VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT
on

Submitted by

PRIYANKA G H (1BM21CS144)

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

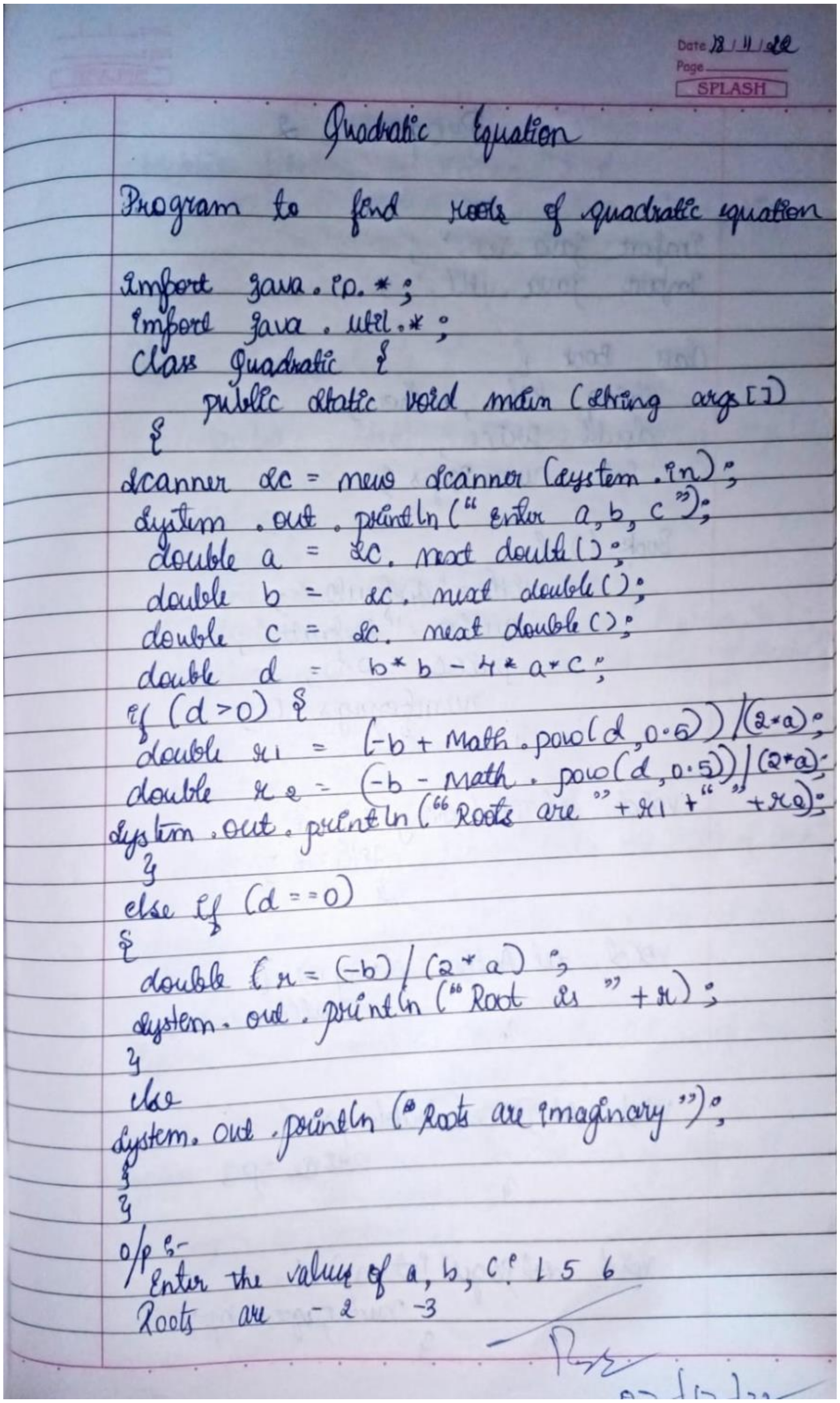
B. M. S. College of Engineering,
Bull Temple Road, Bangalore 560019
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Lab work entitled "Object Oriented Java Programming (21CS3PCOOJ)" carried out by **PRIYANKA G H (1BM21CS144),** 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 (21CS3PCOOJ) work prescribed for the said degree.

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



Date 18/11/22
Page
SPLASH

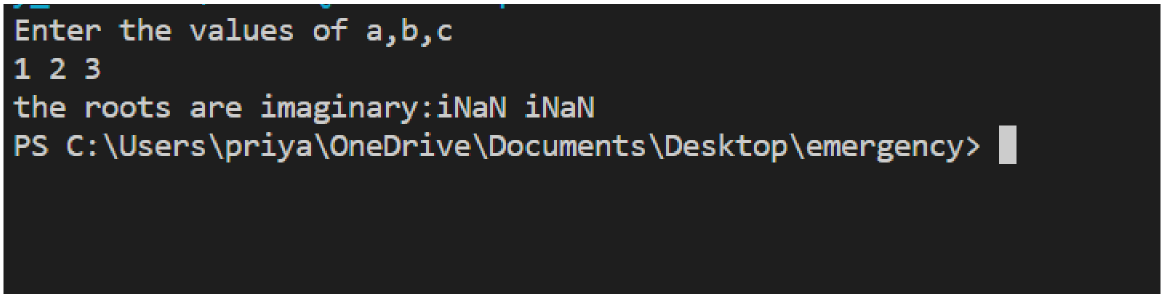
Quadratic Equation

Program to find roots of quadratic equation

```
import java.io.*;
import java.util.*;
class Quadratic {
      public static void main (String args[])
  {
scanner sc = new scanner (system.in);
 system.out.println (" enter a, b, c ");
  double a = sc. next doulb();
  double b = sc. next double();
  double c = sc. next double();
  double d = b*b - 4*a*c;
 if (d>0) {
  double x1 = (-b + Math.pow(d,0.5)) / (2*a);
 double x2 = (-b - Math.pow(d,0.5)) / (2*a);
system.out.println ("Roots are " + x1 + " " + x2);
 }
else if (d==0)
{
 double (x = (-b) / (2*a);
 system.out.println ("Root is " + x);
}
else
system.out.println ("Roots are imaginary");
}
}
```

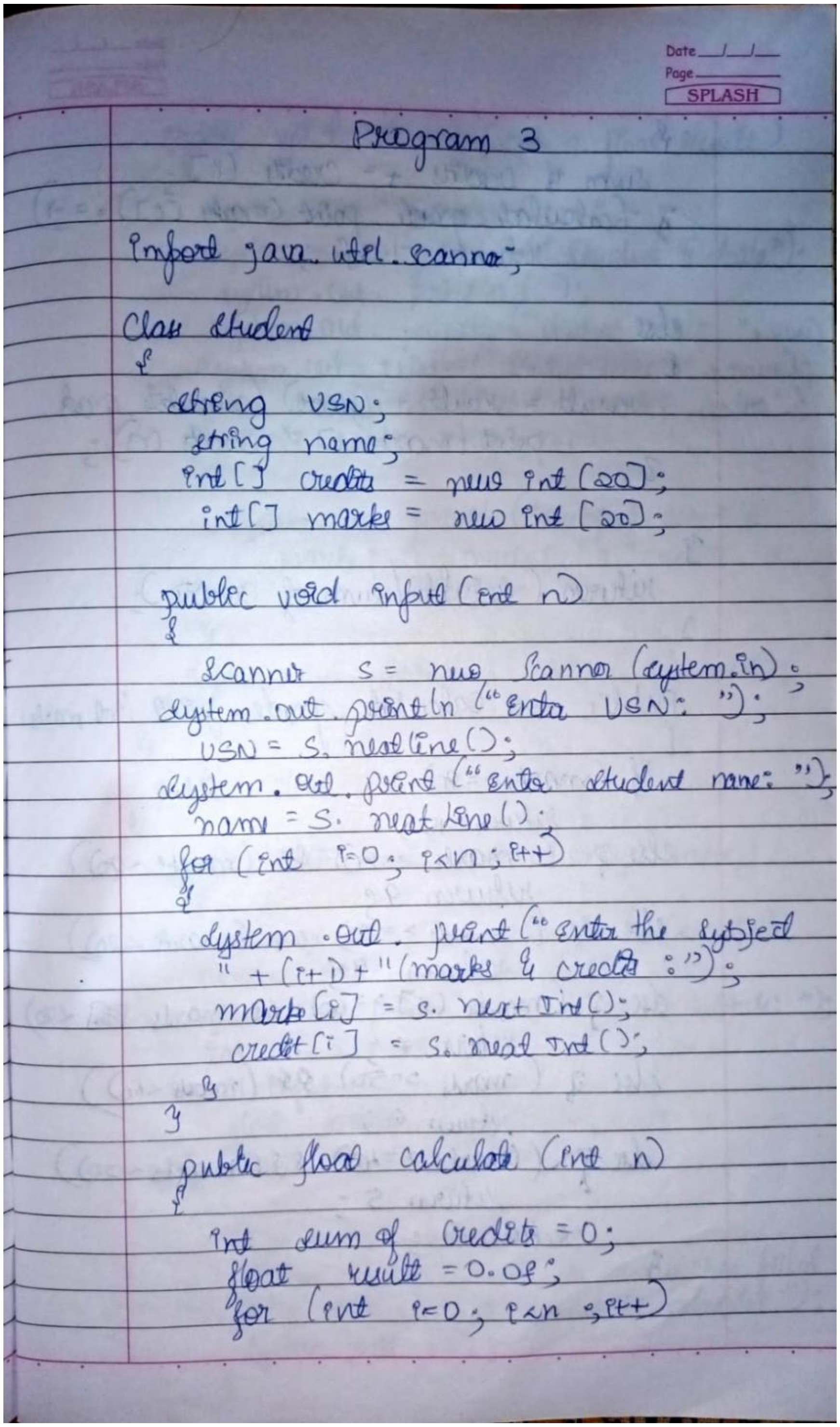
o/p :-

```
Enter the values of a, b, c: 1 5 6
Roots are -2 -3
```



```
Enter the values of a,b,c
1 2 3
the roots are imaginary:iNaN iNaN
PS C:\Users\priya\OneDrive\Documents\Desktop\emergency>
```

