VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT

on

OBJECT ORIENTED JAVA PROGRAMMING

Submitted by

VYLERI KEZHEKE SIDDHARTH

(1BM21CS247)

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 Oct 2022-Feb 2023

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 “DATA STRUCTURES” carried out by Vyleri Kezheke Siddharth(1BM21CS247), 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 during the year 2022-23. The Lab report has been approved as it satisfies the academic requirements in respect of JAVA Lab - (22CS3PCOOJ) work prescribed for the said degree.

Name of the Lab-Incharge	Dr. Jyothi S. Nayak
Designation	Professor and Head
Department of CSE	Department of CSE
BMSCE, Bengaluru	BMSCE, Bengaluru

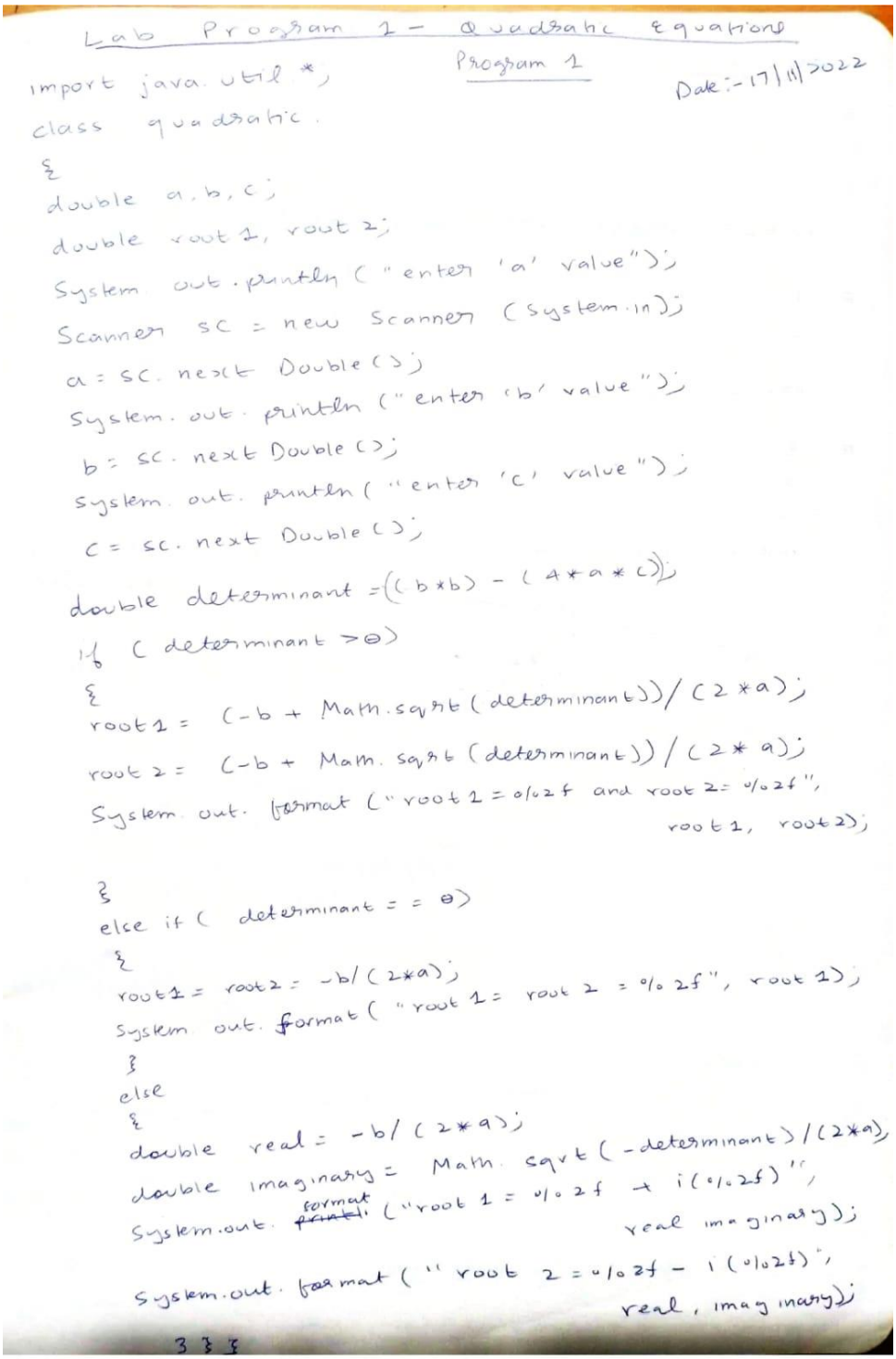
INDEX SHEET

COURSE OUTCOMES

CO1	Apply the knowledge of java concepts to find the solution for a given solution
CO2	Analyze the given java application for correctness
CO3	Develop Java programs for a given requirement
CO4	Conduct practical experiments for demonstrating features of java

LAB PROGRAM 1

Q: 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.



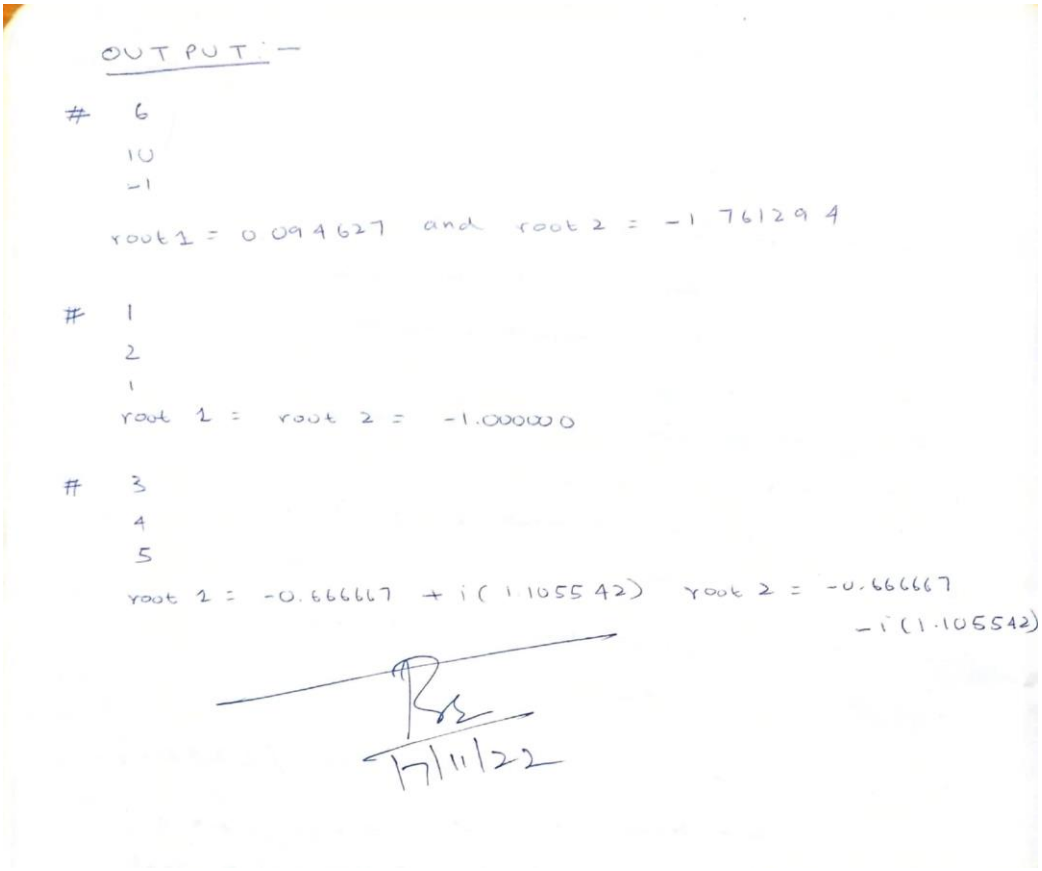
Lab Program 1 – Quadratic equations

Program 1

Date :- 17/11/2022

```
import java.util.*;
class quadratic
{
double a,b,c;
double root1, root2;
System.out.println("enter 'a' value");
Scanner sc = new Scanner(System.in);
a = sc.nextDouble();
System.out.println("enter 'b' value");
b = sc.nextDouble();
System.out.println("enter 'c' value");
c = sc.nextDouble();
double determinant =((b*b) - (4*a*c));
if (determinant>0)
{
root1 = (-b + Math.sqrt(determinant))/(2*a);
root2 = (-b + Math.sqrt(determinant))/(2*a);
System.out.format("root1 = %.2f and root2= %.2f",
                  root1, root2);

}
else if (determinant==0)
{
root1 = root2 = -b/(2*a);
System.out.format("root1 = root2 = %.2f", root1);
}
else
{
double real = -b/(2*a);
double imaginary = Math.sqrt(-determinant)/(2*a);
System.out.format("root1 = %.2f + i(%.2f)",
                  real imaginary);
System.out.format("root2 = %.2f - i(%.2f)",
                  real, imaginary);
}}
```



