B.M.S COLLEGE OF ENGINEERING BENGALURU
Autonomous Institute, Affiliated to VTU

LAB REPORT

OBJECT ORIENTED JAVA PROGRAMMING
23CS3PCOOJ

PROGRAMS WITH OUTPUTS

Submitted by:

DHANUSH C
1BM22CS085

Department of Computer Science and Engineering
B.M.S College of Engineering
Bull Temple Road, Basavanagudi, Bangalore 560 019
2023-2024

LAB PROGRAMS

1 – QUADRATIC EQUATION

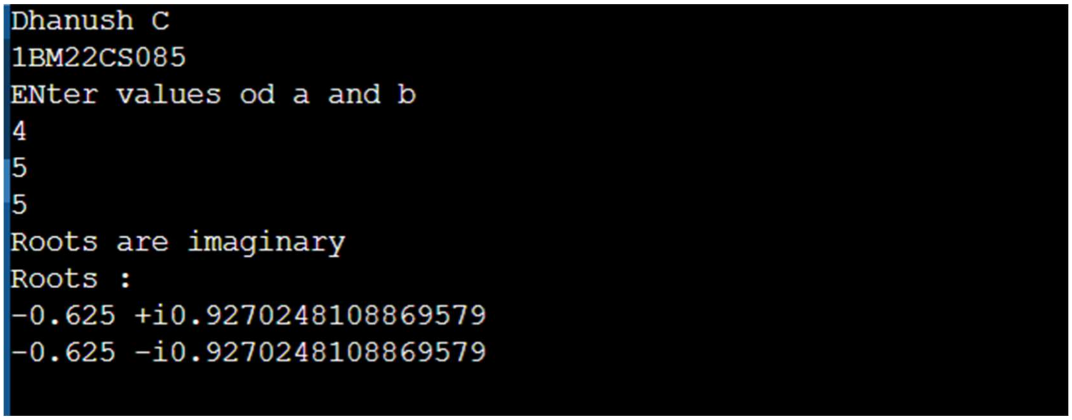
Develop a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in a, b, c and use the quadratic formula.If the discriminate $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.

PROGRAM:

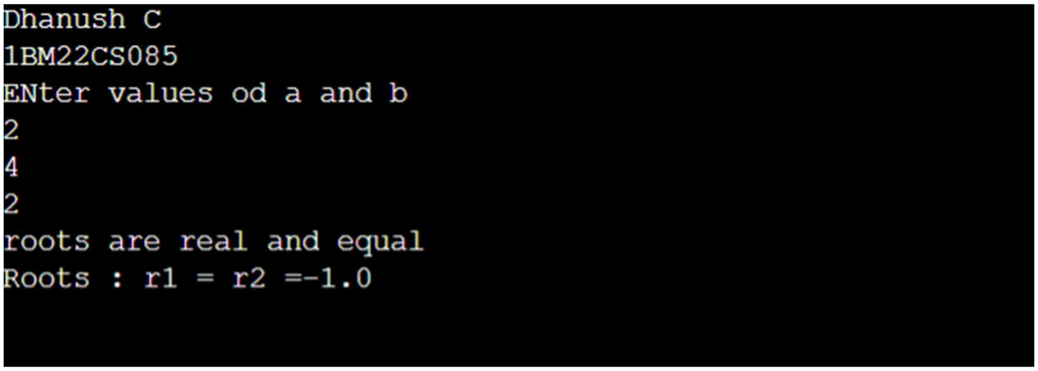
```
package labpgms;

import java.util.Scanner;

class quadra{
   int a,b,c;
   double r1,r2,d;

   void input()
   {
      Scanner sc = new Scanner(System.in);
      System.out.println("ENter values od a and b");
      a = sc.nextInt();
      b = sc.nextInt();
      c = sc.nextInt();

   }

   void compute()
   {
       if (a==0)
      {   Scanner s = new Scanner(System.in);
          System.out.println("not a quadratic equation");
          System.out.println("enter a new value of a");
          a = s.nextInt();

      }

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

       if (d==0)
       {
          System.out.println("roots are real and equal");
          r1 = (-b)/(2*a) ;
```

```
            System.out.println("Roots : r1 = r2 ="+r1);
        }

        else if (d>0)
        {
            System.out.println("roots are real and distinct");
            r1 = (-b + Math.sqrt(d))/(2.0*a);
            r2 = (-b - Math.sqrt(d))/(2.0*a);
            System.out.println("Roots :");
            System.out.println("r1="+r1);
            System.out.println("r2="+r2);
        }

        else
        {
           System.out.println("Roots are imaginary");
           r1 = -b/(2.0*a);
           r2 = Math.sqrt(Math.abs(d))/(2.0*a);
           System.out.println("Roots : ");
           System.out.println(+r1+ " +i" +r2);
           System.out.println(+r1+ " -i" +r2);

        }
    }
}
public class quad
{
    public static void main(String[] args) {
        System.out.println("Dhanush C");
        System.out.println("1BM22CS085");
        quadra q = new quadra();
        q.input();
        q.compute();

    }
}
```

OUTPUT:



```
Dhanush C
1BM22CS085
ENter values od a and b
4
5
5
Roots are imaginary
Roots :
-0.625 +i0.9270248108869579
-0.625 -i0.9270248108869579
```



```
Dhanush C
1BM22CS085
ENter values od a and b
2
4
2
roots are real and equal
Roots : r1 = r2 =-1.0
```

LAB PROGRAM – 2 – SGPA OF A STUDENT

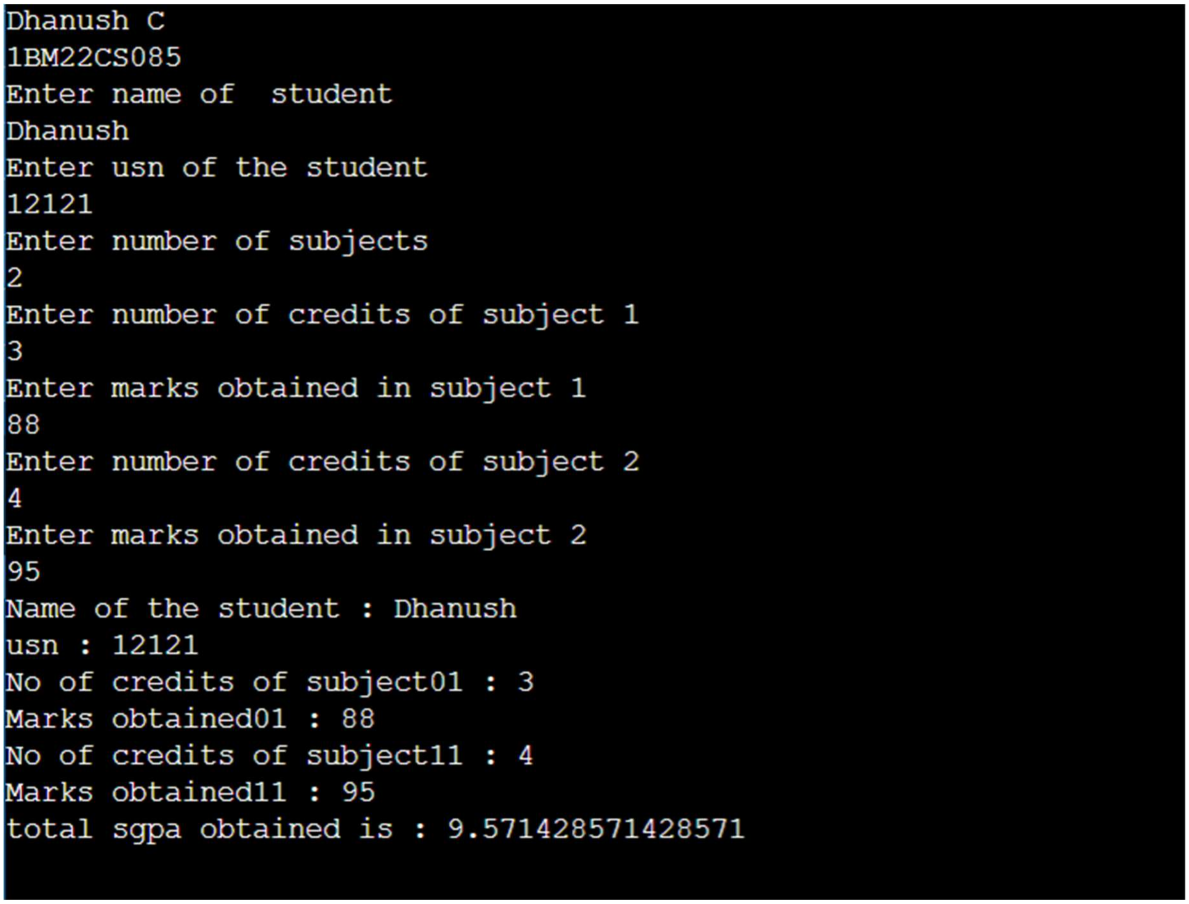
Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to acceptand display details and a method to calculate SGPA of a student.