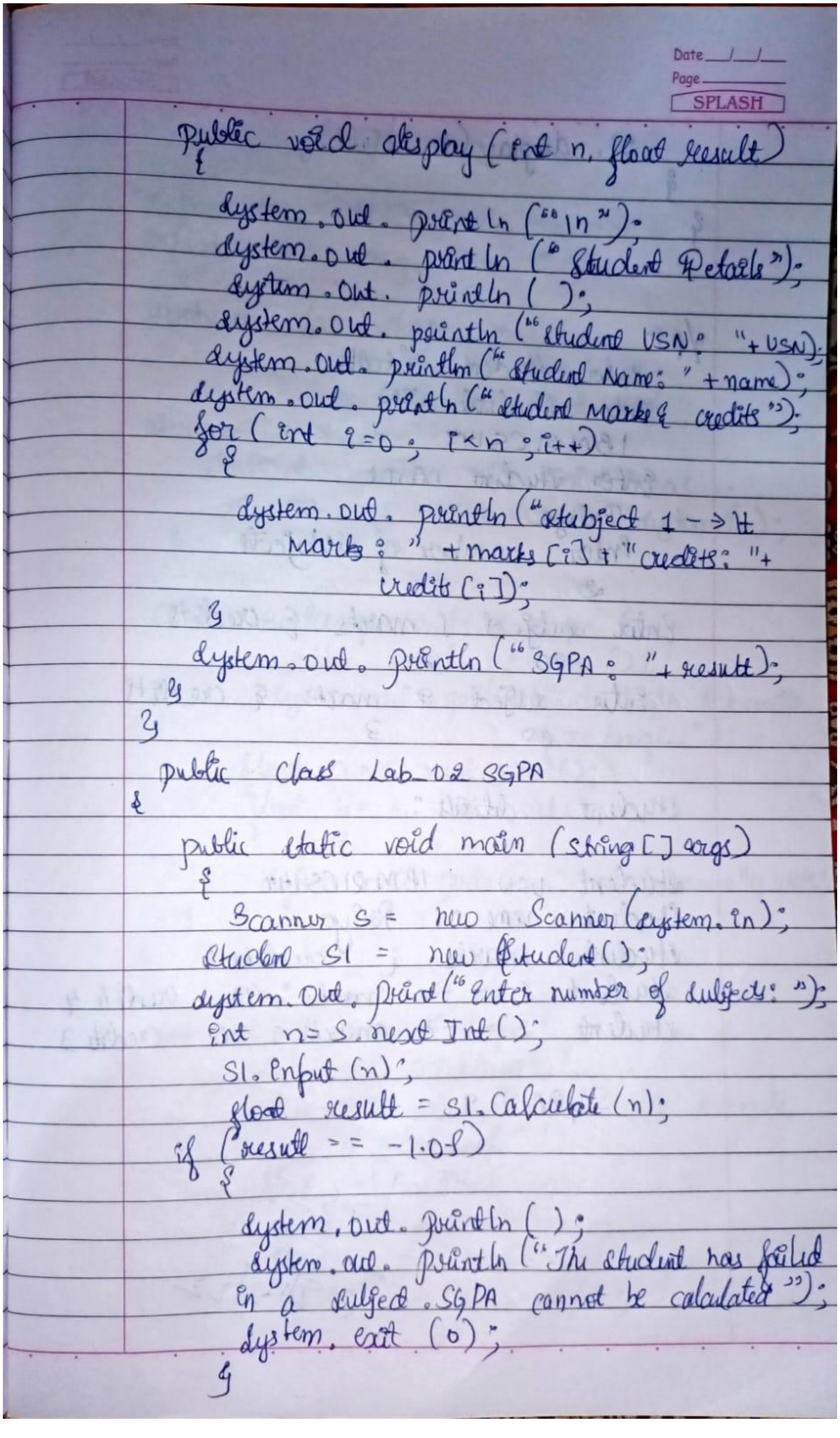
2. Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.



Date __/__/__
Page ______
SPLASH

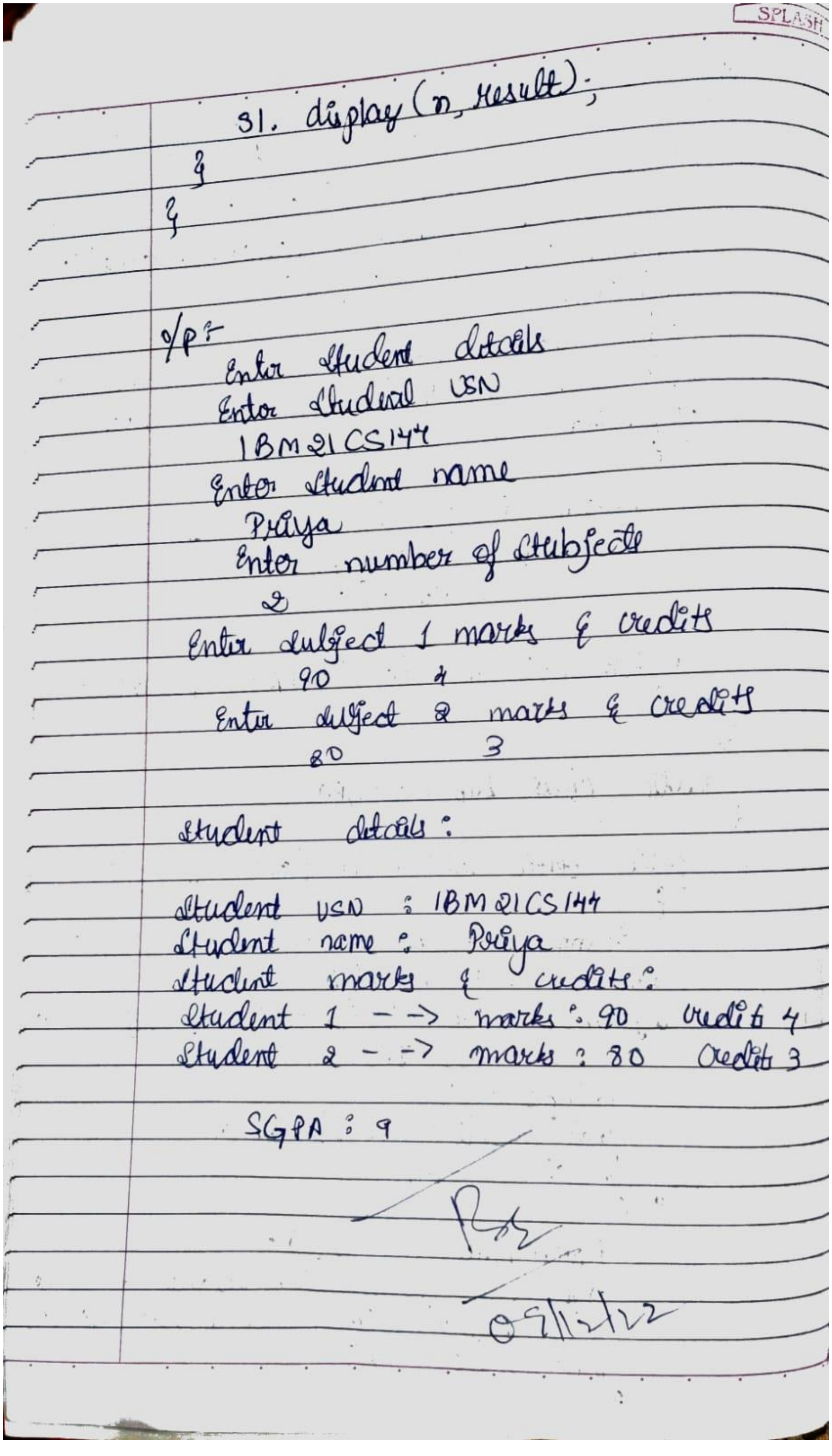
Program 3

```
import java.util.scanner;

Class student
{
  string USN;
   string name;
   int[] credits = new int [20];
    int[] marks = new int [20];

  public void input (int n)
  {
    Scanner  s = new Scanner (system.in);
  system.out.println ("Enter USN: ");
    USN = S. nextLine();
  system. out. print ("Enter student name: ");
    name = S. next Line();
   for (int i=0; i<n; i++)
   {
     system .out. print ("Enter the subject
      " + (i+1) + "(marks & credits :");
     marks[i] = s. next Int();
      credit [i] = s. next Int();
    }
  }
  public float calculate (int n)
  {
     int sum of credits = 0;
     float result = 0.0f;
     for (int i=0; i<n ; i++)
{
  sum of credits += credits [i].
  if (calculate_grade_point (marks [i]) == -1)
          return -1.0f ;

else
{
  result = result + (float) (calculate_grade_
          point (marks[i] * credits [i]);
  }

}
  return ( result / sum of credits );
}

public int calculate_grade_point (int marks)
{
  if (marks >= 90)
        return 10;
else if ((marks >= 80) && (marks <90))
        return 9;
else if ((marks >= 70) && (mark <80))
        return 8;
else if (marks ~~[i]~~ >= 60) && marks ~~[i]~~ <70)
        return 7;
else if ( marks >=50) && (marks <60))
        return 6 ;
 else if ((marks >=40) && (marks <50))
        return 5 ;
      return -1 ;
  }
```



Date __/__/__
Page ____
SPLASH

```
Public void display (int n, float result)
{
   system. out. println ("\n");
   system. out. println ("Student Details");
   system. out. println ( );
   system. out. println ("Student USN: "+ USN);
   system. out. println ("Student Name: " + name);
  system. out. println ("Student Marks & credits");
  for ( int i=0 ; i<n ; i++)
   {
      system. out. println ("subject 1 -> \t
        Marks : " + marks [i] + "credits: "+
              credits [i]);
   }
  system. out. println ("SGPA : "+ result);
 }
}
 public class Lab_02 SGPA
{
  public static void main (String [] args)
   {
    Scanner s = new Scanner (system. in);
   Student S1 = new Student ();
   system. out. print ("Enter number of subjects: ");
   int n= s. next Int ();
    S1. input (n);
    float result = S1. Calculate (n);
  if (result == -1.0f)
   {
     system. out. println ( );
     system. out. println ("The student has failed
     in a subject. SGPA cannot be calculated");
     system. exit (0);
   }
```