Output:

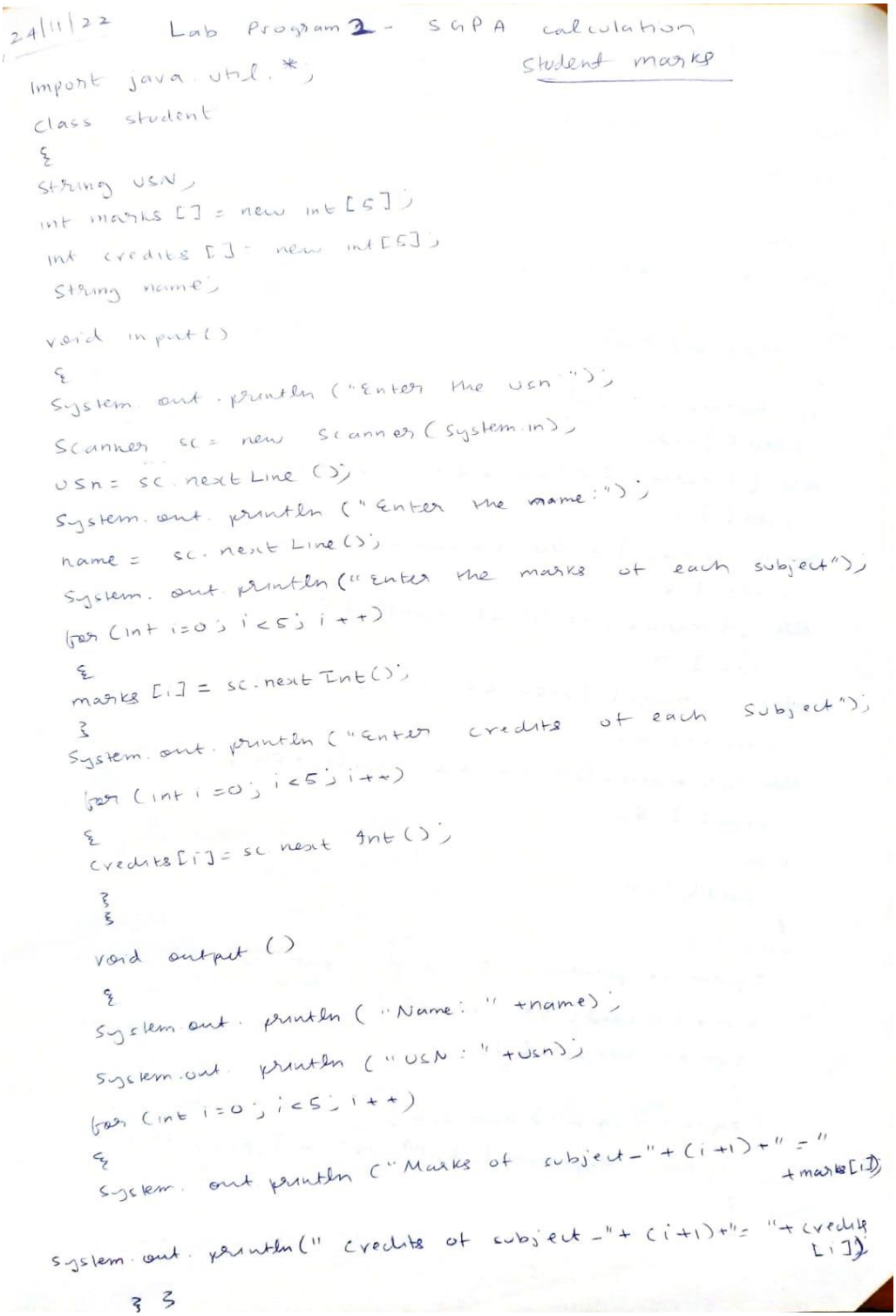
```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac quad.java } ; if ($?) {
 java quad }
enter 'a' value
6
enter 'b' value
10
enter 'c' value
-1
root1 = 0.094627 and root 2 = -1.761294
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac quad.java } ; if ($?) {
 java quad }
enter 'a' value
1
enter 'b' value
2
enter 'c' value
1
root 1 = root 2 = -1.000000
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac quad.java } ; if ($?) {
 java quad }
enter 'a' value
3
enter 'b' value
4
enter 'c' value
5
root 1 = -0.666667 + i(1.105542)root 2 = -0.666667 - i(1.105542)
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

LAB PROGRAM 2

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



```
24/11/22    Lab Program 2 - SGPA calculation
                                Student marks
import java.util.*;
class student
{
String USN;
int marks[] = new int[5];
int credits[] = new int[5];
String name;

void input()
{
System.out.println("Enter the usn");
Scanner sc = new Scanner(System.in);
usn = sc.nextLine();
System.out.println("Enter the name:");
name = sc.nextLine();
System.out.println("Enter the marks of each subject");
for(int i=0; i<5; i++)
{
marks[i] = sc.nextInt();
}
System.out.println("Enter credits of each subject");
for(int i=0; i<5; i++)
{
credits[i] = sc.next Int();
}
}
void output()
{
System.out.println("Name: " +name);
System.out.println("USN: " +usn);
for(int i=0; i<5; i++)
{
System.out.println("Marks of subject-" + (i+1) + " = "
+marks[i]);
System.out.println("credits of subject-" + (i+1) + "= " + credits
[i]);
}
}
```

