DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

B M S COLLEGE OF ENGINEERING (AUTONOMOUS COLLEGE UNDER VTU, BELGAUM)

BANGALORE – 560019 2023-24

LAB REPORT OF OBJECT-ORIENTED JAVA PROGRAMMING (23CS3PCOOJ)

LAB REPORT

BY

NAME	USN
Harshith. B.	2023BMS02519

Course Instructor

Shravya. A. R., Assistant Professor,

Dept. of CSE, BMSCE

LAB PROGRAM 1:

Develop a Java program that prints all real solutions to the quadratic equation $ax^2+bx+c = 0$. Read in 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.

CODE:

```java
import java.util.Scanner;

class QuadRoots {
    double a, b, c, firstroot, secondroot;

    QuadRoots(double a, double b, double c) {
        this.a = a;
        this.b = b;
        this.c = c;
    }

    void Eval() {
        double det = b * b - 4 * a * c;

        if (det > 0) {
            firstroot = (-b + Math.sqrt(det)) / (2 * a);
            secondroot = (-b - Math.sqrt(det)) / (2 * a);
            System.out.format("First Root = %.2f and Second Root = %.2f", firstroot, secondroot);
        }
        else if (det == 0) {
            firstroot = secondroot = -b / (2 * a);
            System.out.format("First Root = Second Root = %.2f;", firstroot);
        }
        else {
            double real = -b / (2 * a);
            double img = Math.sqrt(-det) / (2 * a);
            System.out.printf("First Root = %.2f+(%.2f)i", real, img);
            System.out.printf("\nSecond Root = %.2f-(%.2f)i", real, img);
        }
    }

}


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

        System.out.println("NAME: HARSHITH B");
        System.out.println("USN:  2023BMS02519");

        double a, b, c;
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a : ");
        a = sc.nextDouble();
        System.out.print("Enter b : ");
        b = sc.nextDouble();
        System.out.print("Enter c : ");
        c = sc.nextDouble();

        QuadRoots q = new QuadRoots(a, b, c);
        q.Eval();

        sc.close();
    }
}
```

OUTPUT:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac Quadratic.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java QRun
NAME: HARSHITH B
USN:  2023BMS02519
Enter a : 3
Enter b : 25
Enter c : -18
First Root = 0.67 and Second Root = -9.00
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java QRun
NAME: HARSHITH B
USN:  2023BMS02519
Enter a : 2
Enter b : 0
Enter c : -15
First Root = 2.74 and Second Root = -2.74
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java QRun
NAME: HARSHITH B
USN:  2023BMS02519
Enter a : 5
Enter b : -5
Enter c : -10
First Root = 2.00 and Second Root = -1.00
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java QRun
NAME: HARSHITH B
USN:  2023BMS02519
Enter a : 1
Enter b : -2
Enter c : 10
First Root = 1.00+(3.00)i
Second Root = 1.00-(3.00)i
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```

LAB PROGRAM 2:

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

CODE:

```java
import java.util.Scanner;

class Student {
    private String usn;
    private String name;
    private int[] credits;
    private 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 sc) {
        System.out.print("Enter USN:  ");
        this.usn = sc.next();

        System.out.print("Enter Name:  ");
        this.name = sc.next();
        sc.next();

        this.marks = new int[credits.length];
        for (int i = 0; i < credits.length; i++) {
            System.out.print("Enter marks for subject " + (i + 1) + ":  ");
            this.marks[i] = sc.nextInt();
        }
    }

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

```

```java

    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 mark) {
        if (mark >= 90) return 10;
        else if (mark >= 80) return 9;
        else if (mark >= 70) return 8;
        else if (mark >= 60) return 7;
        else if (mark >= 50) return 6;
        else if (mark >= 40) return 5;
        else return 0;
    }
}

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

        System.out.println("NAME: HARSHITH B");
        System.out.println("USN:  2023BMS02519\n");

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the number of subjects:  ");
        int numOfSubjects = sc.nextInt();

        int[] credits = new int[numOfSubjects];
        System.out.println("Enter credits for each subject:");
        for (int i = 0; i < numOfSubjects; i++) {
            credits[i] = sc.nextInt();
        }

        Student student = new Student("", "", credits, new int[numOfSubjects]);
        student.acceptDetails(sc);
        student.displayDetails();
        System.out.println("SGPA: " + student.calculateSGPA());

        sc.close();
    }
}

