VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Object Oriented Java Programming
(22CS3PCOOJ)

Submitted by

Dhavan SK (1BM21CS054)

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
October-2022 to 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

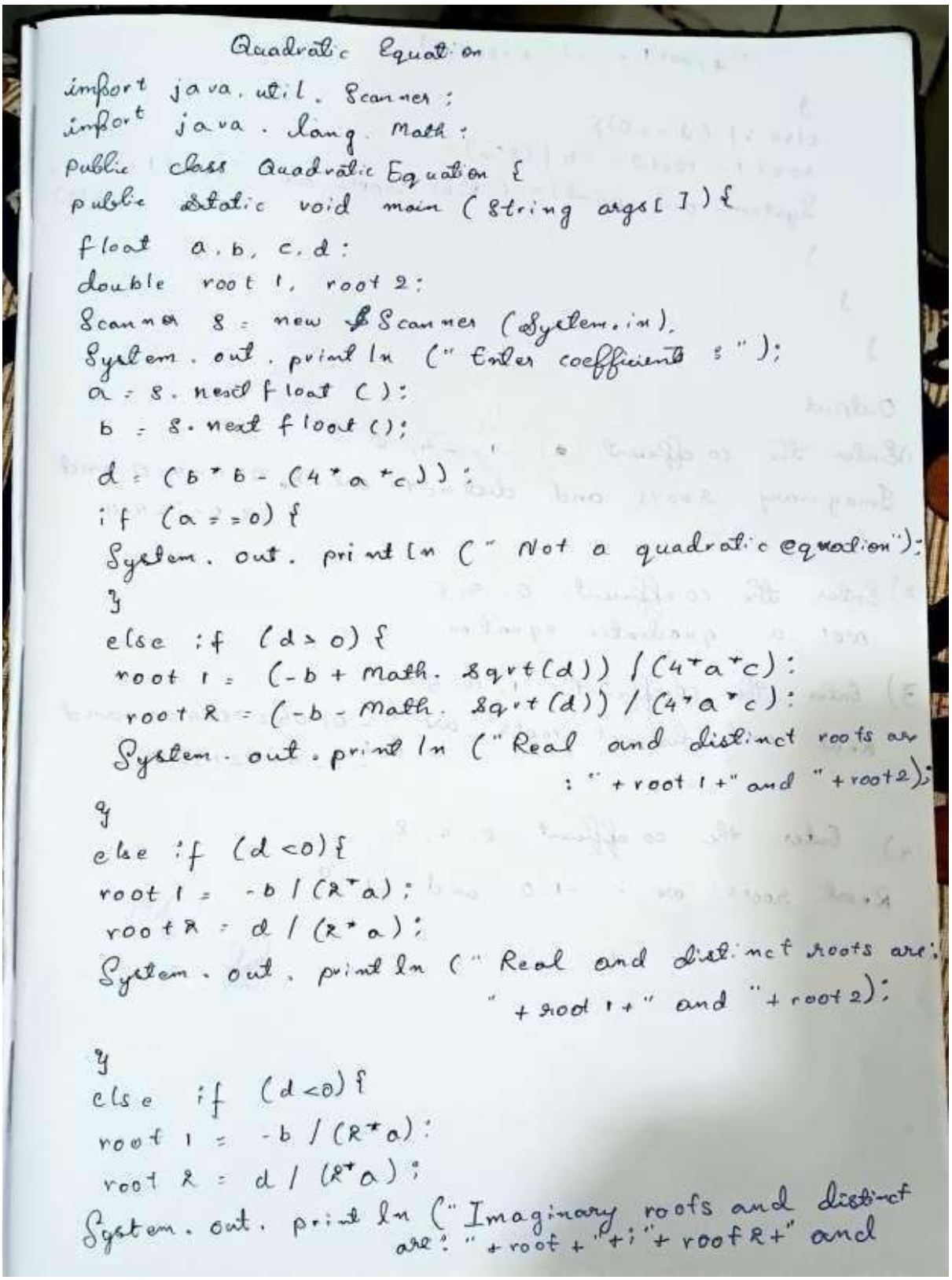
This is to certify that the Lab work entitled "Object Oriented Java Pragramming (22CS3PCOOJ)" carried out by **Dhavan SK (1BM21CS054),** who is bonafide student of **B. M. S. College of Engineering.** It is in partial fulfillment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visvesvaraya Technological University, Belgaum during the year 2022. The Lab report has been approved as it satisfies the academic requirements in respect of a Object Oriented Java Programming(22CS3PCOOJ) work prescribed for the said degree.

Dr.NANDHINI VINEETH
Assistant Professor
Department of CSE
BMSCE, Bengaluru

Dr. Jyothi S Nayak
Professor and Head
Department of CSE
BMSCE, Bengaluru

`

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.



```
Quadratic Equation
import java.util.Scanner;
import java.lang.Math;
Public class Quadratic Equation {
public static void main (String args[]){
float a, b, c, d;
double root1, root2;
Scanner s = new Scanner (System.in);
System.out.println (" Enter coefficients : ");
a = s.nextfloat ();
b = s.nextfloat ();
d = (b*b - (4*a*c));
if (a==0) {
System.out.println (" Not a quadratic equation");
}
else if (d>0) {
root1 = (-b + Math.Sqrt(d)) / (4*a*c);
root2 = (-b - Math.Sqrt(d)) / (4*a*c);
System.out.println (" Real and distinct roots are
: " + root1 + " and " + root2);
}
else if (d<0){
root1 = -b / (2*a);
root2 = d / (2*a);
System.out.println (" Real and distinct roots are:
" + root1 + " and " + root2);

}
else if (d<0) {
root1 = -b / (2*a);
root2 = d / (2*a);
System.out.println (" Imaginary roots and distinct
are: " + root + "+i" + root2 + " and
```

```
" + root1 + " - i " + root2);
    }
    else if (d == 0){
    root1 = root2 = -b / (2*a);
    System.out.println ("Real roots are : " + root1 + " and " + root2);
    }
  }
}
```

