B.M.S. COLLEGE OF ENGINEERING

Basavanagudi, Bengaluru- 560019

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LAB OBSERVATION

On

Object-Oriented Java Programming

(23CS3PCOOJ)

Submitted By:

Dhanush T

In partial fulfilment of

BACHELOR OF ENGINEERING

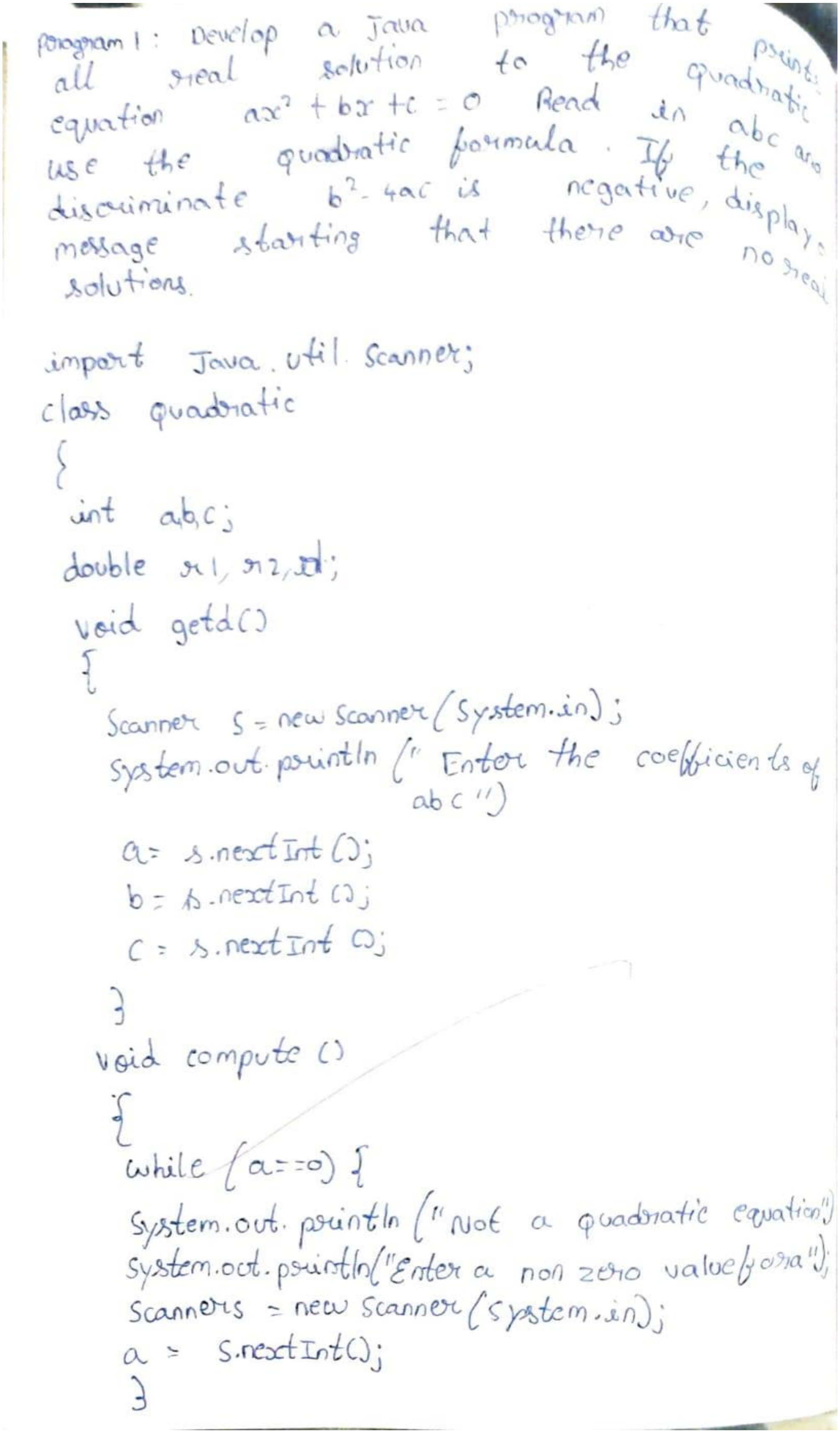
In

COMPUTER SCIENCE AND ENGINEERING

2023-24

Program 1

Develop a Java program that prints all real solutions to the quadratic equation ax2+bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b^24ac



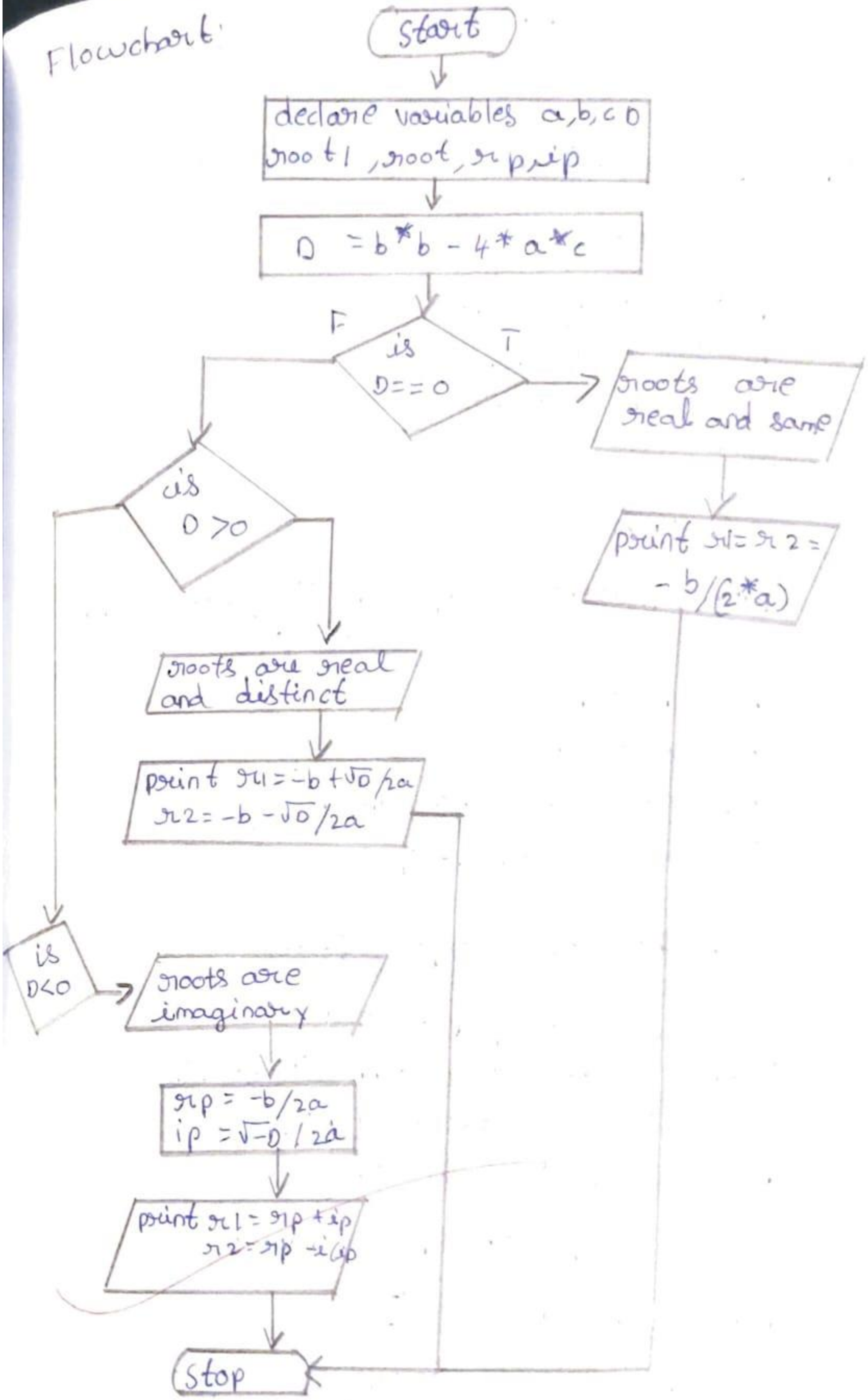
program 1: Develop a Java program that prints all real solution to the quadratic equation $ax^2 + bx + c = 0$ Read in abc and use the quadratic formula. If the discriminate $b^2 - 4ac$ is negative, display message starting that there are no real solutions.

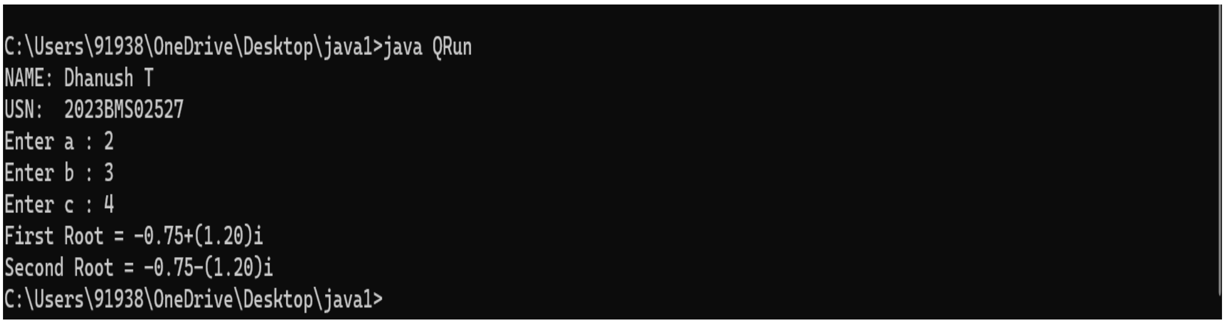
```
import Java.util.Scanner;
class quadratic
{
 int a,b,c;
 double r1, r2, d;
 void getd()
 {
  Scanner s = new Scanner(System.in);
  System.out.println(" Enter the coefficients of
                       abc ")
   a= s.nextInt();
   b= s.nextInt();
   c= s.nextInt();
  }
 void compute ()
  {
   while (a==0) {
   System.out.println ("Not a quadratic equation")
   System.out.println("Enter a non zero value for a");
   Scanners = new Scanner(System.in);
   a = s.nextInt();
   }
d = b*b-4 *a*c;
if (d==0)
{
 r1=(-b)/(2*a);
system.out.println ("Roots are real and equal");
system.out.println("Root1= Root2 = "+r1);

}
else if (d>0)
{ r1= ((-b)+ (math.sqrt(d)))/(double) (2*a);
 r2= ((-b) - math.sqrt(d)))/ (double) (2*a);
 system.out.println ("Roots are real and distinct");
 system.out.println("Root ="+r1+ root2= "+r2);

}
else if (d<0)
{
 system.out.println("Roots are imaginary");
 r1= (-b)/ 2*a);
 r2= math.sqrt(-d)/ 2*a);
 system.out.println("Root1= "+r1+"i"+r2);
 system.out.println ("Root1= "+r1+"-i"+r2);
 }
}
}
class Java {
     public static void main (string [] args) {
          quadratic q=new quadratic ();
          g.getd();
     }
}
```