```

OUTPUT:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac Student.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java SRun
NAME: HARSHITH B
USN:  2023BMS02519

Enter the number of subjects:  6
Enter credits for each subject:
4
3
4
4
2
1
Enter USN:  2023BMS02519
Enter Name:  Harshith B
Enter marks for subject 1:  99
Enter marks for subject 2:  97
Enter marks for subject 3:  78
Enter marks for subject 4:  89
Enter marks for subject 5:  67
Enter marks for subject 6:  98
USN: 2023BMS02519
Name: Harshith
Credits: 4, 3, 4, 4, 2, 1
Marks: 99, 97, 78, 89, 67, 98
SGPA: 9.0
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```

LAB 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.

CODE:

```java
import java.util.Scanner;

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

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

    public String toString() {
        return "Book Name: " + name + "\n" +
               "Author Name: " + author + "\n" +
               "Price: " + price + "\n" +
               "Number of Pages: " + numPages + "\n";
    }
}

class BRun{
    public static void main(String[] args) {
        System.out.println("NAME: HARSHITH B");
        System.out.println("USN: 2023BMS02519\n");

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

```

```
34
35          System.out.print("Enter the number of books: ");
36          n = sc.nextInt();
37          sc.nextLine();
38
39          Books[] b = new Books[n];
40
41          for(int i = 0; i < n; i++) {
42              System.out.println("Books " + (i + 1) + ": ");
43              System.out.print("Enter name of the book: ");
44              name = sc.nextLine();
45              System.out.print("Enter Author: ");
46              author = sc.nextLine();
47              System.out.print("Enter price: ");
48              price = sc.nextInt();
49              sc.nextLine();
50              System.out.print("Enter number of pages: ");
51              numPages = sc.nextInt();
52              sc.nextLine();
53              b[i] = new Books(name, author, price, numPages);
54          }
55
56          for (int i = 0; i < n; i++) {
57              System.out.println("Book: " + (i + 1) + "\n" + b[i]);
58          }
59
60          sc.close();
61      }
62  }
63
```

OUTPUT:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac Book.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java BRun
NAME: HARSHITH B
USN: 2023BMS02519

Enter the number of books: 2
Books 1:
Enter name of the book: IKIGAI: The Japanese secret to a lond and happy life.
Enter Author: Hector Garcia AND Francesc Miralles
Enter price: 99
Enter number of pages: 88
Books 2:
Enter name of the book: The Psychology of Money
Enter Author: Morgan Housel
Enter price: 149
Enter number of pages: 192
Book: 1
Book Name: IKIGAI: The Japanese secret to a lond and happy life.
Author Name: Hector Garcia AND Francesc Miralles
Price: 99
Number of Pages: 88

Book: 2
Book Name: The Psychology of Money
Author Name: Morgan Housel
Price: 149
Number of Pages: 192

PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```

LAB 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.

CODE:

```java
abstract class Shape {
    public int side1, side2;
    abstract void printArea();
}


class Rectangle extends Shape {
    Rectangle(int length, int breadth) {
        this.side1 = length;
        this.side2 = breadth;
    }
    void printArea() {
        System.out.println("The Area of Rectangle : " + (side1 * side2));
    }
}

class Triangle extends Shape {
    Triangle(int base, int height) {
        this.side1 = base;
        this.side2 = height;
    }
    void printArea() {
        System.out.println("The Area of Triangle : " + (0.5 * side1 * side2));
    }
}

class Circle extends Shape {
    Circle(int rad) {
        this.side1 = this.side2 = rad;
    }
    void printArea() {
        System.out.println("The Area of Circle : " + (3.14 * side1 * side2));
    }
}


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

        System.out.println("NAME: HARSHITH B");
        System.out.println("USN:  2023BMS02519\n");

        Rectangle r = new Rectangle(10, 10);
        Triangle t = new Triangle(5, 10);
        Circle c = new Circle(5);

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

OUTPUT:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac Shape.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java SRun
NAME: HARSHITH B
USN:  2023BMS02519

The Area of Rectangle : 100
The Area of Triangle : 25.0
The Area of Circle : 78.5
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```

LAB 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.

CODE:

```
import java.util.Scanner;

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

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

    abstract void deposit(double amount);

    abstract void displayBalance();

    abstract void computeInterest();

    abstract void withdraw(double amount);
}
```

```java
class SavingsAccount extends Account {
    SavingsAccount(String customerName, int accountNumber, String accountType, double balance) {
        super(customerName, accountNumber, accountType, balance);
    }

    void deposit(double amount) {
        balance += amount;
        System.out.println("Amount deposited: " + amount);
    }

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

    void computeInterest() {
        double interestRate = 0.05;
        double interest = balance * interestRate;
        balance += interest;
        System.out.println("Interest added: " + interest);
    }

    void withdraw(double amount) {
        if (balance < amount) {
            System.out.println("Insufficient balance");
        } else {
            balance -= amount;
            System.out.println("Amount withdrawn: " + amount);
        }
    }
}
```

```java
class CurrentAccount extends Account {
    double minimumBalance = 1000;
    double serviceCharge = 50;

    CurrentAccount(String customerName, int accountNumber, String accountType, double balance) {
        super(customerName, accountNumber, accountType, balance);
    }

    void deposit(double amount) {
        balance += amount;
        System.out.println("Amount deposited: " + amount);
    }

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

    void computeInterest() {
        System.out.println("Current account does not earn interest");
    }

    void withdraw(double amount) {
        if (balance - amount < minimumBalance) {
            System.out.println("Insufficient balance");
            balance -= serviceCharge;
            System.out.println("Service charge: " + serviceCharge);
        } else {
            balance -= amount;
            System.out.println("Amount withdrawn: " + amount);
        }
    }
}
```

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

        System.out.println("NAME: HARSHITH B");
        System.out.println("USN:  2023BMS02519\n");

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter customer name: ");
        String customerName = sc.nextLine();

        System.out.print("Enter account number: ");
        int accountNumber = sc.nextInt();

        System.out.print("Enter account type (savings/current): ");
        String accountType = sc.next();

        System.out.print("Enter initial balance: ");
        double balance = sc.nextDouble();

        Account account;
        if (accountType.equals("savings")) {
            account = new SavingsAccount(customerName, accountNumber, accountType, balance);
        } else {
            account = new CurrentAccount(customerName, accountNumber, accountType, balance);
        }

        System.out.println("\n###-MENU-###");
        System.out.println("1. Deposit");
        System.out.println("2. Display balance");
        System.out.println("3. Compute interest");
        System.out.println("4. Withdraw");
        System.out.println("5. Exit\n");

        while (true) {
            System.out.print("Enter choice: ");
            int choice = sc.nextInt();

            switch (choice) {
                case 1:
                    System.out.print("\nEnter amount to deposit: ");
                    double amount = sc.nextDouble();
                    account.deposit(amount);
                    break;
                case 2:
                    account.displayBalance();
                    break;
                case 3:
                    account.computeInterest();
                    break;
                case 4:
                    System.out.print("\nEnter amount to withdraw: ");
                    amount = sc.nextDouble();
                    account.withdraw(amount);
                    break;
                case 5:
                    sc.close();
                    System.exit(0);
                    break;
                default:
                    System.out.println("\nInvalid choice");
            }
        }
    }
}
```

OUTPUT:

OUTPUT1:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac Bank.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java Brun
NAME: HARSHITH B
USN:  2023BMS02519