Output

1) Enter the co-efficient a, b, c 1, -4, 6
Imaginary roots and distinct are 2.0 + i -4.0 and 2.0 - i -4.0

2) Enter the co-efficient 0, 5, 6
Not a quadratic equation

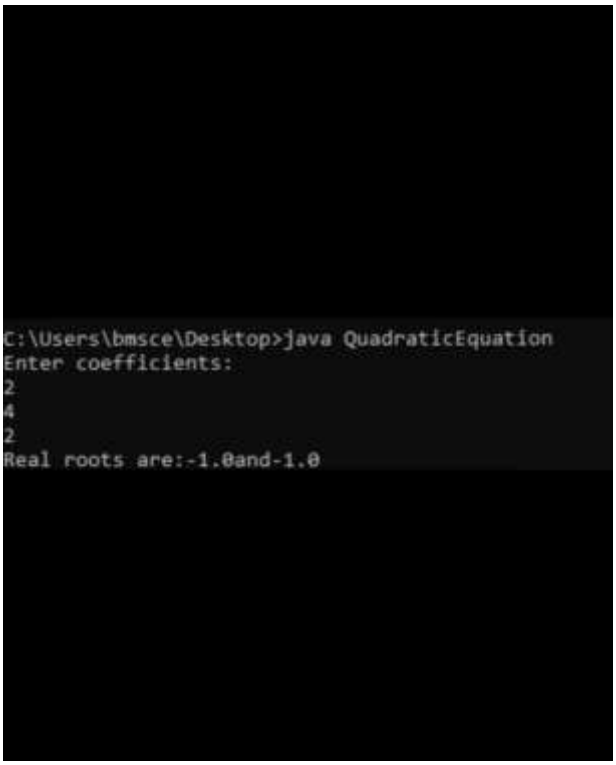
3) Enter the co-efficient 1, 10, 5
Real and distinct roots are : -0.05278404 and -0.94721359

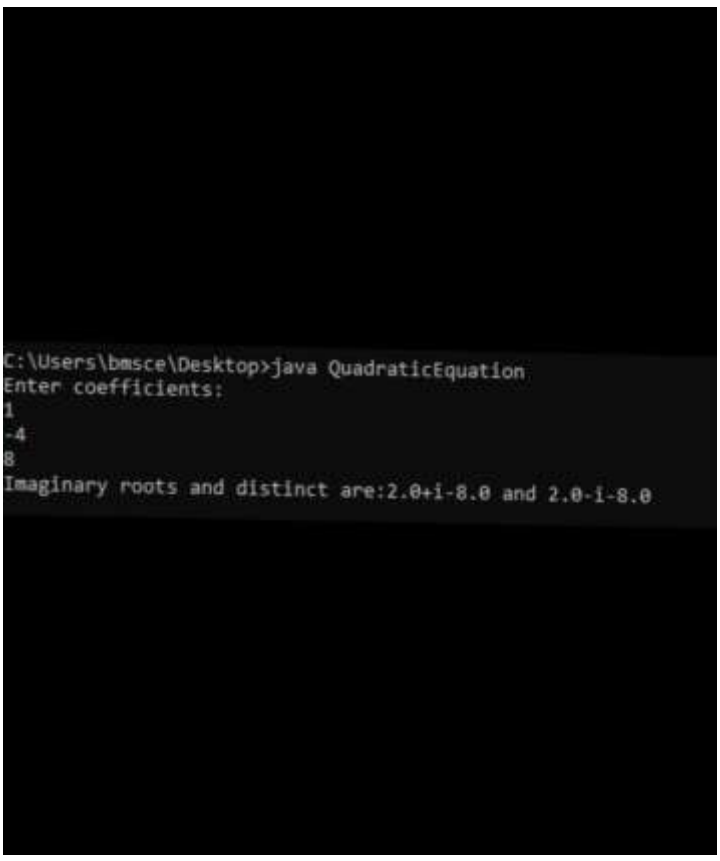
4) Enter the co-efficient 2, 4, 2.
Real roots are : -1.0 and -1.0