→ Algorithm

```
Step 1: start
Step 2: Declare variables a, b, c D, root 1, c, root2
Step 3: calculate discriminant (D = b * b - 4*a*c)
Step 4: if D==0 roots are real and same
            r1 = r2 = -b/2a
step 5: if D>1 root are real and distinct
            r1 = $(-b + \sqrt{D})/2a$
            r2 = $(-b - \sqrt{D}/2a$
step 6: if D<1 roots are imaginary
            rp = -b/2a
            ip = $\sqrt{-D}/2a$
            root1 = rp+ ip
            root 2 = rp-iip
step7: End
```

Flowchart:

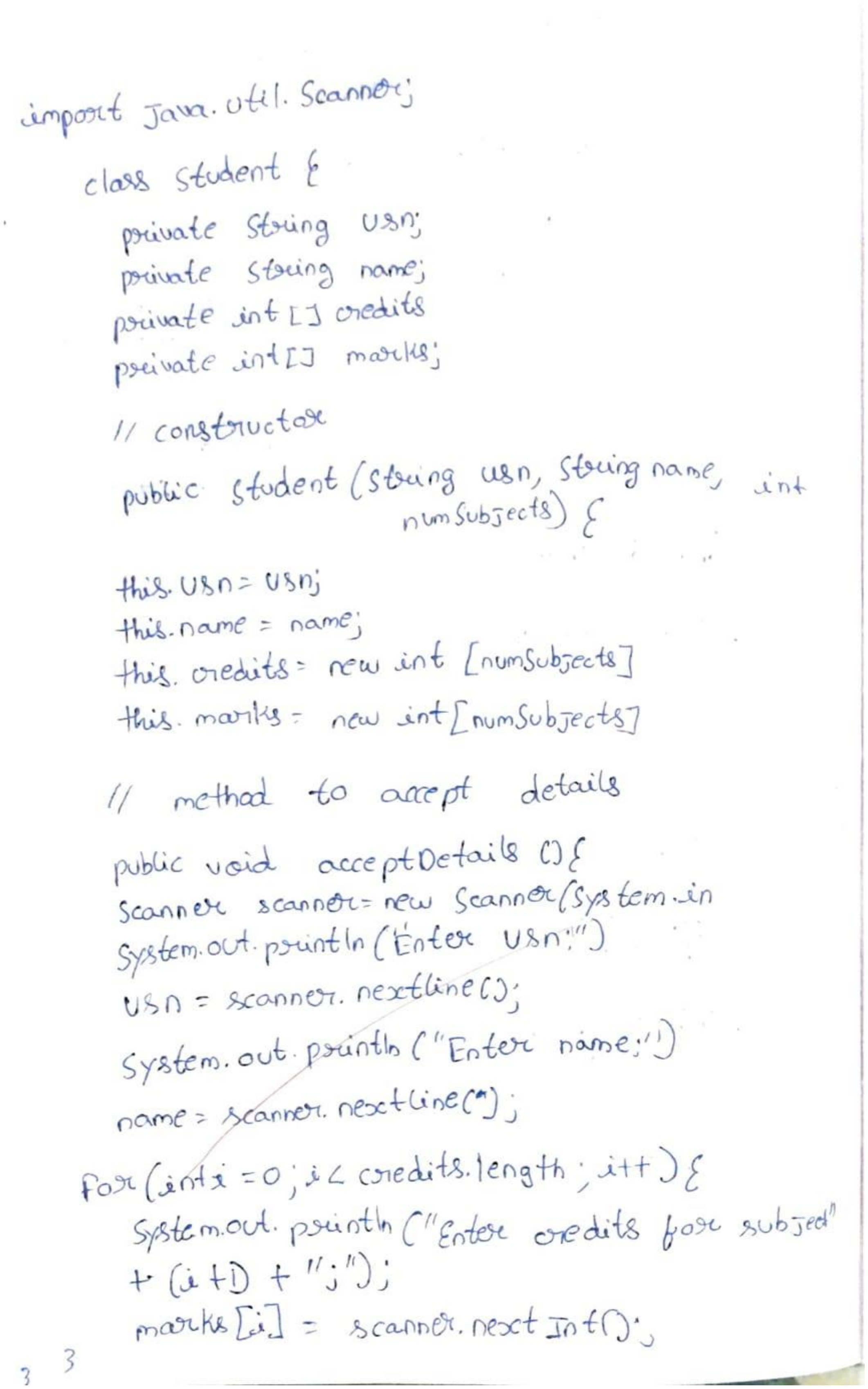

Start
declare variables a,b,c D
root1 ,root, rp,ip
D =b*b - 4*a*c
is
D==0
F
T
roots are
real and same
print r1=r2=
-b/(2*a)
is
D>0
roots are real
and distinct
print r1=-b+√D/2a
r2=-b-√D/2a
is
D<0
roots are
imaginary
rp = -b/2a
ip =√-D / 2a
print r1=rp+ip
r2=rp-i(ip
Stop



```
C:\Users\91938\OneDrive\Desktop\java1>java QRun
NAME: Dhanush T
USN:  2023BMS02527
Enter a : 2
Enter b : 3
Enter c : 4
First Root = -0.75+(1.20)i
Second Root = -0.75-(1.20)i
C:\Users\91938\OneDrive\Desktop\java1>
```

Program 2

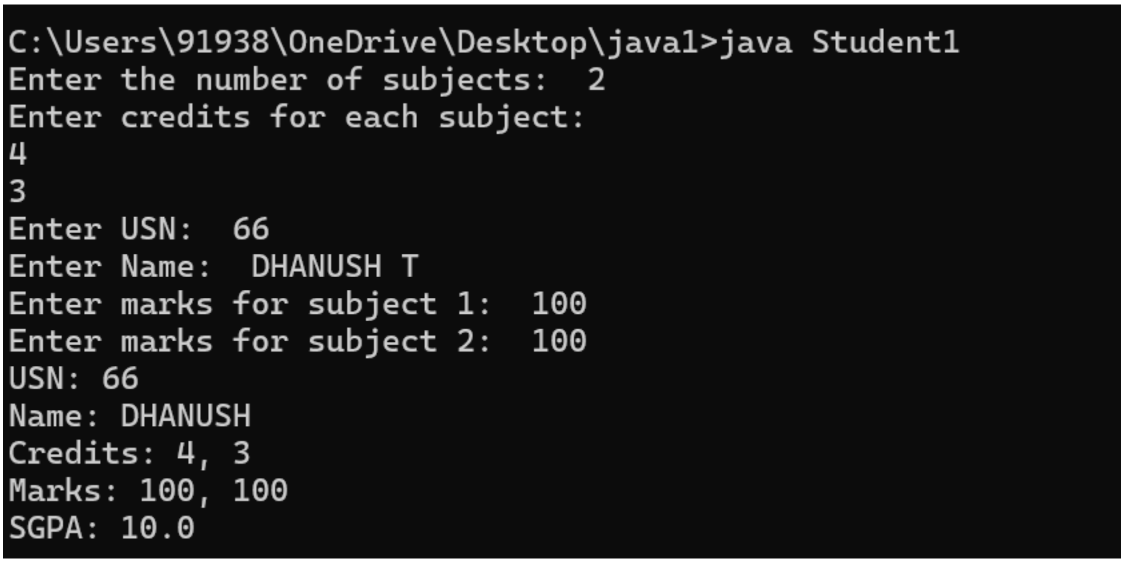
Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.



```
import Java.util.Scanner;
   class student {
     private String usn;
     private string name;
     private int [] credits
     private int[] marks;
     // constructor
     public student (String usn, String name, int
                         numSubjects) {
     this.usn = usn;
     this.name = name;
     this.credits = new int [numSubjects]
     this.marks = new int[numSubjects]
     // method to accept details
     public void acceptDetails (){
     Scanner scanner= new Scanner(System.in
     System.out.println ("Enter usn:")
     usn = scanner.nextline();
     System.out.println ("Enter name:")
     name = scanner.nextLine();
  For (int i =0; i< credits.length; i++){
     System.out.println ("Enter credits for subject"
     + (i+1) + ":");
     marks[i] = scanner.nextInt();
```

```
//                     of the studen

public void displayDetails () {
    System.out.println("USN: " + usn);
    System.out.println(" Name:"+ name);
    System.out.println(" subject -wise Details");
  for(int i=0; i< credits.length; i++) {
     System.out.println ("subject " + (i+1) +"
      - credits: " + credits[i] + ", Marks:" +
       marks[i]

public double SGPA () {
    double total credits = 0
    double total gradepoints = 0;

  for (int i=0; i< credits.length; i++) {
   total credits += credits [i];
   total Gradepoints+= calculateGradepoints
      (marks[i] * credits[i];

      }
      return total gradepoints/ total credits

  }

private double   calculateGradepoints (int
                                      marks)
    if (Marks>=90) {
         return 10.0;
   }else if (marks >= 80) {
         return 9.0;
   } else if (marks >=70) {
         return 8.0
   } elseif (marks >=60) {
         return 7.0
```

