B.M.S. College of Engineering

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)

Bull Temple Road, Basavanagudi, Bengaluru – 560019

Department of

Computer Science & Engineering

Lab Programs Report

Course Title: Object Oriented Java Programming

Course Code: 23CS3PCOOJ

By,

Sparsha Kadaba (1BM22CS287)

B.M.S. College of Engineering

(Autonomous College Affiliated to Visvesvaraya Technological University, Belgaum)

Bull Temple Road, Basavanagudi, Bengaluru – 560019

Department of
Computer Science & Engineering

CERTIFICATE

This is to certify that the report on "**Java Lab Programs**" has been carried out by **Sparsha Kadaba** (**1BM22CS287)** as a part of AAT for the course **Object Oriented Java Programming** with course code **23CS3PCOOJ**, Computer Science and Engineering from Visvesvaraya Technological University, Belgaum during the year 2023–24. It is certified that all corrections/suggestions indicated for Internal Assessments have been incorporated in the report.

Sparsha Kadaba
1BM22CS287

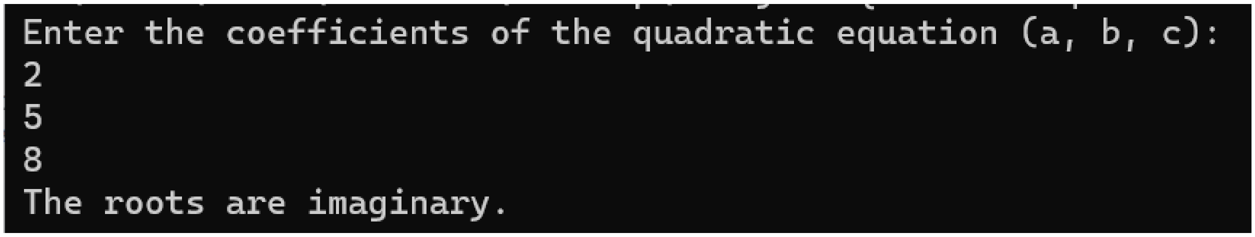
Shravya AR
Assistant Professor
Department of CSE
BMSCE,

INDEX

PROGRAM 1

Develop a Java program that prints all real solutions to the quadratic equation ax2+bx+c=0. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions.

```
import java.util.Scanner;

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

        System.out.println("Enter the coefficients of the quadratic equation (a, b, c):");
        double a = scanner.nextDouble();
        double b = scanner.nextDouble();
        double c = scanner.nextDouble();

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

        if (discriminant < 0) {
            System.out.println("The roots are imaginary.");
        } else if (discriminant == 0) {
            System.out.println("The roots are real and equal.");
            double root = -b / (2 * a);
            System.out.println("The only real solution is: " + root);
        } else {
            System.out.println("The roots are real and unequal.");
            double root1 = (-b + Math.sqrt(discriminant)) / (2 * a);
            double root2 = (-b - Math.sqrt(discriminant)) / (2 * a);
            System.out.println("The real solutions are: " + root1 + " and " + root2);
        }

        scanner.close();
    }
}
```

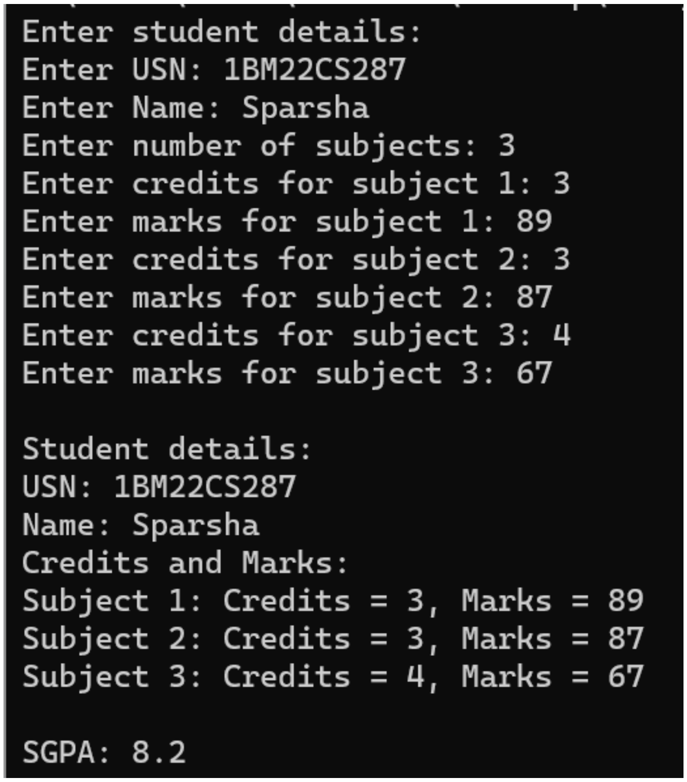


```
Enter the coefficients of the quadratic equation (a, b, c):
2
5
8
The roots are imaginary.
```