```
void cgpa
{
int cred[] = new int[5];
double sgpa;
int sum = 0;
int num = 0;
for (int i=0; i<5; i++)
{
if (marks[i] > 0)
{
 if (marks[i] >= 90)
  cred[i] = 10;
 else if (marks[i] >= 80 && marks[i] < 90)
  cred[i] = 9;
 else if (marks[i] >= 70 && marks[i] < 80)
  cred[i] = 8;
 else if (marks[i] >= 60 && marks[i] < 70)
  cred[i] = 7;
 else if (marks[i] >= 50 && marks[i] < 60)
  cred[i] = 6;
 else if (marks[i] >= 40 && marks[i] < 50)
  cred[i] = 5;
 else
  cred[i] = 0;
}
else
 System.out.println("In valid input");
 num += (cred[i] * credits[i]);
 sum += credits[i];
 }
 sgpa = (double) num/sum;
 System.out.println("SGPA is:" + sgpa);
 }
}
```

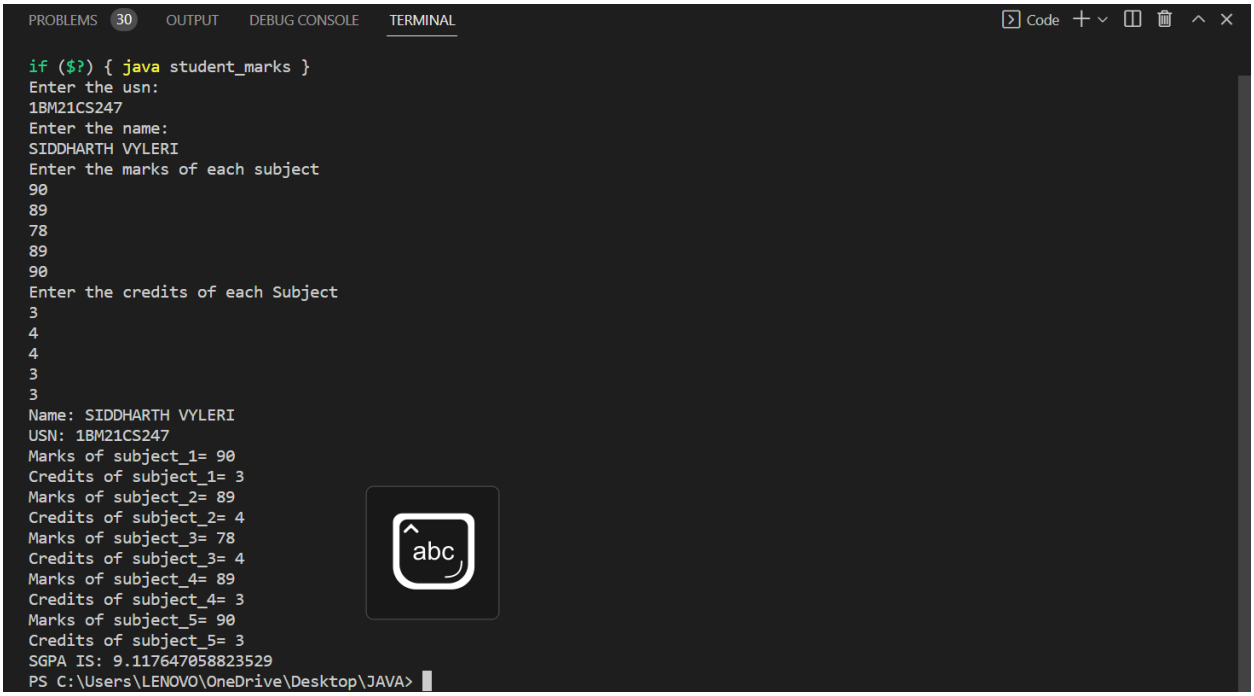
```
class sgpa
{
public static void main (String args[])
{
student vs = new student();
vs.input();
vs.output();
vs.cgpa();
}
}
```

OUTPUT: —

```
Enter the marks of each subject
90
89
78
89
90
Enter the credits of each subject
3
4
4
3
3
Name: VYLERI KEZHEKE SIDDHARTH
USN: 1BM21CS247
Marks of subject-1 = 90
Credits of subject-1 = 3
Marks of subject-2 = 89
credits of subject-2 = 4
Marks of subject-3 = 78
credits of subject-3 = 4
Marks of subject-4 = 89
credits of subject-4 = 3
Marks of subject-5 = 90
credits of subject-5 = 3
SGPA IS: 9.11764705882353
```

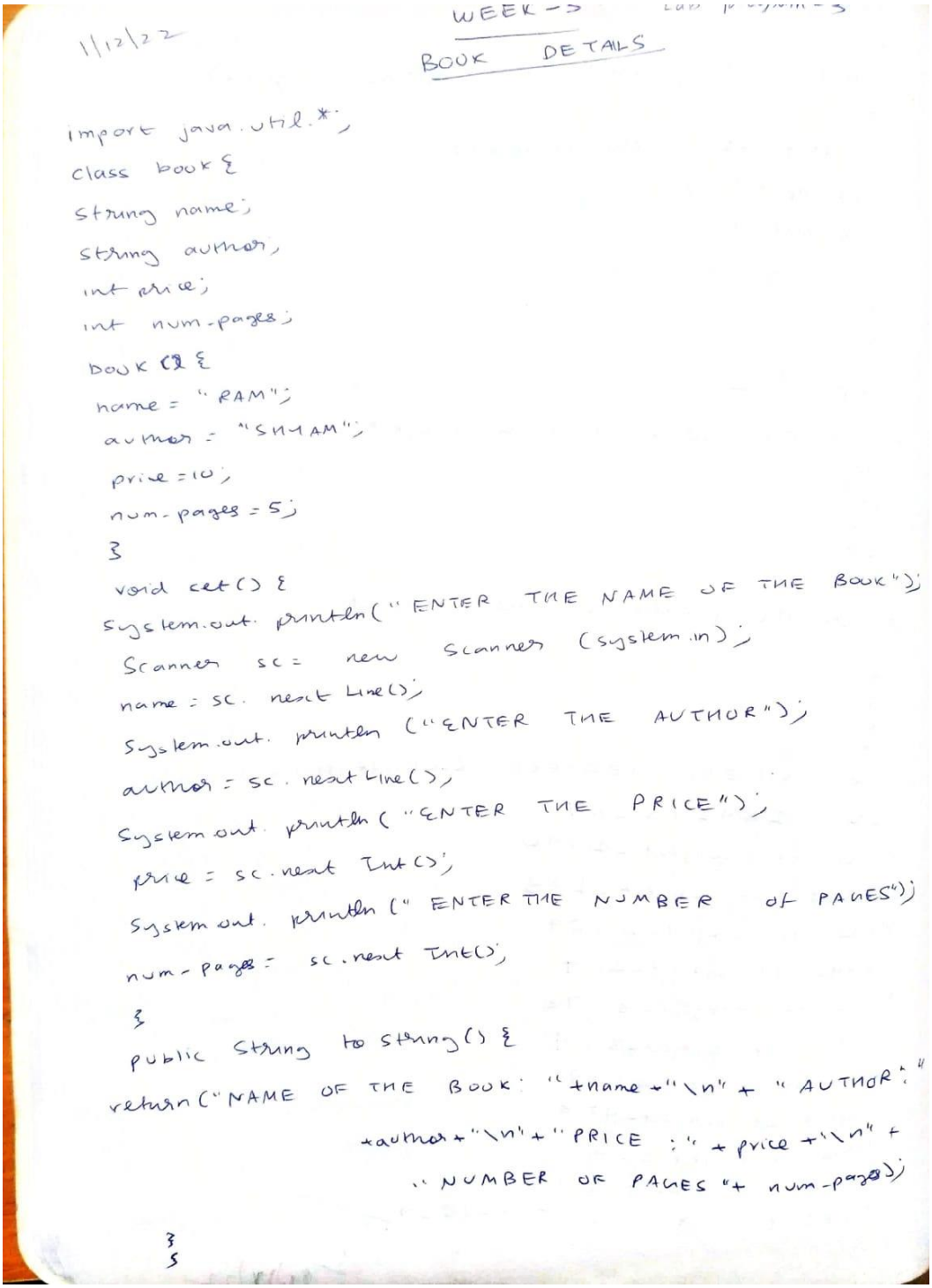
28/11/22

Output:


PROBLEMS 30
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
if ($?) { java student_marks }
Enter the usn:
1BM21CS247
Enter the name:
SIDDHARTH VYLERI
Enter the marks of each subject
90
89
78
89
90
Enter the credits of each Subject
3
4
4
3
3
Name: SIDDHARTH VYLERI
USN: 1BM21CS247
Marks of subject_1= 90
Credits of subject_1= 3
Marks of subject_2= 89
Credits of subject_2= 4
Marks of subject_3= 78
Credits of subject_3= 4
Marks of subject_4= 89
Credits of subject_4= 3
Marks of subject_5= 90
Credits of subject_5= 3
SGPA IS: 9.117647058823529
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>

LAB PROGRAM 3

Q: 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.