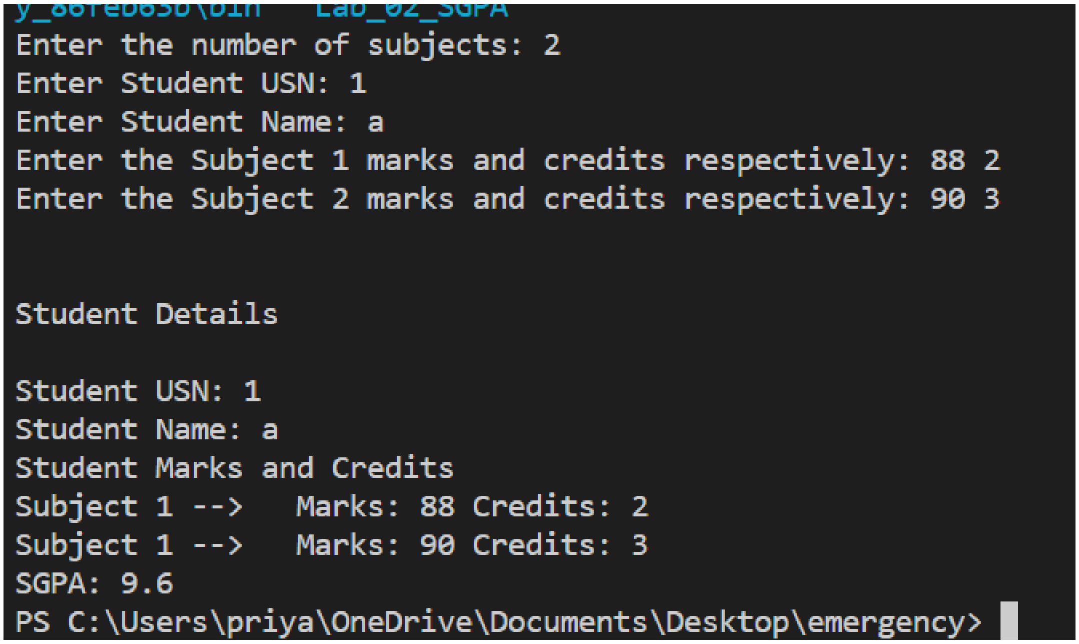


```
s1. display(n, result);
  }
}
```

o/p:

```
  Enter Student details
  Enter Student USN
   1BM21CS144
  Enter Student name
   Priya
   Enter number of Subjects
    2
 Enter subject 1 marks & credits
      90         4
  Enter subject 2 marks & credits
      80          3

Student details:

Student USN : 1BM21CS144
Student name : Priya
Student marks & credits:
Student 1 --> marks: 90  credit 4
Student 2 --> marks: 80  credit 3

  SGPA : 9
```

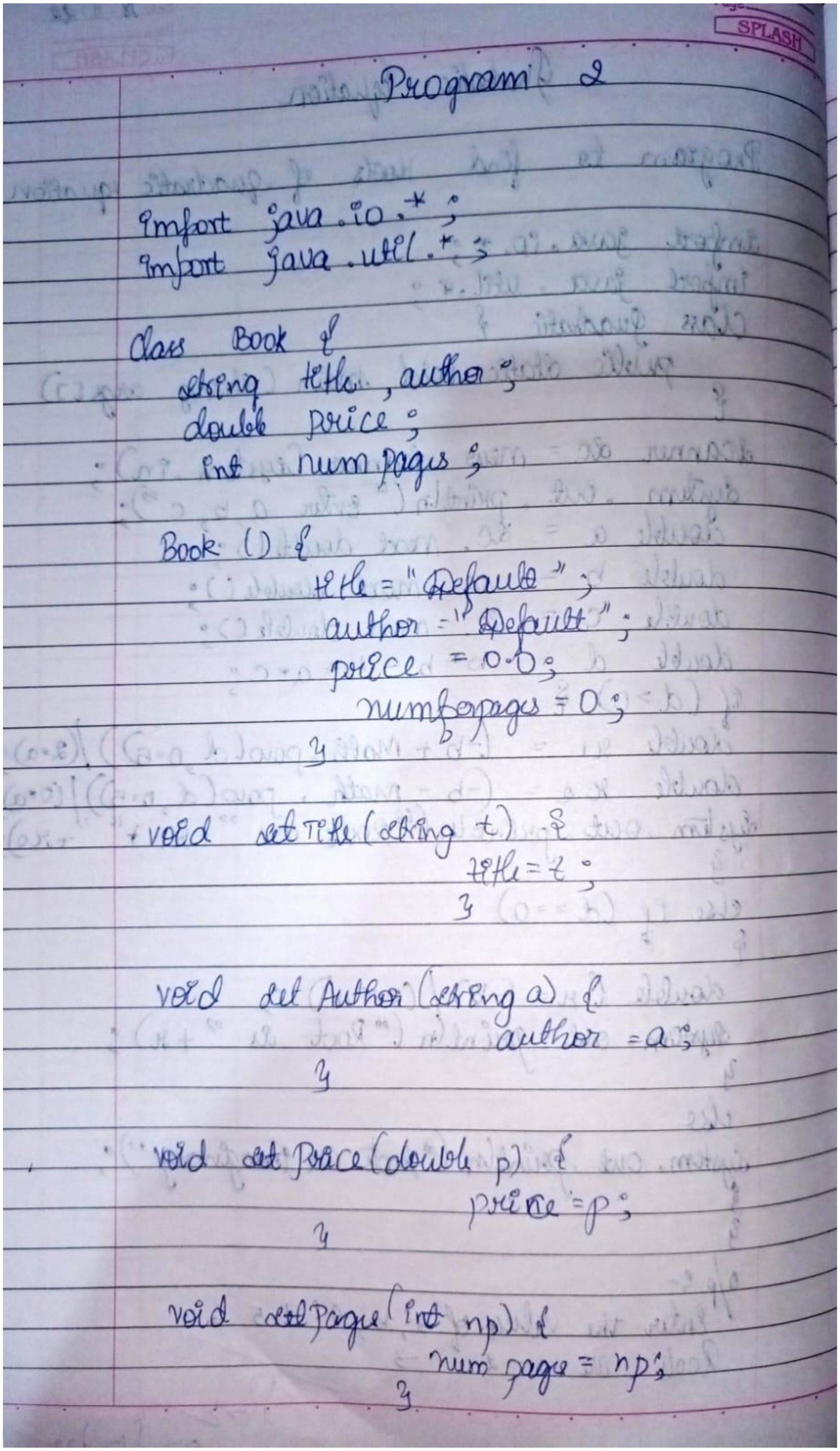
09/12/22



```
y_86feb83b\bin   Lab_02_SGPA
Enter the number of subjects: 2
Enter Student USN: 1
Enter Student Name: a
Enter the Subject 1 marks and credits respectively: 88 2
Enter the Subject 2 marks and credits respectively: 90 3


Student Details

Student USN: 1
Student Name: a
Student Marks and Credits
Subject 1 -->   Marks: 88 Credits: 2
Subject 1 -->   Marks: 90 Credits: 3
SGPA: 9.6
PS C:\Users\priya\OneDrive\Documents\Desktop\emergency>
```

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.



Program 2

```
import java.io.* ;
import java.util.* ;

class Book {
  string title, author ;
  double price ;
  int num pages ;

  Book () {
           title = "Default" ;
           author = "Default" ;
           price = 0.0 ;
             numofpages = 0 ;
         }

  void setTitle (string t) {
                   title = t ;
                 }

  void set Author (string a) {
                   author = a ;
           }

  void set Price (double p) {
                   price = p ;
           }

  void setPage (int np) {
                 num page = np ;
           }
```

```
Public String to String () {
    return title +"\t" + author + "\t"+ price+
        "\t"+ numpages +"\n";
    }

Class BookDetails {

    public static void main (String args [])
{
    String t, a;
    double p;
    int np, n;
    Scanner sc = new Scanner (System.in);
  System.out.println ("Enter the no. of Books");
    n = sc.nextInt();
  Book b[] = new Book [n];
  for (int ~~a~~ i=0; i<n; i++)
  {
  System.out.println ("Enter the Title of Books");
      t = sc.next();
  System.out.println ("Enter the Author of the
                      Books");
      a = sc.next();
  System.out.println ("Enter the Price of the
                        Books");
      p = sc.nextDouble();
System.out.println ("Enter the no of pages of
                the Books");
    np = sc.nextInt();
```

Page

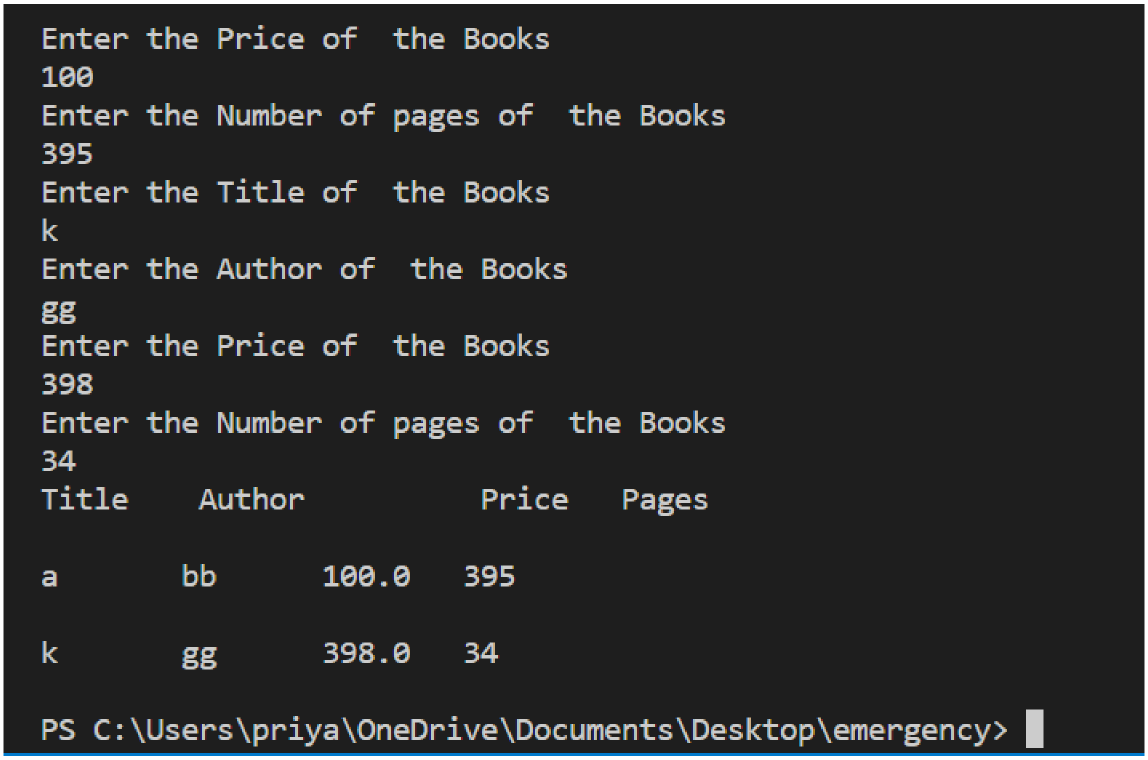
SPLASH

```
        b[i] = neuo Book();
        b[i] =set Title(t);
        b[i] . set Author(a);
        b[i] . set Price (p);
        b[i] . set Pages (np);
}

system . out . println(" Title \t Author \t
                price \t pages \n");
for(int i=0 ; i<n ;i++)
 {
     system . out . println( b[i]);
}
}
```