Output

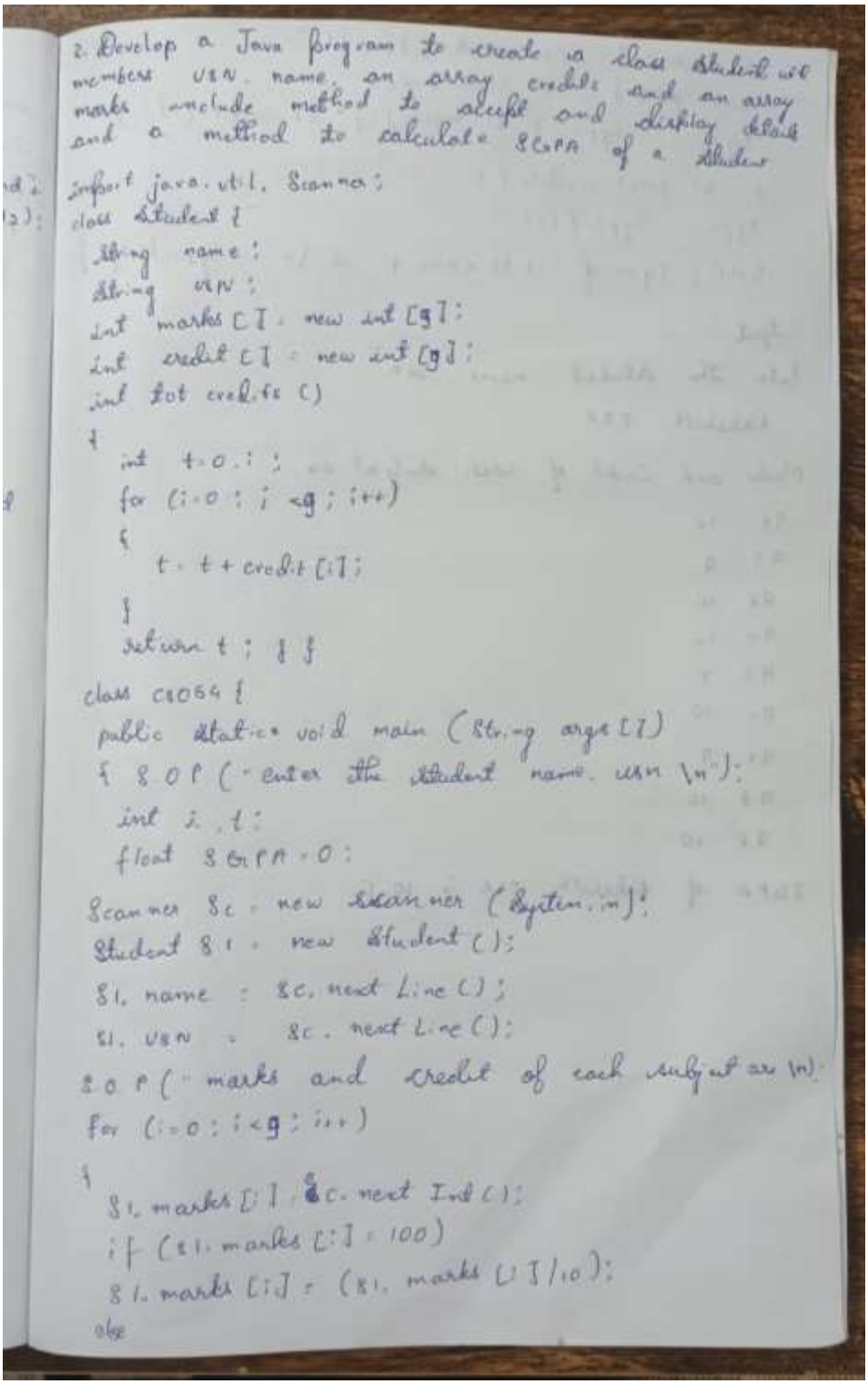
```
C:\Users\bmsce\Desktop>java QuadraticEquation
Enter coefficients:
0
5
6
Not a quadratic equation
```

```
C:\Users\bmsce\Desktop>java QuadraticEquation
Enter coefficients:
1
-4
6
Imaginary roots and distinct are:2.0+i-4.0 and 2.0-i-4.0
```


C:\Users\bmsce\Desktop>java QuadraticEquation
Enter coefficients:
2
4
2
Real roots are:-1.0and-1.0


C:\Users\bmsce\Desktop>java QuadraticEquation
Enter coefficients:
1
-4
8
Imaginary roots and distinct are:2.0+i-8.0 and 2.0-i-8.0

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.



```
2. Develop a Java program to create a class Student with
members USN, name, an array credits and an array
marks include method to accept and display details
and a method to calculate SGPA of a Student

import java.util.Scanner;
class Student {
  String name;
  String usn;
  int marks[] = new int[9];
  int credit[] = new int[9];
  int tot_credits()
  {
    int t=0,i;
    for(i=0; i<9; i++)
    {
      t = t + credit[i];
    }
    return t; } }
class cse64 {
  public static void main (String args[])
  { SOP("enter the student name, usn \n");
    int i, t;
    float SGPA=0;
  Scanner sc = new Scanner (System.in);
  Student s1 = new Student();
  s1.name = sc.nextLine();
  s1.usn = sc.nextLine();
SOP("marks and credit of each subject are \n");
for(i=0; i<9; i++)
{
  s1.marks[i] = sc.nextInt();
  if(s1.marks[i] > 100)
  s1.marks[i] = (s1.marks[i]/10);
else
```

```
        s1.marks[i] = (s1.marks[i]/10)+1;
        s1.credit[i] = sc.nextInt();
        sgpa = sgpa + s1.marks[i]*s1.credit[i]; }
        t = s1.totalcredits();
        sgpa = sgpa/t;
        S.O.P("Sgpa of " + s1.name + " is \n" + sgpa); } }
```

Output :