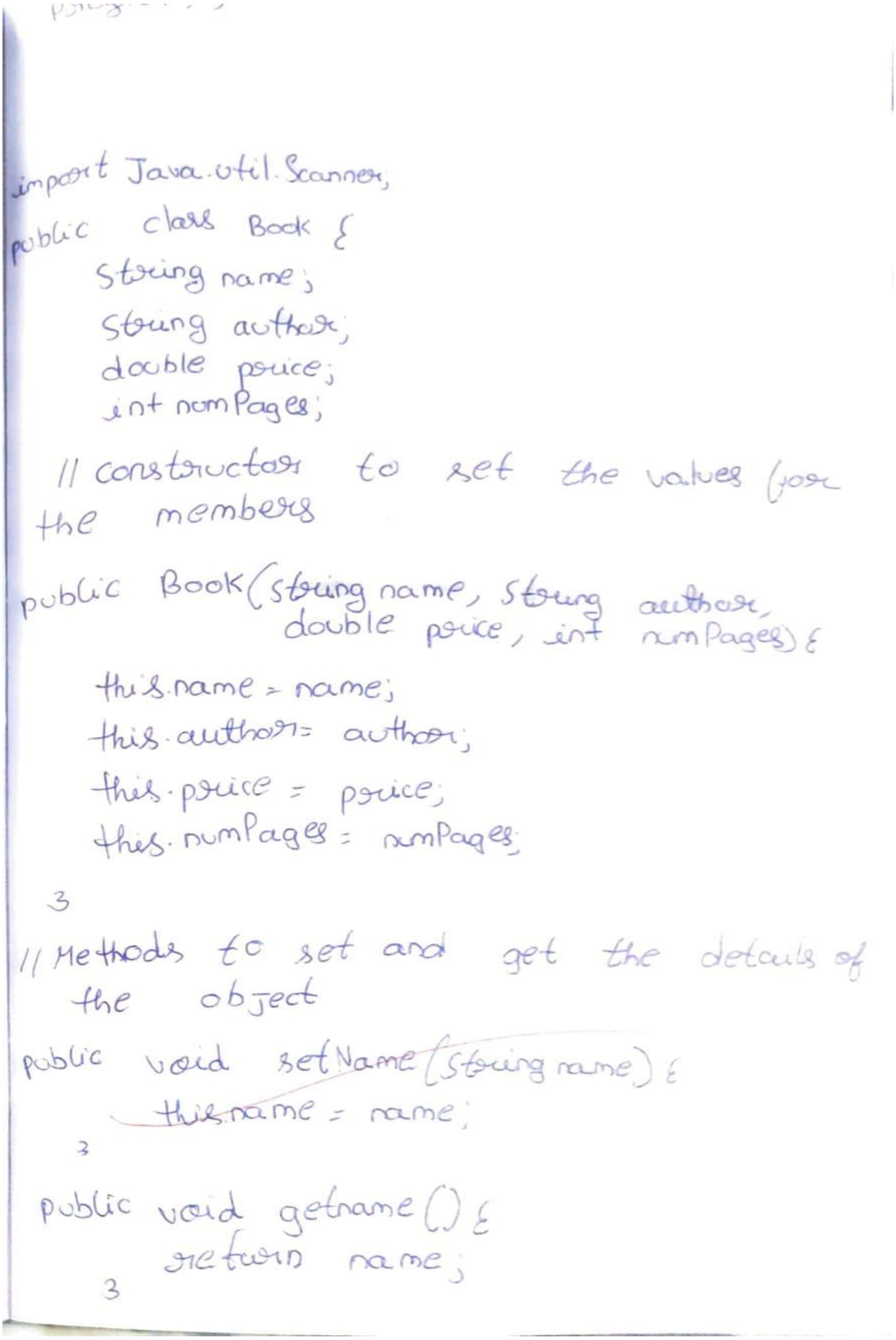
```
        }else if(marks
            return 6.0;

        }else{
            return 0.0;//Fail

        }
    }
}
public class StudentDemo{
    public static void main(String [] args){
        Scanner scanner = new Scanner(System.in);
    System.out.println("Enter the number of
                        subjects:");
        int numSubjects= scanner.nextInt();

        Studen student = new Student("","",
                        numSubjects);

        student.acceptDetails();

    Systemout.println("\nStudent Details:")
        student.displayDetails();

    System.out.println.
```



```
C:\Users\91938\OneDrive\Desktop\java1>java Student1
Enter the number of subjects:  2
Enter credits for each subject:
4
3
Enter USN:  66
Enter Name:  DHANUSH T
Enter marks for subject 1:  100
Enter marks for subject 2:  100
USN: 66
Name: DHANUSH
Credits: 4, 3
Marks: 100, 100
SGPA: 10.0
```

Program 3

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



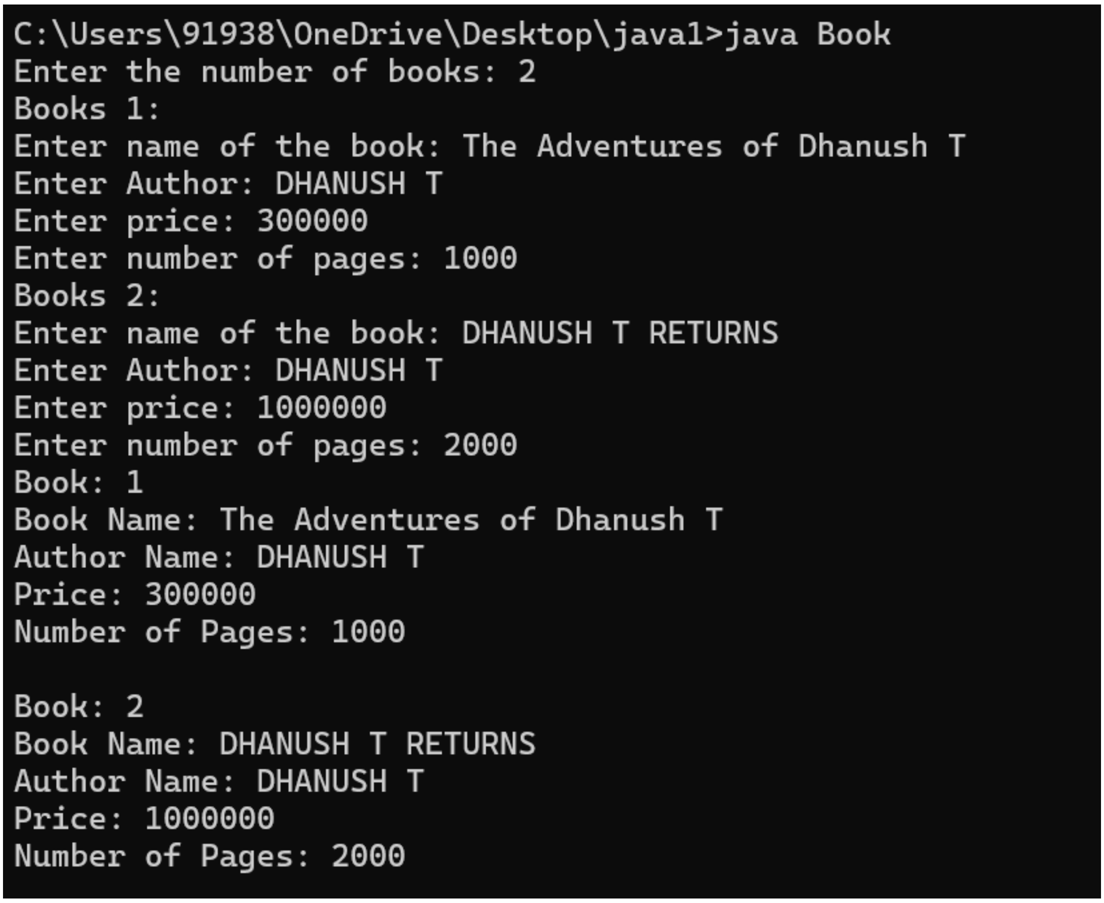
```
import Java.util.Scanner;
public class Book {
    string name;
    string author;
    double price;
    int numPages;

  // constructor to set the values for
 the members

public Book(string name, string author,
            double price, int numPages) {

    this.name = name;
    this.author = author;
    this.price = price;
    this.numPages = numPages;

 }
// Methods to set and get the details of
   the object
public void setName(string name) {
     this.name = name;
  }

 public void getname() {
      return name;
  }
```

```
public void setAuthor(String author){
        this.author = author;

   }
public string getAuthor(){
       returnauthor;

   }
public void setprice(double price){
       this.price = price;

     }
public void getPrice(){
      return price;

public void setnumpage(int numpages){
      this.numpage=numpages;
      }
public void getnumpages( ){
      return numpages;

     }
```

12/11/2

```
public string toString()
    return "Book Details; \n Name:"+
name + "\n Author:" + author + "\n Price
: " + Price + "\nNumber of pages:"+
          numpages.
```

