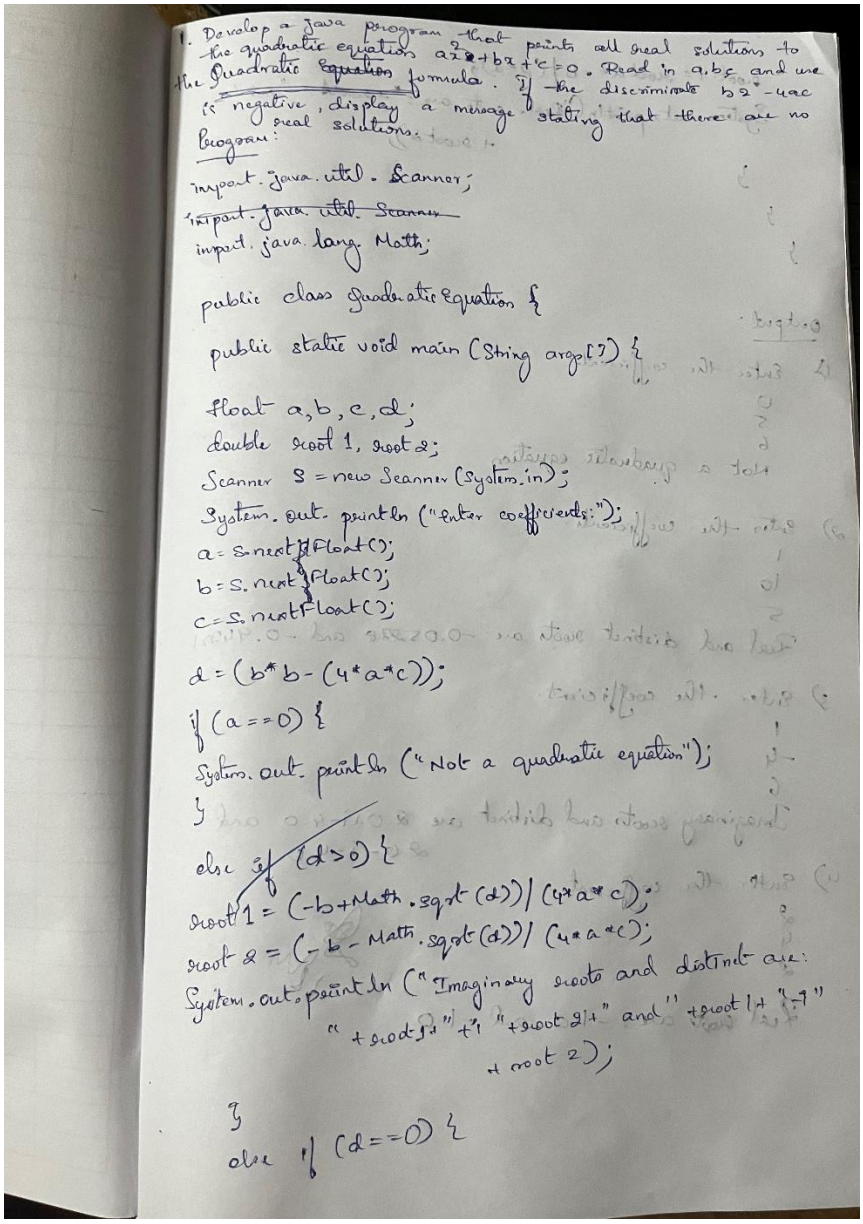
1.



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

Program:

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

public class QuadraticEquation {

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();
c= 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 ("Imaginary roots and distinct are:
" +root1+"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 coefficients:
   0
   5
   6
   Not a quadratic equation

2) Enter the coefficients:
   1
   10
   5
   Real and distinct roots are: -0.05278 and -0.94721

3) Enter the coefficients.
   1
   -4
   6
   Imaginary roots and distinct are 2.0+i-4.0 and
   2.0-i-4.0

4) Enter the coefficient
   2
   4
   2
   Real roots are: -1.0 and -1.0

Outputs:

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

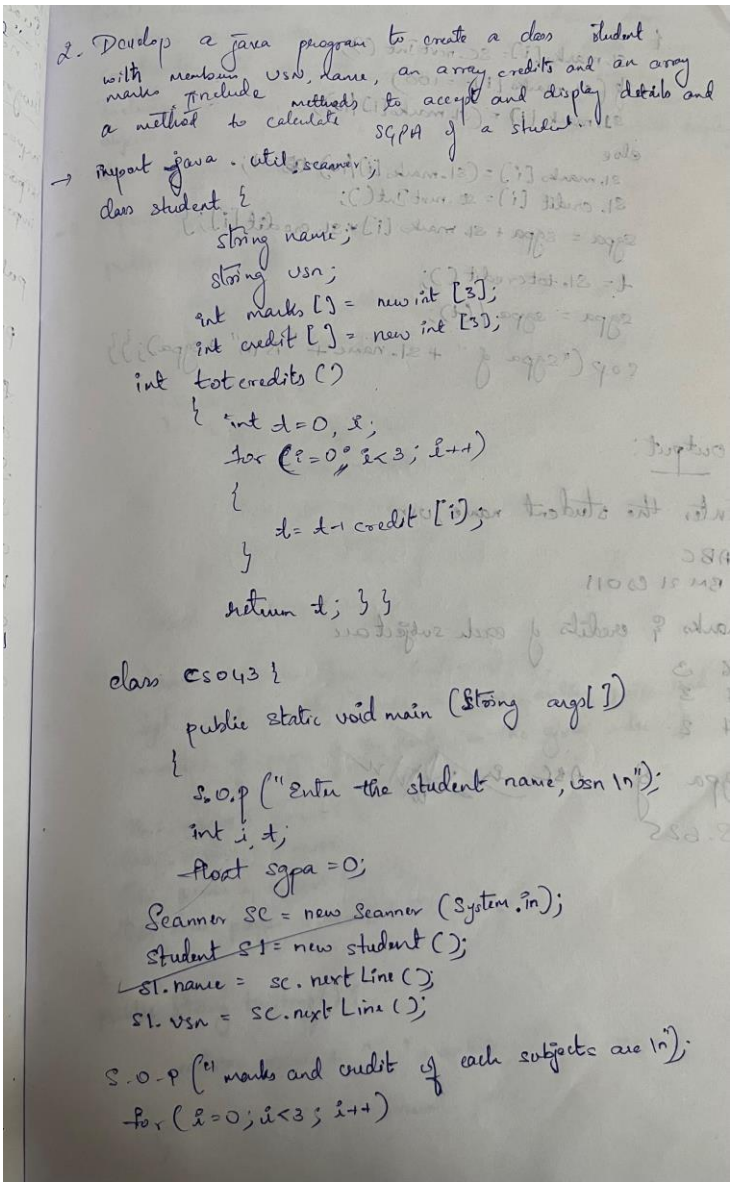
```
C:\Users\bmsce\Desktop>java QuadraticEquation
Enter coefficients:
1
10
5
Real and distinct roots are:-0.05278640450004204 and -0.947213595499958
```

```
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
```

2.