Enter the Student name, usn

Rakshith 03R

Marks and Credit of each Subject are

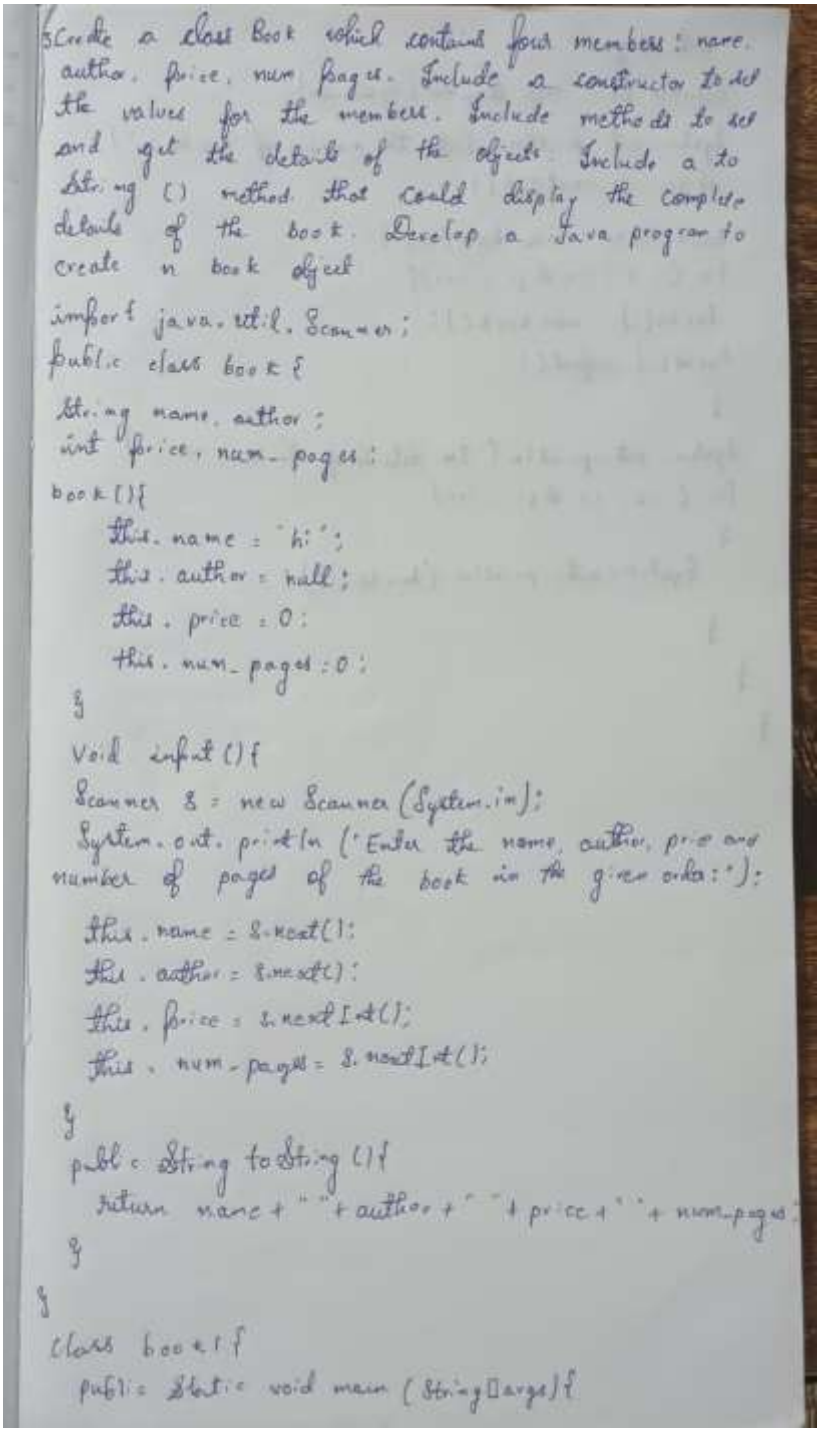
98 10
97 9
98 10
94 10
93 9
94 10
97 9
98 10
98 10

SGPA of Rakshith 03R is 10.0

Output:

```
C:\Users\thris\OneDrive\Desktop\dhavan sk>cd C:\Users\thris\OneDrive\Desktop\dhavan sk

C:\Users\thris\OneDrive\Desktop\dhavan sk>javac student.java

C:\Users\thris\OneDrive\Desktop\dhavan sk>java cs054
Enter the student name,usn

Rakshith 032

Marks and credit of each subject are

98 10
97 9
98 10
94 10
93 9
94 10
97 9
98 10
98 10
sgpa ofRakshith 032is
10.0
```

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



```
/*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 to
String() method that could display the complete
details of the book. Develop a Java program to
create n book object*/
import java.util.Scanner;
public class book {
String name, author;
int price, num_pages;
book(){
    this.name = "hi";
    this.author = null;
    this.price = 0;
    this.num_pages = 0;
}
Void input(){
Scanner s = new Scanner(System.in);
System.out.println("Enter the name, author, price and
number of pages of the book in the given order:");
  this.name = s.next();
  this.author = s.next();
  this.price = s.nextInt();
  this.num_pages = s.nextInt();
}
public String toString(){
  return name + " " + author + " " + price + " " + num_pages;
}
}
class book1{
  public static void main(String[]args){
```

```
    int size;
    Scanner ss = new Scanner(System.in);
    System.out.println("Enter the number of books : ");
    size = ss.nextInt();
    book books = new book[size];
    for (i=0; i< size ; i++){
     books[i] = new book();
    books[i].infput();
     }
    System.out.println("The details of the book are :");
    for (i=0; i< size ; i++)
    {
      System.out.println(books[i]);
     }
  }
}
```

output :

Enter the number of books

2

Enter the name, author, price and number of pages of the book in given order

Shivaprasad Tagore 80 100

Enter the name, author, price and number of pages of the book in given order

Manish Kalam 80 100

The details of the book are :