O/p :-

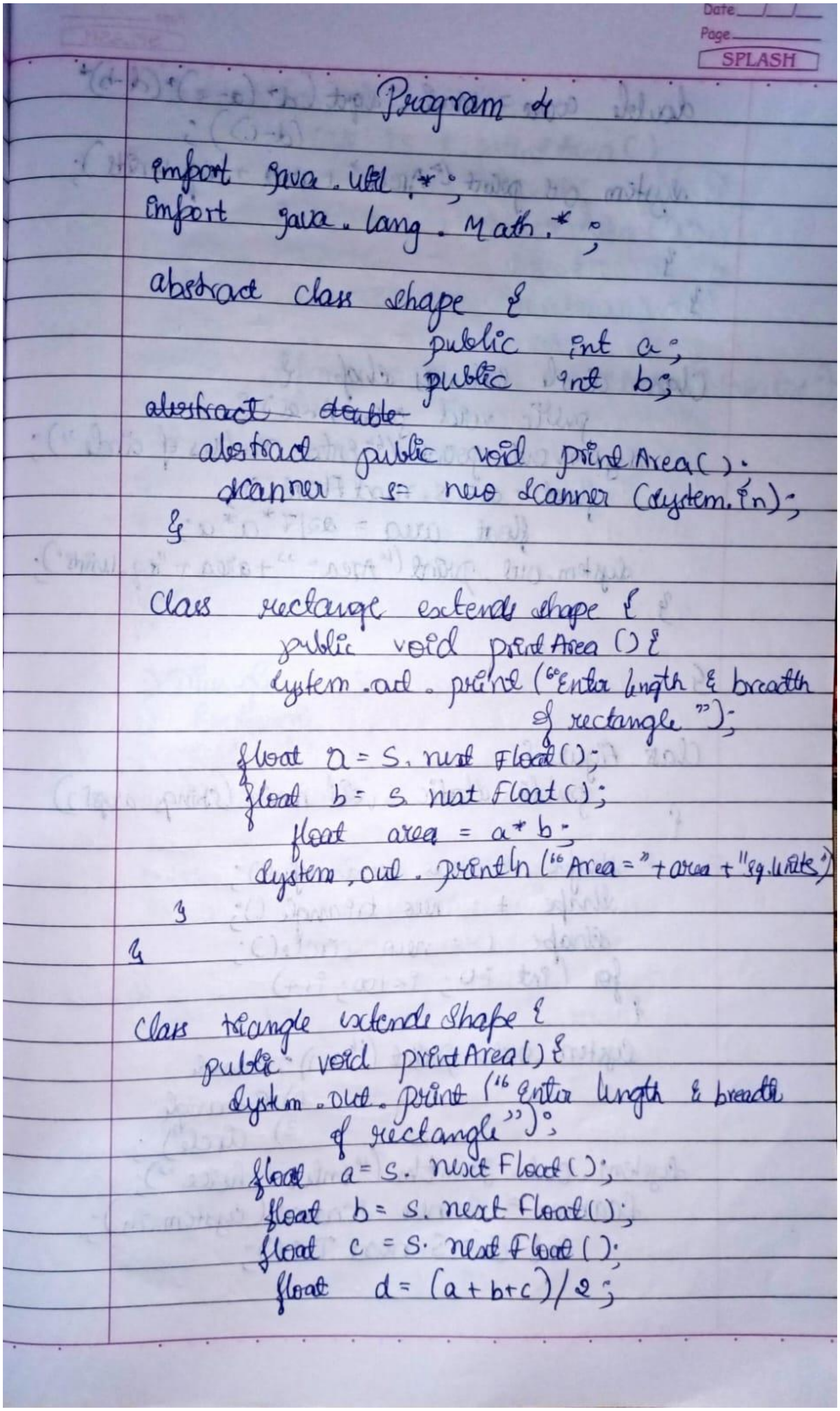
```
Title      Author      price      Pages
P          pqh         300.0      190
Q          grs         400.0      300
```

02/12/22



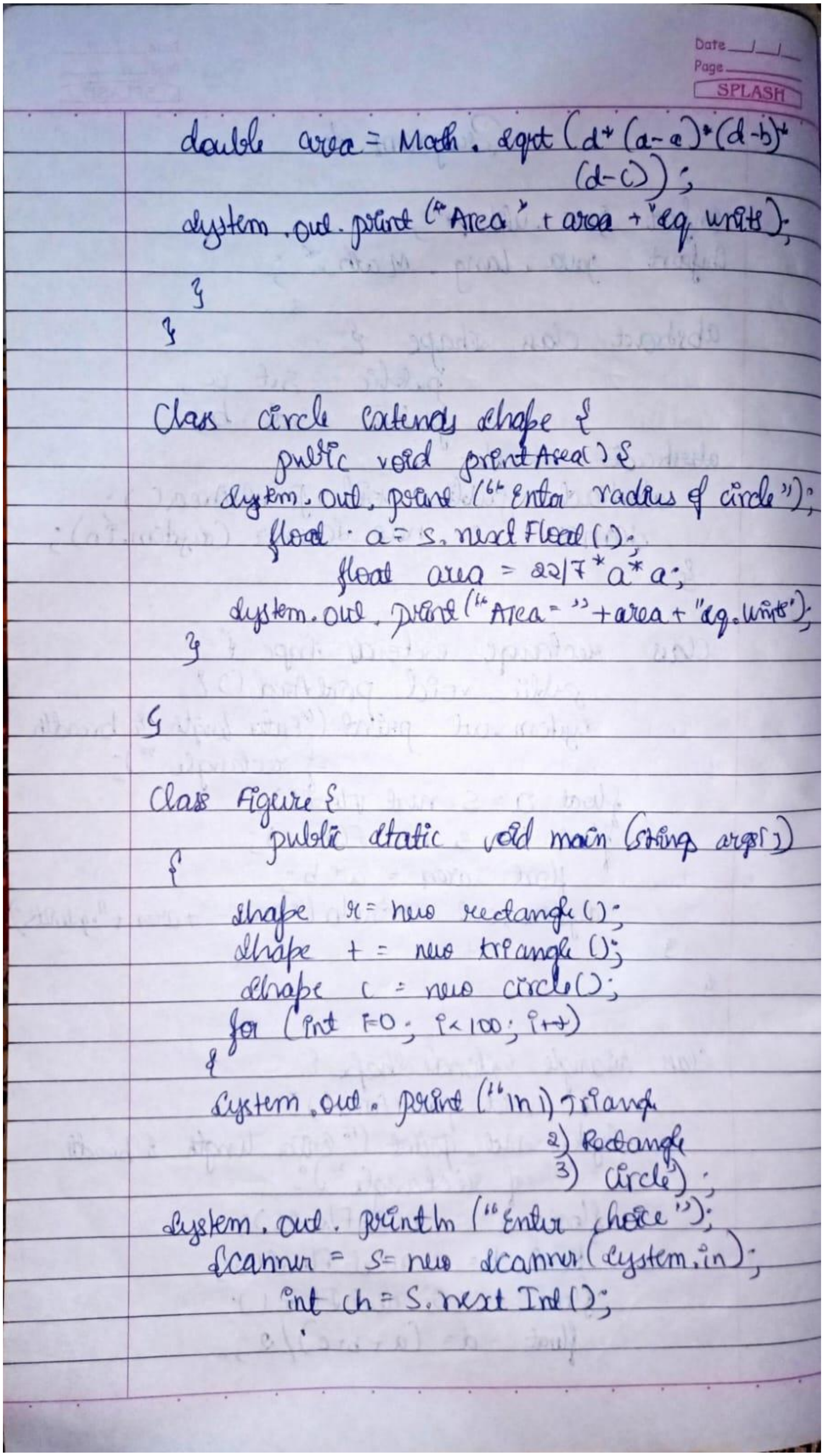
```
Enter the Price of  the Books
100
Enter the Number of pages of  the Books
395
Enter the Title of  the Books
k
Enter the Author of  the Books
gg
Enter the Price of  the Books
398
Enter the Number of pages of  the Books
34
Title    Author          Price   Pages

a       bb      100.0   395

k       gg      398.0   34

PS C:\Users\priya\OneDrive\Documents\Desktop\emergency>
```

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.



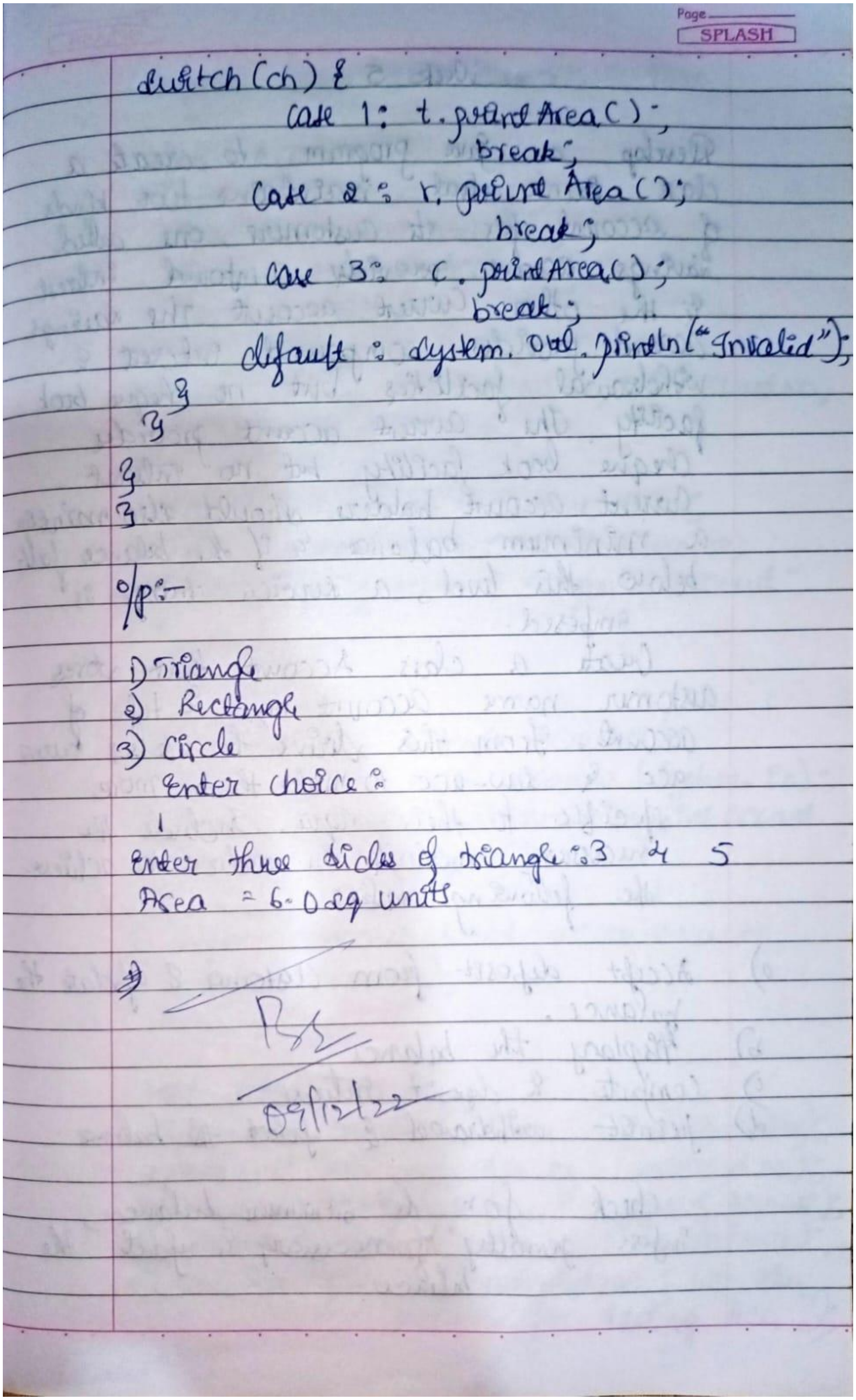
Program 4

```
import java.util.*;
import java.lang.Math.*;

abstract class shape {
                    public int a;
                    public int b;
~~abstract double~~
    abstract public void print Area().
       scanner s= new scanner (system.in);
  }

class rectange extends shape {
          public void print Area () {
          system.out. print ("Enter length & breadth
                              of rectangle ");
       float a = s. next Float();
       float b= s next Float();
           float area = a* b;
       system.out. println ("Area ="+area +"sq.units")
   }
}

class triangle extends Shape {
     public void printArea() {
       system.out. print ("Enter length & breadth
                  of rectangle");
         float a= s. next Float();
         float b= s. next Float();
         float c = s. next Float();
         float d= (a+b+c)/2;
```