PROGRAM 2

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 a method to calculate SGPA of a student.

```
import java.util.Scanner;

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

    public Student(String usn, String name, int[] credits, int[] marks) {
        this.usn = usn;
        this.name = name;
        this.credits = credits;
        this.marks = marks;
    }

    public void acceptDetails() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter USN: ");
        this.usn = scanner.nextLine();
        System.out.print("Enter Name: ");
        this.name = scanner.nextLine();
        System.out.print("Enter number of subjects: ");
        int numSubjects = scanner.nextInt();
        this.credits = new int[numSubjects];
        this.marks = new int[numSubjects];
        for (int i = 0; i < numSubjects; i++) {
            System.out.print("Enter credits for subject " + (i + 1) + ": ");
            this.credits[i] = scanner.nextInt();
            System.out.print("Enter marks for subject " + (i + 1) + ": ");
            this.marks[i] = scanner.nextInt();
        }
    }

    public void displayDetails() {
        System.out.println("USN: " + this.usn);
        System.out.println("Name: " + this.name);
        System.out.println("Credits and Marks:");
        for (int i = 0; i < credits.length; i++) {
            System.out.println("Subject " + (i + 1) + ": Credits = " + credits[i] + ", Marks = " + marks[i]);
        }
    }

    public double calculateSGPA() {
        double totalCredits = 0;
        double totalGradePoints = 0;
        for (int i = 0; i < credits.length; i++) {
            totalCredits += credits[i];
            totalGradePoints += calculateGradePoint(marks[i]) * credits[i];
        }
        return totalGradePoints / totalCredits;
    }

    private double calculateGradePoint(int marks) {
        if (marks >= 90) {
            return 10;
        } else if (marks >= 80) {
            return 9;
        } else if (marks >= 70) {
            return 8;
        } else if (marks >= 60) {
            return 7;
        } else if (marks >= 50) {
            return 6;
        } else if (marks >= 40) {
            return 5;
        } else {
            return 0;
        }
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter student details:");
        System.out.print("Enter USN: ");
        String usn = scanner.nextLine();
        System.out.print("Enter Name: ");
        String name = scanner.nextLine();
        System.out.print("Enter number of subjects: ");
        int numSubjects = scanner.nextInt();
        int[] credits = new int[numSubjects];
        int[] marks = new int[numSubjects];
        for (int i = 0; i < numSubjects; i++) {
            System.out.print("Enter credits for subject " + (i + 1) + ": ");
            credits[i] = scanner.nextInt();
            System.out.print("Enter marks for subject " + (i + 1) + ": ");
            marks[i] = scanner.nextInt();
        }
        Student student = new Student(usn, name, credits, marks);
        System.out.println("\nStudent details:");
        student.displayDetails();
        double sgpa = student.calculateSGPA();
        System.out.println("\nSGPA: " + sgpa);
        scanner.close();
    }
}
```

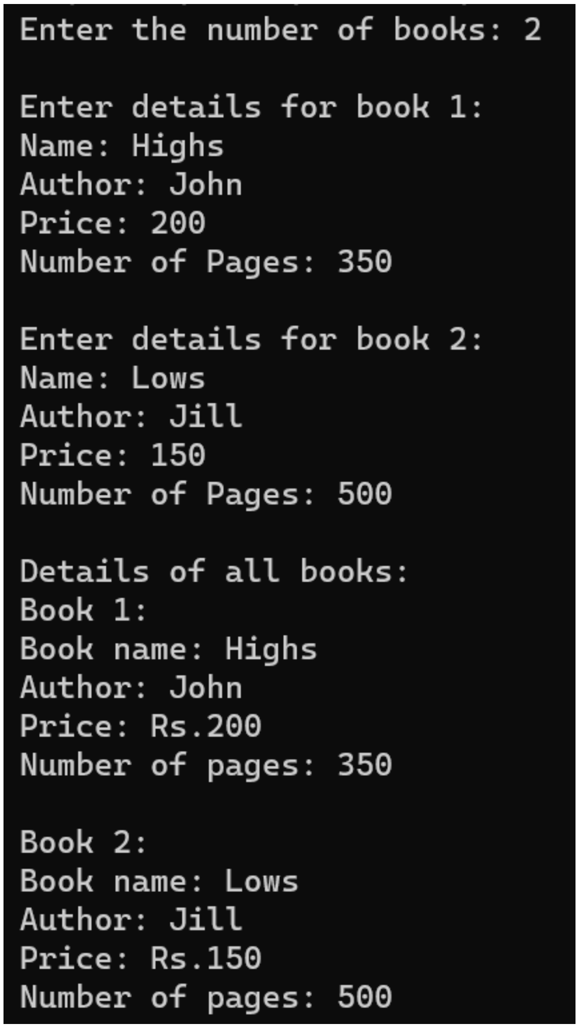


```
Enter student details:
Enter USN: 1BM22CS287
Enter Name: Sparsha
Enter number of subjects: 3
Enter credits for subject 1: 3
Enter marks for subject 1: 89
Enter credits for subject 2: 3
Enter marks for subject 2: 87
Enter credits for subject 3: 4
Enter marks for subject 3: 67

Student details:
USN: 1BM22CS287
Name: Sparsha
Credits and Marks:
Subject 1: Credits = 3, Marks = 89
Subject 2: Credits = 3, Marks = 87
Subject 3: Credits = 4, Marks = 67

SGPA: 8.2
```

