OBJECT ORIENTED JAVA LAB RECORD

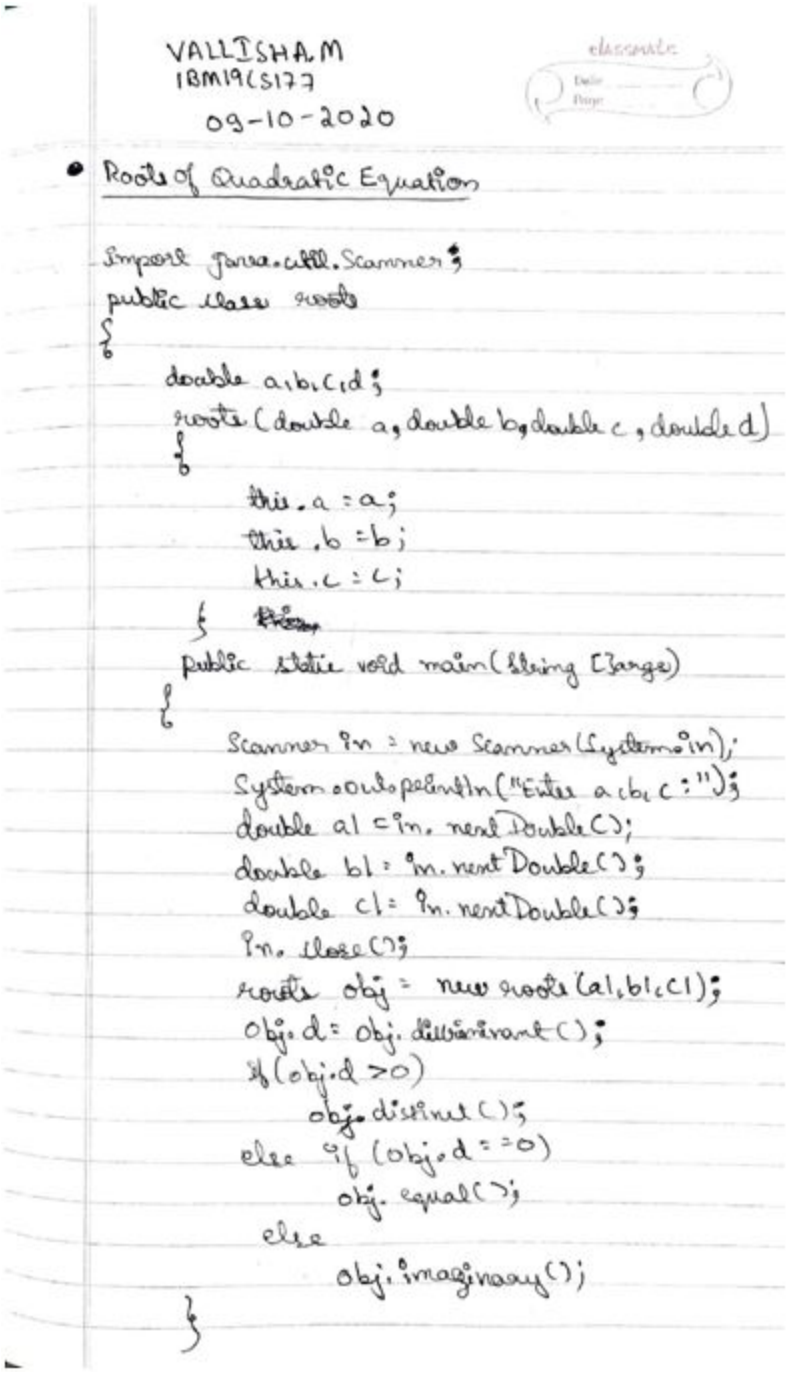
NAME :- VALLISHA M

USN :- 1BM19CS177

SECTION :- 3D

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 formula. If the discriminant b^2-4ac is negative, display a message stating that there are no real solutions.