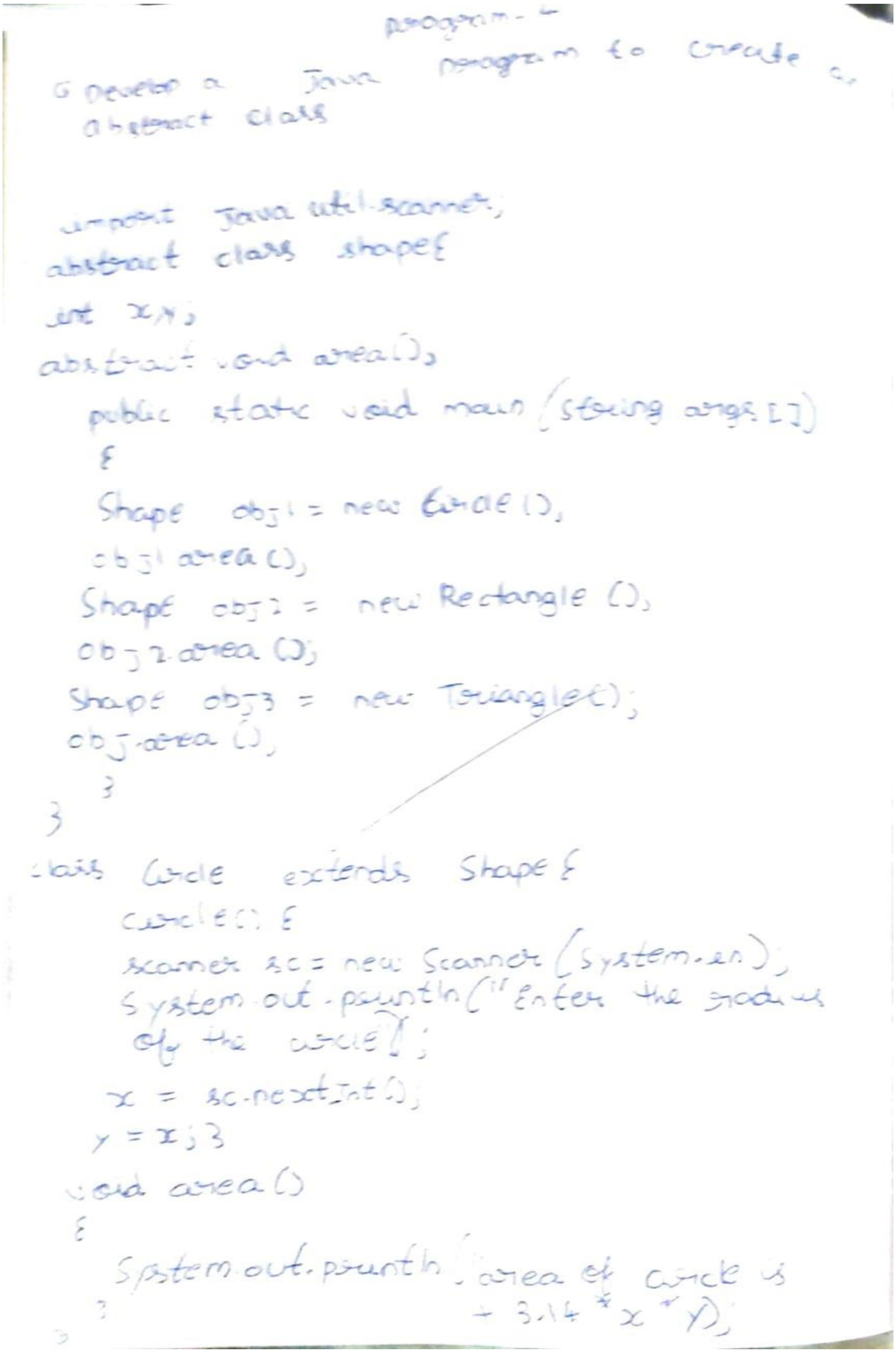
```
public static void main(string []args){
        Scanner sc = new Scanner(system.in),
  system.out.println(" Enter the number of
                          books");
 int n: scanner.nextInt();

  Book [] books = new Book [n];

 for (int i=0; i<n; i++){
   system.out.println("Enter details of the
                          book # ' +(i+1));
    System.out.print(" Name:");
    string name = scanner.nextLine(),

    System.out.print("Author"),
    String name = scanner.nextLine():
   system.out.println("Enter the number of
                          pages:");
   int numpages = sc.nextInt();
    books [i] = new Book (name, author, price
                          numpages);
   }
for(Book book: books){
       System.out.println(book.tostring()),
       System.out.println();
    }
 }
}
```



```
C:\Users\91938\OneDrive\Desktop\java1>java Book
Enter the number of books: 2
Books 1:
Enter name of the book: The Adventures of Dhanush T
Enter Author: DHANUSH T
Enter price: 300000
Enter number of pages: 1000
Books 2:
Enter name of the book: DHANUSH T RETURNS
Enter Author: DHANUSH T
Enter price: 1000000
Enter number of pages: 2000
Book: 1
Book Name: The Adventures of Dhanush T
Author Name: DHANUSH T
Price: 300000
Number of Pages: 1000

Book: 2
Book Name: DHANUSH T RETURNS
Author Name: DHANUSH T
Price: 1000000
Number of Pages: 2000
```

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.

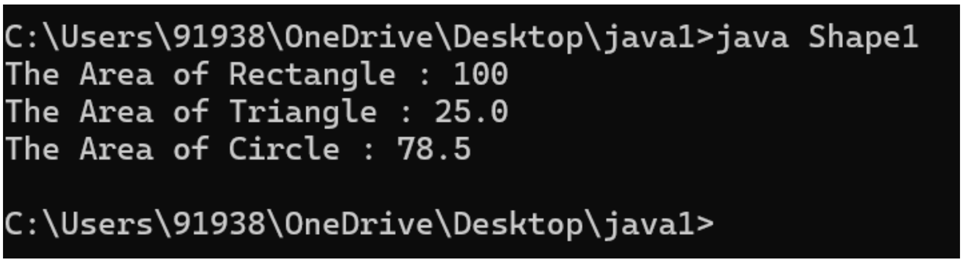


program-4

6 Develop a Java program to create a abstract class

```
import Java util.scanner;
abstract class shape{
int x,y;
abstract void area();
   public static void main(string args[])
   {
   Shape obj1 = new Circle(),
   obj1 area(),
  Shape obj2 = new Rectangle (),
  obj2.area ();
 Shape obj3 = new Triangle();
 obj.area (),
   }
}
class Circle extends Shape{
    circle() {
    scanner sc = new Scanner (system.in);
    System.out.println("Enter the radius
     of the circle");
   x = sc.nextInt();
  y = x; }
 void area()
 {
  System.out.println("area of circle is
                       + 3.14 * x * y);
  }
}
```

```
class Rectangle extends Shape {
    Rectangle (){
    Scanner sc = new Scanner (System.in);
    System.out.println("Enter the length
                       and breadth of the
                       rectangle");

    x = sc.nextInt();
    y = sc.nextInt();
    }
  void area()
    System.out.println("area of ~~circle~~ rectangle is -
                        + x * y); ~~[illegible]~~

  }
 }
class Triangle extends shape {
  Triangle() {
  Scanner sc = new Scanner(System.in);
  System.out.println("Enter the base an d
                 height of the Triangle");
  x = sc.nextInt();
  y = sc.nextInt(); }
  void area()
   {
     System.out.println("area of
                    Triangle is [illegible]
  }
 } o/p Area of Rectangle : 00
       Area of Triangle : [illegible]
       Area of circle : [illegible]
```



```
C:\Users\91938\OneDrive\Desktop\java1>java Shape1
The Area of Rectangle : 100
The Area of Triangle : 25.0
The Area of Circle : 78.5

C:\Users\91938\OneDrive\Desktop\java1>
```

Program 5

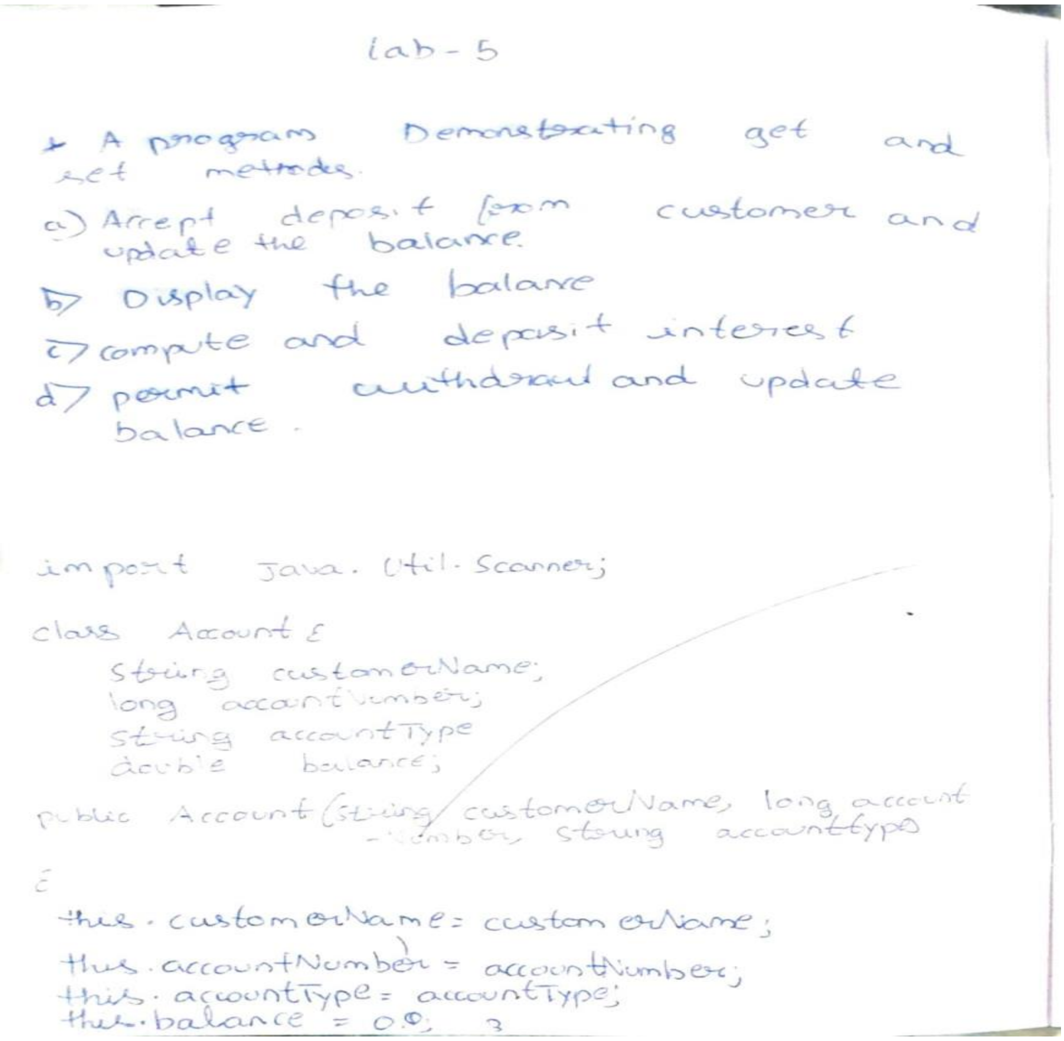
Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book 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.



lab-5

* A program Demonstrating get and set methodes.

a) Accept deposit from customer and update the balance.