Date __/__/__
Page ____
SPLASH

```
double area = Math. sqrt (d* (a-a)*(d-b)*
                              (d-c));
system.out.print ("Area" + area + "sq. units");

  }
}


Class circle extends shape {
        public void printArea() {
    system.out.print ("Enter radius of circle");
      float a = s.nextFloat();
            float area = 22/7*a*a;
   system.out.print("Area = "+area+"sq.units");
  }

}

Class Figure {
        public static void main (String args[])
 {
      shape r = new rectangle();
      shape t = new triangle();
      shape c = new circle();
      for (int i=0; i<100; i++)
    {
    system.out.print ("\n1) Triangle
                          2) Rectangle
                          3) circle");
 system.out.println ("Enter choice");
    scanner = s= new scanner(system.in);
        int ch = S.next Int();
```

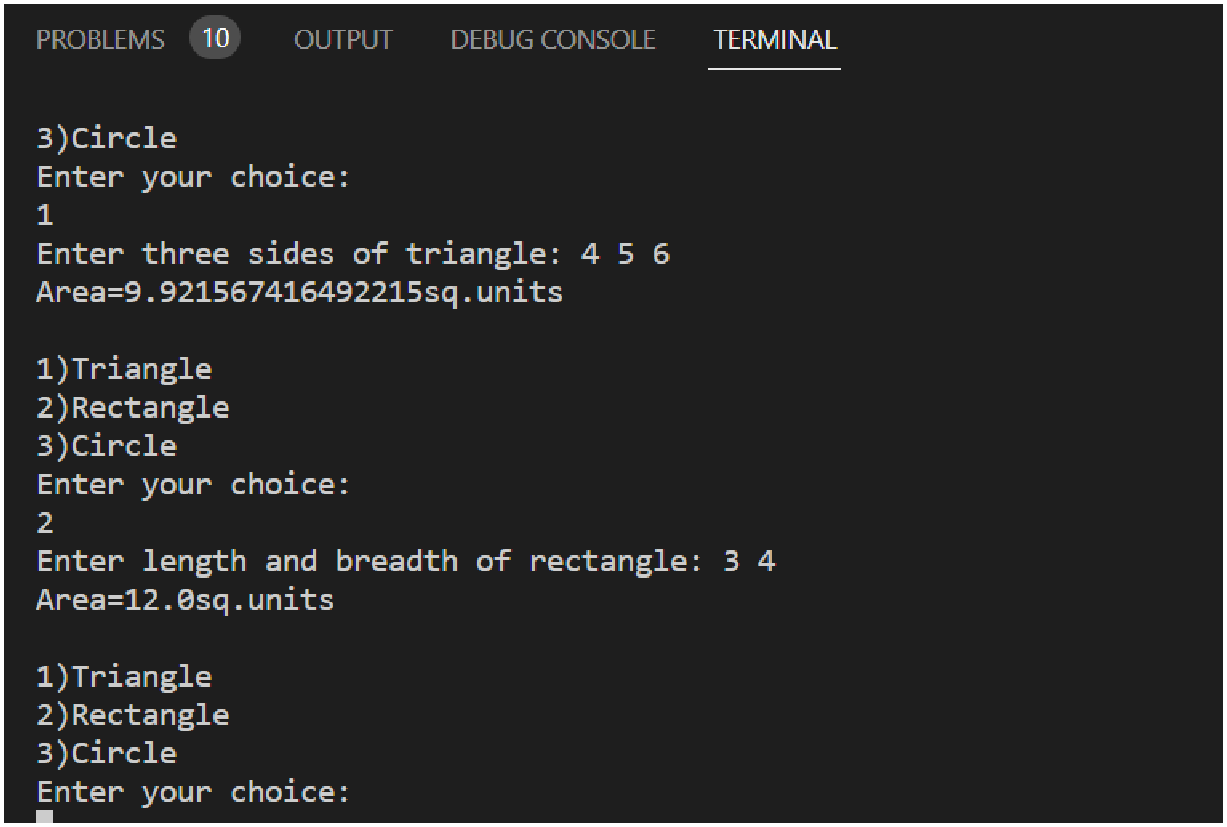


```
switch (ch) {
            case 1: t.printArea();
                    break;
        case 2 : r. printArea();
                    break;
        case 3:   c. printArea();
                    break;
        default : System. out. println("Invalid");
    }
  }
}
}
```

o/p:-

```
1) Triangle
2) Rectangle
3) Circle
  Enter choice :
  1
Enter three sides of triangle : 3   4   5
Area  = 6.0 sq units
```

09/12/22



```
3)Circle
Enter your choice:
1
Enter three sides of triangle: 4 5 6
Area=9.921567416492215sq.units

1)Triangle
2)Rectangle
3)Circle
Enter your choice:
2
Enter length and breadth of rectangle: 3 4
Area=12.0sq.units

1)Triangle
2)Rectangle
3)Circle
Enter your choice:
```

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.

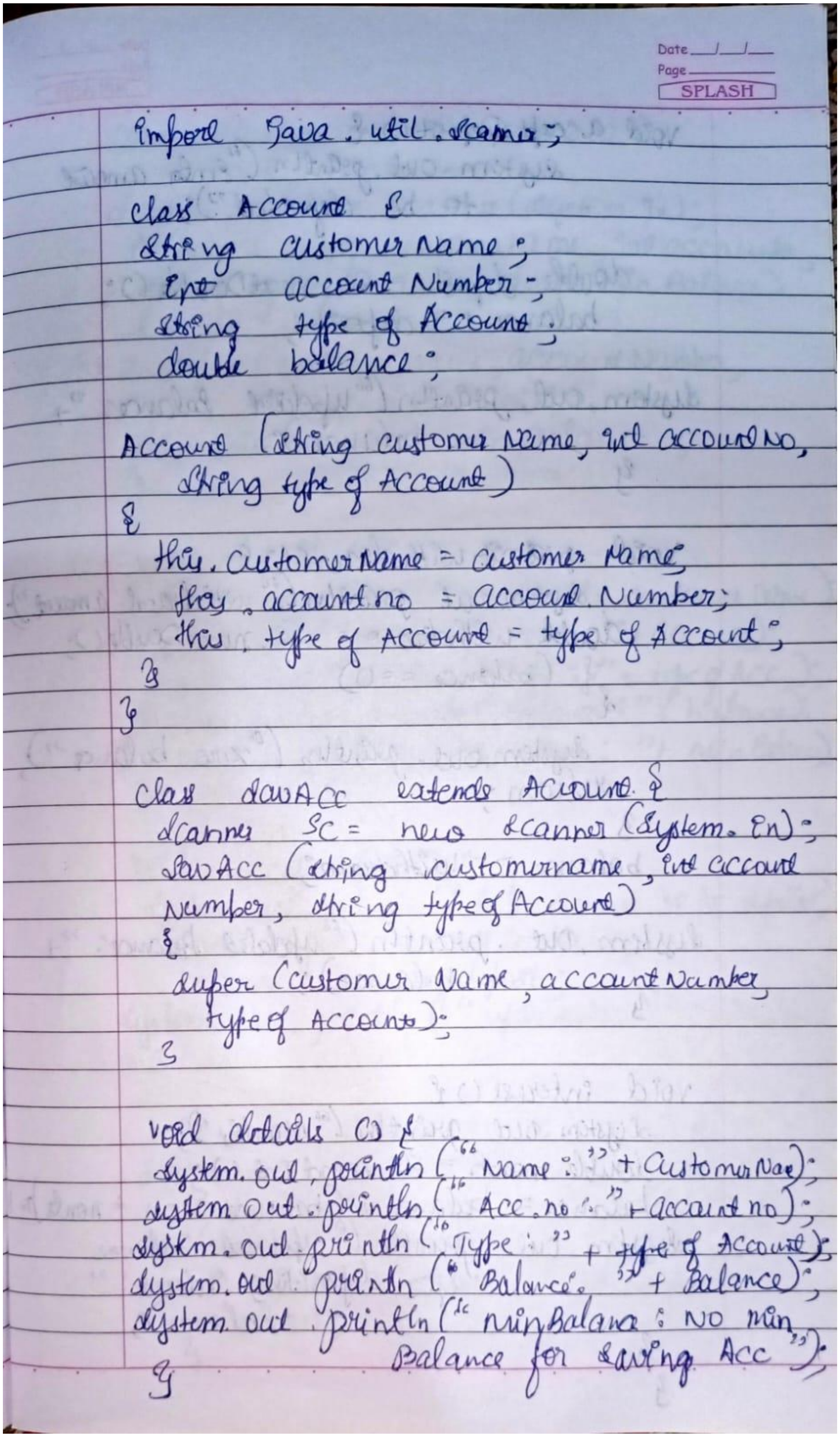
Create a class Account that stores customer name, account number and type of account. From this derive the classes Cur-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

a) Accept deposit from customer and update the balance.

b) Display the balance.