Writeup;



VALLISHAM
1BM19CS177
09-10-2020

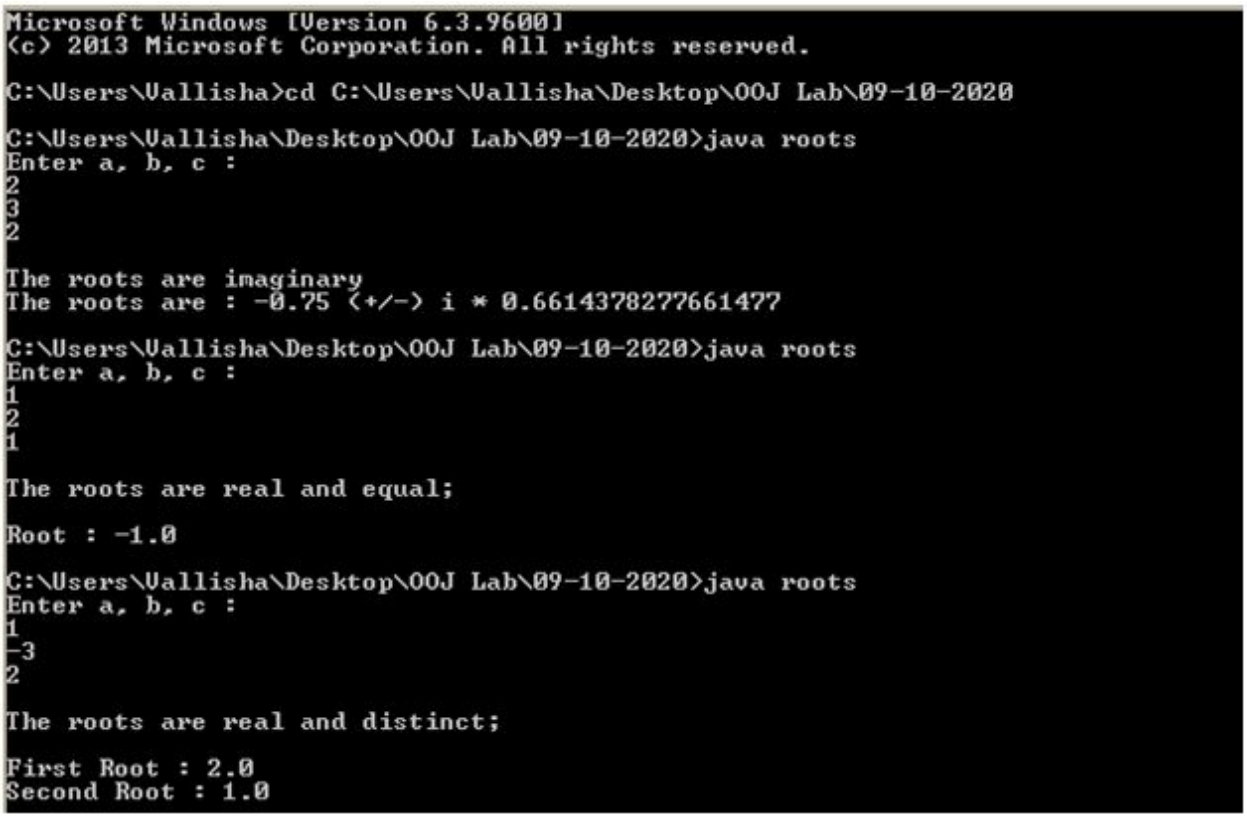
• Roots of Quadratic Equation

```
import java.util.Scanner;
public class roots
{
    double a,b,c,d;
    roots(double a, double b, double c, double d)
    {
        this.a = a;
        this.b = b;
        this.c = c;
    }
    public static void main(String [args)
    {
        Scanner in = new Scanner(System.in);
        System.out.println("Enter a,b,c :");
        double a1 = in.nextDouble();
        double b1 = in.nextDouble();
        double c1 = in.nextDouble();
        in.close();
        roots obj = new roots(a1,b1,c1);
        obj.d = obj.discriminant();
        if(obj.d >0)
            obj.distinct();
        else if (obj.d ==0)
            obj.equal();
        else
            obj.imaginary();
    }
```

```
public double discriminant()
{
    return (b*b - Math.sqrt(4*a*c));
}
public void distinct()
{
    double x = Math.sqrt(d);
    double r1 = (-1*b + x)/(2*a);
    double r2 = (-1*b - x)/(2*a);
    System.out.println("\n The roots are
    real and distinct;\n");
    System.out.println("First root : "+r1);
    System.out.println("Second root : "+r2);
}
public void equal()
{
    double x = Math.sqrt(d);
    double r1 = (-1*b + x)/(2*a);
    System.out.println("\n The roots are
    real and equal;\n");
    System.out.println("Root : "+r1);
}
public void imaginary()
{
    System.out.println("\n The roots are
    imaginary");
    double x = Math.Sqrt(-d);
    double r = (-b)/(2*a);
    double i = Math.abs(x/(2*a));
```