b> Display the balance

c> compute and deposit interest

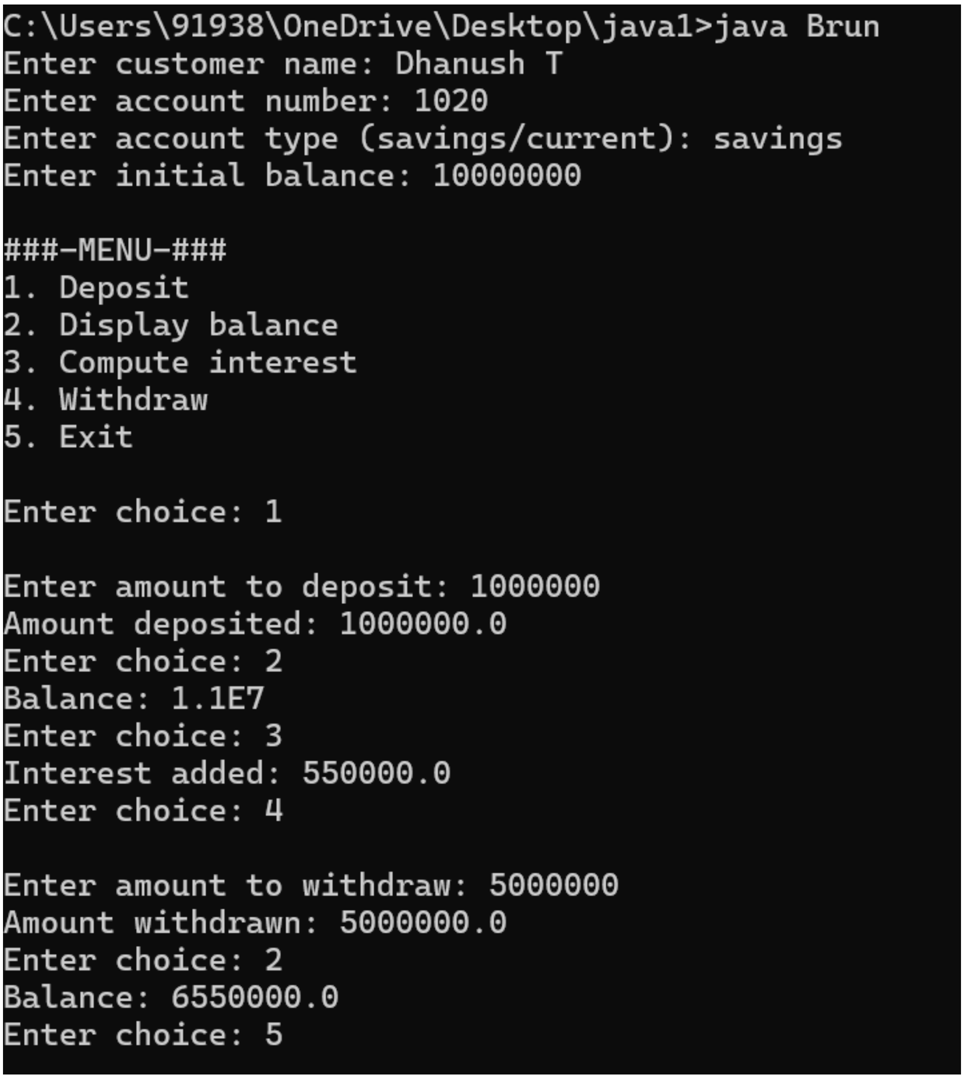
d> permit withdrawl and update balance.

```
import Java. Util. Scanner;

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

public Account(String customerName, long account
                    -Number, String accountType)

{
 this.customerName= customerName;
 this.accountNumber = accountNumber;
 this.accountType= accountType;
 this.balance = 0.0;   3
```

```
public void deposit(double amount) {
    balance += amount;
    System.out.println("Deposit successful.
                updated balance:" + balance);
    }
public void display balance() {
    System.out.println("Accont Balance: "+ balance);
  }
}
class CurrAcct extends Account {
    double minBalance = 1000.0; //
    double serviceCharge = 50.0;
public CurrAcct(String customerName, long account
                -number) {
    this.customerName= customerName;
    this.accountNumber = accountNumber;
    this.accontType = "Current";
    this.balance = 0.0;
    }
  @override
  public void deposit(double amount) {
    balance += amount;
    System.out.println("Deposit successful
     updated balance: " + balance);
    CheckMinBalance();
  }
```

```
public void checkMinBalance(){
    if (balance < minBalance) {
        balance -= serviceCharge;
        System.out.println("Service charge
    imposed. updated balance:" + balance);

    }
  }
}

class SavAcct extends Account{
     double interestRate 0.05;

 public SavAcct(String customerName, long
                    accountNumber){

  this.customerName= customerName,
  this accountNumber = accountNumber
  this.accountType = "Savings";
  this balance = 0.0;
  }

public void computeAndDepositInterest(){
      double interest = balance * interestRate;
       balance += interest;
       System.out.println("Interest computed and
       deposited updated balance:" + balance);

  }
  @override
public void deposit(double amount){
        balance += amount;
        System.out.println("withdrawal
          successful
system out.println(" insufficient funds for
withdrawal:");
  }
 }
}
public class Bank{
   public static void main(string []args){
      currAcct currentAccount = new CurrAcct("
              "Dhanush T", 123456789);
     savAcct("John wick", 987665321),

     currentAccount.deposit(1500.0);
     currentAccount displayBalance();


     SavingsAccount deposit(1500.0);
     SavingsAccount display Balance();


     Savings Account withdraw(500.0);
     Saving Acccont.checkMinBalance (),

    }
   }
```

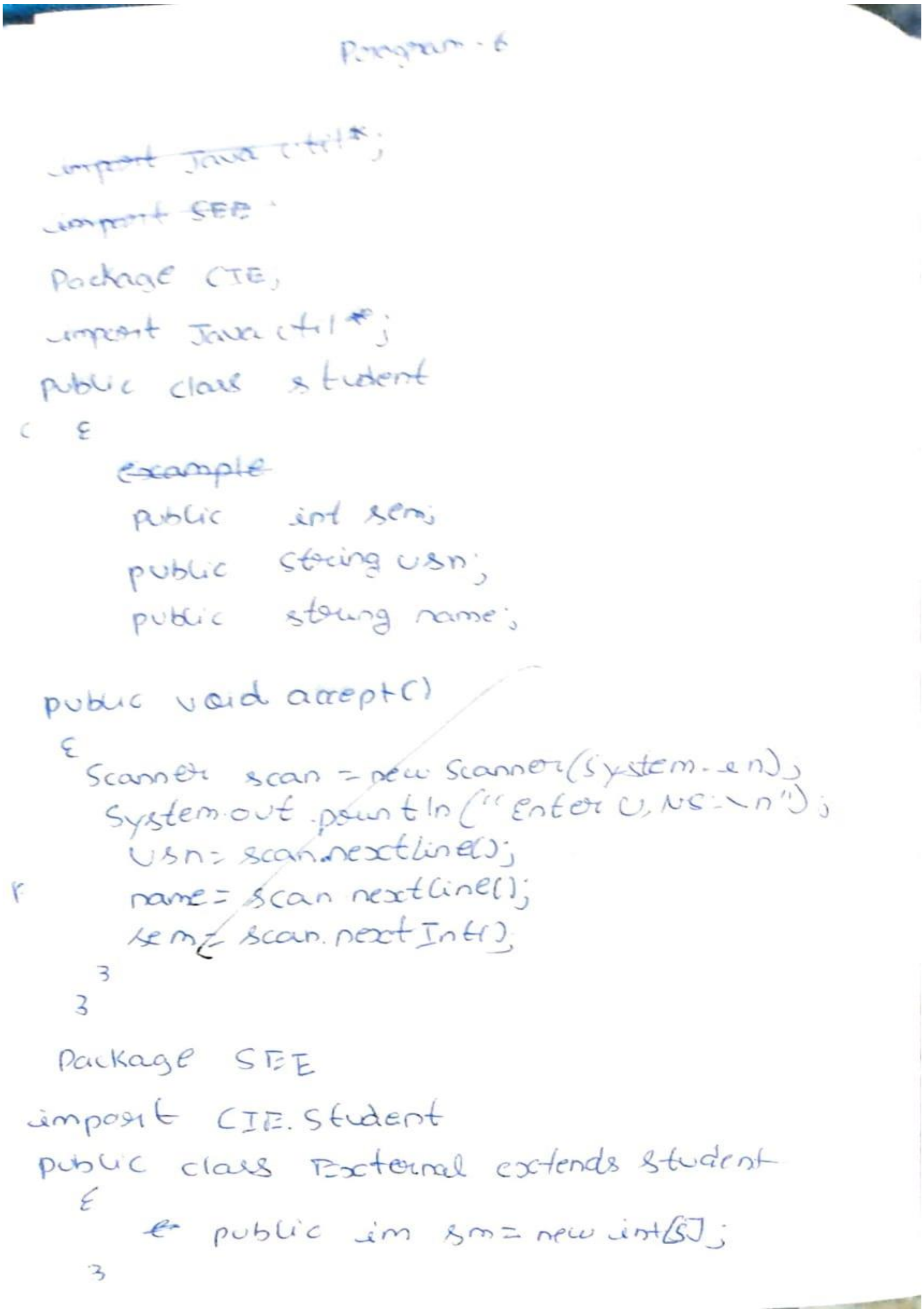
19/01/20



```
C:\Users\91938\OneDrive\Desktop\java1>java Brun
Enter customer name: Dhanush T
Enter account number: 1020
Enter account type (savings/current): savings
Enter initial balance: 10000000

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

Enter choice: 1

Enter amount to deposit: 1000000
Amount deposited: 1000000.0
Enter choice: 2
Balance: 1.1E7
Enter choice: 3
Interest added: 550000.0
Enter choice: 4

Enter amount to withdraw: 5000000
Amount withdrawn: 5000000.0
Enter choice: 2
Balance: 6550000.0
Enter choice: 5
```