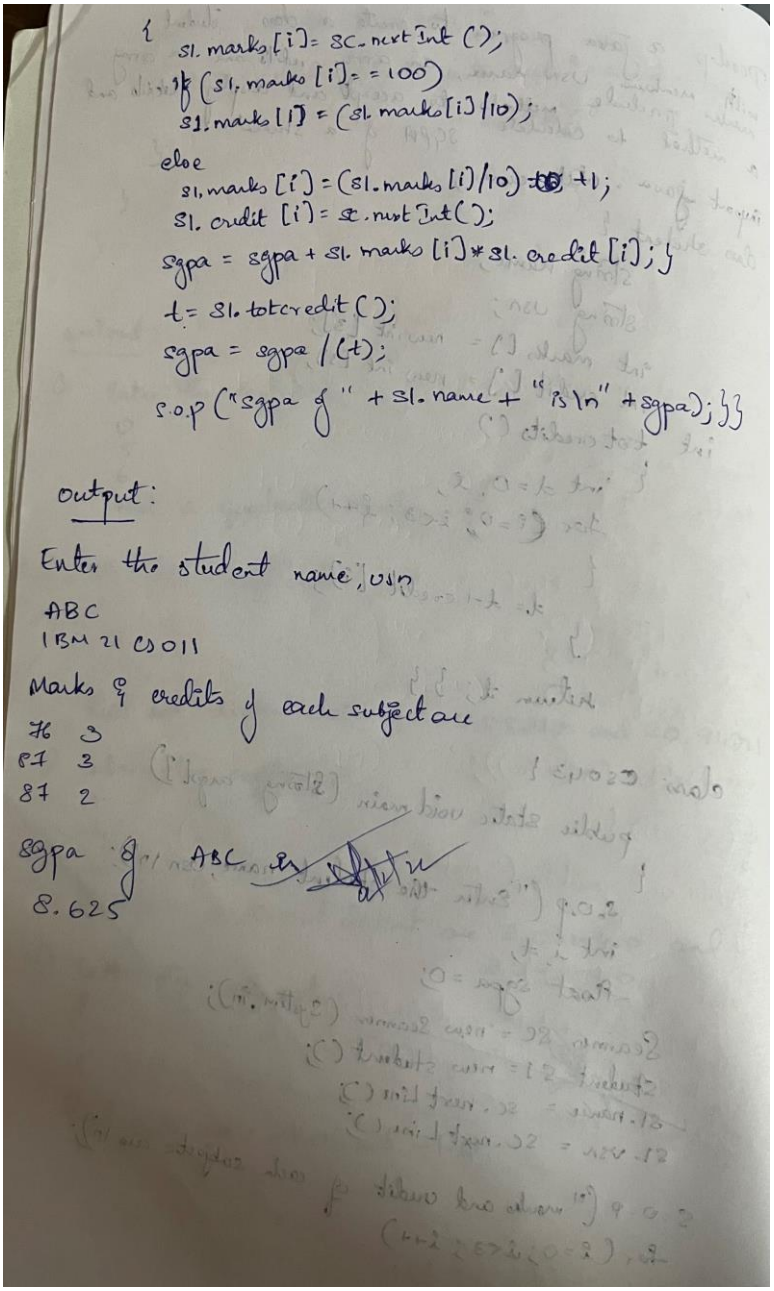
Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.



2. Develop a java program to create a class student with members usn, name, an array credits and an array marks. Include methods 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 [3];
    int credit[] = new int [3];
    int totcredits()
    { int t=0, i;
      for(i=0; i<3; i++)
      {
          t= t+ credit[i];
      }
      return t; } }

class cse043 {
    public static void main (String args[])
    {
      S.O.P("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();
S.O.P("marks and credit of each subjects are \n");
for (i=0; i<3; i++)
```



```
{
    sl.marks[i]= sc.nextInt();
    if(sl.marks[i]==100)
    sl.marks[i] = (sl.marks[i]/10);
    else
    sl.marks[i] = (sl.marks[i]/10) +1;
    sl.credit[i] = s.nextInt();
    sgpa = sgpa + sl.marks[i]*sl.credit[i];}
    t= sl.totcredit();
    sgpa = sgpa /(t);
    s.o.p("sgpa of " + sl.name + "is\n" +sgpa);}}
```

Output:

Enter the student name, usn

ABC

1BM21CS011

Marks & credits of each subject are

76 3

87 3

87 2

sgpa of ABC is

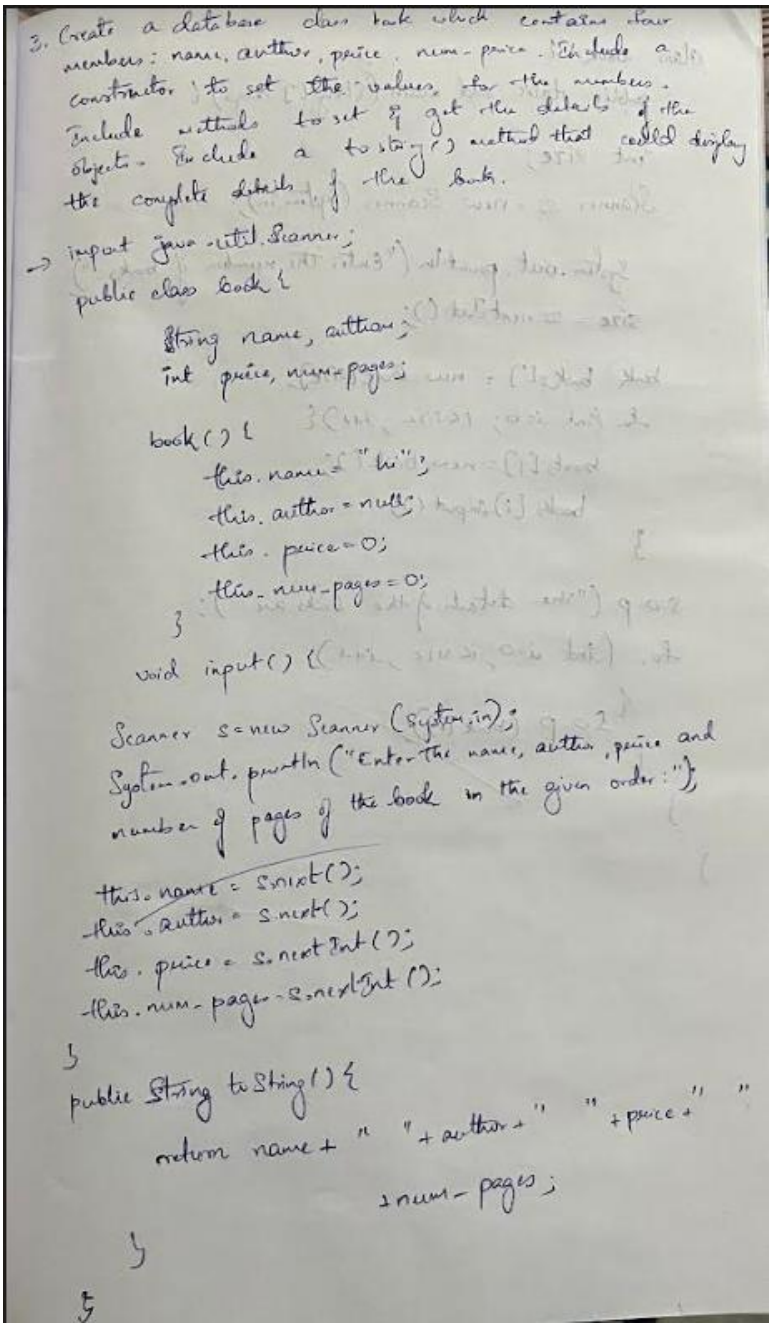
8.625

Output

```
C:\Users\bmsce\Desktop\1BM21CS043>java cs043
Enter the student name,usn

abc
1BM21CS011
Marks and credit of each subject are

76 3
87 3
87 2
sgpa ofabcis
8.625
```

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.