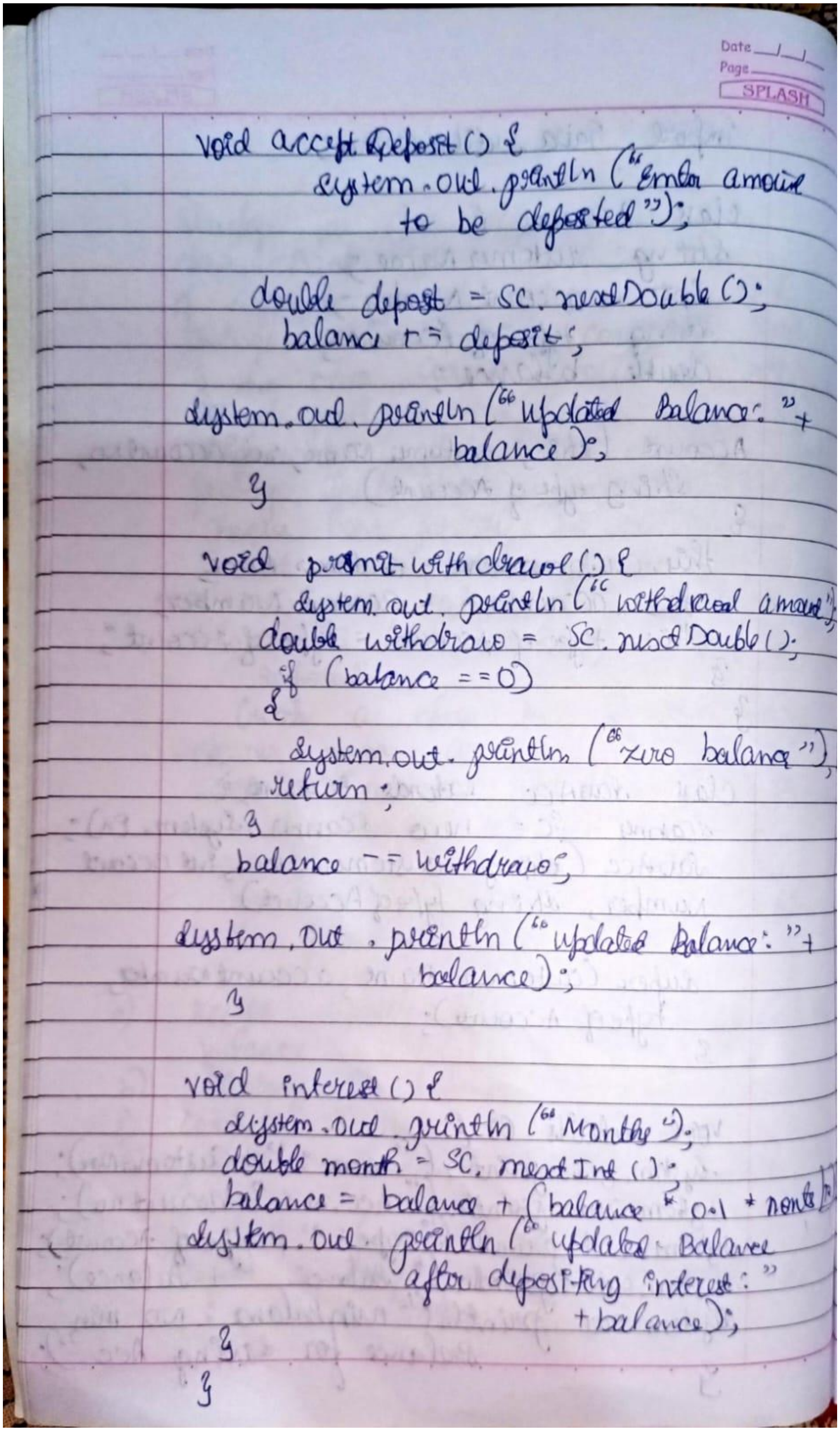
c) Compute and deposit interest

d) Permit withdrawal and update the balance

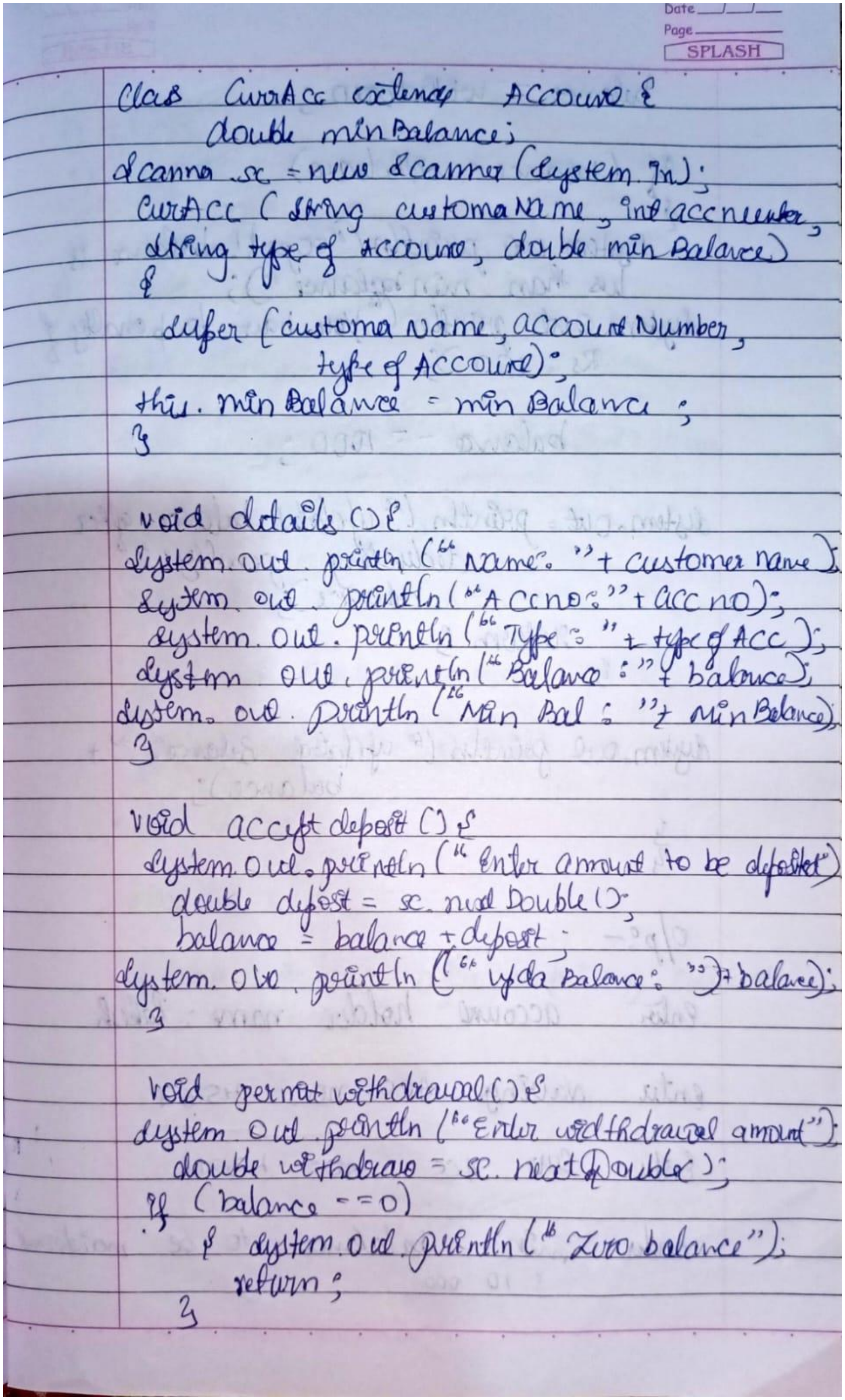


Date __/__/__
Page ______
SPLASH

```
import Java. util. Scanner;

class Account {
String customer Name;
 int account Number;
 String type of Account;
 double balance;

Account (String customer Name, int accountNo,
   String type of Account)
{
 this. customerName = customer Name;
  this. account no = account Number;
  this. type of Account = type of Account;
 }
}

class SavAcc extends Account {
Scanner sc = new Scanner (System. in);
SavAcc (String customername, int account
Number, String typeof Account)
{
super (customer Name, account Number,
  typeof Account);
}

void details () {
System. out. println (" Name : " + CustomerNae);
System out. println (" Acc. no : " + account no);
System. out println (" Type : " + type of Account);
System. out println (" Balance: " + Balance);
System. out println (" minBalance : No min
            Balance for Saving Acc ");
}
```



```
void accept Deposit() {
        system.out.println("Enter amount
              to be deposited");

     double deposit = sc.nextDouble();
      balance += deposit;

system.out.println("updated Balance: "+
               balance);
  }

 void permit withdrawl() {
   system.out.println("withdrawal amount");
   double withdraw = sc.nextDouble();
    if (balance ==0)
   {
      system.out.println("zero balance");
   return;
   }
  balance -= withdraw;

system.out.println("updated Balance: "+
              balance);
  }

void interest() {
  system.out.println("Months");
  double month = sc.nextInt();
  balance = balance + (balance * 0.1 * month)
system.out.println("updated Balance
          after depositing interest:"
                   +balance);
  }
}
```