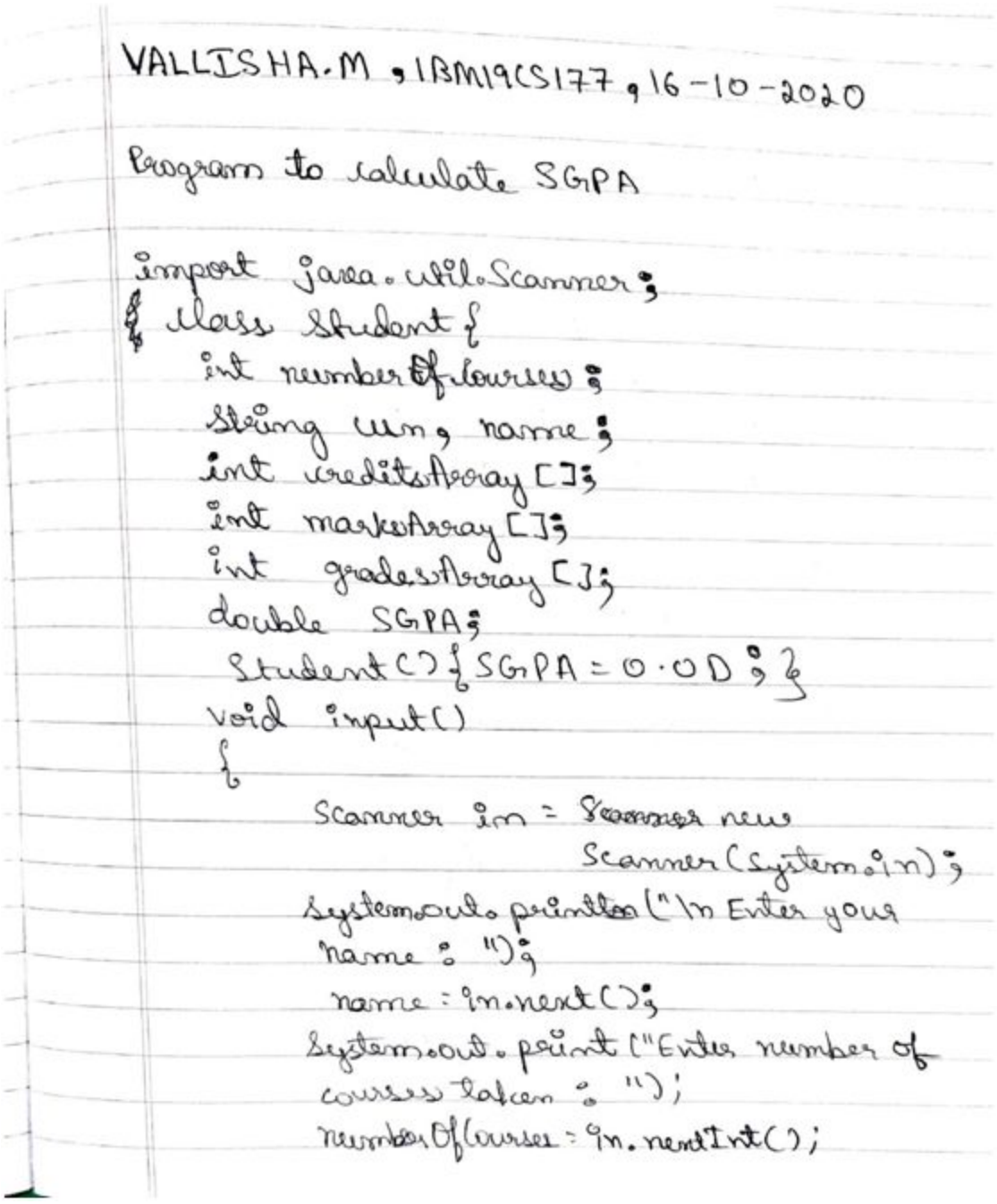
```
if(r==0) r=0;
System.out.println("The roots are :" +r+"
(+|-) i + " + i +");
}
}
```