```
11/12/22                    WEEK -> 3
                            BOOK DETAILS

import java.util.*;
class book {
String name;
String author;
int price;
int num_pages;
book () {
name = "RAM";
author = "SHYAM";
price = 10;
num_pages = 5;
}
void set() {
System.out.println("ENTER THE NAME OF THE BOOK");
Scanner sc = new Scanner (system.in);
name = sc.nextLine();
System.out.println("ENTER THE AUTHOR");
author = sc.nextLine();
System.out.println("ENTER THE PRICE");
price = sc.nextInt();
System.out.println("ENTER THE NUMBER OF PAGES");
num_pages = sc.nextInt();
}
public String toString() {
return("NAME OF THE BOOK: "+name+"\n" + "AUTHOR:"
+author+"\n"+"PRICE :"+price+"\n" +
"NUMBER OF PAGES "+num_pages);
}
}
```

```
class bookDetails {
    public static void main (String args[]) {
    int n;
    System.out.println (" ENTER THE NUMBER OF Book");
    Scanner vb = new Scanner (System.in);
     n = vb.nextInt();
     book b1[] = new Book[n];
    for (int i=0 ; i<n; i++) {
      b1[i] = new book ();
       b1[i].set();
       }
       for(int i=0 ; i<n; i++){
    System.out.println ( "DETAILS OF BOOK" +(i+1));
    System.out.println (b1[i]);
}
}
}
```

OUTPUT:—

```
ENTER THE NUMBER OF BOOKS
2
ENTER THE NAME OF THE BOOK
The murder on the orient esxpress
ENTER THE AUTHOR
agatha christie.
ENTER THE PRICE
 500
 ENTER THE NUMBER OF PAGES
  250
 ENTER THE NAME OF THE BOOK
  james bond.
 ENTER THE AUTHOR
  arthur conan
 ENTER THE PRICE
   400
```

ENTER THE NUMBER OF PAGES
172.
DETAILS OF BOOK 1
NAME OF THE BOOK: The murder on the orient express.

AUTHOR : agatha christie
PRICE : 500
NUMBER OF PAGES : 250.
DETAILS OF BOOK 2
NAME OF THE BOOK :- JAMES BOND
PRICE :- 400.
NUMBER OF PAGES :- 172

Rs

01/12/22

Output:

```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac book_details.java } ; i
f ($?) { java book_details }
ENTER THE NUMBER OF BOOKS
2
ENTER THE NAME OF THE BOOK
The murder on the orient express
ENTER THE AUTHOR
Agatha Christie
ENTER THE PRICE
500
ENTER THE NUMBER OF PAGES
250
ENTER THE NAME OF THE BOOK
James Bond
ENTER THE AUTHOR
Arthur Conan
ENTER THE PRICE
400
ENTER THE NUMBER OF PAGES
172
DETAILS OF BOOK 1
NAME OF THE BOOK : The murder on the orient express
AUTHOR : Agatha Christie
PRICE : 500
NUMBER OF PAGES250
DETAILS OF BOOK 2
NAME OF THE BOOK : James Bond
AUTHOR : Arthur Conan
PRICE : 400
NUMBER OF PAGES172
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

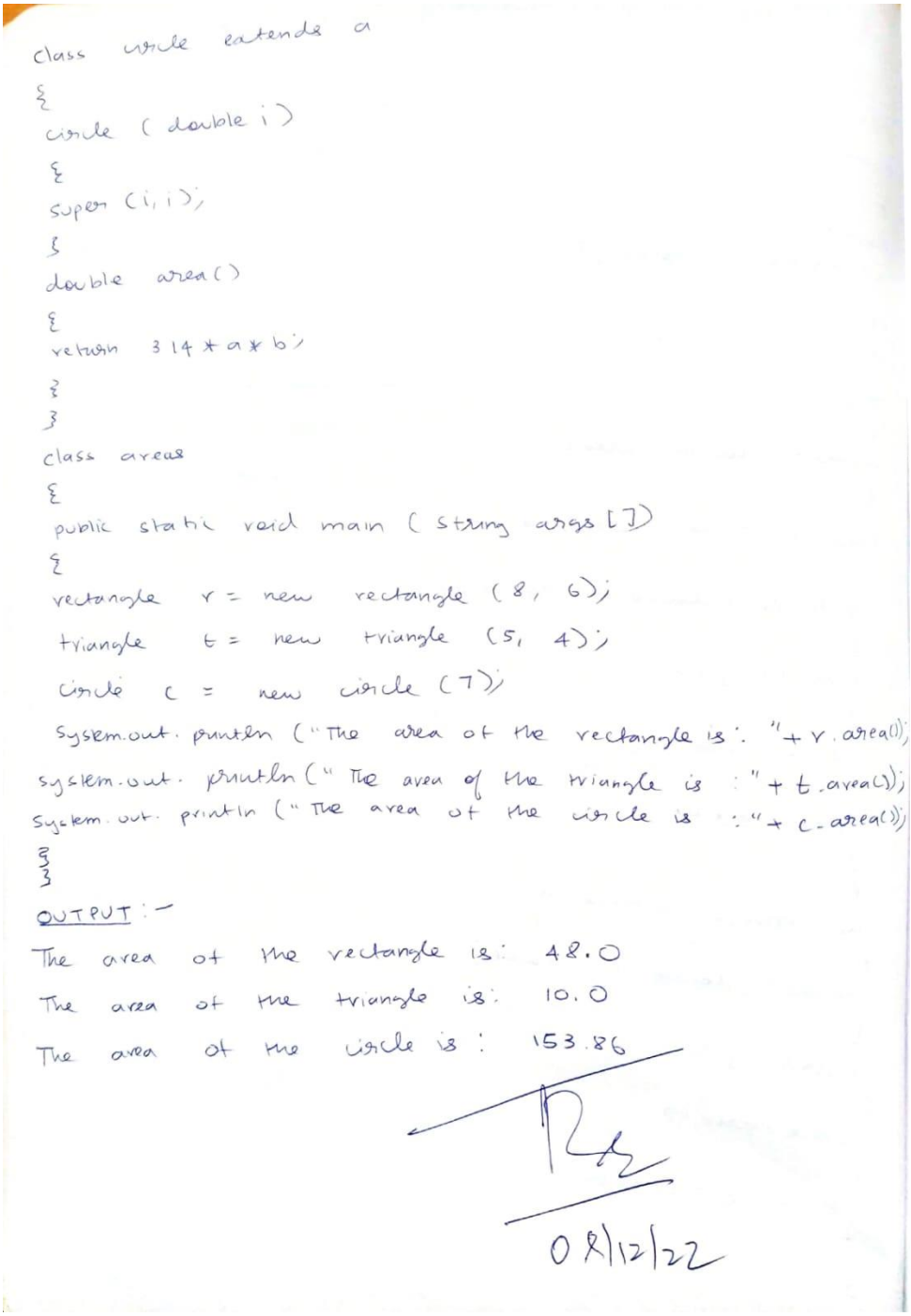
LAB PROGRAM 4

Q: 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.

8/12/22 WEEK-4

Lab program - 4

```
import java.util.*;
abstract class a
{
double a,b;
a( double i, double j)
{
a=i;
b=j;
}
abstract double area();
}
Class rectangle extends a
{
rectangle (double i, double j)
{
super(i,j);
}
double area()
{
return a*b;
}
}
class triangle extends a
{
triangle (double i, double j)
{
super(i,j);
}
double area()
{
return 0.5*a*b;
}
}
```



```
class circle extends a
{
circle ( double i )
{
super (i, i);
}
double area()
{
return 3.14 * a * b;
}
}
class areas
{
public static void main ( String args [])
{
rectangle r = new rectangle (8, 6);
triangle t = new triangle (5, 4);
circle c = new circle (7);
System.out.println ("The area of the rectangle is: " + r.area());
system.out.println (" The area of the triangle is : " + t.area());
System.out.println (" The area of the circle is : " + c.area());
}
}
```

OUTPUT :-

```
The area of the rectangle is: 48.0
The area of the triangle is: 10.0
The area of the circle is : 153.86
```

08/12/22

OUTPUT:

```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac area.java } ; if ($?) {
 java area }
The area of the rectangle is: 48.0
The area of the triangle is: 10.0
The area of the circle is: 153.86
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

LAB PROGRAM 5

Q: 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.