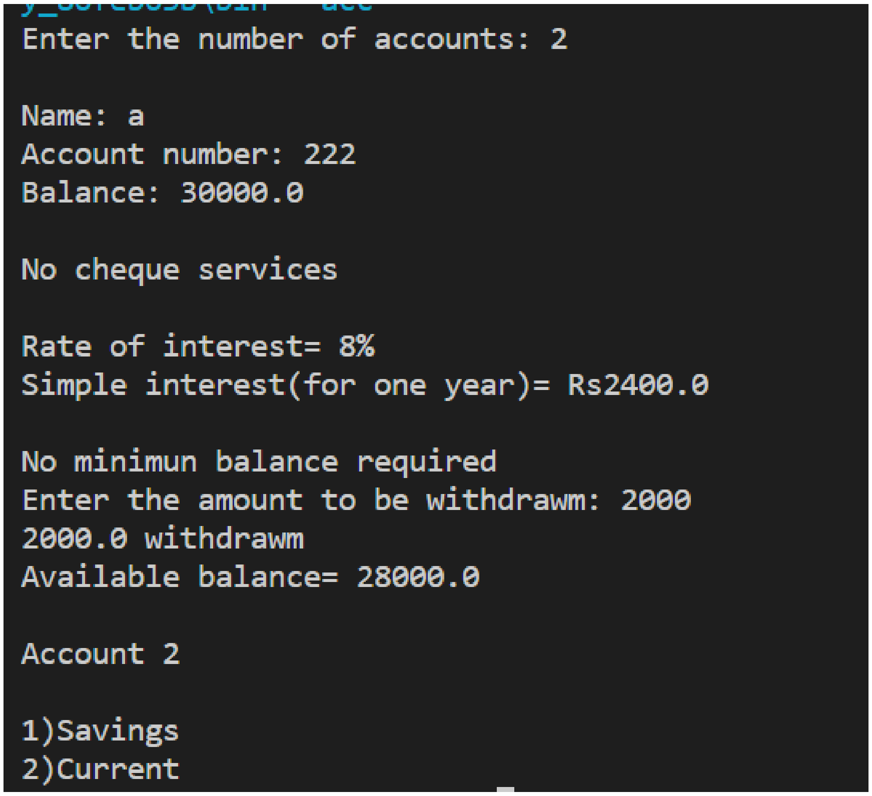


Date __/__/__
Page ____
SPLASH

```
Clas CurrAcc extends Account {
      double minBalance;
Scanna sc = new Scanner(System.In);
 CurrACC (String customaName, int accnumber,
 String type of Account, double min Balance)
 {
   super (customa Name, account Number,
            type of Account);
 this. min Balance = min Balance ;
 }

 void details (){
System.out. println ("Name: "+ customer name);
 System. out. println("A ccno: "+ acc no);
  System. out. println ("Type : " + type of ACC);
 System out. println("Balance : "+ balance);
System. out. println ("Min Bal : "+ Min Balance);
 }

 void accept deposit () {
  System.out. println (" Enter amount to be deposited")
   double deposit = sc. next Double ();
   balance = balance + deposit ;
System. out. println (" Upda Balance: ")+ balance);
  }

 void permit withdrawal(){
System.out.println ("Enter withdrawal amount");
  double withdraw = sc. next Double();
 if (balance ==0)
    { System.out. println (" Zero balance");
      return;
   }
balance -= withdraw ;

if (balance < withdraw)
{
  system.out.println ("oops !! balance is
       less than min balance ");
system.out.println (" You have to penalty of
       Rs : 100 ");

        balance -= 1000 ;

system.out.println (" updated balance after
            deducting penalty: "
                 + balance );
     return ;
  }

system.out.println (" updated Balance: " +
                      balance);

  }
}
```

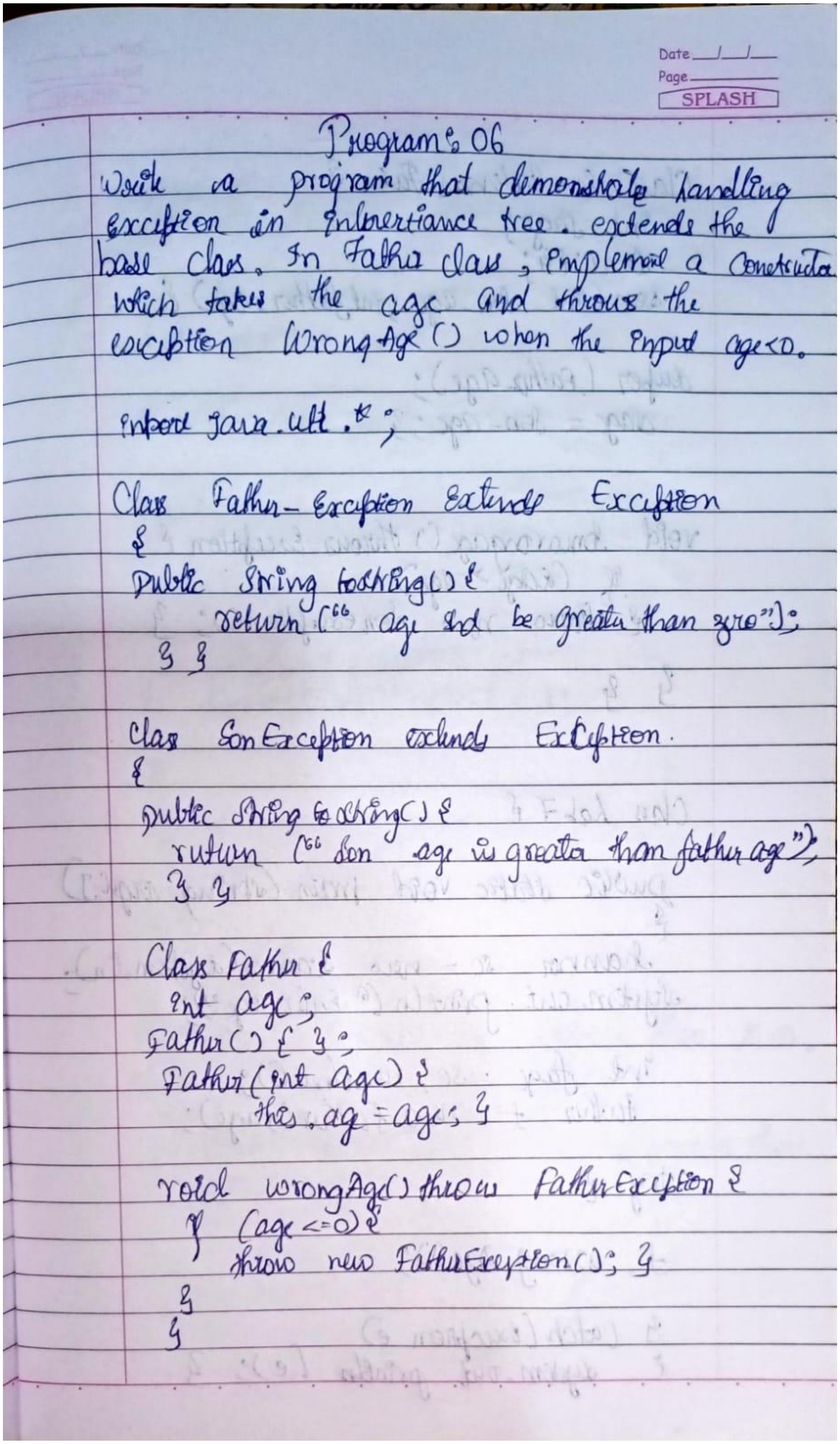
O/p:-

```
Enter   account holder name : Rakesh

enter  savings  ACC  no : 4567

Enter  curr  ACC  no : 43216

Enter  min  balalance  to  be  maintained
      : 10 000
```



```
Enter the number of accounts: 2

Name: a
Account number: 222
Balance: 30000.0

No cheque services

Rate of interest= 8%
Simple interest(for one year)= Rs2400.0

No minimun balance required
Enter the amount to be withdrawm: 2000
2000.0 withdrawm
Available balance= 28000.0

Account 2

1)Savings
2)Current
```

6.Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception WrongAge( ) when the input age<0. In Son class, implement a constructor that cases both father and son’s age and throws an exception if son’s age is >=father’s age.



Date __/__/__
Page ____
SPLASH

Program: 06

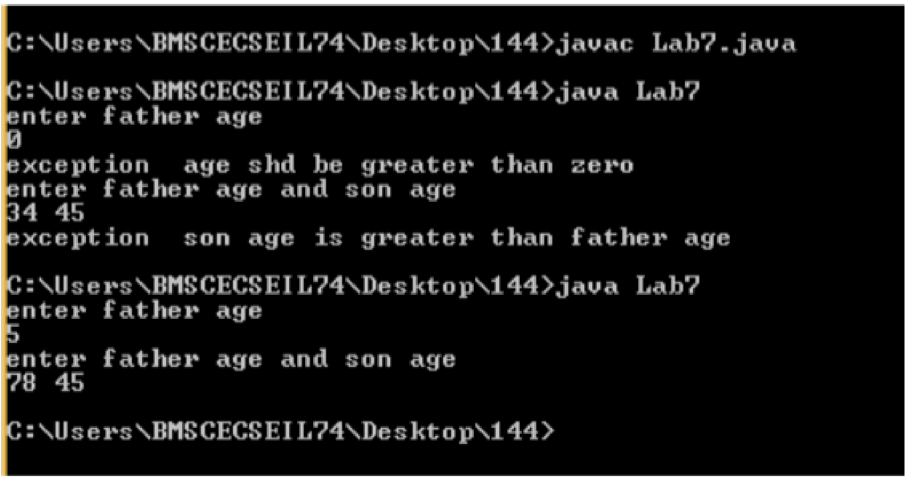
Write a program that demonstraite handling exception in Inheritiance tree. extends the base class. In Father class, implement a Constructor which takes the age and throws the exception WrongAge() when the input age<0.

```
import java.util.*;

Class Father-Exception Extends Exception
{
public String toString(){
    return ("age shd be greater than zero");
 } }

Class SonException extends Exception.
{
public String toString(){
  return ("son age is greater than father age");
 } }

Class Father {
 int age;
Father() { };
 Father(int age) {
      this.age = age; }

 void wrongAge() throw FatherException {
   if (age <= 0) {
     throw new FatherException(); }
  }
 }
```

```
Class Son Extends Father {
  int sage;
  son(){};
  son (int son_age, int father_age) {

super (Father_age);
  sage = son_age; }


void sonwrongage () throws Exception {
     if (sage >= age)
     { throw new sonException(); }

} }


Class Lab7 {

  public static void main (string args[])
  {
   Scanner sc = new Scanner (system.in);
system.out.println ("Enter age");

   int fage = sc.nextInt();
   Father f= new Father(fage);

  try {

   f. wrong Age();

   } catch (exception e)
  { system.out.println (e); }
dystem.out.println ("enter fathr & gson
age ");

int fa= sc.next Int();
int sa = sc. next Int ();

Son s = new son(sa, fa);

try {

    S. Sonwrongage();

  } catch (exception e)
{  System.out.println(e);

 }
 }
 }
```