```
3. Create a database 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 & get the details of the
objects. Include a to string() method that could display
the complete details of the book.

→ 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 book12
    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 (int i=0; i<size; i++){
        books[i] = new book ();
        books[i].input ();
    }
S.O.P ("The details of the books are:");
for (int i=0; i<size; i++)
    {S.O.P (book[i]);
    }
}
}
```

output:

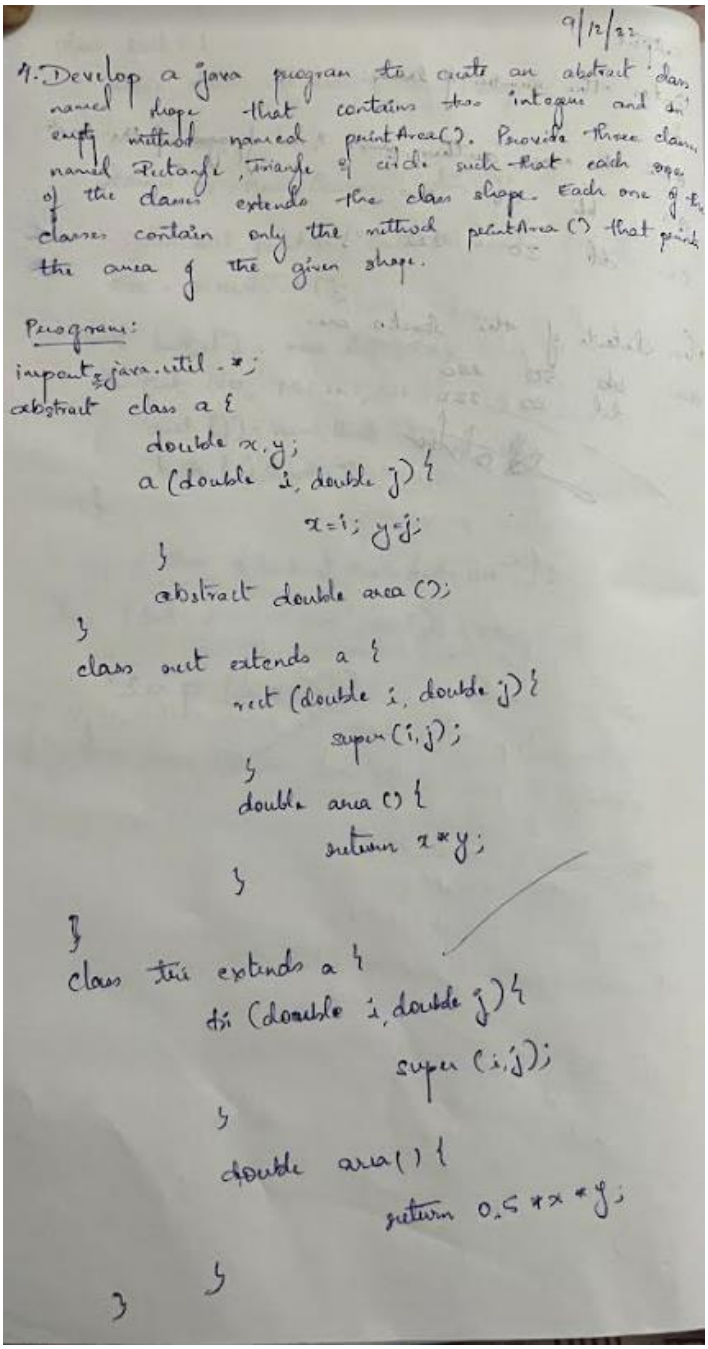
```
Enter the number of books:
2
Enter the name, author, price & no. of pages of the book
aa  bb  50  200
cc  dd  30  222

The details of the books are:
aa  bb  50  200
cc  dd  30  222
```

Output

```
C:\Users\bmsce\Desktop\1BM21CS043>java book1
Enter the number of books:
2
Enter the name, author, price and number of pages of the book in the given order:
aa bb 290 222
Enter the name, author, price and number of pages of the book in the given order:
cc dd 322 432
The details of the book are:
aa bb 290 222
cc dd 322 432
```

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.



9/12/22

1. 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 & 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:

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

```
class cir extends a {
        cir (double i) {
            super (i,i);
        }
        double area() {
            System.out.println(3.14*r*r);
        }
}
class area {
    psvm (String args[]) {
    Scanner sc=new Scanner (System.in);
    System.out.println("Enter the length and breadth of
                        rectangle : ");
    double l=sc.nextInt();
    double b=sc.nextInt();
    s.o.p ("Enter height and base of triangle: ");
    double h=sc.nextInt();
    double ba=sc.nextInt();
    s.o.p ("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);
    s.o.p ("Area of rectangle is" +r.area());
    s.o.p ("Area of triangle is" +t.area());
    s.o.p ("Area of circle is" +c.area());
    }
}
```

Output:

```
Enter the length and breadth of rectangle:
23
43
Enter the length height and base of triangle:
65
34
Enter the radius of circle:
65
Area of rectangle is 989.0
Area of triangle is 1105.0
Area of circle is 13266.5
```

Output