Program 6

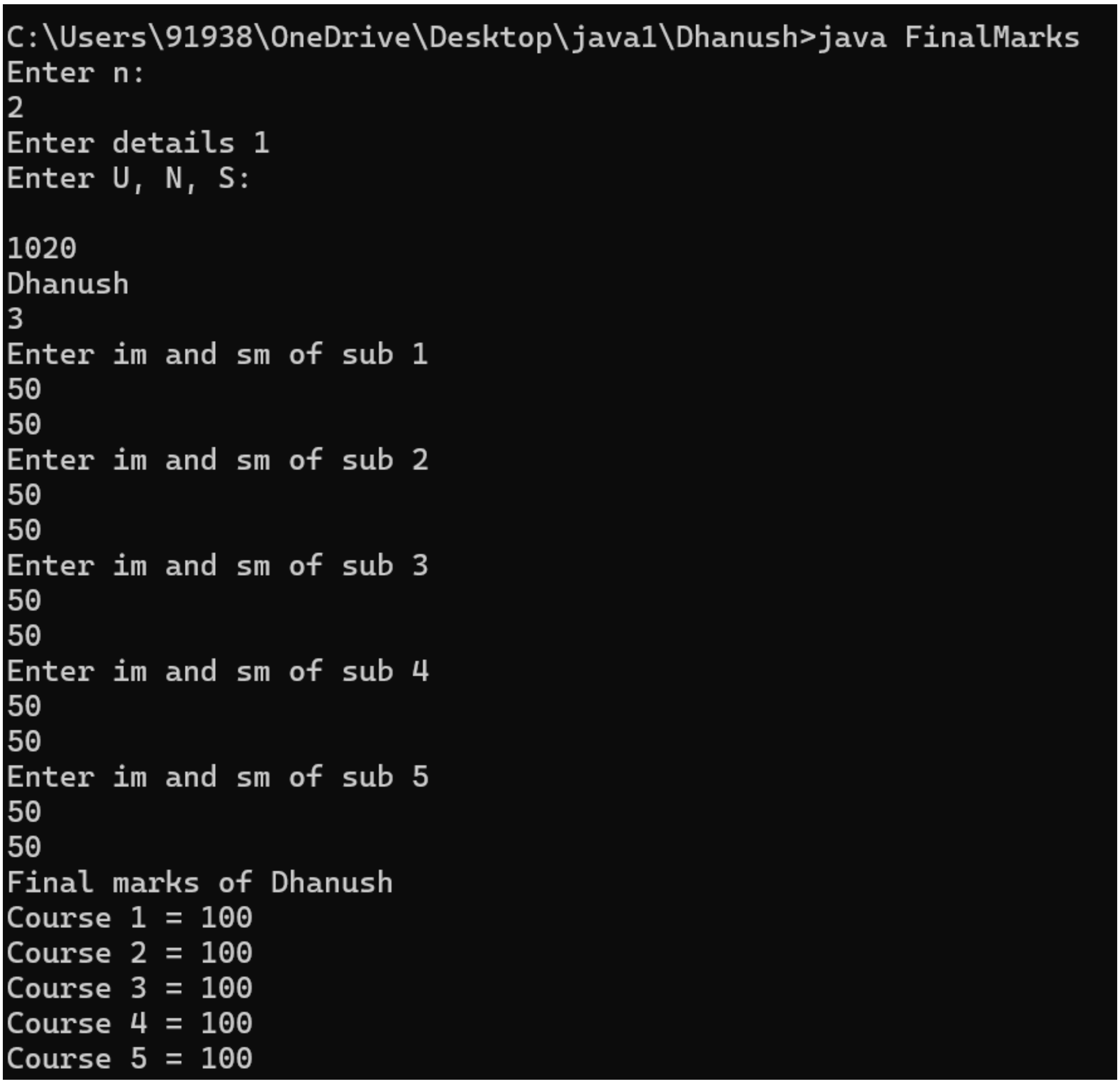
Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External 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.



Program - 6

```
~~import Java util*;~~
~~import SEE~~
Package CIE,
import Java util*;
public class student
{
    ~~example~~
    public    int sem;
    public    string usn;
    public    string name;

public void accept()
 {
  Scanner scan = new Scanner(System.in);
   System.out.println("Enter U,NS:\n");
    usn= scan.nextline();
    name= scan nextline();
    sem= scan.nextInt();
  }
 }

 Package SEE
import CIE.Student
public class External extends Student
 {
    ~~e~~ public int sm= new int[5];
 }
```

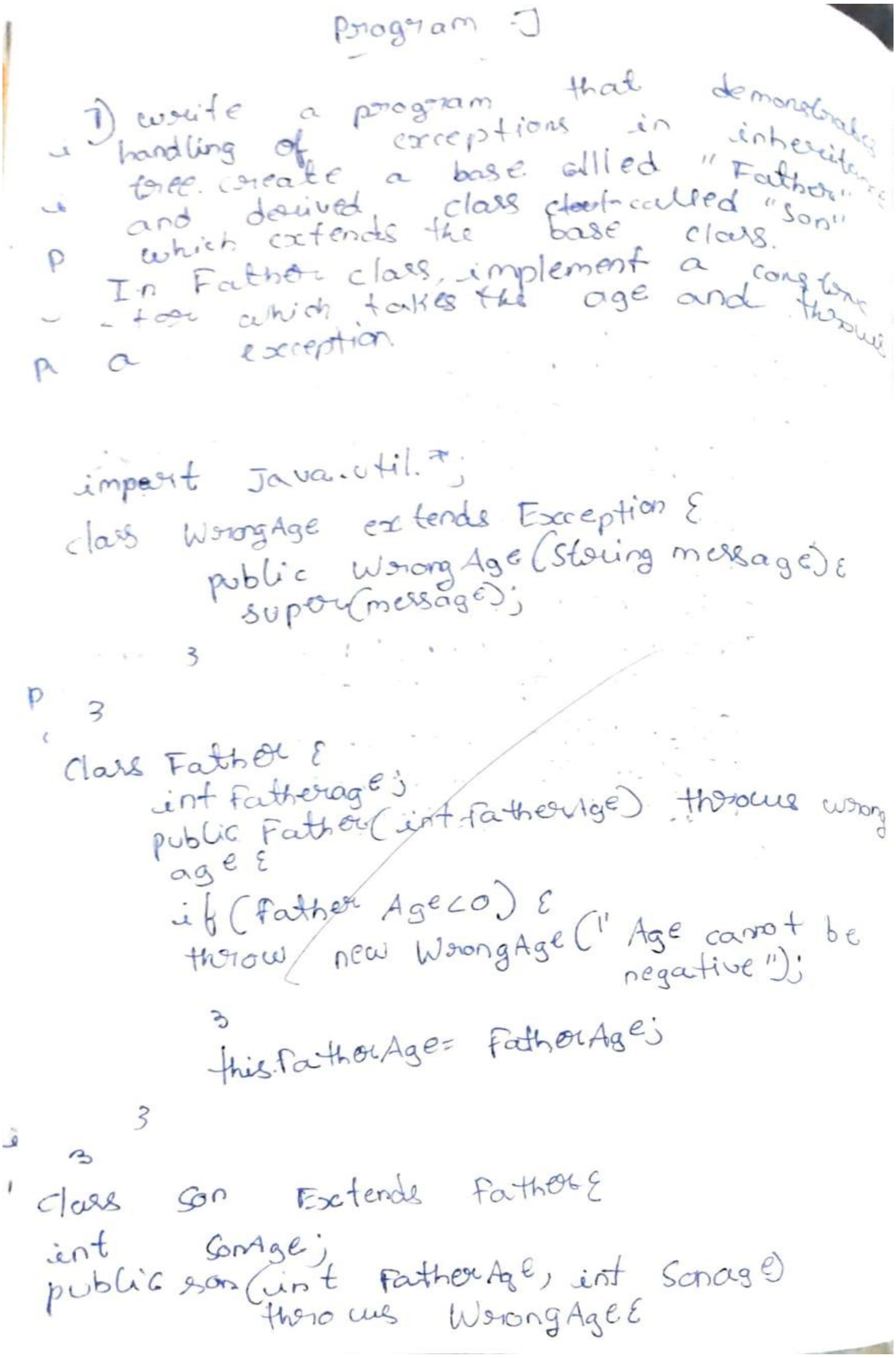
```
import Java.util*;
import SEE*;
import CIE*;
public class Final Marks
{
   public void static voidmain(String args[])
   int fm[] = new int[5];
   Scanner sc = new Scanner(System.in);
   system.out.println("Enter n ");
   int n = sc.nextInt();
   see.External st[] = new SEE.External[n]
   cie.internal ct[] = new CIE.Internal[n]
   For(int i=0;i<n;i++)
   { st[i]=new SEE.External();
     s[i] = new.CIE.Internal();
     System.out.println(" Enter im and
            sm of subject");
      s[i].im[i] = sc.nextInt();
      st[i]sm[i]= sc.nextInt();
      fm[i] = s[i].im[i]+st[i].sm[i];
    }
  }
 }
}
```



```
C:\Users\91938\OneDrive\Desktop\java1\Dhanush>java FinalMarks
Enter n:
2
Enter details 1
Enter U, N, S:

1020
Dhanush
3
Enter im and sm of sub 1
50
50
Enter im and sm of sub 2
50
50
Enter im and sm of sub 3
50
50
Enter im and sm of sub 4
50
50
Enter im and sm of sub 5
50
50
Final marks of Dhanush
Course 1 = 100
Course 2 = 100
Course 3 = 100
Course 4 = 100
Course 5 = 100
```

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=father's age.



Program 7

1) write a program that demonstrates handling of exceptions in inheritance tree. create a base called "Father" and derived class ~~clast~~ called "Son" which extends the base class. In Father class, implement a constructor for which takes the age and throws a exception.

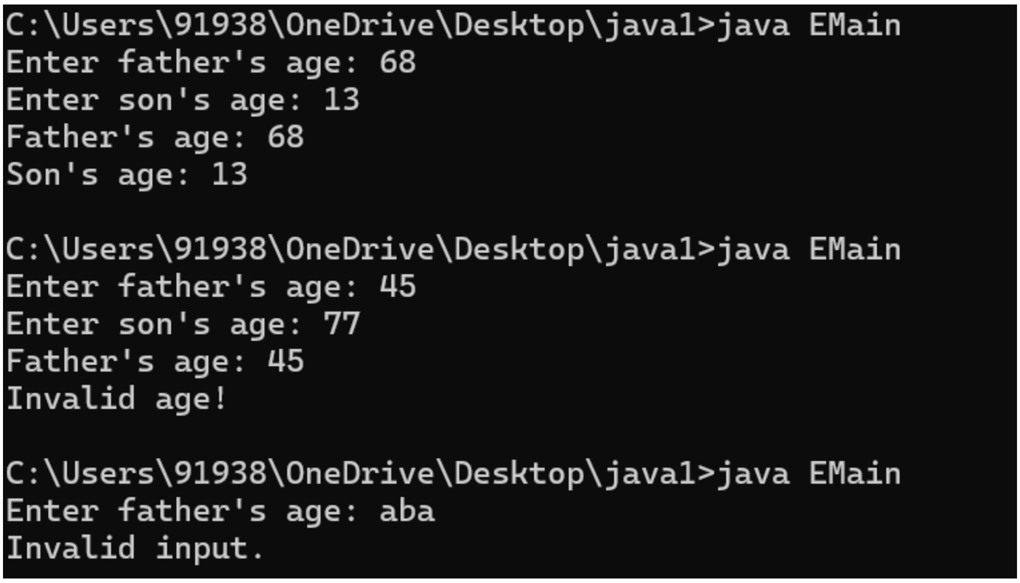
```
import Java.util.*;
class WrongAge extends Exception {
    public WrongAge(String message){
     super(message);
    }
}
Class Father {
    int fatherage;
    public Father(int fatherAge) throws wrong
    age {
     if (father Age<0) {
     throw new WrongAge(" Age cannot be
                          negative");
     }
     this.FatherAge= FatherAge;
    }
}
class Son Extends Father{
int SonAge;
public son(int FatherAge, int SonAge)
     throws WrongAge{
```

```
super (FatherAge);
  if (SonAge >= FatherAge)
  throw new WrongAge (" Sons age must be
  less   than  Father's  Age");

  }
   this. SonAge = sonAge;


  }
 }
public   class FatherSon{
 public  static    void main (String []args) {
     Scanner sc= new Scanner(system.in);
     System.out.println(" Enter Father's Age
          an  son's Age");
     int Fa = sc.nextInt();
     int sa = sc.nextInt();

   try {
       Son s = new son(Fa, sa);
       System.out.println(" Father's age "+
        s. Father Age);
       System.out. println("Son's Age "+ s.sonAge);
       System.out. println (" Error");
     }

   }

}
```