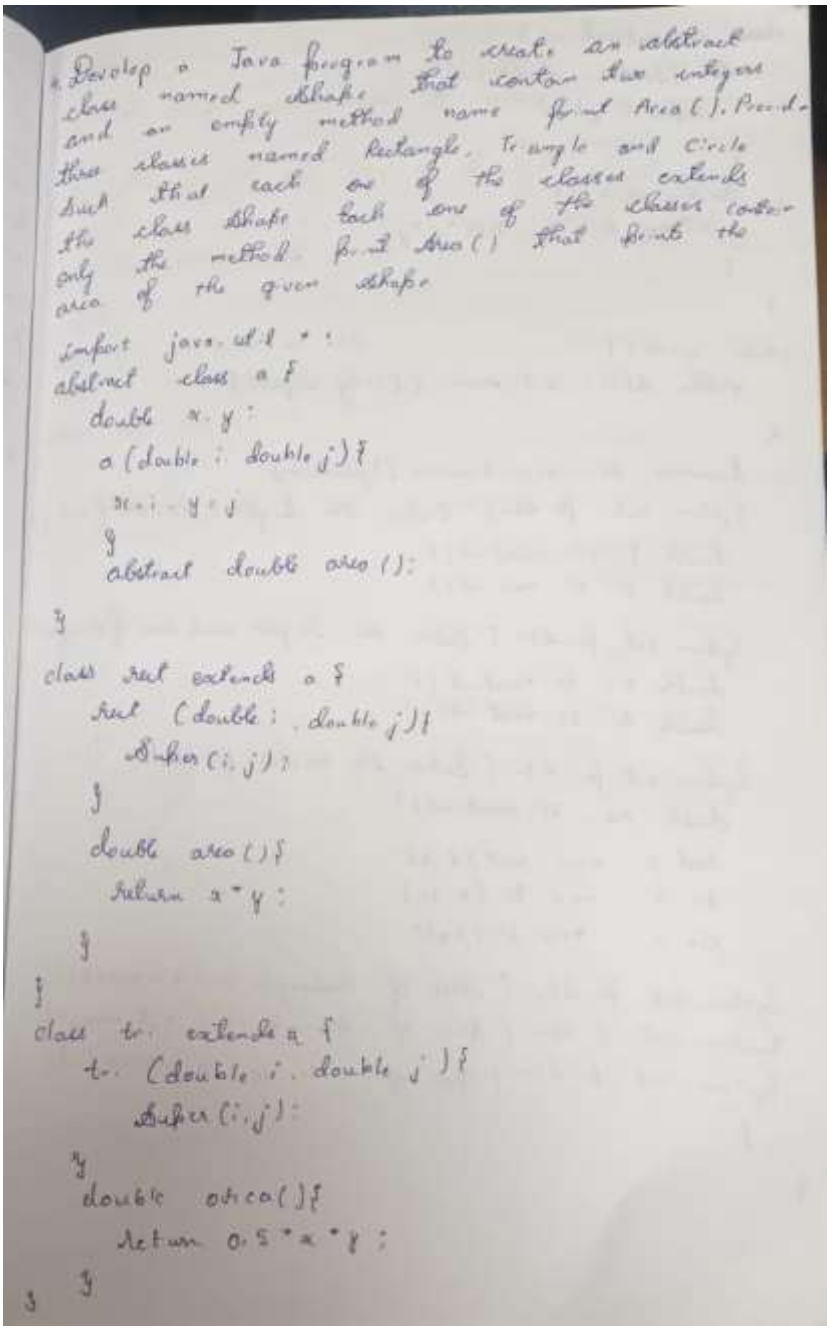
Shivaprasad Tagore 80 100

Manish Kalam 80 100

Output:

```
C:\Users\thris\OneDrive\Desktop\dhavan sk>cd C:\Users\thris\OneDrive\Desktop\dhavan sk

C:\Users\thris\OneDrive\Desktop\dhavan sk>javac book.java

C:\Users\thris\OneDrive\Desktop\dhavan sk>java book1
Enter the number of books:
2
Enter the name, author, price and number of pages of the book in the given order:
Shivaprasad  Tagore 80 100
Enter the name, author, price and number of pages of the book in the given order:
Manish  Kalam 80 100
The details of the book are:
Shivaprasad Tagore 80 100
Manish Kalam 80 100
```

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.



```
4. Develop a Java program to create an abstract
class named Shape that contain two integers
and an empty method name print Area(). Provid
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 print Area() that prints the
area of the given shape.

import java.util.*;
abstract class a {
   double x, y;
   a (double i, double j) {
    x = i; y = j;
   }
   abstract double area();
}
class rect extends a {
   rect (double i, double j) {
     super(i, j);
   }
   double area() {
     return x * y;
   }
}
class tr extends a {
   tr (double i, double j) {
     super(i, j);
   }
   double area() {
     return 0.5 * x * y;
   }
}
```

```
class cir extends a {
  cir tri (double i, double j){
     super (i, j);
  }
  double area(){
    return 0.5*x* 3.14 *x *y;
  }
}

class week4 {
   public static void main (String args[])
{
   Scanner sc = new Scanner (System.in);
   System.out.println(" Enter the length & breadth of rectangle
    double l = sc.nextint();
    double b = sc.nextint();

   System.out.println("Enter the height and base of triangle:");
    double h = sc.nextint();
    double b = sc.nextint();

  System.out.println("Enter the radius of circle:");
    double ra = sc.nextint();

    rect r = new rect (l, b);
    tri t = new tri (h, ba);
    cir c = new cir(ra);

System.out.println("Area of rectangle is " + r.area());
System.out.println("Area of triangle is " + t.area());
System.out.println("Area of circle is " + c.area());
}
```

6. Output:

Enter the length and breadth of rectangle:
4 5
Enter the height and base of triangle:
3 6
Enter the radius of circle:
8 0
Area of rectangle is 20
Area of triangle is 18
Area of circle is 20096

Output:

```
Enter the length and breadth of rectangle:
3 4
Enter the height and base of triangle:
5 7
Enter the radius of circle:
6
Area of rectangle is  12.0
Area of triangle is  17.5
Area of circle is  113.03999999999999
```

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. Currentaccount 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 ofaccount. 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 andupdate the balance.