```
C:\Users\bmsce\Desktop\1BM21CS043>java area
Enter the length and breadth of rectangle:
20 43
Enter the height and base of triangle:
33 76
Enter the radius of circle:
44
Area of rectangle is860.0
Area of triangle is1254.0
Area of circle is6079.04
```

5.

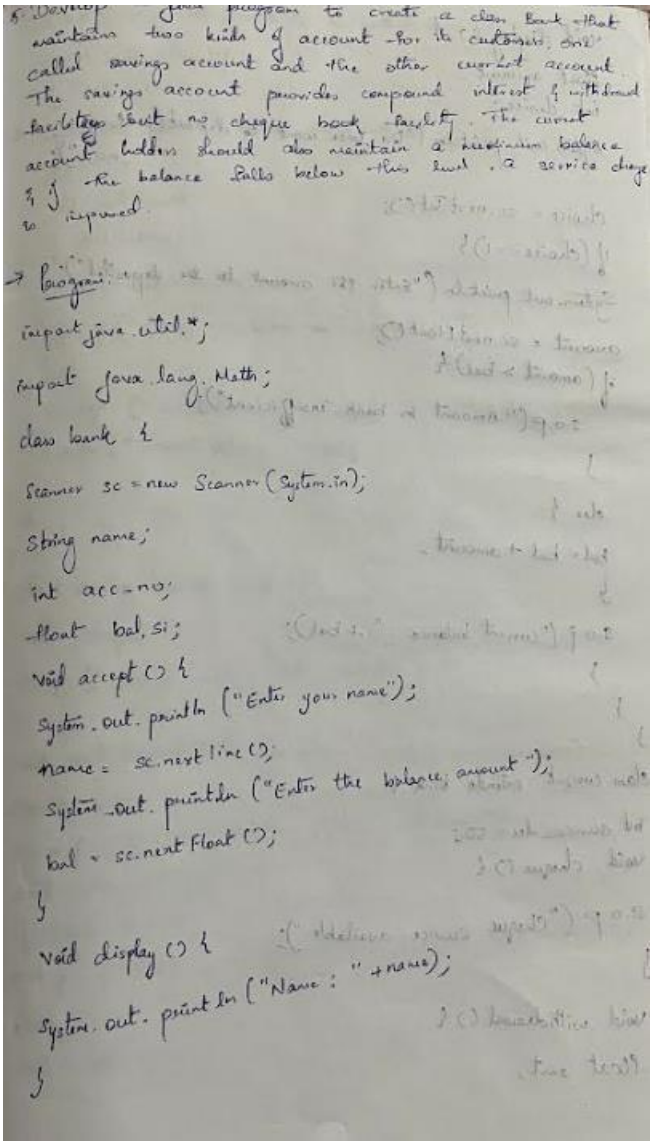
Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book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

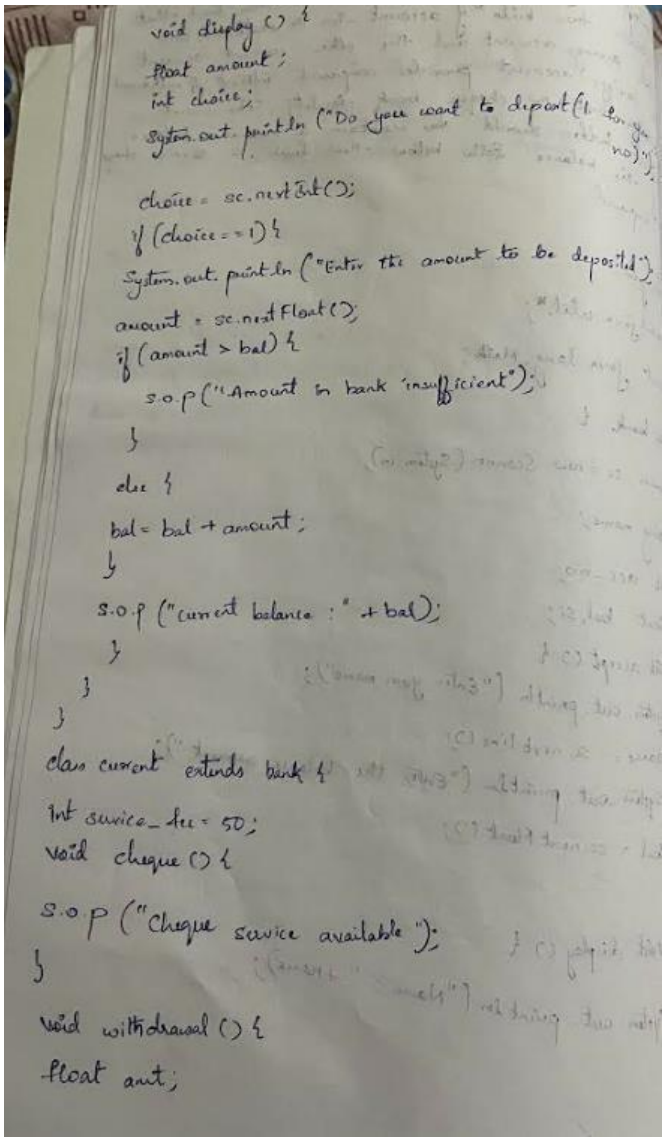
b) Display the balance.

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.



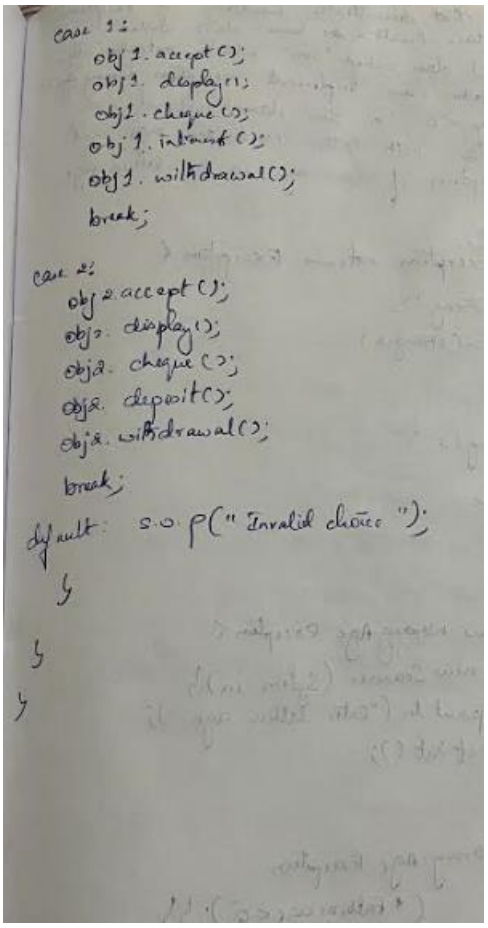
```
5. Develop a Java program to create a class Bank that
maintains two kinds of account for its customers, one
called savings account and the other current account.
The savings account provides compound interest & withdrawal
facilities but no cheque book facility. The current
account holders should also maintain a minimum balance
& if the balance falls below this level, a service charge
is imposed.

→ Program:

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 display () {
    float amount;
    int choice;
    System.out.println ("Do you want to deposit(1 for yes, 0 for no)");
    choice = sc.nextInt();
    if (choice == 1) {
    System.out.println ("Enter the amount to be deposited");
    amount = sc.nextFloat();
    if (amount > bal) {
      s.o.p ("Amount in bank insufficient");
    }
    else {
    bal = bal + amount;
    }
    s.o.p ("current balance : " + bal);
    }
  }
}
class current extends bank {
int service_fee = 50;
void cheque () {
s.o.p ("Cheque service available");
}
void withdrawal () {
float amt;
```

```
s.o.p ("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;
  s.o.p ("50 rs is taken as service fee");
}
s.o.p ("Withdrawn : " + amt);
s.o.p ("Current Balance : " + bal);
}
}
}
class saving extends bank {
void cheque () {
s.o.p ("Cheque service not available");
}
void withdrawal () {
float amt;
s.o.p ("Enter the amount to be withdrawn");
amt = sc.nextFloat();
if (amt > bal)
s.o.p ("Balance insufficient");
else
bal = bal - amt;
```

```
s.o.p ("withdrawn: " + amt);
s.o.p ("current balance: " + bal);
}
void interest () {
s.o.p ("Enter the rate of interest");
int r = sc.nextInt ();
s.o.p (" Enter the no. of times interest applied per time period");
int n = sc.nextInt ();
s.o.p ("Enter the time elapsed");
int t = sc.nextInt ()
si = bal * (1 + (r/n));
s.o.p (" compound interest is " + (Math.pow (si, n*t)));
}
}
ublic class account {
psvm (String args []) {
Scanner sc = new Scanner (System.in);
savings obj1 = new savings ();
current obj1 = new current ();
o.p ("In 1. savings account In 2. current account");
choice = sc.nextInt ();
tch (choice) {
```