8 programs

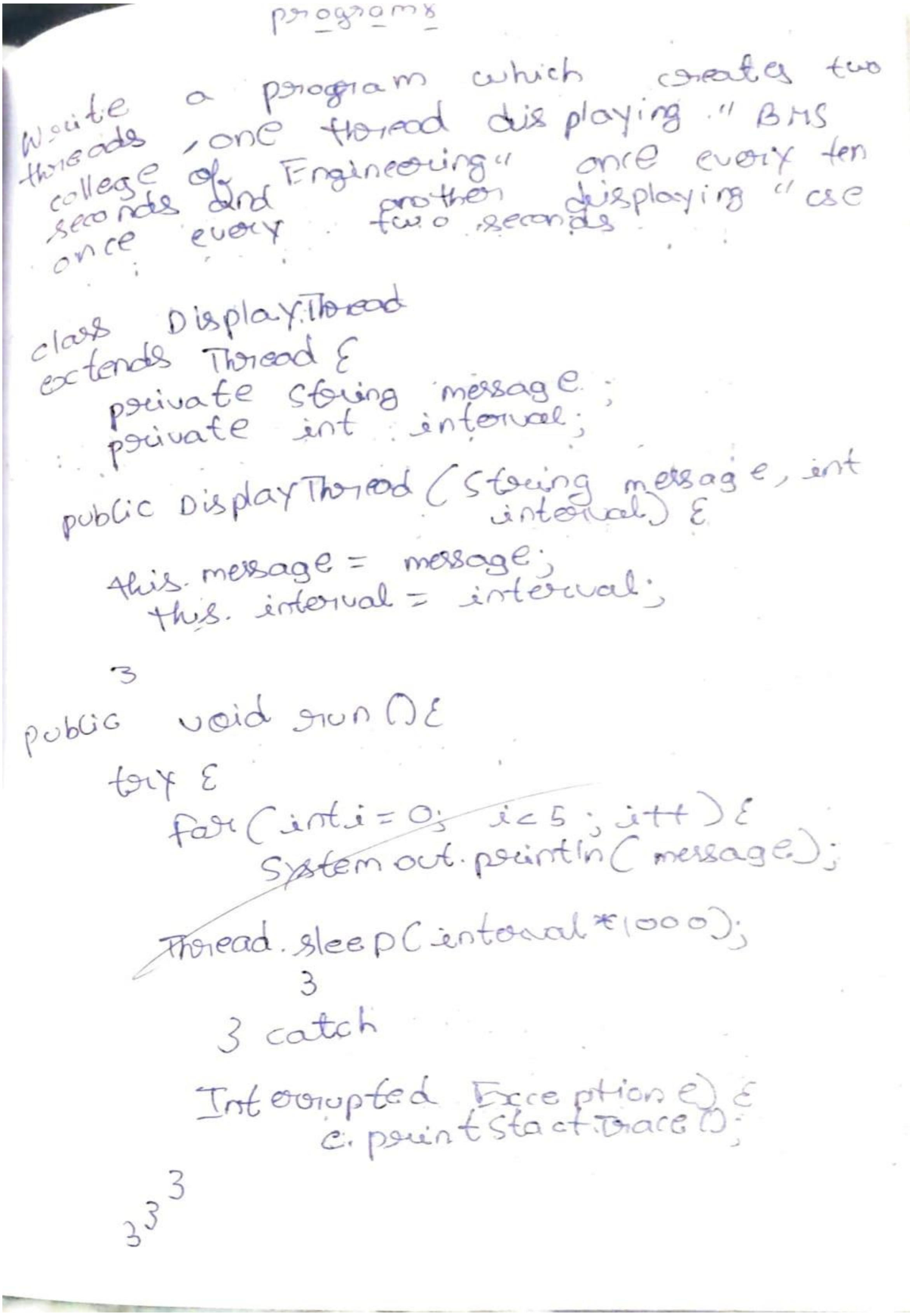
8. Output:

```
Enter Fathers age; 68
Enter Son's age : 13
Father's age : 68
Son's age: 13

Enter Father's age 45
Enter Son's age: 77
Invalid age
```



```
C:\Users\91938\OneDrive\Desktop\java1>java EMain
Enter father's age: 68
Enter son's age: 13
Father's age: 68
Son's age: 13

C:\Users\91938\OneDrive\Desktop\java1>java EMain
Enter father's age: 45
Enter son's age: 77
Father's age: 45
Invalid age!

C:\Users\91938\OneDrive\Desktop\java1>java EMain
Enter father's age: aba
Invalid input.
```

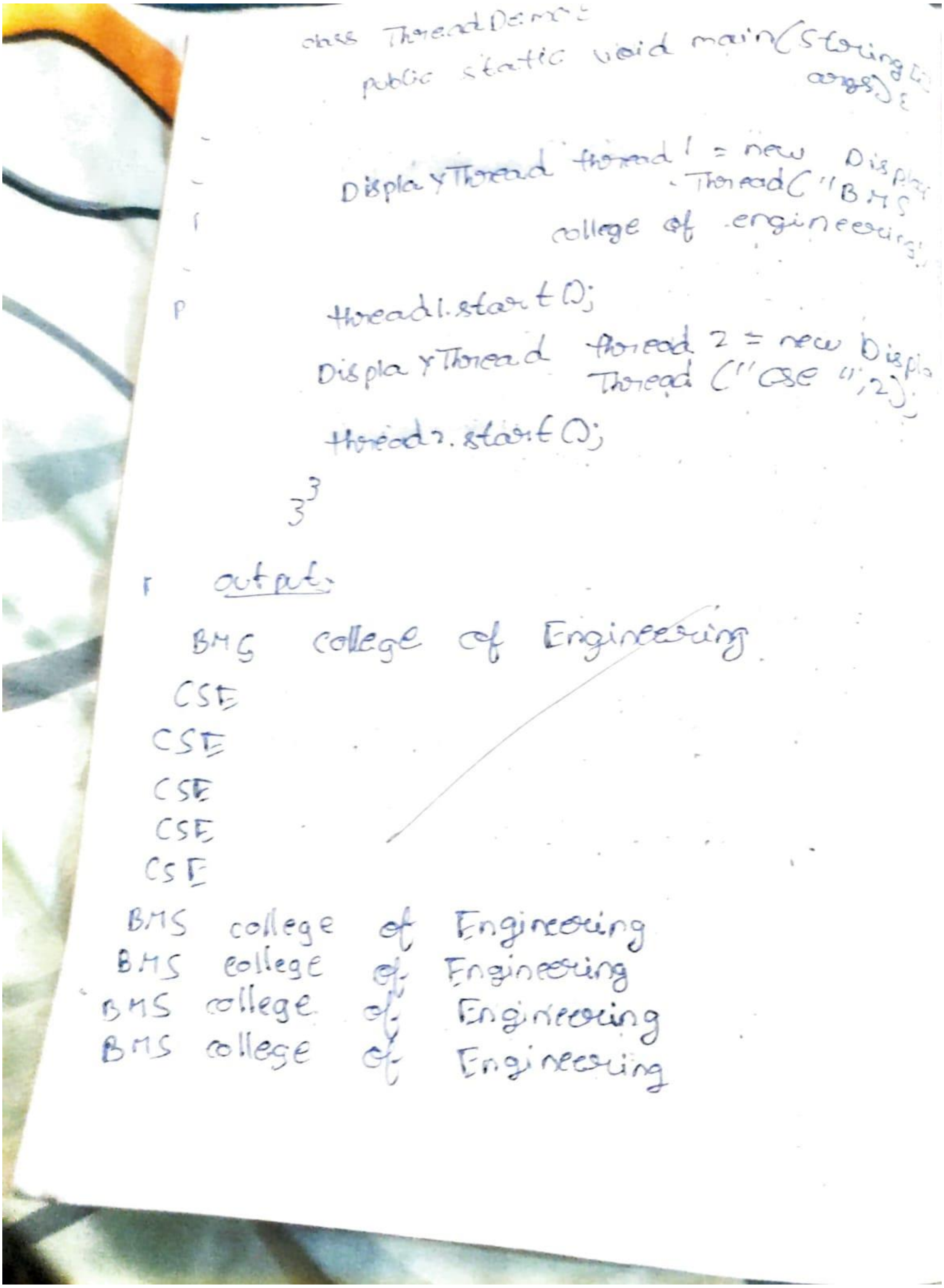
Program 8 write a program which creates two threads, one thread displaying "BMS College of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.



programs

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

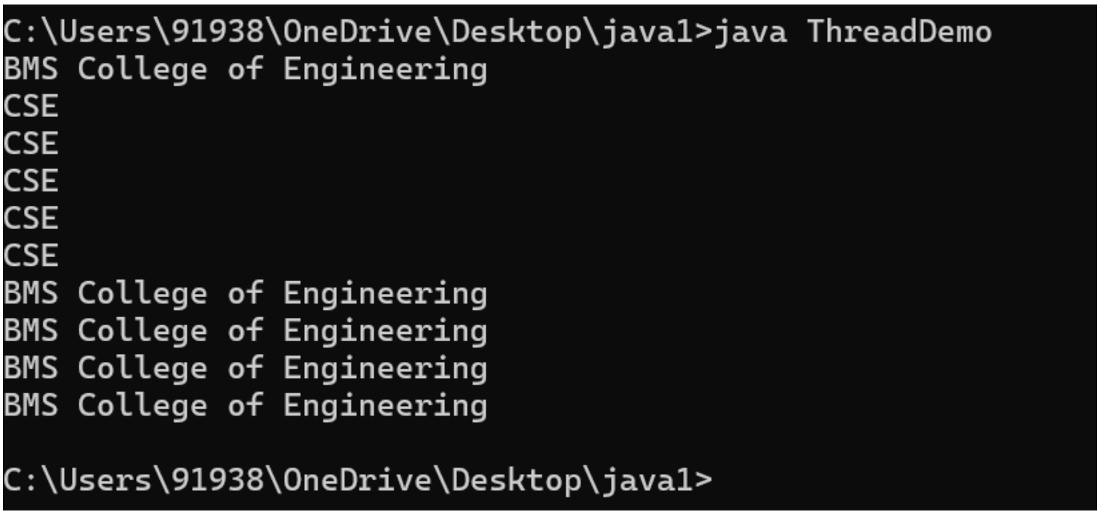
```
class DisplayThread
extends Thread {
    private String message;
    private int interval;

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){

DisplayThread thread1 = new Display
                    Thread("BMS
          college of engineering",

thread1.start();
DisplayThread thread2 = new Display
                 Thread ("CSE ",2);
thread2.start();
}}
```

output:

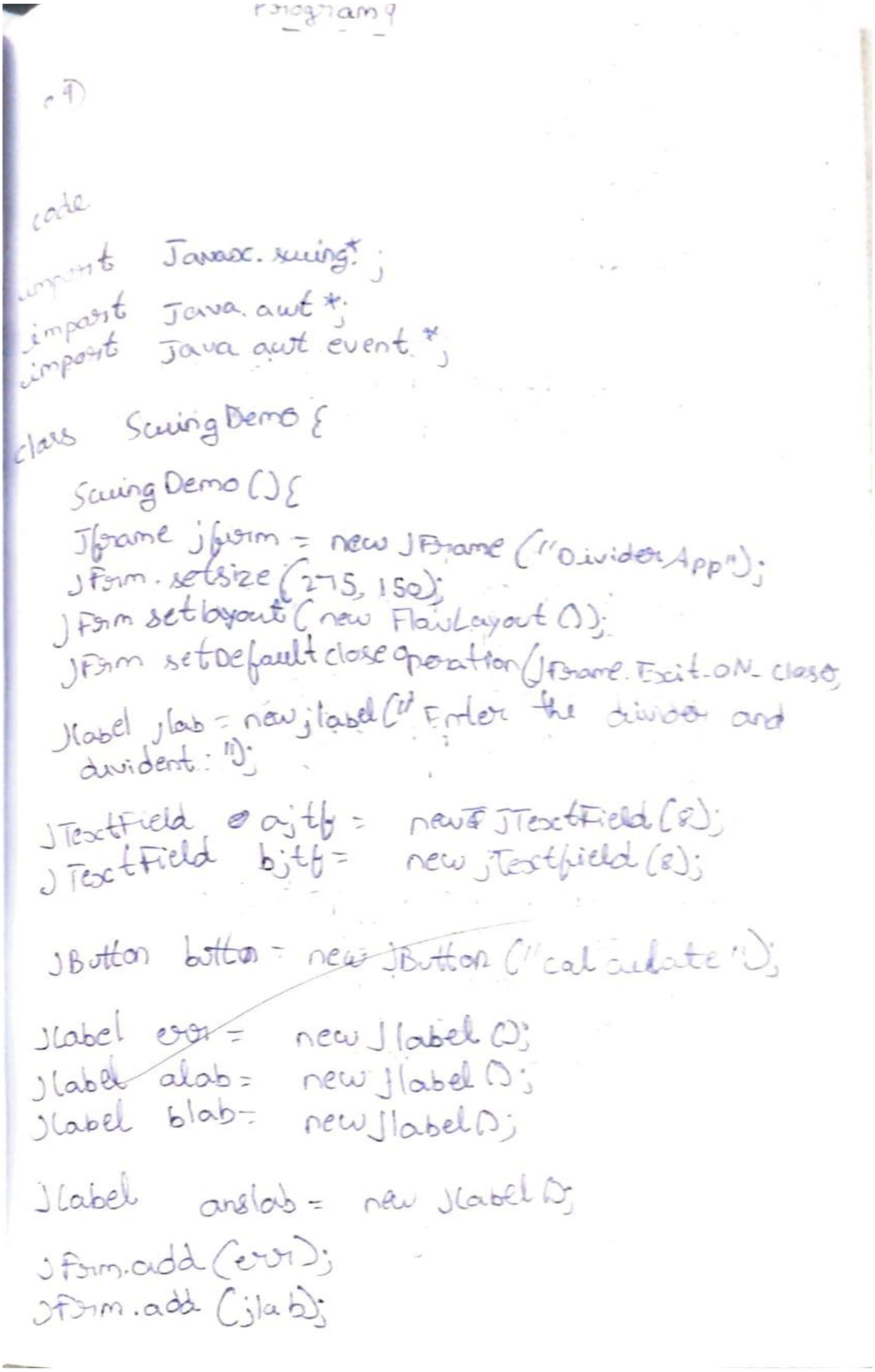
```
BMS college of Engineering
CSE
CSE
CSE
CSE
CSE
BMS college of Engineering
BMS college of Engineering
BMS college of Engineering
BMS college of Engineering
```



```
C:\Users\91938\OneDrive\Desktop\java1>java ThreadDemo
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

C:\Users\91938\OneDrive\Desktop\java1>
```

PROGRAM 9

Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.



Program 9

Q1)

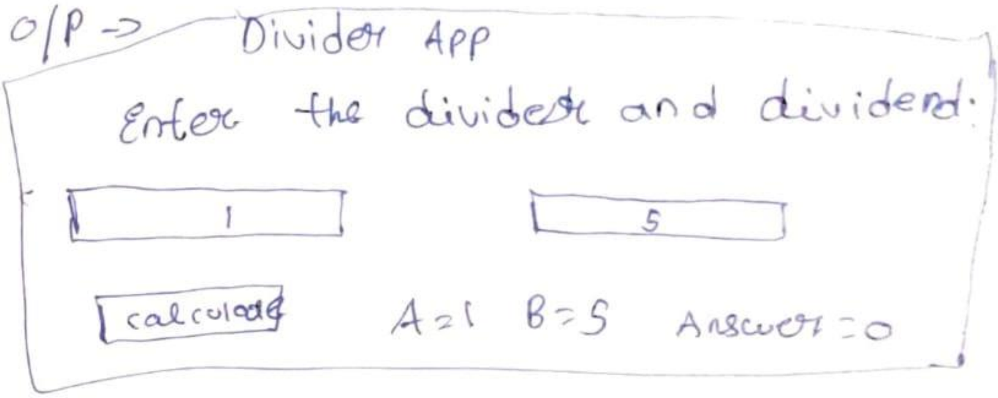
code

```
import  Javax. swing*;
import  Java. awt *;
import  Java awt event *;

class  SwingDemo {
   SwingDemo (){
   Jframe jfrm = new JFrame ("DividerApp");
   JFrm . setsize (275, 150);
   JFrm setlayout ( new FlowLayout ());
   JFrm setDefault close operation (JFrame. Exit-ON- close);
   Jlabel jlab = new jlabel (" Enter the divider and
     divident: ");
   JTextField  ajtf =  new JTextField (8);
   JTextField  bjtf =  new jTextfield (8);
   JButton  button = new JButton ("calculate");
   Jlabel  err =  new Jlabel ();
   Jlabel  alab =  new Jlabel ();
   Jlabel  blab =  new Jlabel();
   Jlabel   anslab = new Jlabel ();
   JFrm.add (err);
   JFrm .add (jlab);
```

```
    JFrm.add (ajtf);
    JFrm.add (bjtf);
    JFrm.add (button);
    JFrm.add(alab);
    JFrm.add (alab);
    JFrm.add (blab);
    JFrm.add (anslab);


button.addActionListener(new ActionListener(){
    public void actionPerformed(Action Event evt){
        try{

            int a= Integer parseInt(ajtf.getText());
            int b= Integer.parse Int(bjtf.getText());
            int ans = a/b;

            alab.setText("\nA= "+a);
            blab.setText("\n B= "+b);
            anslab.setText("\n Ans= "+ans);

        }

        catch(NumberFormatException e){
            alab.setText(" ");
            blab.setText(" ");
            anslab.setText(" ");
            err.setText(" ");
            ~~err.s~~
        }
    }
    });
  JFrm.setvisible(true);

}
```

```
public static void Main(string args [ ]){
Swing Utitilig. invokeLator(new Runnable() {
public void run(){
new SwingDemo();
}
});
}
}
```

O/P ->



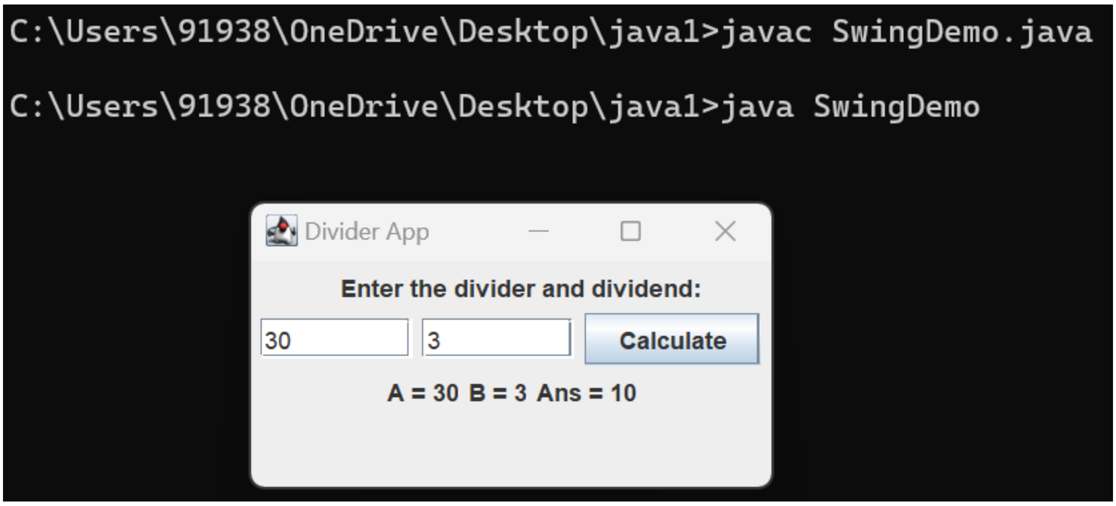
Divider App

Enter the divider and dividend:

| 1 | | 5 |

| calculate | A=1 B=5 Answer=0

23/2/24



```
C:\Users\91938\OneDrive\Desktop\java1>javac SwingDemo.java

C:\Users\91938\OneDrive\Desktop\java1>java SwingDemo
```

Enter the divider and dividend:

30

3

A = 30 B = 3 Ans = 10