PROGRAM:

```
package labpgms;
import java.util.*;
class student
{
   String name;
   String usn;
   int[] marks;
   int[] credits;
   int n;

   public void details(){
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter name of  student");
      name=sc.nextLine();
      System.out.println("Enter usn of the student");
      usn = sc.nextLine();
      System.out.println("Enter number of subjects");
      n = sc.nextInt();


      credits = new int[n];
      marks = new int[n];

      for(int i=0; i<n; i++)
      {
         System.out.println("Enter number of credits of subject " +
(i+1));
```

```
            credits[i] = sc.nextInt();
            System.out.println("Enter marks obtained in subject " + (i+1));
            marks[i] = sc.nextInt();

        }

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

        for(int i=0;i<n;i++)
        {
            System.out.println("No of credits of subject"+i+1+" :
"+credits[i]);
            System.out.println("Marks obtained"+i+1+" : " +marks[i]);

        }
    }

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

    public double sgpa_calculator()
    {
        double totalcreditpoints = 0;
        double totalgradepoints = 0;
        for(int i=0;i<n;i++)
        {
            totalcreditpoints += credits[i];
            totalgradepoints += (calculategradepoint(marks[i])*credits[i]);
        }

        // Check if totalcreditpoints is zero to avoid division by zero
        if(totalcreditpoints == 0) {
            System.out.println("Error: Total credits cannot be zero.");
            return 0;
        }

        double sgpa = totalgradepoints / totalcreditpoints;
        return sgpa;
    }
```

```
}

public class sgpamain
{
    public static void main(String[] args) {
        System.out.println("Dhanush C");
        System.out.println("1BM22CS085");

        student s1 = new student();
        s1.details();
        s1.display();
        double gpa = s1.sgpa_calculator();
        System.out.println("total sgpa obtained is : "+gpa);

    }
}
```



```
Dhanush C
1BM22CS085
Enter name of  student
Dhanush
Enter usn of the student
12121
Enter number of subjects
2
Enter number of credits of subject 1
3
Enter marks obtained in subject 1
88
Enter number of credits of subject 2
4
Enter marks obtained in subject 2
95
Name of the student : Dhanush
usn : 12121
No of credits of subject01 : 3
Marks obtained01 : 88
No of credits of subject11 : 4
Marks obtained11 : 95
total sgpa obtained is : 9.571428571428571
```

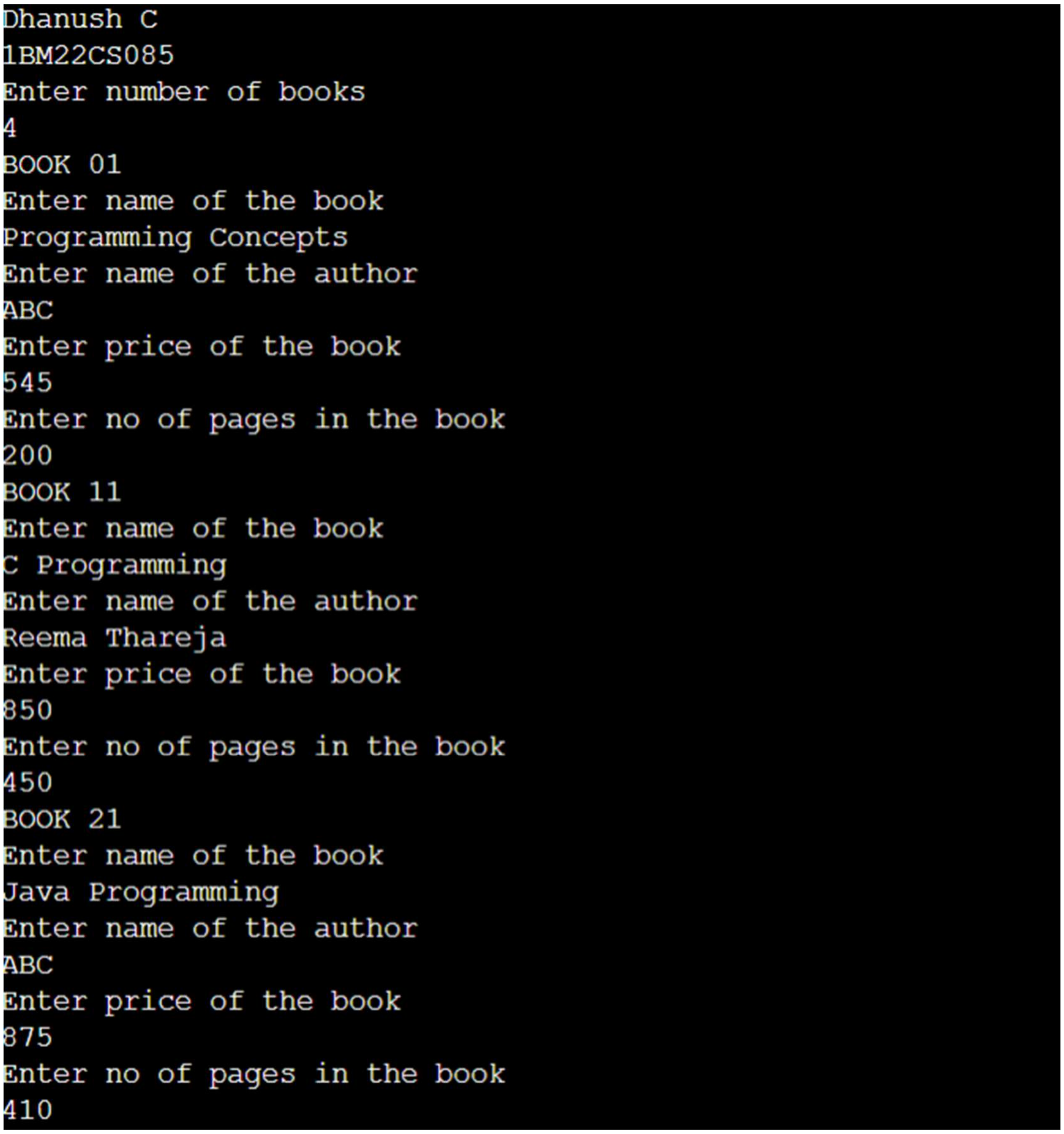
LAB PROGRAM – 3 – BOOK DATA WITH toString()