Output;


Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
C:\Users\Vallisha>cd C:\Users\Vallisha\Desktop\OOJ Lab\09-10-2020
C:\Users\Vallisha\Desktop\OOJ Lab\09-10-2020>java roots
Enter a, b, c :
2
3
2
The roots are imaginary
The roots are : -0.75 (+/-) i * 0.6614378277661477
C:\Users\Vallisha\Desktop\OOJ Lab\09-10-2020>java roots
Enter a, b, c :
1
2
1
The roots are real and equal;
Root : -1.0
C:\Users\Vallisha\Desktop\OOJ Lab\09-10-2020>java roots
Enter a, b, c :
1
-3
2
The roots are real and distinct;
First Root : 2.0
Second Root : 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 anda method to calculate SGPA of a student.

Writeup;



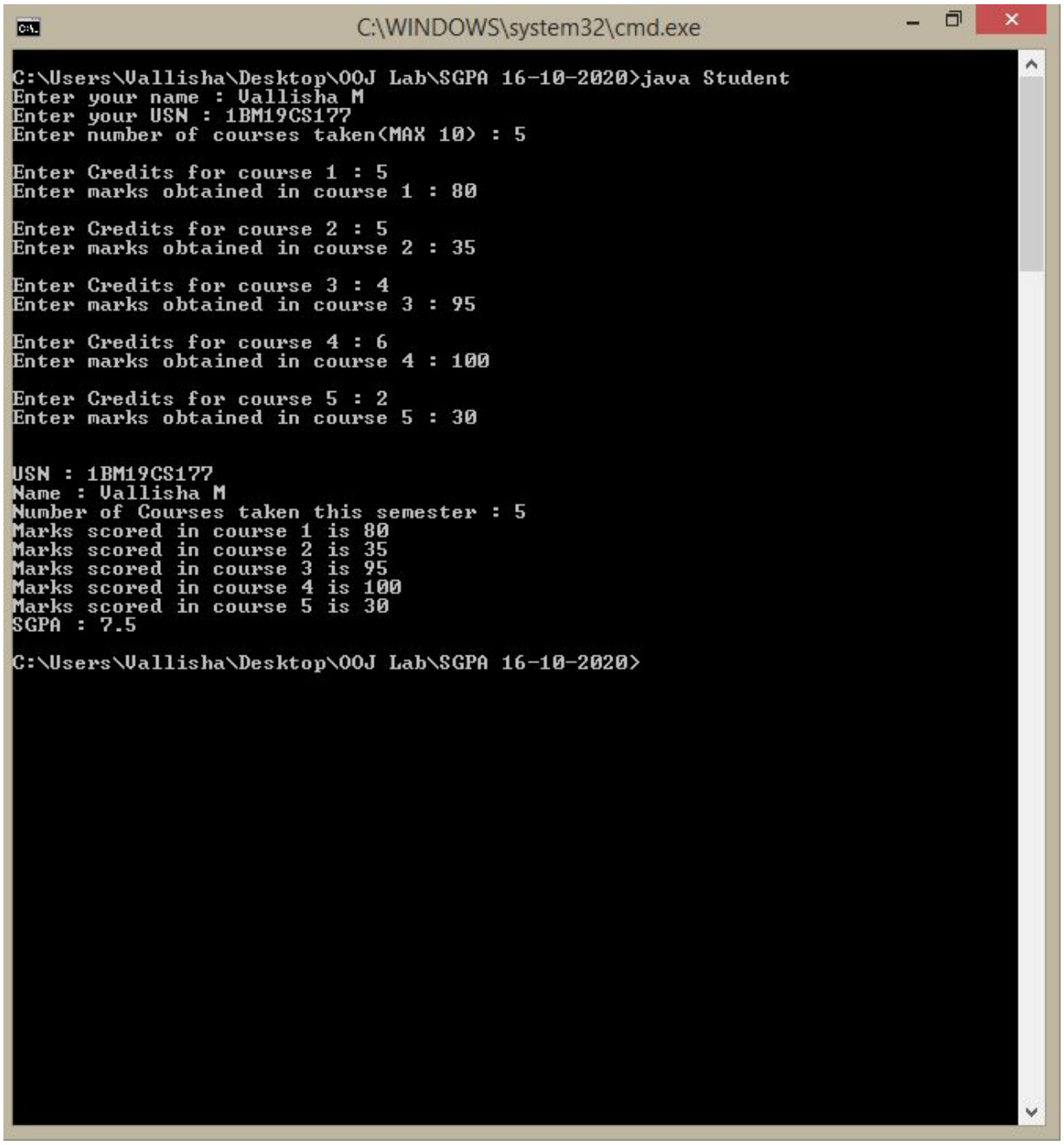
```
VALLISHA.M , 1BM19CS177 , 16-10-2020

Program to calculate SGPA

import java.util.Scanner;
class Student{
    int numberOfCourses;
    String usn, name;
    int creditsArray[];
    int marksArray[];
    int gradesArray[];
    double SGPA;
    Student(){SGPA = 0.0D;}
    void input()
    {
        Scanner in = new
                     Scanner(System.in);
        System.out.println("\n Enter your
        name : ");
        name : in.next();
        System.out.print("Enter number of