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 all 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 deposite interest
d) Permit withdrawal and update the balance

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

~~Complete the of~~

```
import java.util.*;
import java.lang.Math;
class bank {
Scanner sc = new Scanner(System.in);
String name;
int acc-no;
float bal.si;
```

```
Void accept () {
 System.out.println ("Enter your name");
  name = sc.nextline();
 System.out.println ("Enter the balance amount");
  bal = sc.nextfloat();
 }
 Void display () {
 System.out.println ("Name : " + name);
}
Void deposit () {
 float amount;
 int choice;
 System.out.println ("Do you want to deposit (1 for yes,
                                              2 for no)");
choice = sc.nextInd();
if (choice == 1) {
System.out.println ("Enter the amount to be deposited");
 amount = sc.nextfloat();
 if (amount > bal) {
 System.out.println ("Amount in bank insufficient");
}
else {
 bal = bal + amount;
 }
 System.out.println ("Current balance : " + bal);
}
}
}
```

```
class current extends bank {
    int service_fee = 50;
    void cheque() {
    System.out.println ("Cheque Service available");
}
void withdrawal () {
float amt;
System.out.println ("Enter the amount to be withdrawn");
amt = sc.nextfloat();
if (amt > bal)
System.out.println ("Balance insufficient");
else {
bal = bal - amt;
if (bal < 1000) {
bal = bal - service_fee;
System.out.println ("50rs is taken as service fee");
}
System.out.println ("Withdrawn : " + amt);
System.out.println ("Current balance : " + bal);
}
}
}
class Savings extends bank {
void cheque() {
System.out.println ("Cheque service not available");
}
```

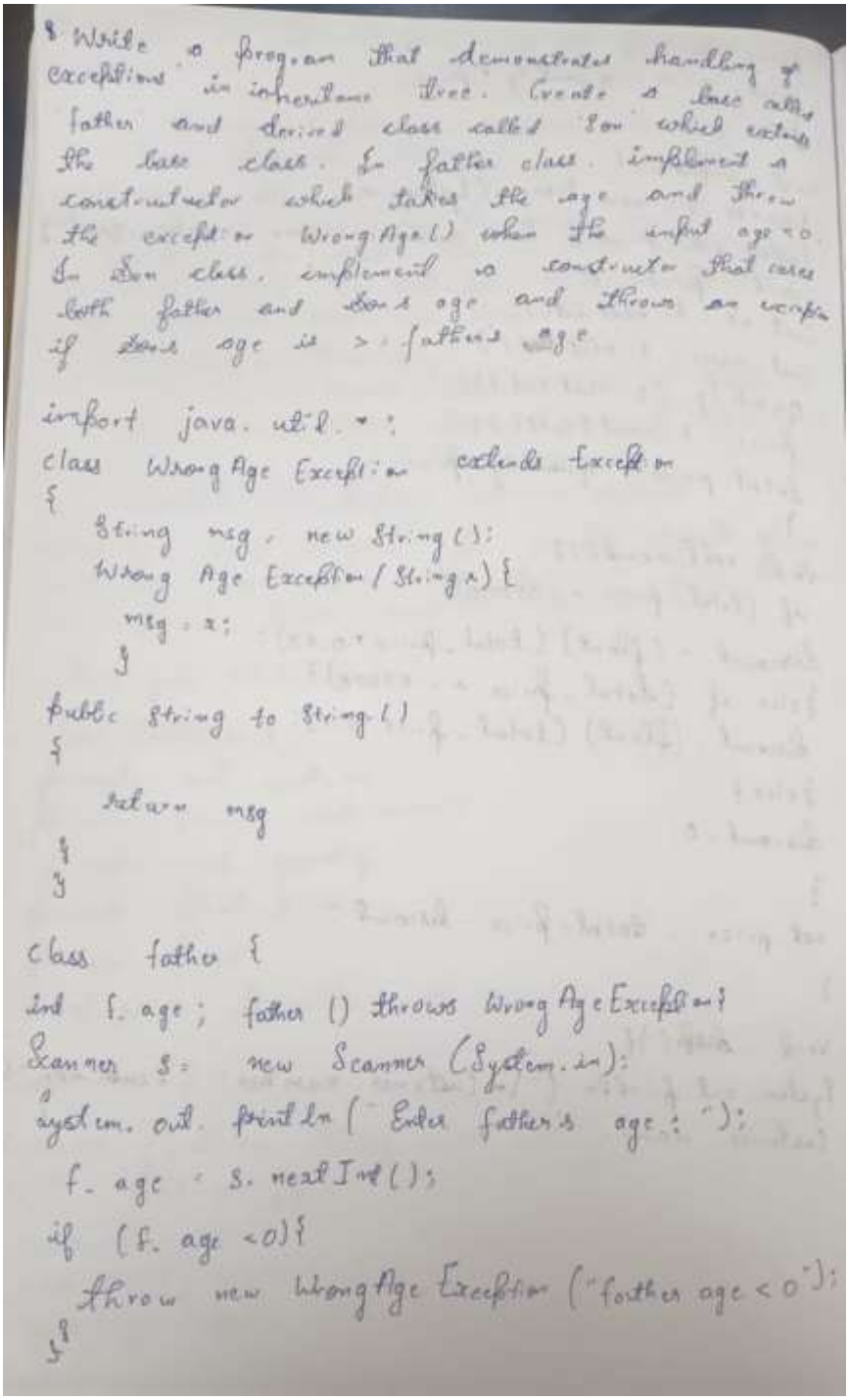
```
Void withdrawal(){
 float amt;
 System.out.println("Enter the amount to be withdrawn");
 amt = sc.nextfloat();
 if (amt > bal)
 System.out.println("Balance insufficient");
 else
  bal = bal - amt;
  System.out.println("Withdrawn : "+amt);
  System.out.println("Current balance : "+bal);
 }
Void interest(){
 System.out.println("Enter the rate of interest");
 int n = sc.nextInt();
System.out.println("Enter the time elapsed");
int t = sc.nextInt();
  si = bal*(1+(r/n));
 System.out.println("Compound interest is "+(Math.pow(si,
                                          n*t)));
 }
}
public class account{
 public static void main(String args[]){
 public Stat
  Scanner sc = new Scanner(System.in);
  Saving obj1 = new Savings();
  current obj2 = new current();
System.out.println("\n1 Savings account \n2.Current accant");
 int choice = sc.nextInt();
```

```
Switch (choice){
case 1:  obj1.accept();
         obj1.display();
         obj1.cheque();
         obj1.deposit();
         obj1.interest();
         obj1.withdrawal();
         break;

case 2:  obj2.accept();
         obj2.display();
         obj2.cheque();
         obj2.deposit();
         obj2.withdrawal();
         break;
default : System.out.println("Invalid choice");
    }
  }

}
```