```
case 1:
    obj1.accept();
    obj1.display();
    obj1.cheque();
    obj1.interest();
    obj1.withdrawal();
    break;

case 2:
    obj2.accept();
    obj2.display();
    obj2.cheque();
    obj2.deposit();
    obj2.withdrawal();
    break;
default: s.o.p("Invalid choice");
  }
}
}
```

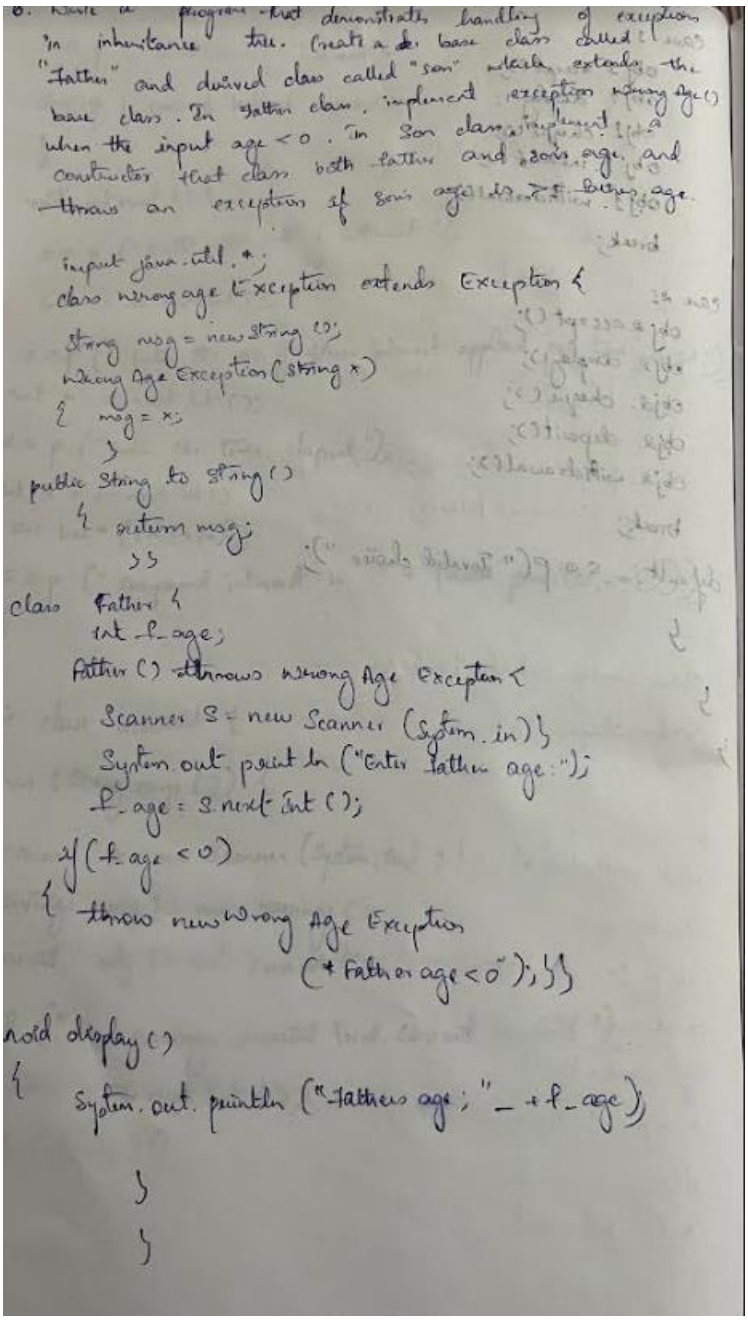
Ouput

```
1.Savings account
2.Current account
2
Enter your name
ABC
Enter the balance amount
34000
Name : ABC
Cheque service available
Do you want to deposit(1 for yes ,2 for no)
1
Enter the amount to be deposited
45000
Amount in bank insufficient
Current balance : 34000.0
Enter the amount to be withdrawn
200
Withdrawn : 200.0
Current balance : 33800.0
```

```
1.Savings account
2.Current account
1
Enter your name
ABC
Enter the balance amount
25000
Name : ABC
Cheque service not available
Do you want to deposit(1 for yes ,2 for no)
1
Enter the amount to be deposited
43000
Amount in bank insufficient
Current balance : 25000.0
Enter the rate of interest
4
Enter the number of times interest applied per time period
45
Enter the time elapsed
33
Compound interest is Infinity
Enter the amount to be withdrawn
7000
Withdrawn : 7000.0
Current balance : 18000.0
```

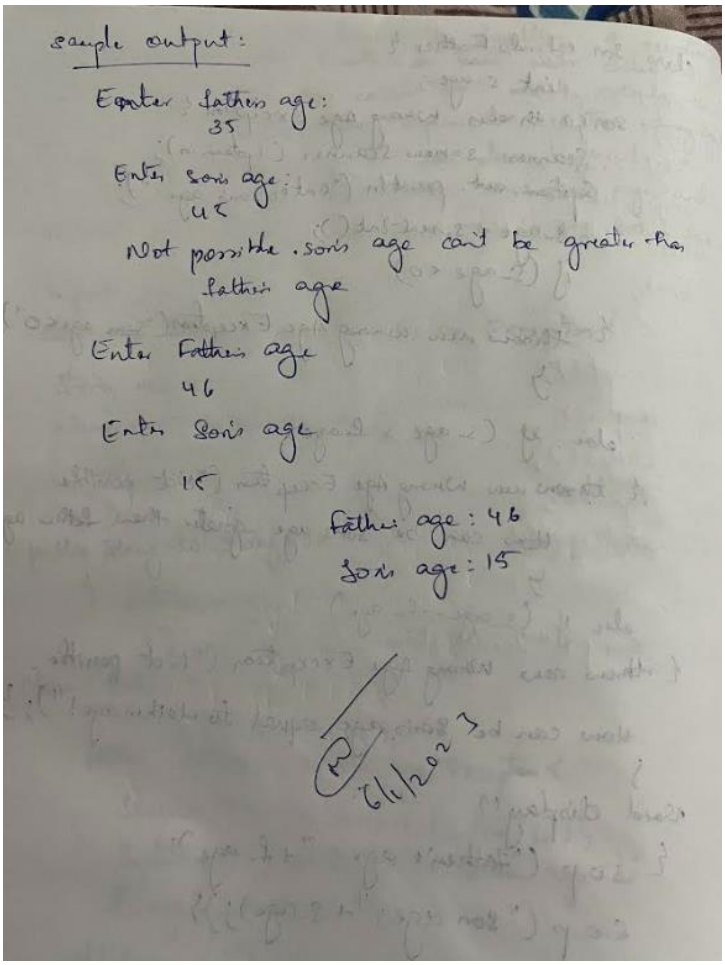
6.

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.



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

input java.util.*;
class wrongAge Exception extends Exception {
String msg = new String ();
WrongAge Exception(String x)
{ msg = x;
}
public String to string()
{ return msg;
}}
class Father {
int f_age;
Father () throws wrong Age Exception {
Scanner S = new Scanner (System.in)};
System out.println ("Enter father age:");
f_age = S.nextInt ();
if (f_age < 0)
{ throw new Wrong Age Exception
("Father age < 0"); }}
void display()
{
System.out.println ("Fathers age; "_ + f_age);
}
}
```

```
class Son extends Father {
        int s_age;
    Son() throws WrongAgeException {
      Scanner s=new Scanner(System.in);
      System.out.println("Enter Son's age:");
       s_age = s.nextInt();
       if (s_age<0)
    { throw new WrongAgeException("Son age<0");
      }
      else if (s_age > f_age)
    { throw new WrongAgeException("Not possible
        How can be Son's age greater than Father age!");
      }
    else if (s_age==f_age)
  { throw new WrongAgeException("Not possible.
    How can be Son's age equal to Father age!"); }
    }
  void display()
   { S.O.P("Father's age : "+f_age);
     S.O.P("Son age : "+s_age);}}
  class Except{
          Sparam (String[] args)
      { try
       { Father f=new Father();
         f.display();
        Son s= new Son();
         s.display();
        }
  Catch (WrongAgeException wae)
      { S.O.P (wae);
        }}}
```