PROGRAM 3

Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.

```
import java.util.Scanner;

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

    Books(String name, String author, int price, int num_pages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.num_pages = num_pages;
    }

    @Override
    public String toString() {
        return "Book name: " + this.name + "\n" +
               "Author: " + this.author + "\n" +
               "Price: Rs." + this.price + "\n" +
               "Number of pages: " + this.num_pages + "\n";
    }
}

public class MainBooks {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of books: ");
        int n = scanner.nextInt();
        scanner.nextLine();
        Books[] books = new Books[n];
        for (int i = 0; i < n; i++) {
            System.out.println("\nEnter details for book " + (i + 1) + ":");
            System.out.print("Name: ");
            String name = scanner.nextLine();
            System.out.print("Author: ");
            String author = scanner.nextLine();
            System.out.print("Price: ");
            int price = scanner.nextInt();
            scanner.nextLine();
            System.out.print("Number of Pages: ");
            int num_pages = scanner.nextInt();
            scanner.nextLine();
            books[i] = new Books(name, author, price, num_pages);
        }

        System.out.println("\nDetails of all books:");
        for (int i = 0; i < n; i++) {
            System.out.println("Book " + (i + 1) + ":\n" + books[i]);
        }

        scanner.close();
    }
}
```



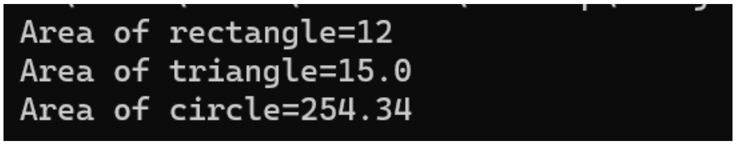
```
Enter the number of books: 2

Enter details for book 1:
Name: Highs
Author: John
Price: 200
Number of Pages: 350

Enter details for book 2:
Name: Lows
Author: Jill
Price: 150
Number of Pages: 500

Details of all books:
Book 1:
Book name: Highs
Author: John
Price: Rs.200
Number of pages: 350

Book 2:
Book name: Lows
Author: Jill
Price: Rs.150
Number of pages: 500
```

PROGRAM 4

Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape.

```
abstract class Shape
{
        int a,b;
        abstract void printArea();
}

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

class Triangle extends Shape
{
        void printArea()
        {
                System.out.println("Area of triangle="+(0.5*a*b));
        }
}

class Circle extends Shape
{
        void printArea()
        {
                System.out.println("Area of circle="+(3.14*a*a));
        }
}

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

```
    {
            Rectangle r=new Rectangle();
            r.a=3; r.b=4;
            r.printArea();

            Triangle t=new Triangle();
            t.a=5; t.b=6;
            t.printArea();