        courses taken : ");
        numberOfCourses = in.nextInt();
```

```
System.out.print("Enter your usn : ");
usn = in.next();
marksArray = new int[numberOfCourses];
creditsArray = new int[numberOfCourses];
gradesArray = new int[numberOfCourses];

int i = 0;
System.out.println();
for(i = 0; i < numberOfCourses; i++)
{
    System.out.print("Enter credits for
    course " + (i+1) + " : ");
    creditsArray[i] = in.nextInt();
    System.out.print("Enter marks obtained
    in " + (i+1) + " : ");
    marksArray[i] = in.nextInt();
}
in.close();
}
void computeGradesArray()
{
int i = 0;
for(i=0; i< numberOfCourses; i++)
{
    if(marksArray[i] == 10)
    {
        gradesArray[i] = 10;
        continue;
    }
```

```
if (marksArray[i] >= 50)
{
    gradesArray[i] = (marksArray[i]/10) + 1;
    continue;
}
if (marksArray[i] >= 35)
{
    gradesArray[i] = 4;
    continue;
}
gradesArray[i] = 0;
}
}
void computeSGPA()
{
    int i = 0;
    int netCredits = 0;
    for(i = 0; i < numberOfCourses; i++)
    {
      SGPA = SGPA + creditsArray[i] * gradesArray[i];
      netCredits = netCredits + creditsArray[i];
    }
    SGPA = SGPA/netCredits;
}
```