Output:

```
1.Savings account
2.Current account
1
Enter your name
Dhavan SK
Enter the balance amount
10000
Name : Dhavan SK
Cheque service not available
Do you want to deposit(1 for yes ,2 for no)
1
Enter the amount to be deposited
5000
Current balance : 15000.0
Enter the rate of interest
5
Enter the number of times interest applied per time period
2
Enter the time elapsed
5
Compound interest is 3.4050628916015623E46
Enter the amount to be withdrawn
4000
Withdrawn : 4000.0
Current balance : 11000.0
```

7) Write a program that demonstrates handling of exceptions in inheritance tree. Create abase class called "Father" and derived class called "Son" which extends the base class. InFather 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 bothfather and son's age and throws an exception if son's age is >=father's age.



```
// 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 throw
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 excep
if Son's age is >= father's age

import java.util.*;
class WrongAge Exception extends Exception
{
   String msg = new String();
   WrongAge Exception(String x){
     msg = x;
    }
public String toString()
{
   return msg
}
}

class father {
int f.age; father() throws WrongAgeException{
Scanner s = new Scanner(System.in);
System.out.println("Enter father's age:");
 f.age = s.nextInt();
if (f.age<0){
  throw new WrongAge Exception("father age<0");
}
```

```
void display(){
  System.out.println("father age: "+f_age);
 }
}
class Son extends father {
int s_age;
 Son() throws Wrong Age Exception {
  Scanner s = new Scanner(System.in);
  System.out.println("Enter Son's age:");
   s_age = s.nextInt();
  if (s_age < 0) {
   throw new Wrong Age Exception("Son age < 0");
  }
  else if (s_age > f_age)
  {
    throw new Wrong Age Exception("Son age is > that father's age!");
  }
 }

Void display()
{
  System.out.println("father age: "+f_age);
  System.out.println("Son age: "+s_age);
 }
}
class excep {
public static void main(String[] args){
 try
 {
```

```
        father f = new father();
        f.display();
        Son s = new Son();
        s.display();
    }
    catch (WrongAge Exception wae)
    {
        System.out.println(wae);;
    }
}
```

Output

Enter father's age : 45

Enter son's age : 67

Son age > that father's age !