            Circle c=new Circle();
            c.a=9;
            c.printArea();
    }
}
```



```
Area of rectangle=12
Area of triangle=15.0
Area of circle=254.34
```

PROGRAM 5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed.

Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

a) Accept deposit from customer and update the balance.

b) Display the balance.

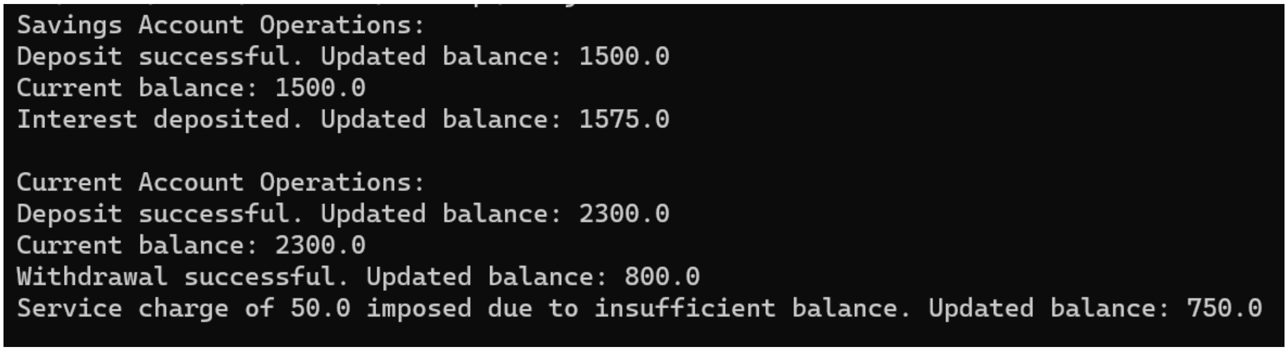
c) Compute and deposit interest

d) Permit withdrawal and update the balance

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

```
import java.util.Scanner;

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

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

    void deposit(double amount) {
        balance += amount;
        System.out.println("Deposit successful. Updated balance: " + balance);
    }

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

    void withdraw(double amount) {
        if (balance - amount < 0) {
            System.out.println("Insufficient balance. Withdrawal failed.");
        } else {
            balance -= amount;
            System.out.println("Withdrawal successful. Updated balance: " + balance);
        }
    }

    void depositInterest() {
        System.out.println("Interest not applicable for current account.");
    }

    void checkMinimumBalance() {
        // Minimum balance check and penalty not applicable for current account
    }
}

class CurrentAccount extends Account {
    double minimumBalance;
    double serviceCharge;

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

    @Override
    void checkMinimumBalance() {
        if (balance < minimumBalance) {
            balance -= serviceCharge;
            System.out.println("Service charge of " + serviceCharge + " imposed due to insufficient balance. Updated balance: " + balance);
        }
    }
}
```

```
class SavingsAccount extends Account {
    double interestRate;

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

    @Override
    void depositInterest() {
        double interest = balance * (interestRate / 100);
        balance += interest;
        System.out.println("Interest deposited. Updated balance: " + balance);
    }
}

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

        SavingsAccount savingsAccount = new SavingsAccount("John Doe", 123456789, "Savings", 1000, 5);

        CurrentAccount currentAccount = new CurrentAccount("Jane Smith", 987654321, "Current", 2000, 1000, 50);

        System.out.println("Savings Account Operations:");
        savingsAccount.deposit(500);
        savingsAccount.displayBalance();
        savingsAccount.depositInterest();

        System.out.println("\nCurrent Account Operations:");
        currentAccount.deposit(300);
        currentAccount.displayBalance();
        currentAccount.checkMinimumBalance();
        currentAccount.withdraw(1500);
        currentAccount.checkMinimumBalance();

        scanner.close();
    }
}
```



```
Savings Account Operations:
Deposit successful. Updated balance: 1500.0
Current balance: 1500.0
Interest deposited. Updated balance: 1575.0

Current Account Operations:
Deposit successful. Updated balance: 2300.0
Current balance: 2300.0
Withdrawal successful. Updated balance: 800.0
Service charge of 50.0 imposed due to insufficient balance. Updated balance: 750.0
```

PROGRAM 6

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

```
//Student.java
package CIE;
import java.util.Scanner;
public class Student {
protected String usn = new String();
protected String name = new String();
protected int sem;

public void inputStudentDetails()
{
Scanner sc=new Scanner(System.in);
System.out.println("Enter student usn");
usn=sc.next();
System.out.println("Enter student name");
name=sc.next();
System.out.println("Enter student semester");
sem=sc.nextInt();
}
public void displayStudentDetails() {
System.out.println("student usn:"+ usn);
System.out.println("student name:"+name);
System.out.println("student sem"+ sem);
}
}