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 tocreate n book objects

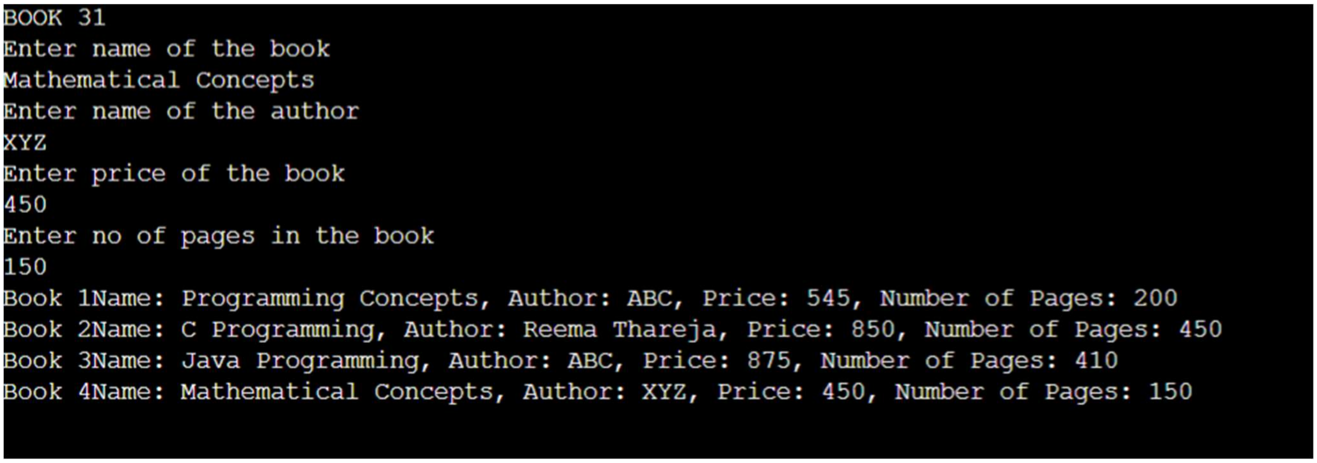
PROGRAM:

```
package labpgms;
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;

    }

    Books()
    {

    }

    Scanner sc = new Scanner(System.in);


    void accept()
    {
        System.out.println("Enter name of the book");
        name = sc.nextLine();
        System.out.println("Enter name of the author");
        author = sc.nextLine();
        System.out.println("Enter price of the book");
        price = sc.nextInt();
        System.out.println("Enter no of pages in the book");
        num_pages = sc.nextInt();

    }
    public String toString() {
        return "Name: " + name + ", Author: " + author + ", Price: " + price + ", Number of
Pages: " + num_pages;
    }


}
public class bookmain
{
    public static void main(String[] args) {
        System.out.println("Dhanush C");
        System.out.println("1BM22CS085");
        Scanner s = new Scanner(System.in);

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

        Books[] books = new Books[n];  //making array of books

        for(int i=0; i<n ;i++)
        {
            System.out.println("BOOK "+i+1);
            books[i] = new Books();  //making objects belonging to class Books
            books[i].accept();
        }

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

        s.close();

    }
}
```

OUTPUT:



```
Dhanush C
1BM22CS085
Enter number of books
4
BOOK 01
Enter name of the book
Programming Concepts
Enter name of the author
ABC
Enter price of the book
545
Enter no of pages in the book
200
BOOK 11
Enter name of the book
C Programming
Enter name of the author
Reema Thareja
Enter price of the book
850
Enter no of pages in the book
450
BOOK 21
Enter name of the book
Java Programming
Enter name of the author
ABC
Enter price of the book
875
Enter no of pages in the book
410
```



```
BOOK 31
Enter name of the book
Mathematical Concepts
Enter name of the author
XYZ
Enter price of the book
450
Enter no of pages in the book
150
Book 1Name: Programming Concepts, Author: ABC, Price: 545, Number of Pages: 200
Book 2Name: C Programming, Author: Reema Thareja, Price: 850, Number of Pages: 450
Book 3Name: Java Programming, Author: ABC, Price: 875, Number of Pages: 410
Book 4Name: Mathematical Concepts, Author: XYZ, Price: 450, Number of Pages: 150
```

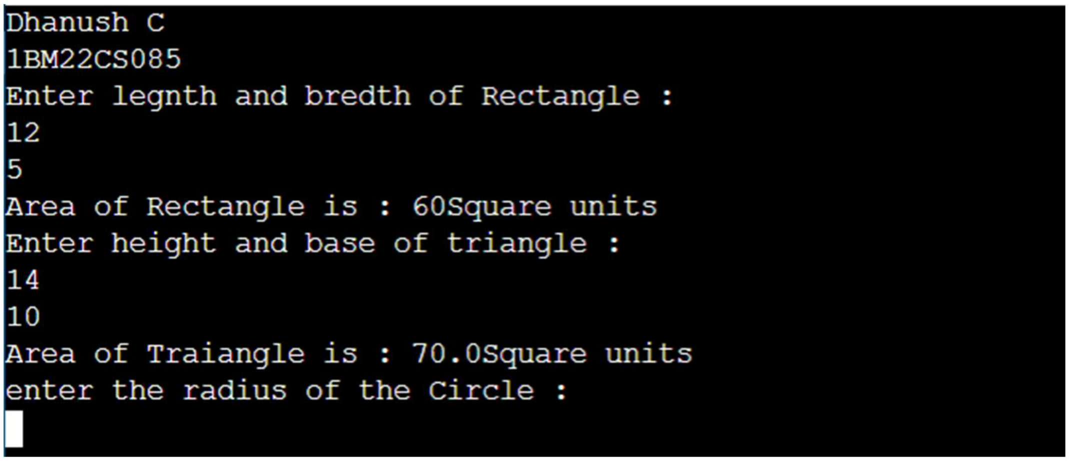
PROGRAM – 4 – ABSTRACT CLASS – SHAPE - AREA

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.

PROGRAM:

```
 package labpgms;
import java.util.Scanner;
abstract class Shape
{
   int x,y;
   Shape(int x,int y)
   {
      this.x=x;
      this.y=y;
   }

   abstract void printArea();
}

class Rectangle extends Shape{
   Rectangle(int x,int y)
   {
      super(x,y);
   }
   void printArea()
   {
      System.out.println("Area of Rectangle is : "+(x*y)+"Square units");
   }
}

class Triangle extends Shape {
    Triangle(int x, int y) {
        super(x, y);
    }

    void printArea() {
        System.out.println("Area of Traiangle is : " + (0.5 * x * y) + "Square
units");
    }

}

class Circle extends Shape
{
    Circle(int x)
    {
        super(x,0);
    }
    void printArea()
    {
        System.out.println("Area of Circle is : "+(3.142*x*x)+"Sqaure units");
    }
}


public class ShapeMain
{
    public static void main(String[] args)
    {
        System.out.println("Dhanush C");
        System.out.println("1BM22CS085");
        Scanner sc = new Scanner(System.in);
        int x,y;
        System.out.println("Enter legnth and bredth of Rectangle : ");
         x = sc.nextInt();
         y = sc.nextInt();
        Rectangle rectangle = new Rectangle(x,y);
        rectangle.printArea();

        System.out.println("Enter height and base of triangle : ");
        x = sc.nextInt();
        y = sc.nextInt();
        Triangle triangle = new Triangle(x,y);
        triangle.printArea();

        System.out.println("enter the radius of the Circle : ");
        x = sc.nextInt();
        Circle circle = new Circle(x);
        circle.printArea();

        sc.close();
    }
}
```

OUTPUT:



```
Dhanush C
1BM22CS085
Enter legnth and bredth of Rectangle :
12
5
Area of Rectangle is : 60Square units
Enter height and base of triangle :
14
10
Area of Traiangle is : 70.0Square units
enter the radius of the Circle :
```

PROGRAM – 5 – BANK – SAVINGS AND CURRENT ACCOUNT

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 achievethe 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.