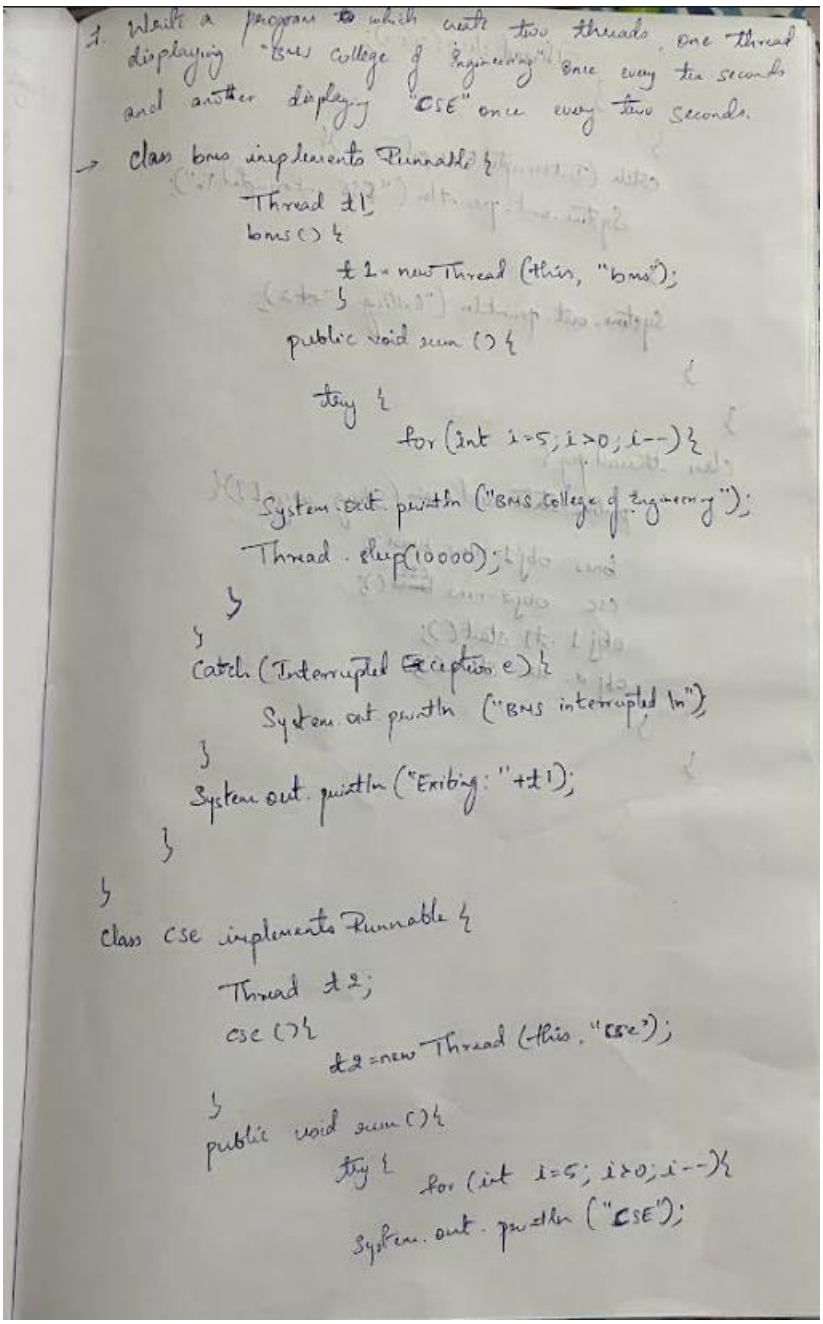


Output:

```
Enter the father's age:35
Enter the son's age:45
Son's age is more than Father's age
```

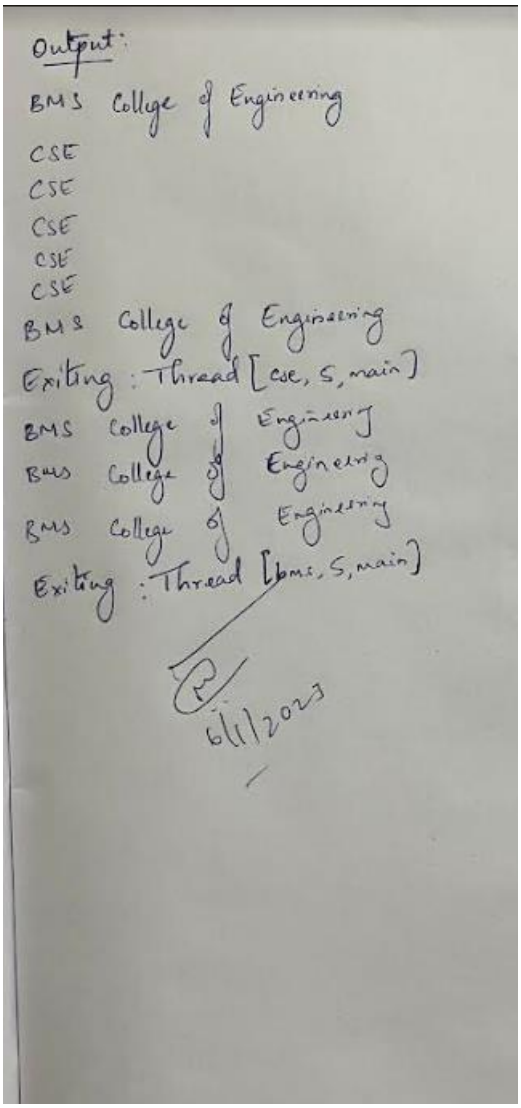
7.