```
15/12/24                    WEEK - 5
                            BANK ACCOUNT
                                lab - program - 5
import java.util.Scanner;
class Account {
  public static int min = 500;
  String name;
  int acc_num;
  public float price;
  Scanner sc = new Scanner (System.in);
  public void get_info ()
  {
   System.out.println ("Enter Name:");
   name = sc.nextLine ();
   System.out.println ("Enter Account Number:");
   acc_num = sc.nextInt ();
   System.out.println ("Enter opening amount (must >500):");
   price = sc.nextFloat ();
   if ( price < 500)
        System.out.println ("Enter opening amount (must >500):"
  }
  public void show ()
  {
   System.out.println ("Name:" +name);
   System.out.println ("Account-number:" + acc_num);
   System.out.println ("Amount :" + price);
  }
}
class Current extends Account {
      float dep, wit, penalty;
      public void deposit ()
      {
       System.out.println ("Enter Amount to deposit");
       dep = sc.nextFloat ();
       show ();
```

```
        price += dep;
        System.out.println("Total Amount is:" + price);
    }
    public void check_Bal()
    {
        if( price < min )
            price = price - 150;
            System.out.println("Account balance is:" + price);
    }
    public void withdraw_Bal()
    {
        System.out.println("Enter Amount to withdraw");
        wit = sc.nextFloat();
        show();
        if( wit < price)
        {
            price -= wit;
            System.out.println("After withdrawal balance
                                amount :" + price);
        }
        else
            System.out.println("Insufficient Balance");
        check_Bal();
    }
}
class Savings extends Account {
    float dep, wit, intr;
    public void deposit()
    {
        System.out.println("Enter amount to be
                            deposited:");
        dep = sc.nextFloat();
        show();
```

```
        price += dep;
        System.out.println("Total Amount:" + price);
    }
    public void check_interest()
    {
        inter = price * 2/100;
        price += inter;
        System.out.println("Total amount with interest:"+price);
    }
    public void withdraw_Bal()
    {
        System.out.println("Enter amount to withdraw:");
        wit = sc.nextFloat();
        show();
        if( wit < price)
        {
            price -= wit;
            System.out.println("After withdrawal balance
                                amount:" + price);
        }
        else
            System.out.println("Insufficient Balance");
    }
}
class labprog5 {
    public static void main(String args[]) {
        String ch;
        int co = 0;
        Scanner sc = new Scanner(System.in);
        Current c1 = new Current();
        Savings s1 = new Savings();
        System.out.println("Choose Account Type:");
        System.out.println("Press c for Current Account:");
        System.out.println("Press s for Savings Account:");
```

```
ch = sc.next();
if (ch.equalsIgnoreCase("c"))
{
    c1.get_info();
    c1.check_Bal();
    while (co != 4)
    {
    System.out.println("1. Display \n 2. Deposit \n
                        3. Withdraw \n 4. Exit");

    System.out.println("enter your choice");
    int cho = sc.nextInt();
    switch (cho)
    {
        case 1: c1.show();
                break;
        case 2: c1.deposit();
                break;
        case 3: c1.withdraw_Bal();
                break;
        case 4: System.exit(0);
                break;
        default: System.out.println("wrong choice");
    }
}
}
else if (ch.equalsIgnoreCase("s"))
{
    s1.get_info();
    while (co1 != 5)
    {
      System.out.println("1. Display \n 2. Deposit \n
                          3. Withdraw \n 4. Interest \n
```

```
System.out.println("Enter your choice");
int cho = sc.nextInt();
switch(cho)
{
    case 1: s1.show();
            break;
    case 2: s1.deposit();
            break;
    case 3: s1.withdraw-Bal();
            break;
    case 4: s1.check-intst();
            break;
    case 5: System.exit(0);
            break;
    default: System.out.println("wrong choice");
}
}
else
    System.out.println("wrong choice!.");
}
}
```

OUTPUT :-

```
choose Account Type:
Press c for Current Account:
Press s for Saving Account:
C
Enter Name:
Siddharth
Enter Account Number
2222
enter opening amount (must>500):
5000
Account balance is 5000.0
1. Display
2. Deposit
3. Withdraw
4. Exit.
Enter your choice
2
Enter Amount to deposit
5000
1. Display
2. Deposit
3. Withdraw
4. Exit
Enter your choice
1
Name : Siddharth
Account_number : 2222
Amount : 10000.0
1. Display
2. Deposit
3. Withdraw
4. Exit
enter your choice
4
```

Choose Account Type -

Press c for Current Account

Press s for Saving Account

s

Enter Name:

Siddharth

Enter Account Number:

2222

Enter opening amount (must >500):

5000

1. Display
2. Deposit
3. Withdraw
4. Interest
5. Exit.

enter your choice.

4

Total amount with interest : 5100.0

1. Display
2. Deposit
3. Withdraw
4. Interest
5. Exit

Enter your choice:

5

29/12/22

OUTPUT:

```
Press c for Current Account:
Press s for Saving Account:
c
Enter Name:
Siddharth
Enter Account Number:
2222
Account balance is:5000.0
1.Display
2.Deposit
3.Withdraw
4.Exit
Enter Your Coice
2
Eneter Ammount to deposit
5000
Name:Siddharth
Account_number:2222
Amount:5000.0
Total Amount is :10000.0
1.Display
2.Deposit
3.Withdraw
4.Exit
Enter Your Coice
1
Name:Siddharth
Account_number:2222
Amount:10000.0
1.Display
2.Deposit
3.Withdraw
4.Exit
Enter Your Coice
4
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

```
Choose Account Type:
Press c for Current Account:
Press s for Saving Account:
s
Enter Name:
Siddharth
Enter Account Number:
2222
Enter opening amount (must>500):
5000
1.Display
2.Deposit
3.Withdraw
4Intrest
5.Exit
Enter Your Coice
4
Total amount with interest:5100.0
1.Display
2.Deposit
3.Withdraw
4Intrest
5.Exit
Enter Your Coice
1
Name:Siddharth
Account_number:2222
Amount:5100.0
1.Display
2.Deposit
3.Withdraw
4Intrest
5.Exit
Enter Your Coice
5
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

LAB PROGRAM 6

Q: 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.

5/1/23 WEEK- 6

Lab program - 6

Age Exception Program:-

SOURCE CODE:-

```
import java.util.*;
class WrongAge extends Exception
{
  public String toString()
   {
    return "Please enter the right age:";
   }
}
class Father
{
  int age;
  Father()
  {
  }
  Father (int age1)
   {
    age = age1;
    System.out.println("Father age:" + age);
   }
}
class Son extends Father
{
  Son ( int age1)
   {
    super();
    System.out.println("Son's age:" + age1);
   }
}
```

```
class age
{
public static void main (String args[]) throws
                                        WrongAge
  {
    Scanner sc = new Scanner (System.in);
    int j, k;
    System.out.println (" enter father's age:");
    j = sc.nextInt();
    System.out.println (" enter sons age:");
    k = sc.nextInt();
    if ( j <= 0 || j <= k)
    {
      throw new WrongAge();
    }
    else
    {
      Father f = new Father (j);
      Son s = new Son (k);
    }
  }
}
```

OUTPUT:-

```
enter fathers age:
50
enter sons age:
14
Father age: 50
son age: 14
```

enter fathers age:
40
enter sons age:
60
Exception in thread "main" Please enter the right age:
at age.main(age.java:45).

enter fathers age:
-12
enter sons age:-
-35
Exception in thread "main". Please enter the right age:
at age.main(age.java:45)

enter fathers age:
-12
enter sons age:
56
Exception in thread "main" Please enter the right age
at age.main(age.java:

05/01/23

OUTPUT:

```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac father_son.java } ; if
($?) { java father_son }
enter fathers age:
50
enter sons age:
14
Father age:50
Son age:14
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

```
($?) { java father_son }
enter fathers age:
40
enter sons age:
60
Exception in thread "main" Please enter the right age:
        at father_son.main(father_son.java:44)
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac father_son.java } ; if
($?) { java father_son }
enter fathers age:
-12
enter sons age:
-35
Exception in thread "main" Please enter the right age:
        at father_son.main(father_son.java:44)
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac father_son.java } ; if
($?) { java father_son }
enter fathers age:
-12
enter sons age:
56
Exception in thread "main" Please enter the right age:
        at father_son.main(father_son.java:44)
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

LAB PROGRAM 7

Q: 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.

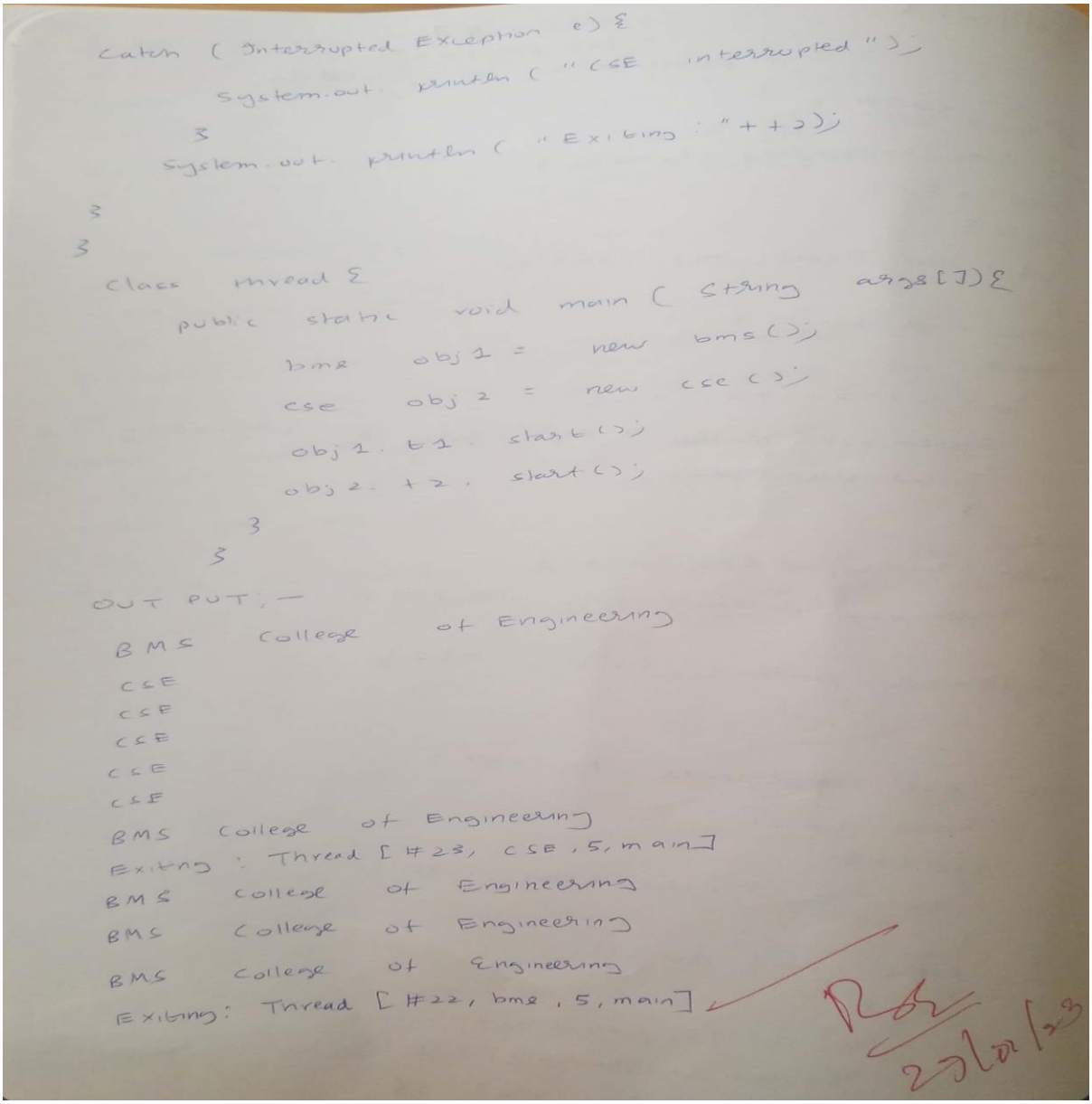
27/1/2023 WEEK 7

lab program - 7

Implementation of Threads.

SOURCE CODE -

```
class bms implements Runnable {
 Thread t1;
 bms() {
  t1 = new Thread( this, "bms");
 }
 public void run () {
  try {
    for (int i = 5; i > 0; i--) {
     System.out.println (" Bms College of Engineering ");
      Thread.sleep (10000);
      }
  }
 catch (InterruptedException e) {
    System.out.println ("BMS interrupted \n");
 }
 System.out.println ( "Exiting :" + t1);
 }
}
class cse implements Runnable {
     Thread t2;
     cse () {
       t2 = new Thread (this, "cse");
     }
     public void run (){
      try {
         for ( int i=5; i >0; i-- ) {
          system.out.println ( "CSE");
           Thread.sleep(2000); }}
```

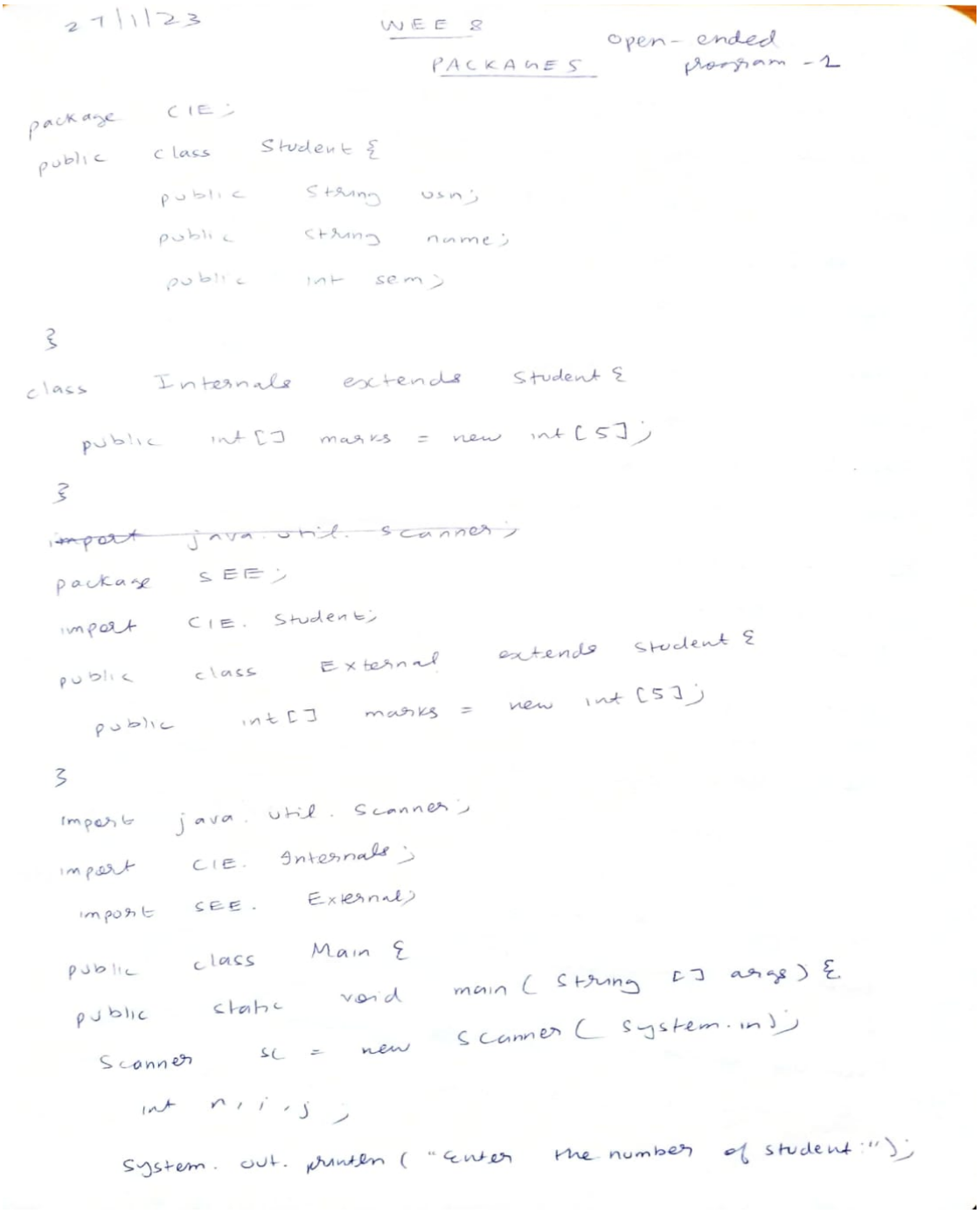


OUTPUT:

```
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA> cd "c:\Users\LENOVO\OneDrive\Desktop\JAVA\" ; if ($?) { javac thread.
java } ; if ($?) { java thread }
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
Exiting:Thread[#23,cse,5,main]
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
Exiting:Thread[#22,bms,5,main]
PS C:\Users\LENOVO\OneDrive\Desktop\JAVA>
```

LAB PROGRAM 8

Q: 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.