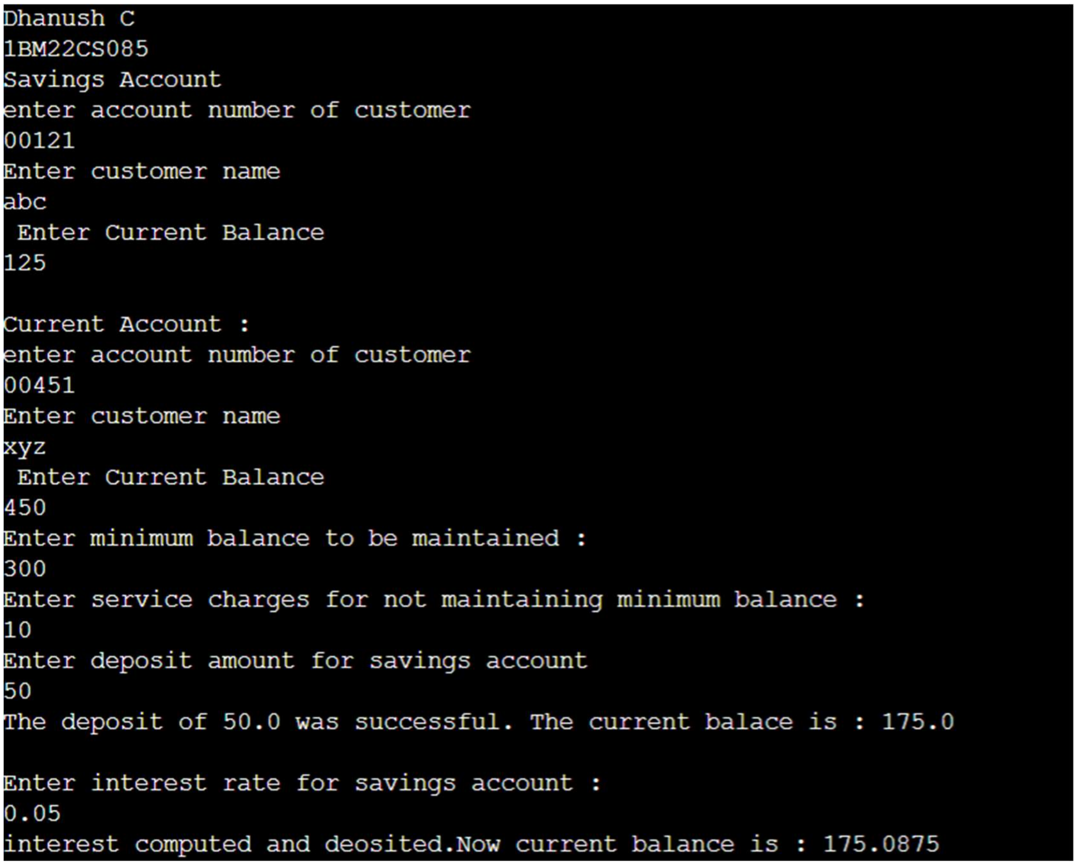
Program:

```
package labpgms;
import java.util.Scanner;
class Account
{
   String customer_name;
   long acc_no;
   String acc_type;
   double balance;

   Account(String customer_name,long acc_no, String acc_type,Double balance)
   {
      this.customer_name = customer_name;
      this.acc_no = acc_no;
      this.acc_type = acc_type;
      this.balance = balance;
   }

   void deposit(double amount)
   {
      balance = balance + amount;
      System.out.println("The deposit of "+amount+" was successful. The current balace is :
"+balance);
   }

   void displayBalance()
   {
      System.out.println("\n Account Number : "+acc_no+"\n Customer Name :
```

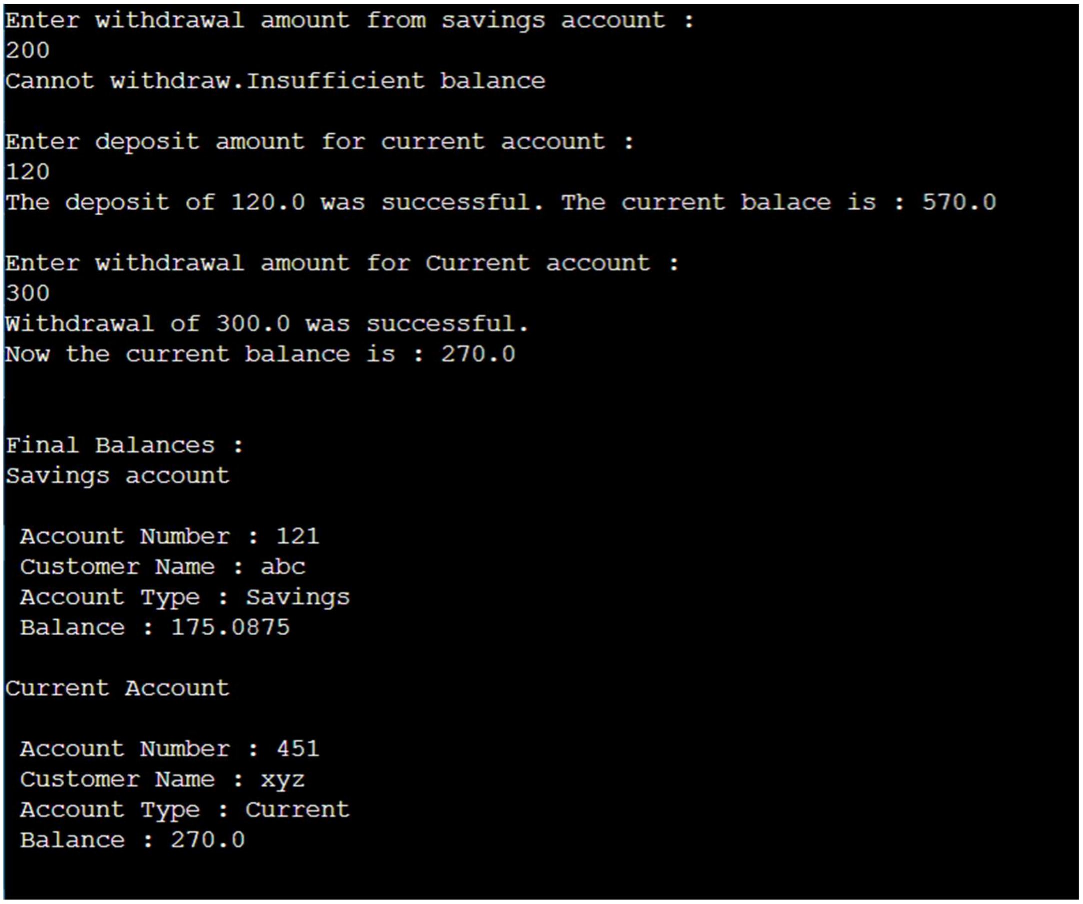
```
"+customer_name+"\n Account Type : "+acc_type+"\n Balance : "+balance);

  }
}

class SavingsAccount extends Account {
  SavingsAccount(String customer_name, long acc_no, double balance) {
    super(customer_name, acc_no, "Savings", balance);
  }

  void interest(double rate) {
    balance = balance + balance * (rate) / 100;
    System.out.println("interest computed and deosited.Now current balance is : " + balance);
  }

  void withdraw(double amount) {
    if (balance >= amount) {
      balance = balance - amount;
      System.out.println("Withdrawal of " + amount + " was successful.\nNow the current
balance is : " + balance);
    } else {
      System.out.println("Cannot withdraw.Insufficient balance");
    }
  }

}


   class CurrentAccount extends Account
  {
    double min_balance;
    double service_charges;

    CurrentAccount(String customer_name,long acc_no,double balance,double
min_balance,double service_charge)
    {
      super(customer_name,acc_no,"Current",balance);
      this.min_balance = min_balance;
      this.service_charges = service_charge;
    }

    void check_min_balance()
    {
      if(balance<min_balance)
      {
        balance = balance - service_charges;
        System.out.println("Minimum balance not maintained!.\nService charges
applied!\nCurrent Balance : "+balance);
      }
      else {
        System.out.println("Minimum balance maintained.\nNO WORRIES!");
      }
    }

    void cheque(double amount)
    {
      if(balance>=amount)
      {
        balance = balance - amount;
        System.out.println("Withdrawal of "+amount+" was successful.\nNow the current
balance is : "+balance);
      }

      else {
        System.out.println("Cannot withdraw.Insufficient balance");
      }
    }

  }



public class BankMain
{
  public static void main(String[] args)
  {
    System.out.println("Dhanush C");
    System.out.println("1BM22CS085");
   Scanner sc = new Scanner(System.in);

   System.out.println("Savings Account ");

    System.out.println("enter account number of customer");
    long num = sc.nextLong();

    System.out.println("Enter customer name");
    String name = sc.nextLine();
```

```
        System.out.println(" Enter Current Balance");
        double bal = sc.nextDouble();

        SavingsAccount SA = new SavingsAccount(name,num,bal);
        System.out.print("\n");



        System.out.println("Current Account : ");

        System.out.println("enter account number of customer");
        long cust_num = sc.nextLong();

        System.out.println("Enter customer name");
        String cust_name = sc.nextLine();

        System.out.println(" Enter Current Balance");
        double cust_bal = sc.nextDouble();

        System.out.println("Enter minimum balance to be maintained : ");
        double min_bal = sc.nextDouble();

        System.out.println("Enter service charges for not maintaining minimum balance : ");
        double ser_char = sc.nextDouble();

        CurrentAccount CA = new
    CurrentAccount(cust_name,cust_num,cust_bal,min_bal,ser_char);

        System.out.println("Enter deposit amount for savings account ");
        double SDA = sc.nextDouble();
        SA.deposit(SDA);
        System.out.print("\n");

        System.out.println("Enter interest rate for savings account : ");
        double SIR = sc.nextDouble();
        SA.interest(SIR);
        System.out.print("\n");

        System.out.println("Enter withdrawal amount from savings account : ");
        double SWD = sc.nextDouble();
        SA.withdraw(SWD);
        System.out.print("\n");

        System.out.println("Enter deposit amount for current account : ");
        double CDA = sc.nextDouble();
        CA.deposit(CDA);
        System.out.print("\n");

        System.out.println("Enter withdrawal amount for Current account : ");
        double CWD = sc.nextDouble();
        CA.cheque(CWD);
        System.out.print("\n");

        System.out.println("\nFinal Balances : ");
        System.out.println("Savings account");
        SA.displayBalance();
        System.out.print("\n");

        System.out.println("Current Account");
        CA.displayBalance();
    }
}
```