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.



```
1. Write a program to which creates two threads, one thread
displaying "BMS College of Engineering" once every ten seconds
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 (int i=5; 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[]){
        bmis obj1=new bmis();
        cse obj2=new cse();
        obj1.t1.start();
        obj2.t2.start();
    }
}
```

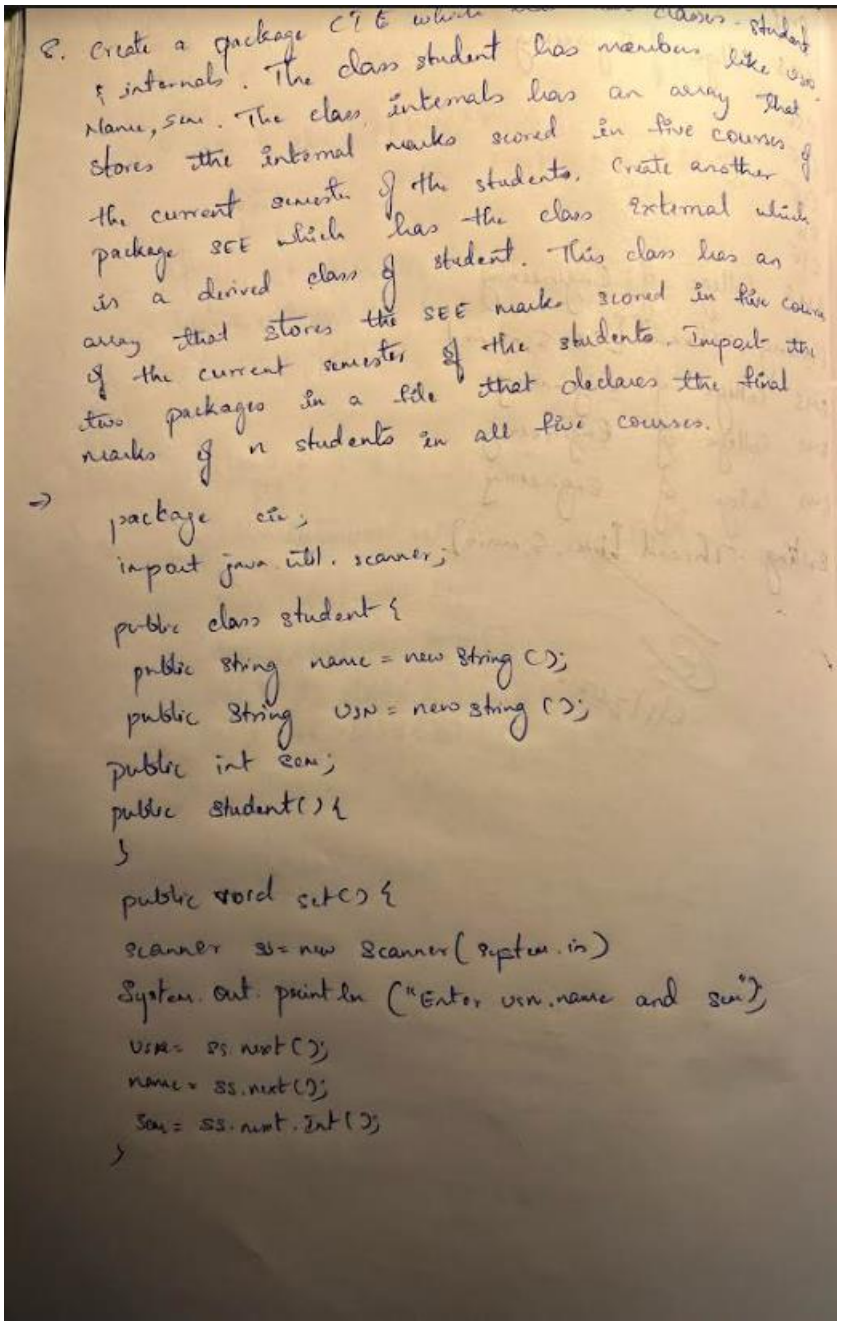

Output:
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
Exiting : Thread [cse, 5, main]
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
Exiting : Thread [bms, 5, main]
6/1/2023

Output:

```
C:\Users\bmsce\Desktop>java threadprg
BMSCE
CSE
CSE
BMSCE
BMSCE
CSE
CSE
BMSCE
CSE
BMSCE
Exiting: Thread[BMSCE,5,main]
Exiting: Thread[CSE,5,main]
```

8.

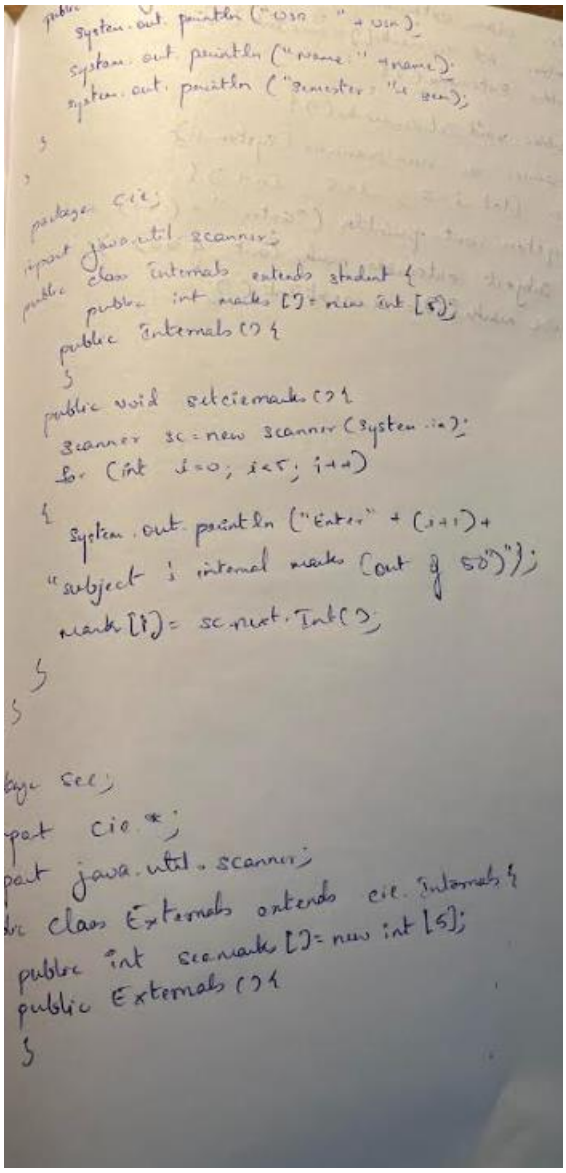
Create a package CIE which has two classes- Student and Internals- a subclass of Student. The class Student has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which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



