InterruptedException
        } catch (InterruptedException 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(Lab2.A a) {
        String name = Thread.currentThread().getName();
        System.out.println(name + " entered B.bar");

        try {
            Thread.sleep(1000); // This can throw InterruptedException
        } catch (InterruptedException e) {
            System.out.println("B Interrupted");
        }

        System.out.println(name + " trying to call A.last()");
        a.last();
    }

    void last() {
        System.out.println("Inside B.last");
    }
}

public class IPS implements Runnable {
    Lab2.A a = new Lab2.A();
    Lab2.B b = new Lab2.B();
}
```



```

IPS() {
    // Set up the main thread and start the new thread
    Thread.currentThread().setName("MainThread");
    Thread t = new Thread(this, "RacingThread");
    t.start();

    // Main thread acquires lock on object a and calls foo
    a.foo(b);

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

@Override
public void run() {
    // This method will be run in the RacingThread
    b.bar(a);
}

public static void main(String[] args) {
    // The main method will create an instance of IPS and trigger the deadlock
    System.out.println("Name: K Raghavendra S Adiga");
    System.out.println("USN: 1BM23CS133");
    new IPS();
}
}

```

Output:

```

"C:\Program Files\Java\jdk-22\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.1.3\lib\idea_rt.jar=61632:
Name: K Raghavendra S Adiga
USN: 1BM23CS133
RacingThread entered B.bar
MainThread entered A.foo
MainThread trying to call B.last()
Inside B.last
Back in main thread
RacingThread trying to call A.last()
Inside A.last

Process finished with exit code 0

```

Program 10.B

Demonstrate Deadlock in Java

Algorithm:

25-11-2024

10. B] Demonstrate Deadlock.

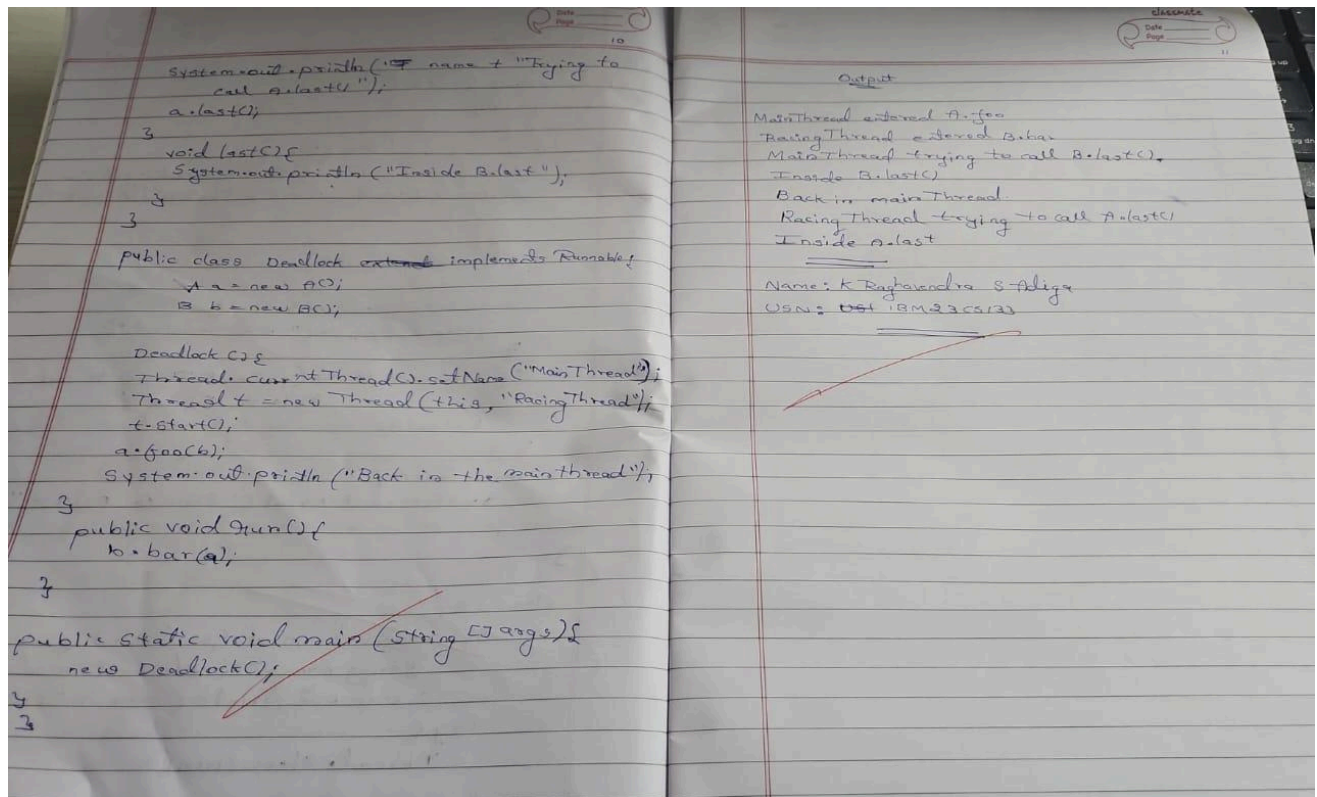
```

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

        try {
            Thread.sleep(1000);
        }
        catch (InterruptedException 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 foo (A a) {
        String name = Thread.currentThread().getName();
        System.out.println (name + "entered B.foo");
        try {
            Thread.sleep(1000);
        }
        catch (InterruptedException e) {
            System.out.println ("B Interrupted");
        }
    }
}

```



Code:

package Lab;

```

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

        try {
            Thread.sleep(1000); // This may throw InterruptedException
        } catch (InterruptedException 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); // This may throw InterruptedException
    } catch (InterruptedException e) {
        System.out.println("B Interrupted");
    }

    System.out.println(name + " trying to call A.last()");
    a.last();
}

void last() {
    System.out.println("Inside B.last");
}
}

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

    Deadlock() {
        // Start the thread and set its name
        Thread.currentThread().setName("MainThread");
        Thread t = new Thread(this, "RacingThread");
        t.start();

        // Main thread acquires lock on a and calls foo
        a.foo(b);

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

    @Override
    public void run() {
        // This method runs in the new thread
        b.bar(a);
    }

    public static void main(String[] args) {
        // Create the Deadlock instance and trigger the deadlock scenario
        System.out.println("Name: K Raghavendra S Adiga");
        System.out.println("USN: 1BM23CS133");
        new Deadlock();
    }
}

```

Output:

```
"C:\Program Files\Java\jdk-22\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.1.3\lib\idea_rt.jar=61816
Name: K Raghavendra S Adiga
USN: 1BM23CS133
RacingThread entered B.bar
MainThread entered A.foo
MainThread trying to call B.last()
Inside B.last
Back in main thread
RacingThread trying to call A.last()
Inside A.last

Process finished with exit code 0
```