OUTPUT:



```
Dhanush C
1BM22CS085
Savings Account
enter account number of customer
00121
Enter customer name
abc
 Enter Current Balance
125

Current Account :
enter account number of customer
00451
Enter customer name
xyz
 Enter Current Balance
450
Enter minimum balance to be maintained :
300
Enter service charges for not maintaining minimum balance :
10
Enter deposit amount for savings account
50
The deposit of 50.0 was successful. The current balace is : 175.0

Enter interest rate for savings account :
0.05
interest computed and deosited.Now current balance is : 175.0875
```



```
Enter withdrawal amount from savings account :
200
Cannot withdraw.Insufficient balance

Enter deposit amount for current account :
120
The deposit of 120.0 was successful. The current balace is : 570.0

Enter withdrawal amount for Current account :
300
Withdrawal of 300.0 was successful.
Now the current balance is : 270.0


Final Balances :
Savings account

 Account Number : 121
 Customer Name : abc
 Account Type : Savings
 Balance : 175.0875

Current Account

 Account Number : 451
 Customer Name : xyz
 Account Type : Current
 Balance : 270.0
```

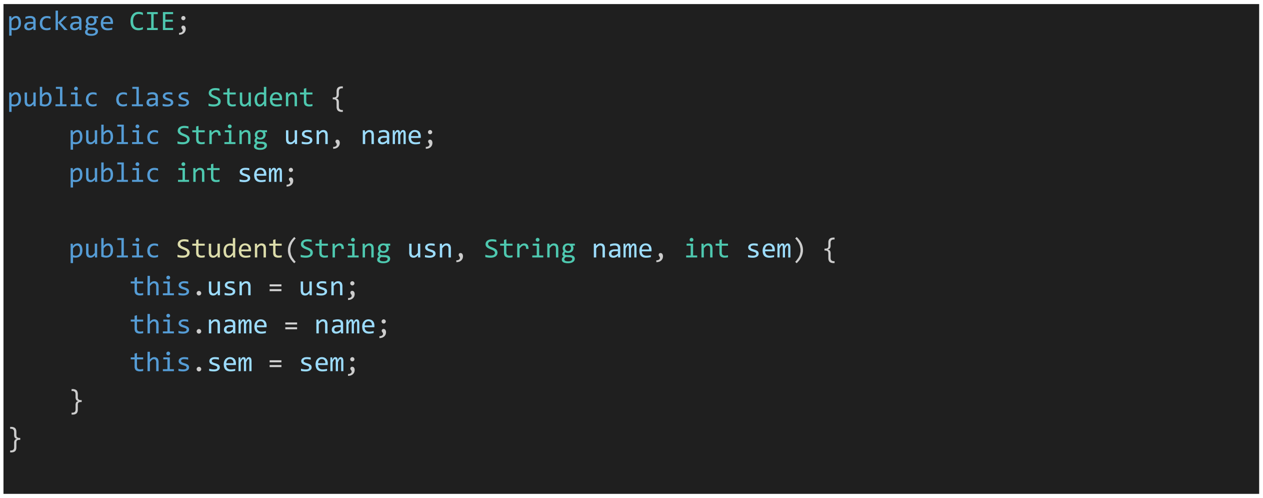
PROGRAM – 6 – CIE AND SEE PACKAGES OF ‘N’ STUDENTS

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.

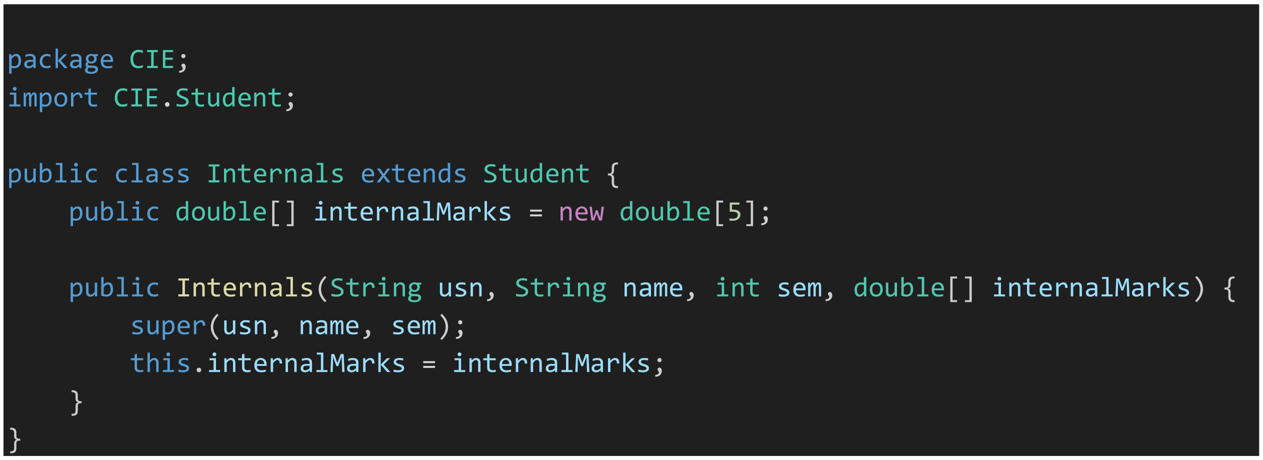
This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.