```
27/11/23                WEEK 8                    open-ended
                        PACKAGES                  program - 2

package CIE;
public class Student {
    public String usn;
    public String name;
    public int sem;
}
class Internals extends Student {
    public int[] marks = new int[5];
}
import java.util.Scanner;
package SEE;
import CIE.Student;
public class External extends Student {
    public int[] marks = new int[5];
}
import java.util.Scanner;
import CIE.Internals;
import SEE.External;
public class Main {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
int n, i, j;
System.out.println("Enter the number of student:");
```

```
n = sc.nextInt();
Internals [] inter = new Internals [n];
External [] exter = new External [n];
for (i=0; i<n; i++){
  inter [i] = new Internals();
  exter [i] = new External();
System.out.println("Enter the details of "+(i+1)+"Stud
System.out.println("Enter USN:");
inter [i].usn = sc.next();
exter [i].usn = inter[i].usn;
System.out.println("Enter Name:");
inter [i].name = sc.next();
exter[i].name = inter[i].name;
System.out.println("Enter Semester");
inter [i].sem = sc.nextInt();
exter[i].sem = inter[i].sem;
System.out.println("Enter the internal marks
  of 5 courses having 3 credits");
for (j=0; j<5; j++){
inter [i].marks [j] = sc.nextInt();
}
System.out.println("Enter the external marks of
  5 courses having 3 credits");
```