~~Enter father's age : 45~~

~~Enter son's age : 47~~

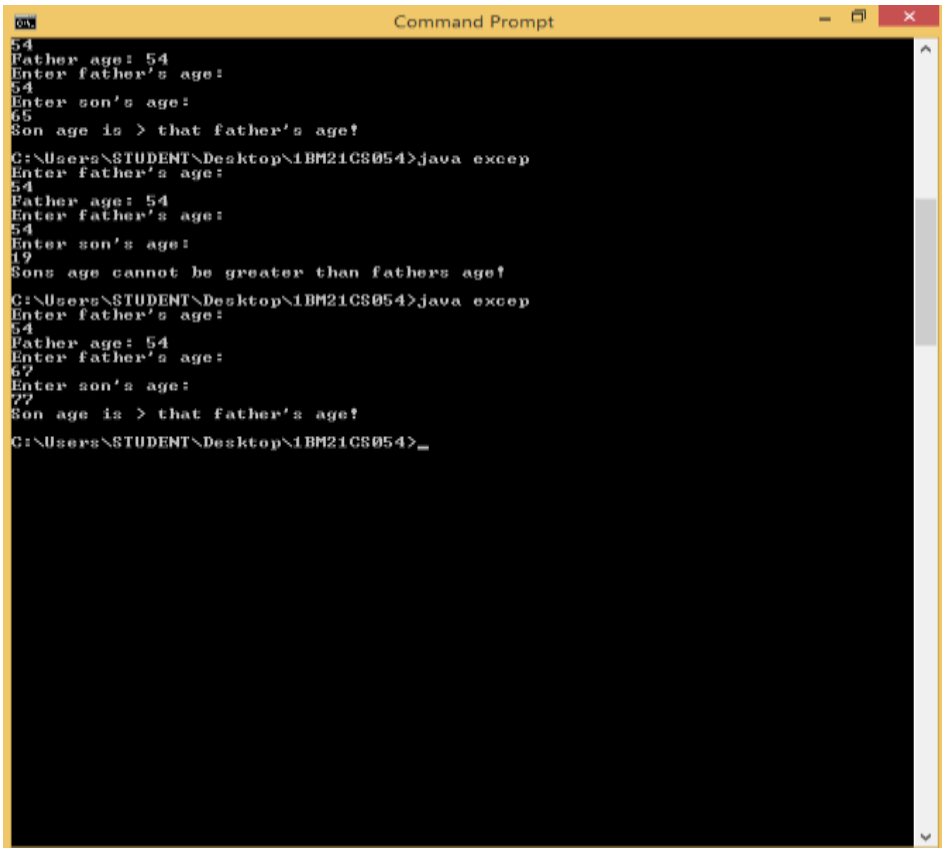
Enter father's age : 45

Enter son's age : 47

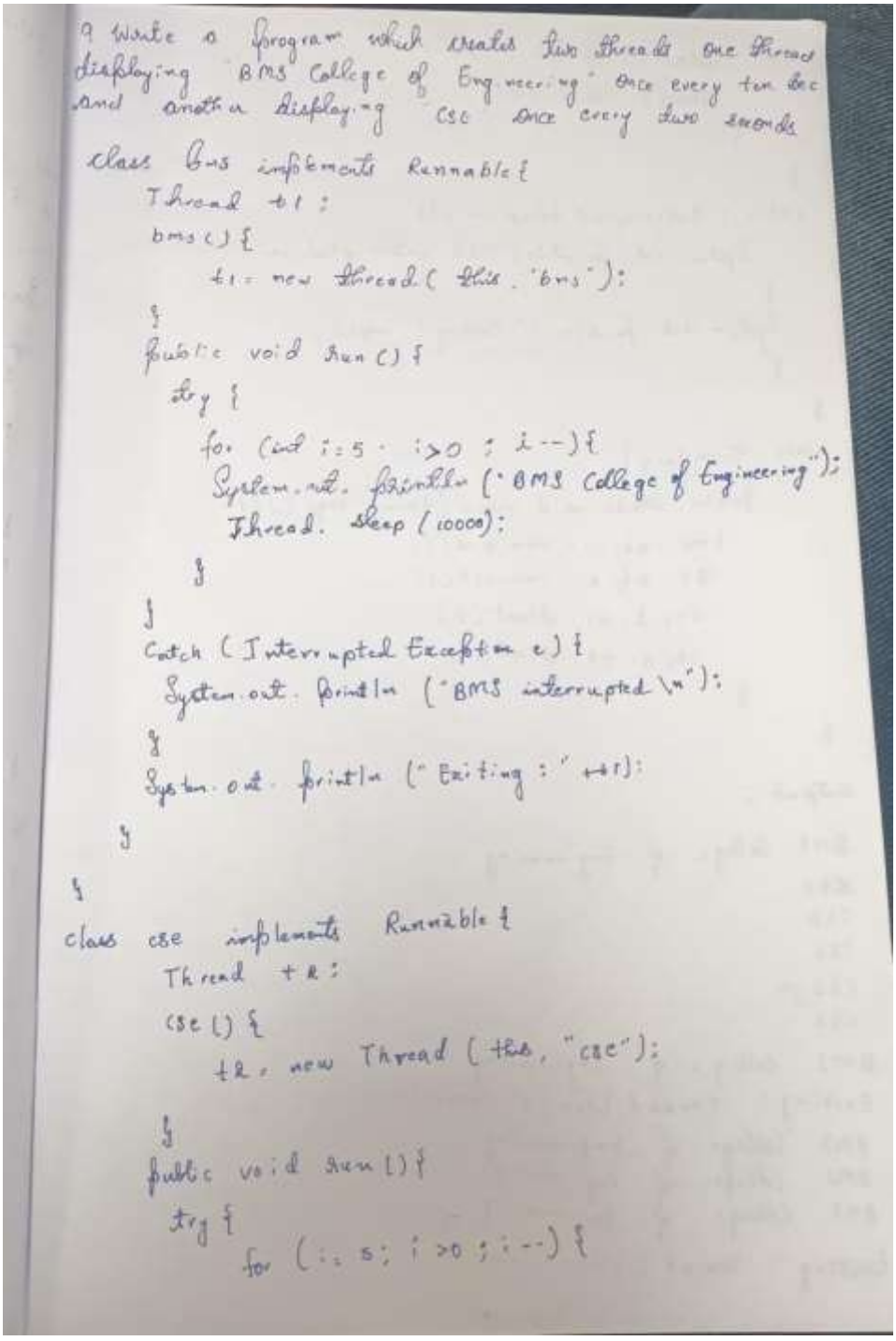
Sons age cannot be greater than fathers age!

30/12/2022

Output:


Command Prompt
54
Father age: 54
Enter father's age:
54
Enter son's age:
65
Son age is > that father's age!
C:\Users\STUDENT\Desktop\1BM21CS054>java excep
Enter father's age:
54
Father age: 54
Enter father's age:
54
Enter son's age:
19
Sons age cannot be greater than fathers age!
C:\Users\STUDENT\Desktop\1BM21CS054>java excep
Enter father's age:
54
Father age: 54
Enter father's age:
67
Enter son's age:
77
Son age is > that father's age!
C:\Users\STUDENT\Desktop\1BM21CS054>_

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.



```
8 Write a program which creates two threads one thread
displaying "BMS College of Engineering" once every ten sec
and another displaying "CSE" once every two seconds

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 (;; s; i>0; i--) {
```

```
            System.out.println("CSE");
            Thread.sleep(2000);
        }
    }
    catch (InterruptedException e){
        System.out.println("CSE interrupted\n");
    }
    System.out.println("Exiting: "+t2);
    }
}
class threadprg{
    public static void main(String args[]){
        bms obj1 = new bms();
        cse obj2 = new cse();
        obj1.t1.start();
        obj2.t2.start();
    }
}
```

output:

```
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
Exiting: Thread[cse,5,main]
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
Exiting: Thread[bms,5,main]
```

Output:


C:\Users\BMSCECSEIL74\Desktop\1BM21CS054>java threadprg
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
Exiting: Thread[cse,5,main]
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
Exiting: Thread[bms,5,main]
C:\Users\BMSCECSEIL74\Desktop\1BM21CS054>