PROGRAM:

In CIE folder:



```java
package CIE;

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

    public Student(String usn, String name, int sem) {
        this.usn = usn;
        this.name = name;
        this.sem = sem;
    }
}
```

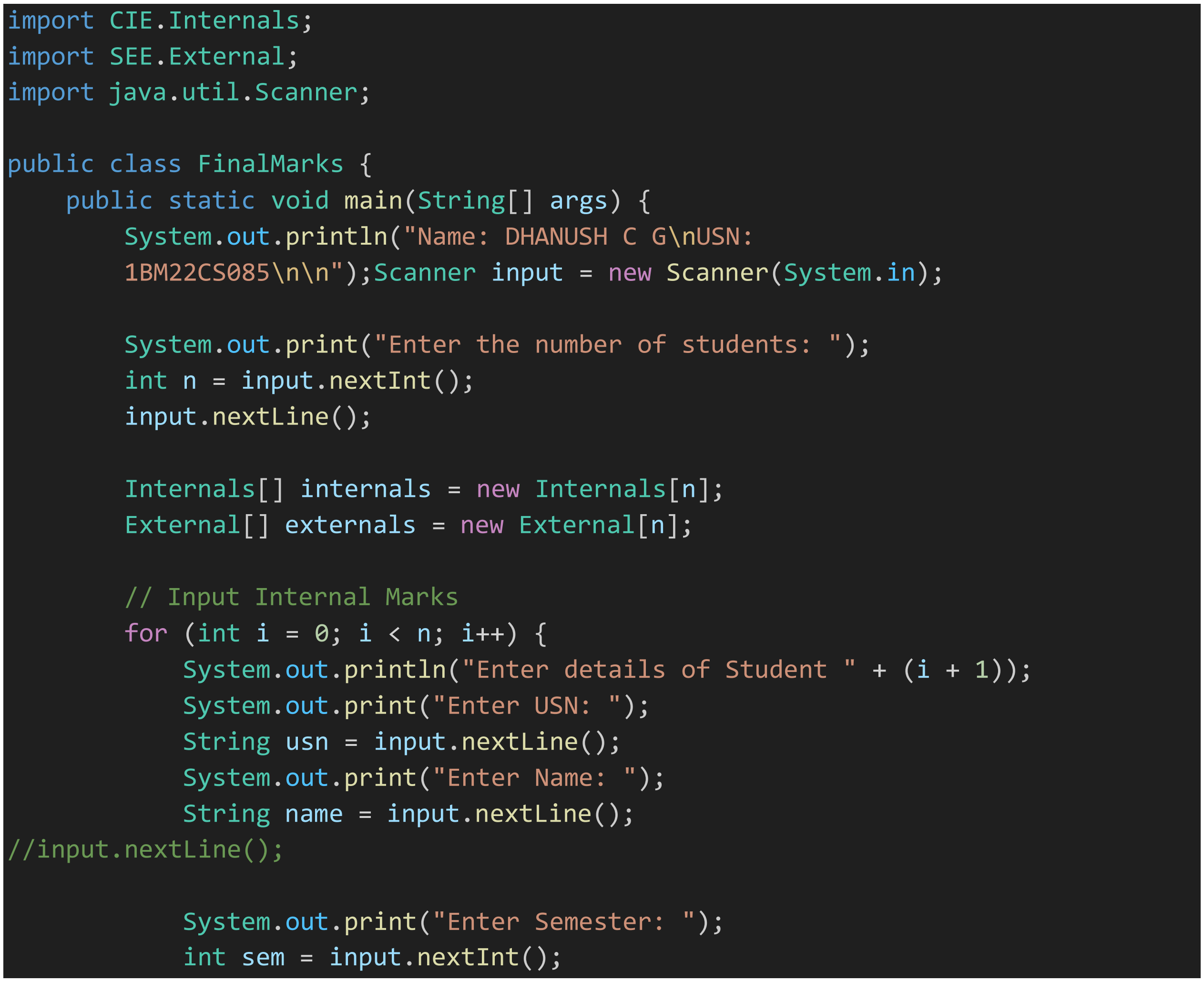


```java
package CIE;
import CIE.Student;

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

    public Internals(String usn, String name, int sem, double[] internalMarks) {
        super(usn, name, sem);
        this.internalMarks = internalMarks;
    }
}
```

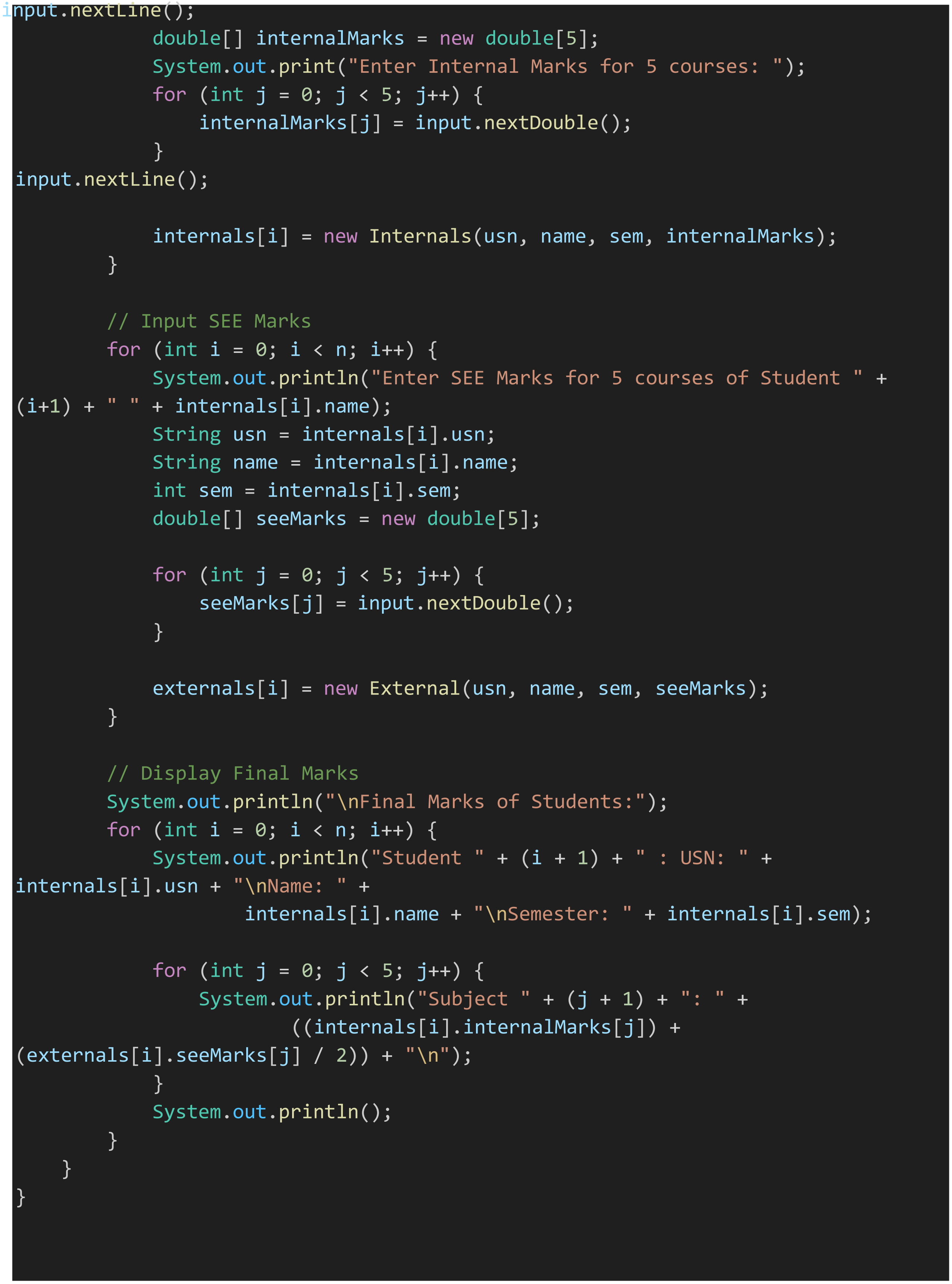
In SEE folder:

```java
package SEE;

import CIE.*;

public class External extends Student {
    public double[] seeMarks = new double[5];

    public External(String usn, String name, int sem, double[] seeMarks) {
        super(usn, name, sem);
        this.seeMarks = seeMarks;
    }
}
```