//Internals.java
package CIE;
import java.util.Scanner;
public class Internals extends Student {
protected int marks[] = new int[5];
public void inputCIEmarks()
{
Scanner sc=new Scanner(System.in);
for (int i=0;i<5;i++)
{
System.out.println("Enter 5 subject marks");
marks[i]=sc.nextInt();
}
}
```

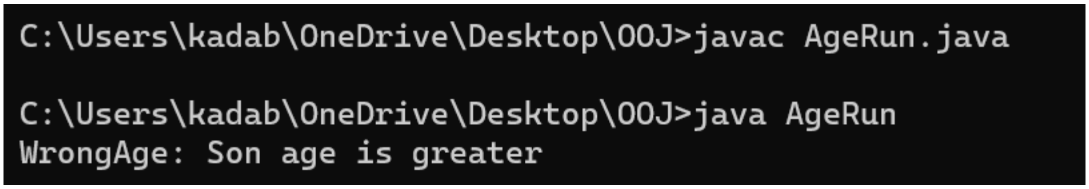
```
}

//Externals.java
package SEE;
import CIE.Internals;
import java.util.Scanner;
public class Externals extends Internals {
protected int marks[];
protected int finalMarks[];
public Externals() {
marks = new int[5];
finalMarks = new int[5];
}
public void inputSEEmarks()
{
Scanner sc = new Scanner(System.in);
for(int i=0;i<5;i++)
{
System.out.print(“Subject “+(i+1)+” marks: “);
marks[i] = sc.nextInt();
}
}
public void calculateFinalMarks() {
for(int i=0;i<5;i++)
finalMarks[i] = marks[i]/2 + super.marks[i];
}
public void displayFinalMarks() {
displayStudentDetails();
for(int i=0;i<5;i++)

System.out.println(“Subject “+ (i+1) + “:” + finalMarks[i]);
}
}

//Main1.java
import SEE.Externals;
class Main1 {
public static void main(String args[])
{
int numOfStudents = 2;
Externals finalMarks[] = new Externals[numOfStudents];
for(int i=0;i<numOfStudents;i++)
{
finalMarks[i] = new Externals();
finalMarks[i].inputStudentDetails();
System.out.println(“Enter CIE marks”);
finalMarks[i].inputCIEmarks();
System.out.println(“Enter SEE marks”);
finalMarks[i].inputSEEmarks();
}
System.out.println(“Displaying data:\n”);
for(int i=0;i<numOfStudents;i++)
{
```

```
finalMarks[i].calculateFinalMarks();
finalMarks[i].displayFinalMarks();
}
}
}
```

PROGRAM 7

Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.

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

class Father
{
int age;
Father(int age) throws WrongAge
{
if(age<0)
{
throw new WrongAge("No negative");
}
this.age=age;
}
}

class Son extends Father
{
int SonAge;
Son(int fatherAge, int SonAge) throws WrongAge
{
super(fatherAge);
if(SonAge>=fatherAge)
{
throw new WrongAge("Son age is greater");
}
this.SonAge=SonAge;
}
}

class AgeRun
{
public static void main( String args[])
{
try
{
Father f= new Father(50);
Son s= new Son(50, 60);
}
catch(WrongAge e)
{
System.out.println(e);
}
}
}
```



```
C:\Users\kadab\OneDrive\Desktop\OOJ>javac AgeRun.java

C:\Users\kadab\OneDrive\Desktop\OOJ>java AgeRun
WrongAge: Son age is greater
```

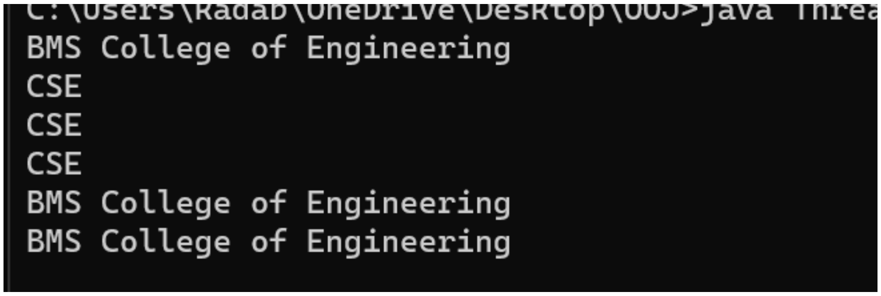
```
}
```

PROGRAM 8

Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

```
import java.io.*;

class B 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(e);
		}

	}
}

class C extends Thread
{
	public void run()
	{
		try
		{
			for(int i = 0; i < 3; i++)
			{
				System.out.println("CSE ");
				Thread.sleep(2000);
			}
		}
		catch (InterruptedException e)
		{
			System.out.println(e);
		}

	}
}

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

```
        {
                B b = new B();
                C c = new C();

                b.start();
                c.start();
        }
}
```



```
C:\Users\Radab\OneDrive\Desktop\OOJ>java Threa
BMS College of Engineering
CSE
CSE
CSE
BMS College of Engineering
BMS College of Engineering
```