Enter customer name: Harshith B
Enter account number: 4128
Enter account type (savings/current): current
Enter initial balance: 1000

###-MENU-###
1. Deposit
2. Display balance
3. Compute interest
4. Withdraw
5. Exit

Enter choice: 1

Enter amount to deposit: 2000
Amount deposited: 2000.0
Enter choice: 2
Balance: 3000.0
Enter choice: 3
Current account does not earn interest
Enter choice: 2
Balance: 3000.0
Enter choice: 4

Enter amount to withdraw: 300
Amount withdrawn: 300.0
Enter choice: 2
Balance: 2700.0
Enter choice: 5
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```

OUTPUT2:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac Bank.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java Brun
NAME: HARSHITH B
USN:  2023BMS02519

Enter customer name: Harshith B
Enter account number: 4128
Enter account type (savings/current): savings
Enter initial balance: 1000

###-MENU-###
1. Deposit
2. Display balance
3. Compute interest
4. Withdraw
5. Exit

Enter choice: 1

Enter amount to deposit: 2000
Amount deposited: 2000.0
Enter choice: 2
Balance: 3000.0
Enter choice: 3
Interest added: 150.0
Enter choice: 2
Balance: 3150.0
Enter choice: 4

Enter amount to withdraw: 300
Amount withdrawn: 300.0
Enter choice: 2
Balance: 2850.0
Enter choice: 5
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```

LAB PROGRAM 6:

Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals 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.

CODES:

CODE1:

```java
package CIE;

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

CODE2:

```java
package CIE;

public class Student {
    public String usn, name;
    public int sem;
}
```

CODE3:

```java
package SEE;

import CIE.Student;

public class External extends Student {
    public int[] seeMarks = new int[5];
}
```

CODE4:

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

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

        System.out.println("NAME: HARSHITH B");
        System.out.println("USN:  2023BMS02519\n");

        Scanner sc = new Scanner(System.in);

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

        Internals[] cieStudents = new Internals[n];
        External[] seeStudents = new External[n];
        for (int i = 0; i < n; i++) {
            cieStudents[i] = new Internals();
            seeStudents[i] = new External();

            System.out.println("Enter details for Student " + (i + 1) + ":");
            System.out.print("USN: ");
            cieStudents[i].usn = seeStudents[i].usn = sc.next();
            System.out.print("Name: ");
            cieStudents[i].name = seeStudents[i].name = sc.next();
            System.out.print("Semester: ");
            cieStudents[i].sem = seeStudents[i].sem = sc.nextInt();

            System.out.println("Enter Internal Marks for 5 courses:");
            for (int j = 0; j < 5; j++) {
                System.out.print("Course " + (j + 1) + ": ");
                cieStudents[i].internalMarks[j] = sc.nextInt();
            }

            System.out.println("Enter External Marks for 5 courses:");
            for (int j = 0; j < 5; j++) {
                System.out.print("Course " + (j + 1) + ": ");
                seeStudents[i].seeMarks[j] = sc.nextInt();
            }
        }

        for (int i = 0; i < n; i++) {
            for (int j = 0; j < 5; j++) {
                seeStudents[i].seeMarks[j] += cieStudents[i].internalMarks[j];
            }
        }

        System.out.println("\nFinal Marks of Students:");
        for (int i = 0; i < n; i++) {
            System.out.println("Student " + (i + 1) + ":");
            System.out.println("USN: " + cieStudents[i].usn);
            System.out.println("Name: " + cieStudents[i].name);
            System.out.println("Semester: " + cieStudents[i].sem);

            System.out.println("Total Marks:");
            for (int j = 0; j < 5; j++) {
                System.out.println("Course " + (j + 1) + ": " + seeStudents[i].seeMarks[j]);
            }

            System.out.println();
        }
        sc.close();
    }
}
```

OUTPUT:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac CIE/Student.java CIE/Internals.java SEE/External.java Main.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java Main
NAME: HARSHITH B
USN:  2023BMS02519