Main

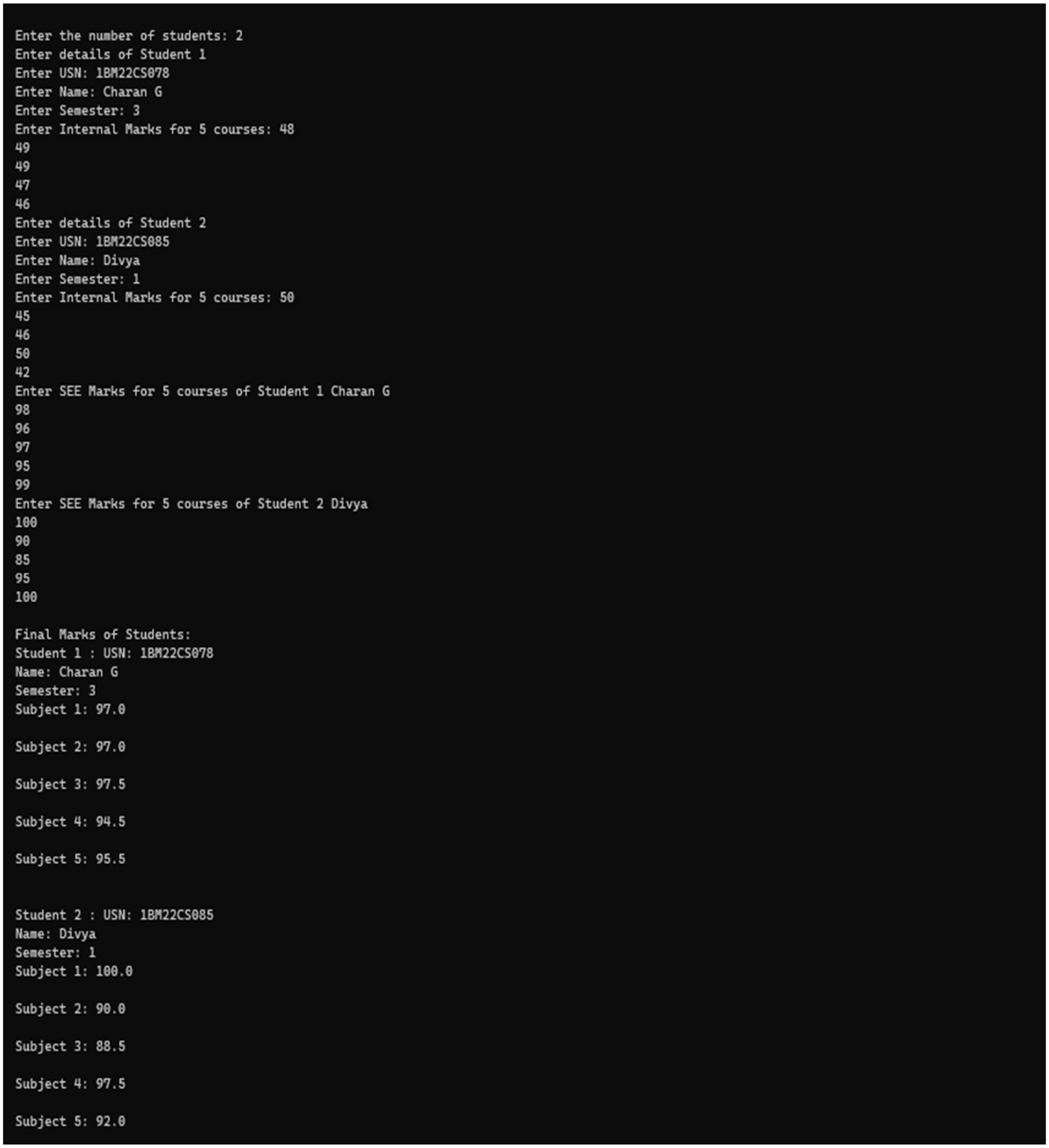


```
import CIE.Internals;
import SEE.External;
import java.util.Scanner;

public class FinalMarks {
    public static void main(String[] args) {
        System.out.println("Name: DHANUSH C G\nUSN:
        1BM22CS085\n\n");Scanner input = new Scanner(System.in);

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

        Internals[] internals = new Internals[n];
        External[] externals = new External[n];

        // Input Internal Marks
        for (int i = 0; i < n; i++) {
            System.out.println("Enter details of Student " + (i + 1));
            System.out.print("Enter USN: ");
            String usn = input.nextLine();
            System.out.print("Enter Name: ");
            String name = input.nextLine();
//input.nextLine();

            System.out.print("Enter Semester: ");
            int sem = input.nextInt();
```



```
input.nextLine();
            double[] internalMarks = new double[5];
            System.out.print("Enter Internal Marks for 5 courses: ");
            for (int j = 0; j < 5; j++) {
                internalMarks[j] = input.nextDouble();
            }
input.nextLine();

            internals[i] = new Internals(usn, name, sem, internalMarks);
        }

        // Input SEE Marks
        for (int i = 0; i < n; i++) {
            System.out.println("Enter SEE Marks for 5 courses of Student " +
(i+1) + " " + internals[i].name);
            String usn = internals[i].usn;
            String name = internals[i].name;
            int sem = internals[i].sem;
            double[] seeMarks = new double[5];

            for (int j = 0; j < 5; j++) {
                seeMarks[j] = input.nextDouble();
            }

            externals[i] = new External(usn, name, sem, seeMarks);
        }

        // Display Final Marks
        System.out.println("\nFinal Marks of Students:");
        for (int i = 0; i < n; i++) {
            System.out.println("Student " + (i + 1) + " : USN: " +
internals[i].usn + "\nName: " +
                    internals[i].name + "\nSemester: " + internals[i].sem);

            for (int j = 0; j < 5; j++) {
                System.out.println("Subject " + (j + 1) + ": " +
                        ((internals[i].internalMarks[j]) +
(externals[i].seeMarks[j] / 2)) + "\n");
            }
            System.out.println();
        }
    }
}
```

OUTPUT:



```
Enter the number of students: 2
Enter details of Student 1
Enter USN: 1BM22CS078
Enter Name: Charan G
Enter Semester: 3
Enter Internal Marks for 5 courses: 48
49
49
47
46
Enter details of Student 2
Enter USN: 1BM22CS085
Enter Name: Divya
Enter Semester: 1
Enter Internal Marks for 5 courses: 50
45
46
50
42
Enter SEE Marks for 5 courses of Student 1 Charan G
98
96
97
95
99
Enter SEE Marks for 5 courses of Student 2 Divya
100
90
85
95
100

Final Marks of Students:
Student 1 : USN: 1BM22CS078
Name: Charan G
Semester: 3
Subject 1: 97.0

Subject 2: 97.0

Subject 3: 97.5

Subject 4: 94.5

Subject 5: 95.5


Student 2 : USN: 1BM22CS085
Name: Divya
Semester: 1
Subject 1: 100.0

Subject 2: 90.0

Subject 3: 88.5

Subject 4: 97.5

Subject 5: 92.0
```