```
for ( j=0 ; j < 5 ; j++ ) {
  enter[i].marks[j]= sc.nextInt();
}
}
System.out.println("Details of student with their final marks :");
for (i=0; i<n; i++) {
System.out.println("student "+(i+1)+" :");
System.out.println("USN :"+ inter[i].usn);
System.out.println("Name :"+ inter[i].name);
System.out.println("semester :"+ inter[i].sem);
System.out.println("Final marks :");
for (j=0; j<5 ;j++){
System.out.println((j+1)+" :"+(inter[i].marks+
                               enter[i].marks[j]/2));
}
}
}
}
```

PTO

OUTPUT: -

Enter USN, NAME & SEM

1BM21CS247

Siddharth

3

Student Details:

USN : 1BM21CS247

NAME:- siddharth

SEM: 3

Enter marks in 5 courses:

40
45
56
41
48

INTERNAL MARKS

subject 0 = 48

subject 1 = 45

subject 2 = 56

subject 3 = 41

subject 4 = 48

Enter external marks scored in 5 courses:

43
45
47
48
41

EXTERNAL MARKS

subject 0 = 43

subject 1 = 45

subject 2 = 47

subject 3 = 48

subject 4 = 41

FINAL MARKS

Subject 0 = 61

subject 1 = 67

subject 2 = 79

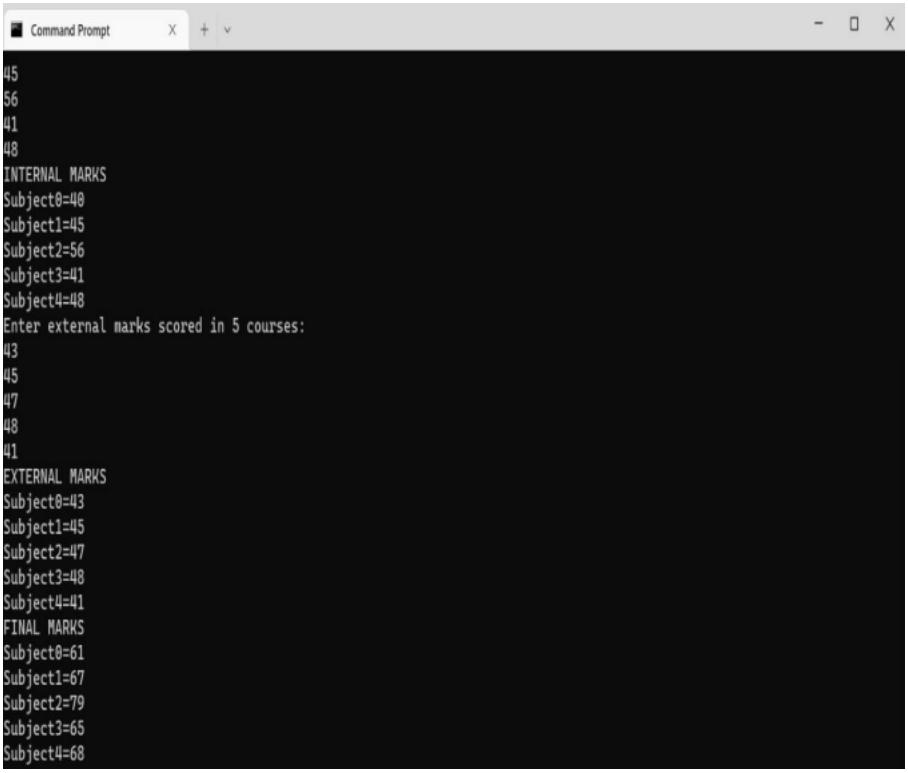
subject 3 = 65

subject 4 = 68

27/01/23

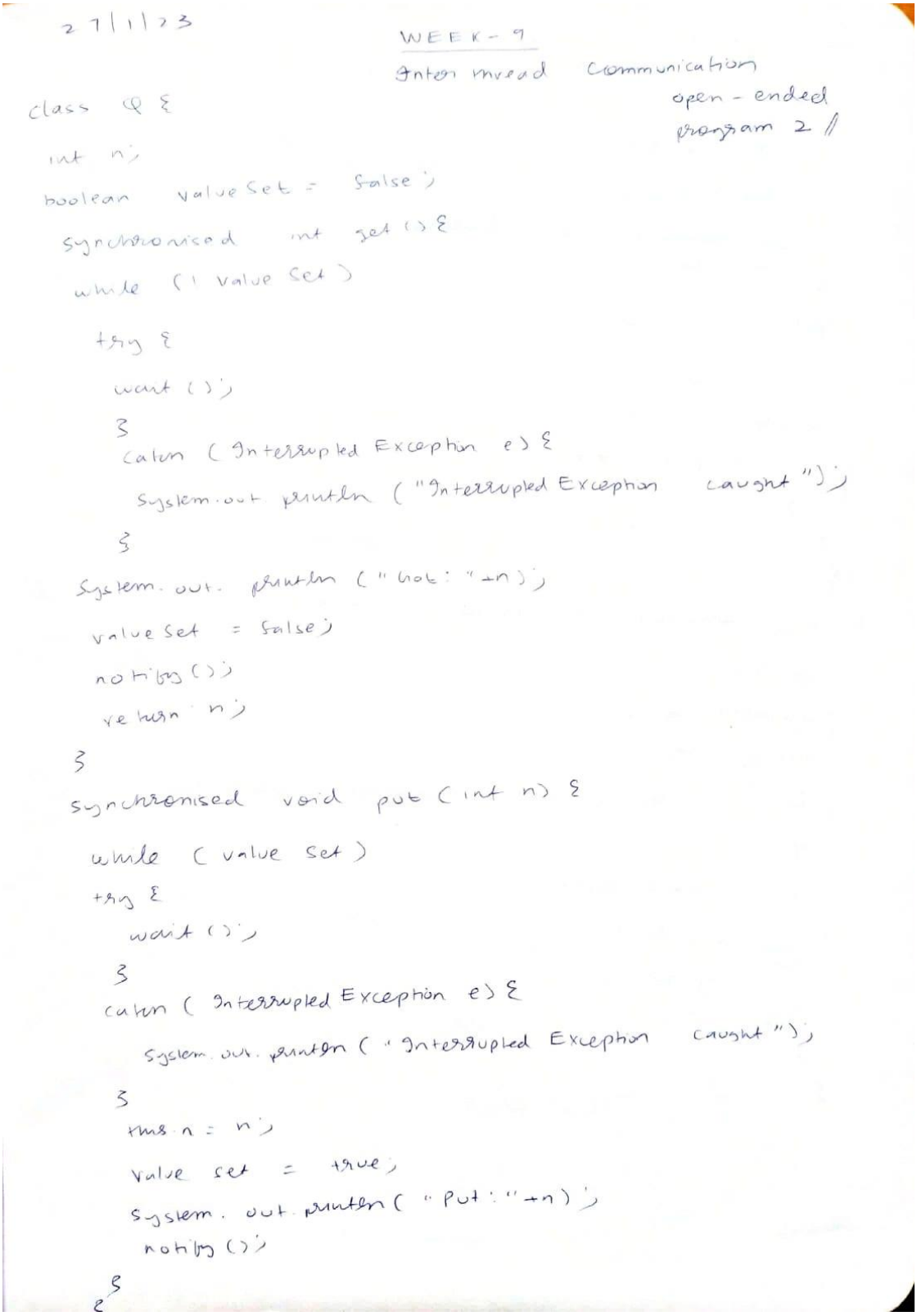
OUTPUT:


Enter USN, NAME & SEM
1BM21CS254
Z
1
Student Details:
USN:1BM21CS254
NAME:Z
SEM:1
Enter marks scored in 5 courses:
40
45
56
41
48
INTERNAL MARKS
Subject0=40
Subject1=45
Subject2=56
Subject3=41
Subject4=48
Enter external marks scored in 5 courses:


Command Prompt
45
56
41
48
INTERNAL MARKS
Subject0=40
Subject1=45
Subject2=56
Subject3=41
Subject4=48
Enter external marks scored in 5 courses:
43
45
47
48
41
EXTERNAL MARKS
Subject0=43
Subject1=45
Subject2=47
Subject3=48
Subject4=41
FINAL MARKS
Subject0=61
Subject1=67
Subject2=79
Subject3=65
Subject4=68

LAB PROGRAM 9

Q) Demonstrate Inter process Communication and deadlock



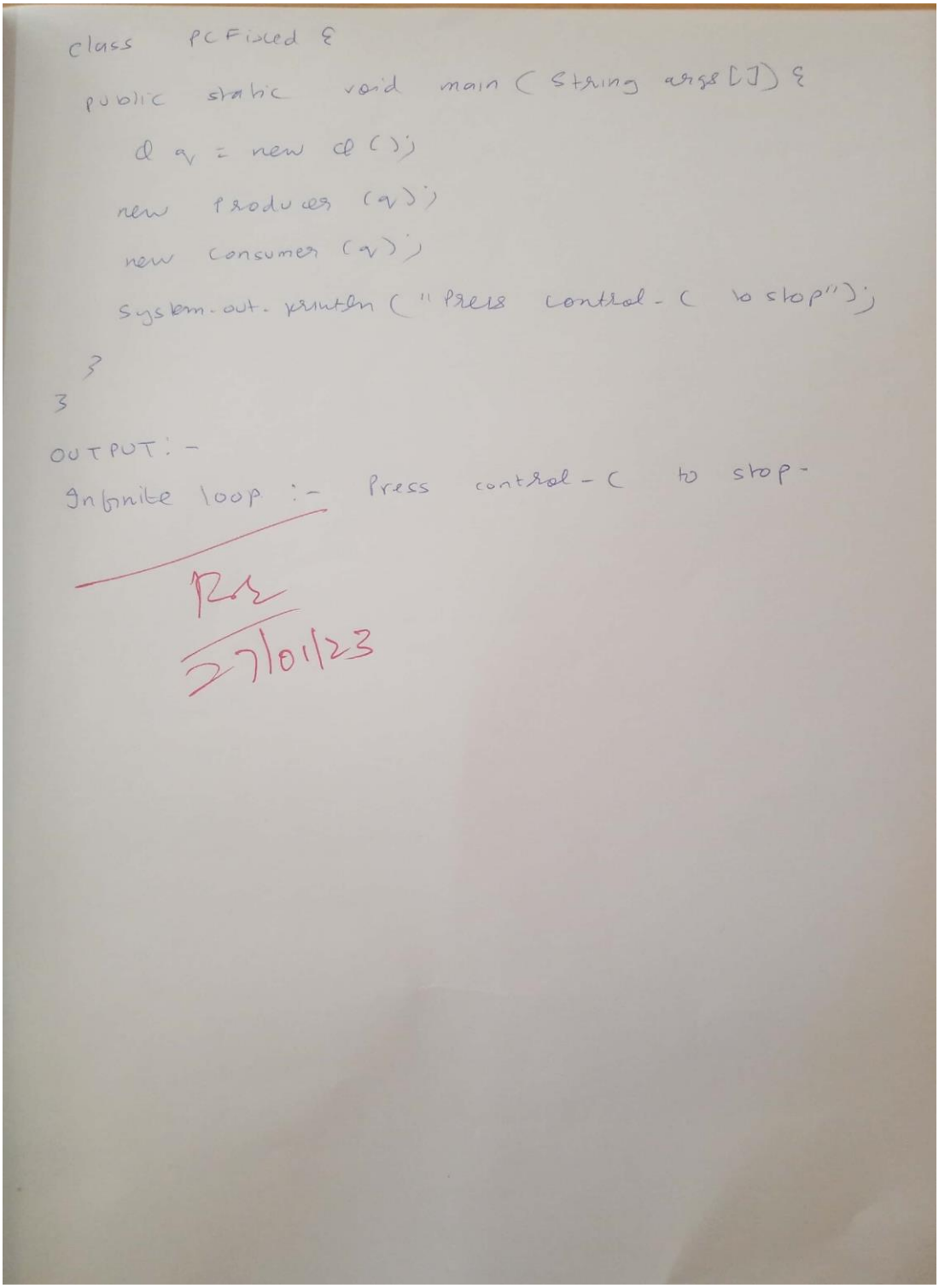
```
27/1/23

WEEK - 9
Inter thread Communication
                                    open - ended
                                    program 2 //

class Q {
 int n;
 boolean valueSet = false;
 synchronised int get() {
  while (!valueSet)
   try {
    wait();
    }
    catch (InterruptedException e) {
     System.out.println ("InterruptedException caught");
    }
  System.out.println ("Got: " + n);
  valueSet = false;
  notify();
  return n;
 }
 synchronised void put(int n) {
  while (valueSet)
  try {
    wait();
   }
   catch (InterruptedException e) {
     System.out.println ("InterruptedException caught");
   }
   this.n = n;
   valueSet = true;
   System.out.println("Put: " + n);
   notify();
 }
```

```
class Producer implements Runnable {
    Q q;
    Producer(Q q) {
        this.q = q;
        new Thread(this, "Producer").start();
    }
    public void run() {
        int i = 0;
        while (true) {
            q.put(i++);
        }
    }
}

class Consumer implements Runnable {
    Q q;
    Consumer(Q q) {
        this.q = q;
        new Thread(this, "Consumer").start();
    }
    public void run() {
        while (true) {
            q.get();
        }
    }
}
```



OUTPUT:

INFINITE LOOP (PRESS CONTROL C TO STOP)