Enter the number of students: 2
Enter details for Student 1:
USN: 2023BMS02519
Name: Harshith
Semester: 3
Enter Internal Marks for 5 courses:
Course 1: 50
Course 2: 49
Course 3: 48
Course 4: 47
Course 5: 50
Enter External Marks for 5 courses:
Course 1: 49
Course 2: 48
Course 3: 48
Course 4: 49
Course 5: 50
```

```
Enter details for Student 2:
USN: 2023BMS02562
Name: Aashirvaad
Semester: 3
Enter Internal Marks for 5 courses:
Course 1: 49
Course 2: 48
Course 3: 47
Course 4: 49
Course 5: 50
Enter External Marks for 5 courses:
Course 1: 50
Course 2: 50
Course 3: 49
Course 4: 47
Course 5: 46
```

```
Final Marks of Students:
Student 1:
USN: 2023BMS02519
Name: Harshith
Semester: 3
Total Marks:
Course 1: 99
Course 2: 97
Course 3: 96
Course 4: 96
Course 5: 100

Student 2:
USN: 2023BMS02562
Name: Aashirvaad
Semester: 3
Total Marks:
Course 1: 99
Course 2: 98
Course 3: 96
Course 4: 96
Course 5: 96

PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```

LAB 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.

CODE:

```java
import java.util.Scanner;

class WrongAge extends Exception {
    public WrongAge() {
        super("Invalid age!");
    }
}

class Father {
    private int age;

    public Father(int age) throws WrongAge {
        if (age < 0) {
            throw new WrongAge();
        }
        this.age = age;
    }

    public int getAge() {
        return age;
    }
}

class Son extends Father {
    private int sonAge;

    public Son(int fatherAge, int sonAge) throws WrongAge {
        super(fatherAge);

        if (sonAge >= fatherAge) {
            throw new WrongAge();
        }
        this.sonAge = sonAge;
    }

    public int getSonAge() {
        return sonAge;
    }
}

```

```java

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

        System.out.println("NAME: HARSHITH B");
        System.out.println("USN:  2023BMS02519\n");

        Scanner sc = new Scanner(System.in);

        try {
            System.out.print("Enter father's age: ");
            int fatherAge = sc.nextInt();

            System.out.print("Enter son's age: ");
            int sonAge = sc.nextInt();

            Father father = new Father(fatherAge);
            System.out.println("Father's age: " + father.getAge());

            Son son = new Son(fatherAge, sonAge);
            System.out.println("Son's age: " + son.getSonAge());
        } catch (WrongAge e) {
            System.out.println(e.getMessage());
        } catch (Exception e) {
            System.out.println("Invalid input.");
        } finally {
            sc.close();
        }
    }
}
```

OUTPUT:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac Excep.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java EMain
NAME: HARSHITH B
USN:  2023BMS02519

Enter father's age: 50
Enter son's age: 25
Father's age: 50
Son's age: 25
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java EMain
NAME: HARSHITH B
USN:  2023BMS02519

Enter father's age: 20
Enter son's age: 30
Father's age: 20
Invalid age!
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```

LAB 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.

CODE:

```java
class DisplayThread extends Thread {
    private String message;
    private int interval;

    public DisplayThread(String message, int interval) {
        this.message = message;
        this.interval = interval;
    }

    public void run() {
        try {
            for(int i = 0; i < 5; i++) {
                System.out.println(message);
                Thread.sleep(interval * 1000);
            }
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}

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

        System.out.println("NAME: HARSHITH B");
        System.out.println("USN:  2023BMS02519\n");

        DisplayThread thread1 = new DisplayThread("BMS College of Engineering", 10);
        thread1.start();

        DisplayThread thread2 = new DisplayThread("CSE", 2);
        thread2.start();
    }
}
```

OUTPUT:

```
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> javac MultiThreads.java
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE> java ThreadDemo
NAME: HARSHITH B
USN:  2023BMS02519

BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
PS A:\CODE\HARS\JAVA_PROGRAMS_BMSCE>
```