PROGRAM – 7 – EXCEPTION IN INHERITANCE TREE

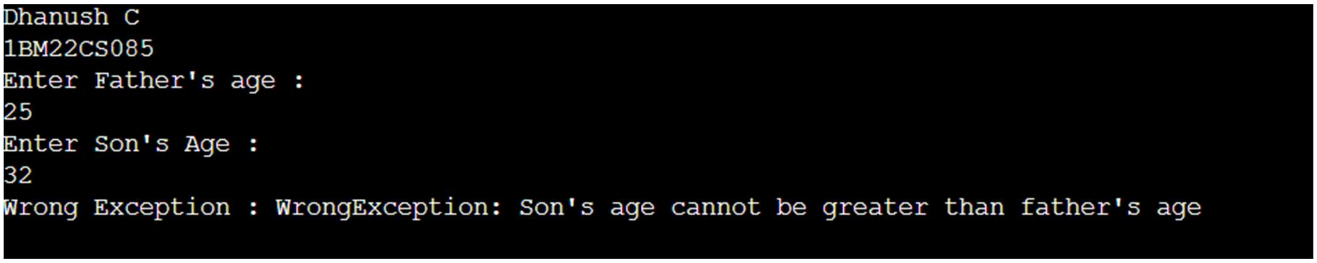
Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age=father’s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

PROGRAM:

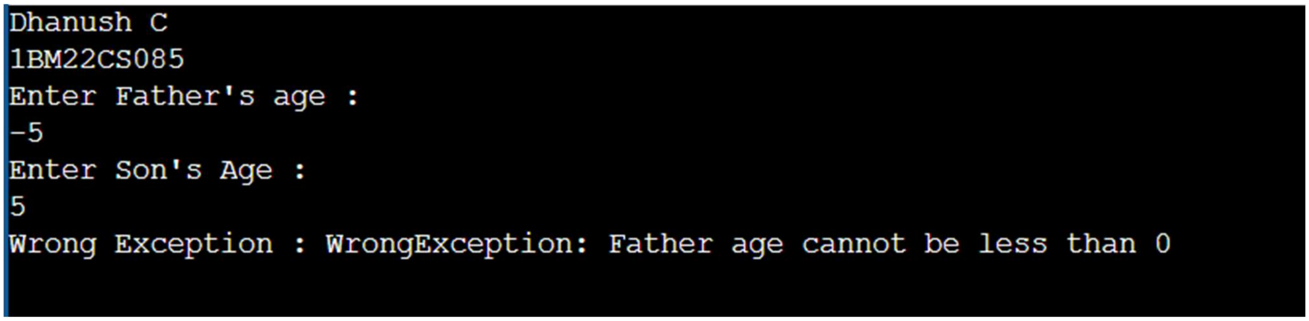
```
package labpgms;
import java.util.Scanner;
class WrongException extends Exception{
   WrongException(String str)
   {
      super(str);
   }
}

class Father{
   int FatherAge;

   Father(int FatherAge) throws WrongException
   {
      this.FatherAge = FatherAge;

      if (FatherAge < 0)
      {
         throw new WrongException("Father age cannot be less than 0");
      }
   }
}

class Son extends Father{
   int SonAge;
   Son(int FatherAge, int SonAge) throws WrongException
```

```
    {
        super(FatherAge);
        this.SonAge = SonAge;

        if(SonAge < 0)
        {
            throw new WrongException("Son's Age cannot be less than 0");
        }

        else if(FatherAge<SonAge)
        {
            throw new WrongException("Son's age cannot be greater than father's age");
        }
    }
}

public class ExceptionMain {
    public static void main(String[] args) {
        System.out.println("Dhanush C");
        System.out.println("1BM22CS085");

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter Father's age : ");
        int father_age = sc.nextInt();

        System.out.println("Enter Son's Age : ");
        int son_age = sc.nextInt();
        try {
            Son son = new Son(father_age, son_age);
            System.out.println("Father's and Son's age are valid!");
        } catch (WrongException e) {
            System.out.println("Wrong Exception : " + e);
        }
    }
}
```

OUTPUT:



```
Dhanush C
1BM22CS085
Enter Father's age :
25
Enter Son's Age :
32
Wrong Exception : WrongException: Son's age cannot be greater than father's age
```



```
Dhanush C
1BM22CS085
Enter Father's age :
-5
Enter Son's Age :
5
Wrong Exception : WrongException: Father age cannot be less than 0
```

.

PROGRAM – 8 - MULTITHREADING

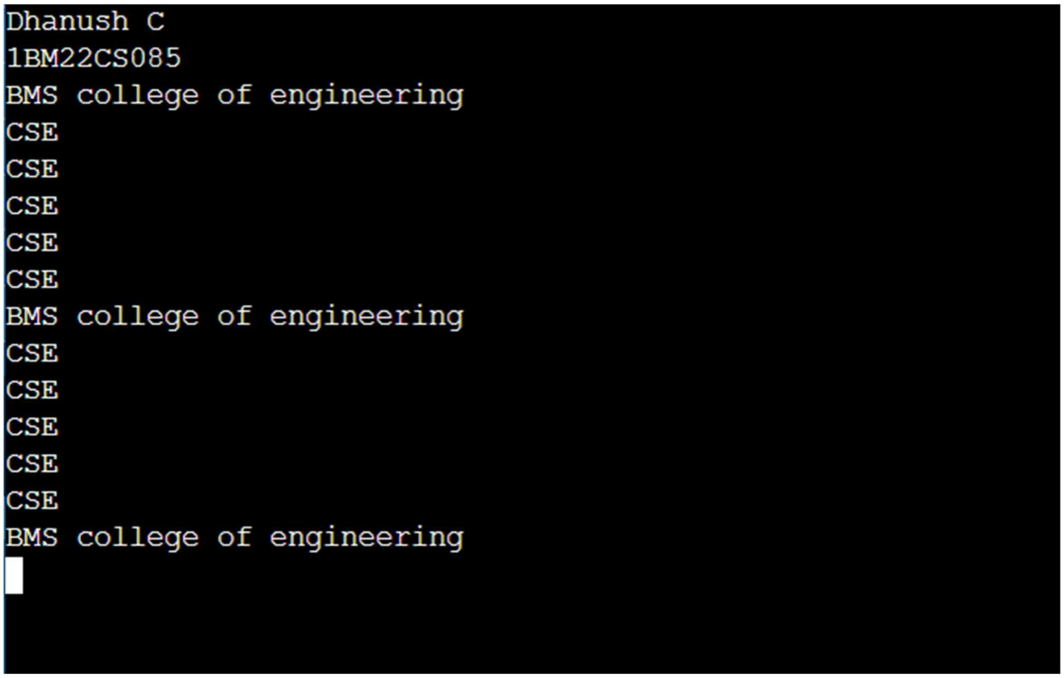
Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

PROGRAM:

```
package labpgms;
import java.util.*;
class BmsDisplay extends Thread{
   public void run()
   {
      for(int i=0;i<3;i++)
      {
         System.out.println("BMS college of engineering");

            try {
               Thread.sleep(10000);
            } catch (InterruptedException e) {
               System.out.println("Exception : "+e);
            }
         }

      }
}

class CseDisplay extends Thread{
   public void run()
   {  for(int i=0;i<10;i++){
      System.out.println("CSE");
      try{
         Thread.sleep(2000);
      }catch(InterruptedException e){
         System.out.println("Exception : "+e);
      }

   }

   }
}
```

```
public class threads
{
   public static void main(String[] args)
   {
      System.out.println("Dhanush C");
      System.out.println("1BM22CS085");
     BmsDisplay obj1 = new BmsDisplay();
     CseDisplay obj2 = new CseDisplay();

     obj1.start();
     obj2.start();
   }
}
```

OUTPUT:



```
Dhanush C
1BM22CS085
BMS college of engineering
CSE
CSE
CSE
CSE
CSE
BMS college of engineering
CSE
CSE
CSE
CSE
CSE
BMS college of engineering
```