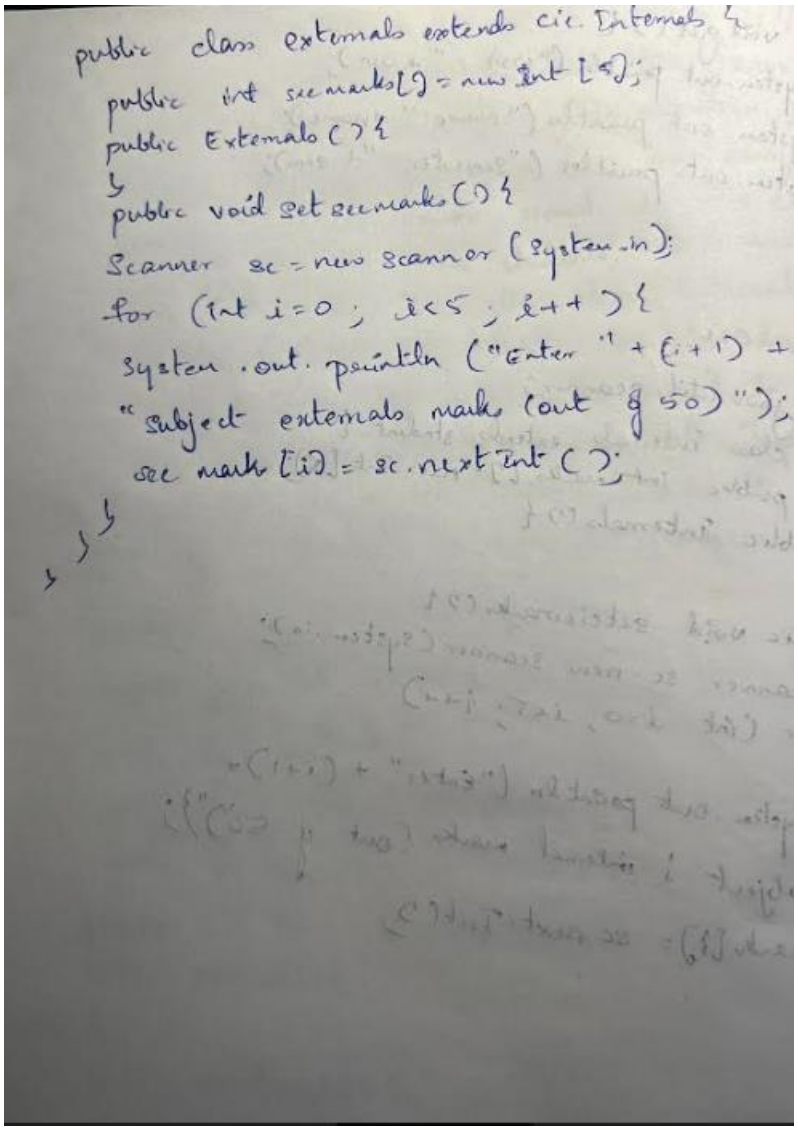
8. Create a package CIE which has two classes - Student & internals. The class student has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the students. Create another package SEE which has the class external 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 students. Import the two packages in a file that declares the final marks of n students in all five courses.

→

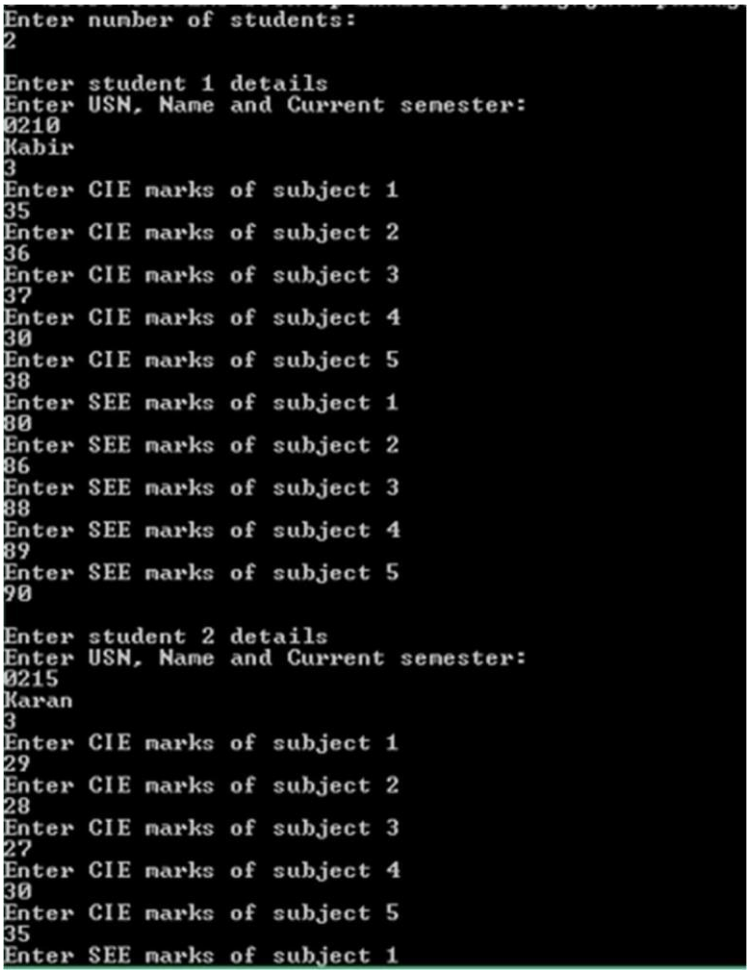
```
package cie;
import java.util.scanner;
public class student {
 public string name = new String();
 public String usn = new string();
public int sem;
public student() {
}
public void set() {
scanner ss = new Scanner(System.in)
System.out.println("Enter usn, name and sem");
usn = ss.next();
name = ss.next();
sem = ss.next.Int();
}
```



```
        System.out.println("USN  " + usn);
        System.out.println("Name: " + name);
        System.out.println("Semester: " + sem);
    }
}

package cie;
import java.util.Scanner;
public class Internals extends Student {
    public int marks[] = new int[5];
    public Internals() {
    }
    public void setCIEmarks() {
    Scanner sc = new Scanner(System.in);
    for (int i=0; i<5; i++)
    {
        System.out.println("Enter" + (i+1) +
    "subject's internal marks (out of 50)");
        marks[i] = sc.nextInt();
    }
}

ge see;
pat cie.*;
pat java.util.Scanner;
lic class Externals extends cie.Internals {
public int seemarks[] = new int[5];
public Externals() {
}
```



```
public class extemarks extends cie.Intemarks {
  public int see marks[] = new int [5];
  public Extemarks () {
  }
  public void set see marks () {
  Scanner sc = new scanner (System.in);
  for (int i=0 ; i<5 ; i++ ) {
   System.out.println ("Enter " + (i+1) +
   " subject extemarks marks (out of 50) ");
    see mark [i] = sc.nextint ();
  }
}
```


Enter number of students:
2
Enter student 1 details
Enter USN, Name and Current semester:
0210
Kabir
3
Enter CIE marks of subject 1
35
Enter CIE marks of subject 2
36
Enter CIE marks of subject 3
37
Enter CIE marks of subject 4
30
Enter CIE marks of subject 5
38
Enter SEE marks of subject 1
80
Enter SEE marks of subject 2
86
Enter SEE marks of subject 3
88
Enter SEE marks of subject 4
89
Enter SEE marks of subject 5
90
Enter student 2 details
Enter USN, Name and Current semester:
0215
Karan
3
Enter CIE marks of subject 1
29
Enter CIE marks of subject 2
28
Enter CIE marks of subject 3
27
Enter CIE marks of subject 4
30
Enter CIE marks of subject 5
35
Enter SEE marks of subject 1

```
Enter student 2 details
Enter USN, Name and Current semester:
0215
Karan
3
Enter CIE marks of subject 1
29
Enter CIE marks of subject 2
28
Enter CIE marks of subject 3
27
Enter CIE marks of subject 4
30
Enter CIE marks of subject 5
35
Enter SEE marks of subject 1
32
Enter SEE marks of subject 2
31
Enter SEE marks of subject 3
21
Enter SEE marks of subject 4
29
Enter SEE marks of subject 5
20

Details of student 1

Student Details
Name: Kabir
USN: 0210
Semester: 3
Total marks in subject 1: 115
Total marks in subject 2: 122
Total marks in subject 3: 125
Total marks in subject 4: 119
Total marks in subject 5: 128


Details of student 2

Student Details
Name: Karan
USN: 0215
Semester: 3
Total marks in subject 1: 111
Total marks in subject 2: 109
Total marks in subject 3: 118
Total marks in subject 4: 109
Total marks in subject 5: 125


C:\Users\STUDENT\Desktop\1bm1cs034\packg>
```