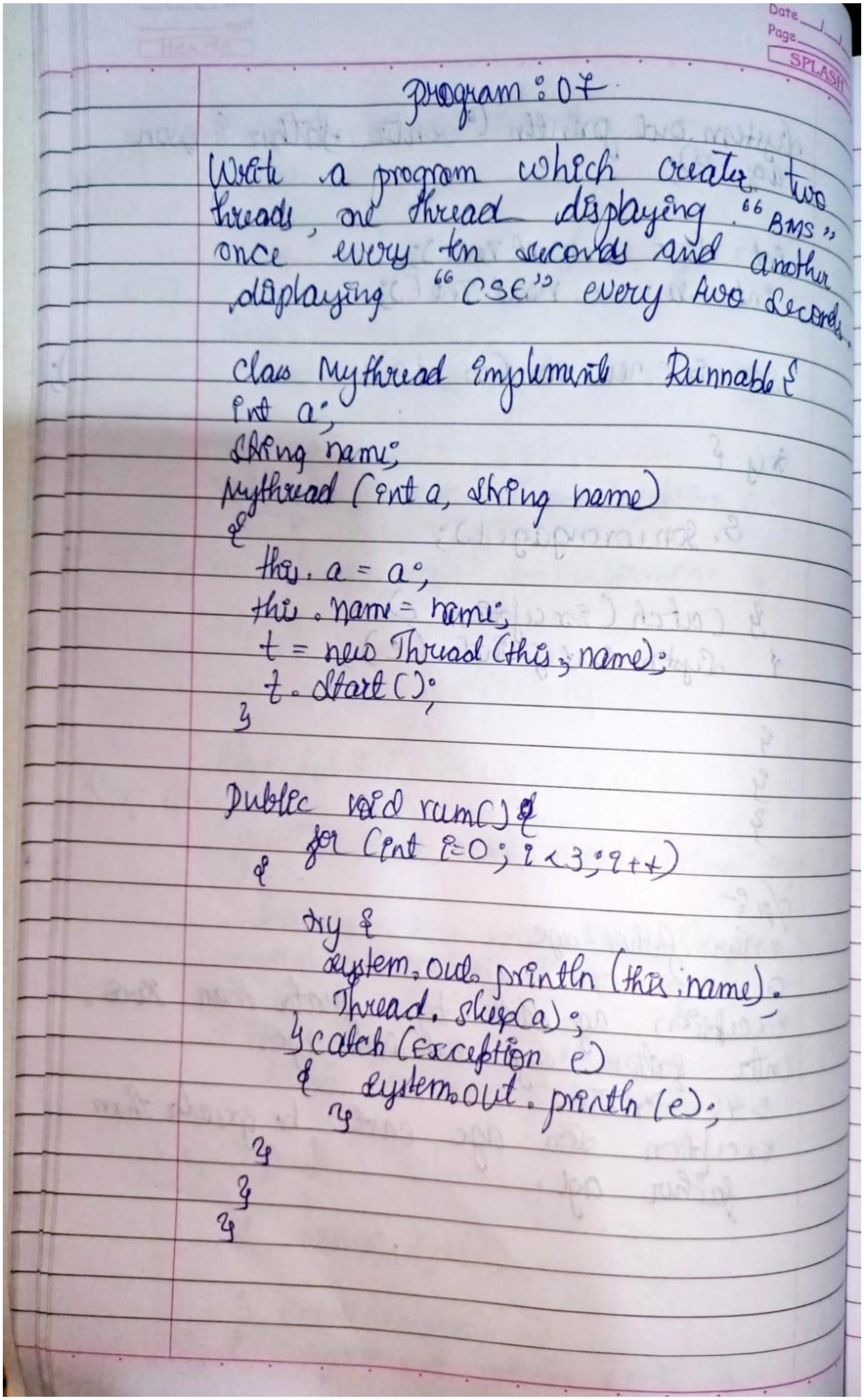
o/p:-

```
enter father age
0
Exception age shd be greater than zero.
enter father age and son age.
 34  45
exception son age cant be greater than
  father age.
```



```
C:\Users\BMSCECSEIL74\Desktop\144>javac Lab7.java

C:\Users\BMSCECSEIL74\Desktop\144>java Lab7
enter father age
0
exception  age shd be greater than zero
enter father age and son age
34 45
exception  son age is greater than father age

C:\Users\BMSCECSEIL74\Desktop\144>java Lab7
enter father age
5
enter father age and son age
78 45

C:\Users\BMSCECSEIL74\Desktop\144>
```

7. 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.

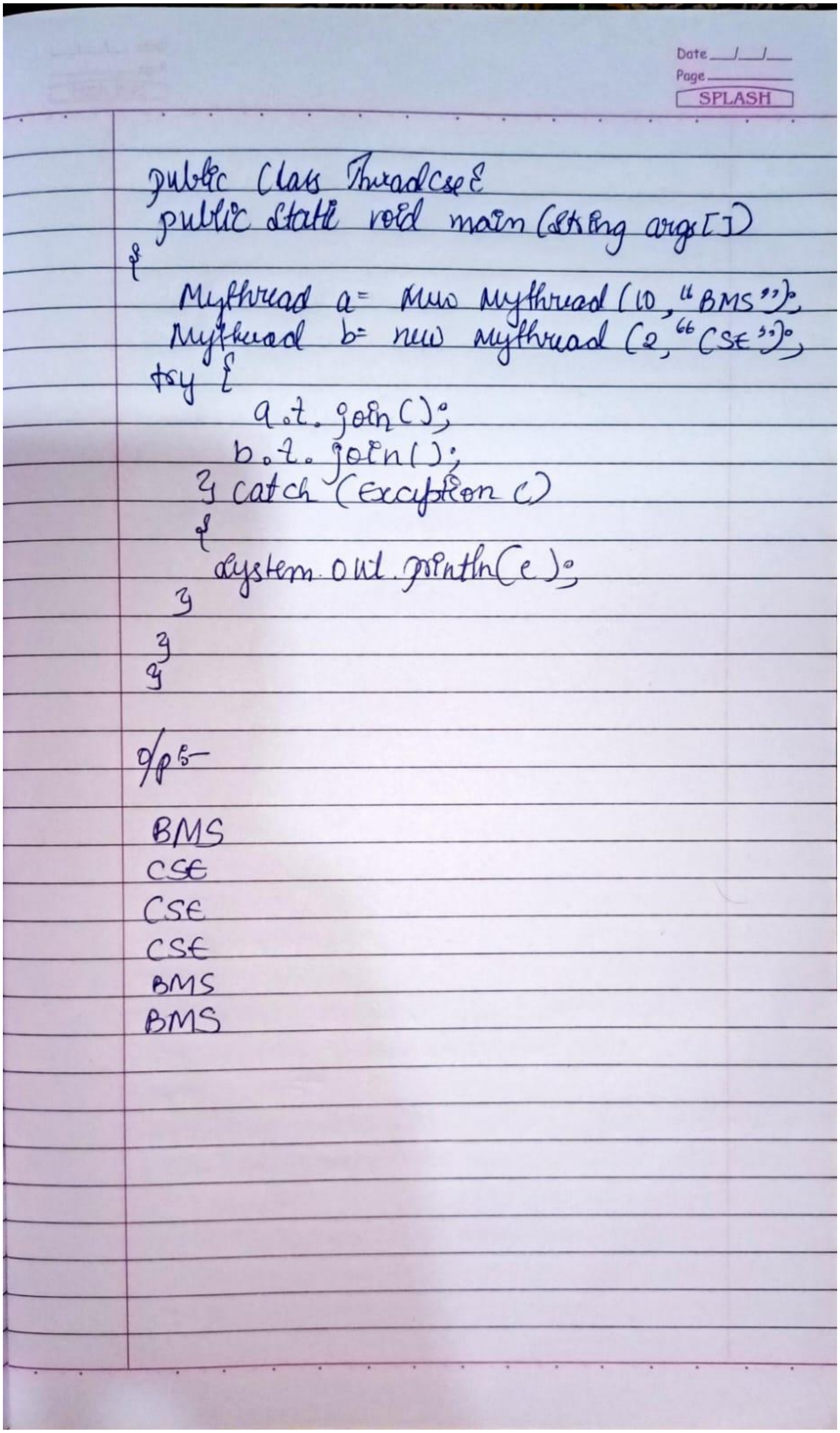


Date / /
Page

program : 07

Write a program which create two threads, one thread displaying "BMS" once every ten seconds and another displaying "CSE" every two seconds.

```
Class Mythread implements Runnable {
int a;
String name;
Mythread (int a, String name)
{
  this. a = a;
  this . name = name;
  t = new Thread (this, name);
  t. start ();
}

public void run() {
      for (int i=0; i<3; i++)
  {
      try {
        system. out. println (this.name).
         Thread. sleep(a);
      } catch (exception e)
       { system.out. println (e);
       }
  }
 }
}
```

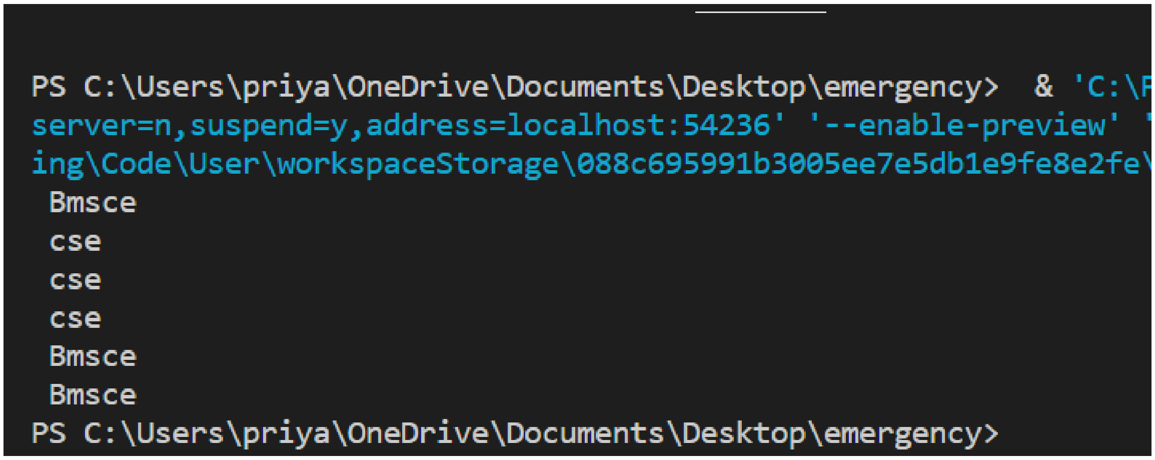


Date ___/___/___
Page ______
SPLASH

```
public Class Threadcse{
 public static void main(String args[])
{
  Mythread a= new Mythread(10,"BMS");
  Mythread b= new Mythread(2,"CSE");
 try {
       a.t.join();
      b.t.join();
    } catch (Exception e)
    {
     System.out.println(e);
   }
 }
}
```

o/p:-

```
BMS
CSE
CSE
CSE
BMS
BMS
```



```
PS C:\Users\priya\OneDrive\Documents\Desktop\emergency>  & 'C:\P
server=n,suspend=y,address=localhost:54236' '--enable-preview' '
ing\Code\User\workspaceStorage\088c695991b3005ee7e5db1e9fe8e2fe\
 Bmsce
 cse
 cse
 cse
 Bmsce
 Bmsce
PS C:\Users\priya\OneDrive\Documents\Desktop\emergency>
```