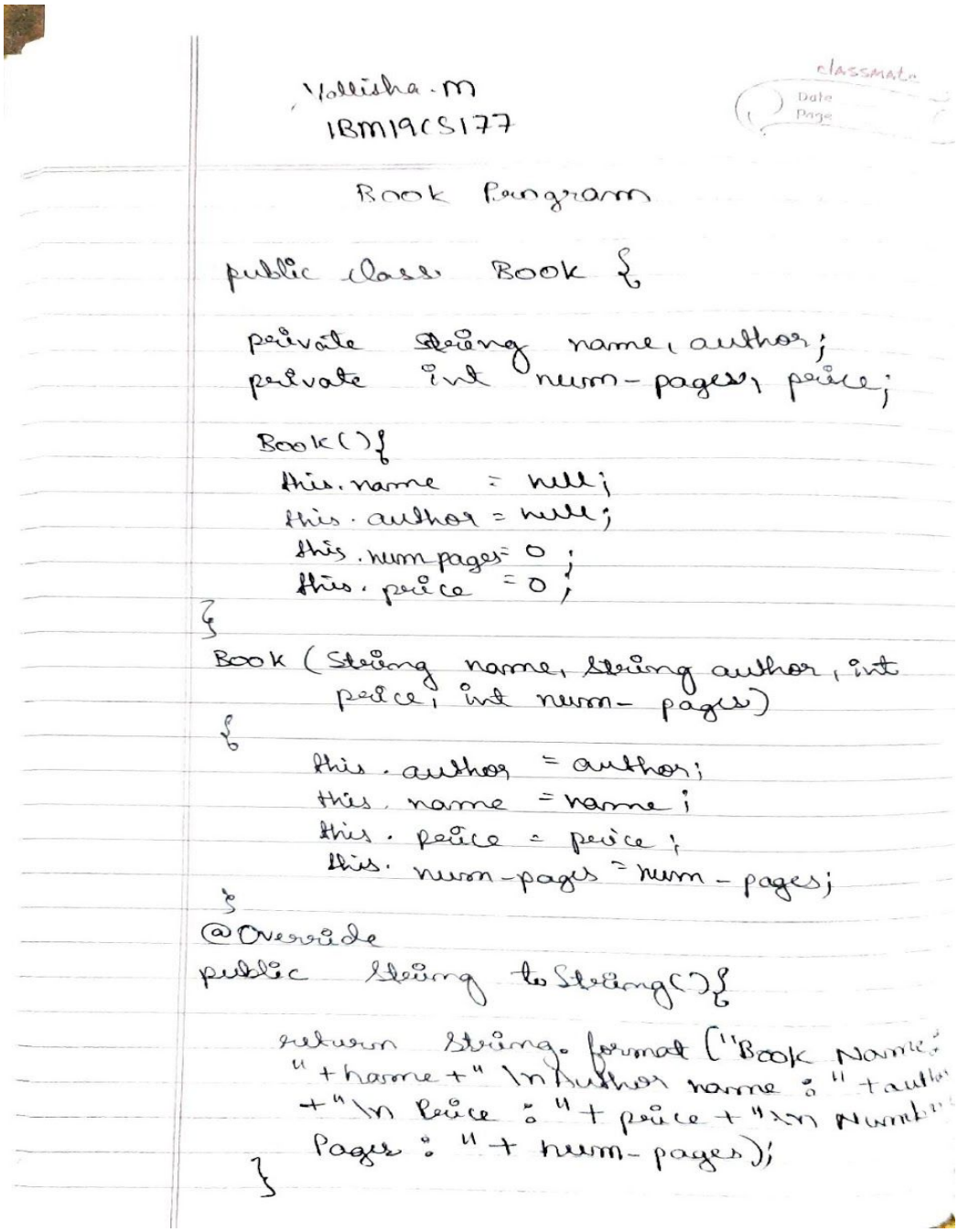
```
void display()
{
    System.out.println();
    System.out.print("USN : " +usn);
    System.out.println("Name : " +name);
    System.out.println("Number of courses
    taken this semester : " +numberOfCourses);
    int i=0;
    for(i=0; i<numberOfCourses; i++)
      System.out.println("Marks scored
      in course " +(i+1)+ " is " +marksArray[i]);
    System.out.println("SGPA : "+SGPA);
}
public static void main(String[] args)
{
   Student object = new Student();
   object.input();
   object.computeGradesArray();
   object.computeSGPA();
   object.display();
}}
```

Output;


C:\WINDOWS\system32\cmd.exe
C:\Users\Vallisha\Desktop\OOJ Lab\SGPA 16-10-2020>java Student
Enter your name : Vallisha M
Enter your USN : 1BM19CS177
Enter number of courses taken(MAX 10) : 5
Enter Credits for course 1 : 5
Enter marks obtained in course 1 : 80
Enter Credits for course 2 : 5
Enter marks obtained in course 2 : 35
Enter Credits for course 3 : 4
Enter marks obtained in course 3 : 95
Enter Credits for course 4 : 6
Enter marks obtained in course 4 : 100
Enter Credits for course 5 : 2
Enter marks obtained in course 5 : 30
USN : 1BM19CS177
Name : Vallisha M
Number of Courses taken this semester : 5
Marks scored in course 1 is 80
Marks scored in course 2 is 35
Marks scored in course 3 is 95
Marks scored in course 4 is 100
Marks scored in course 5 is 30
SGPA : 7.5
C:\Users\Vallisha\Desktop\OOJ Lab\SGPA 16-10-2020>

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.

Writeup;



Vallisha.M
1BM19CS177

Book Pangram

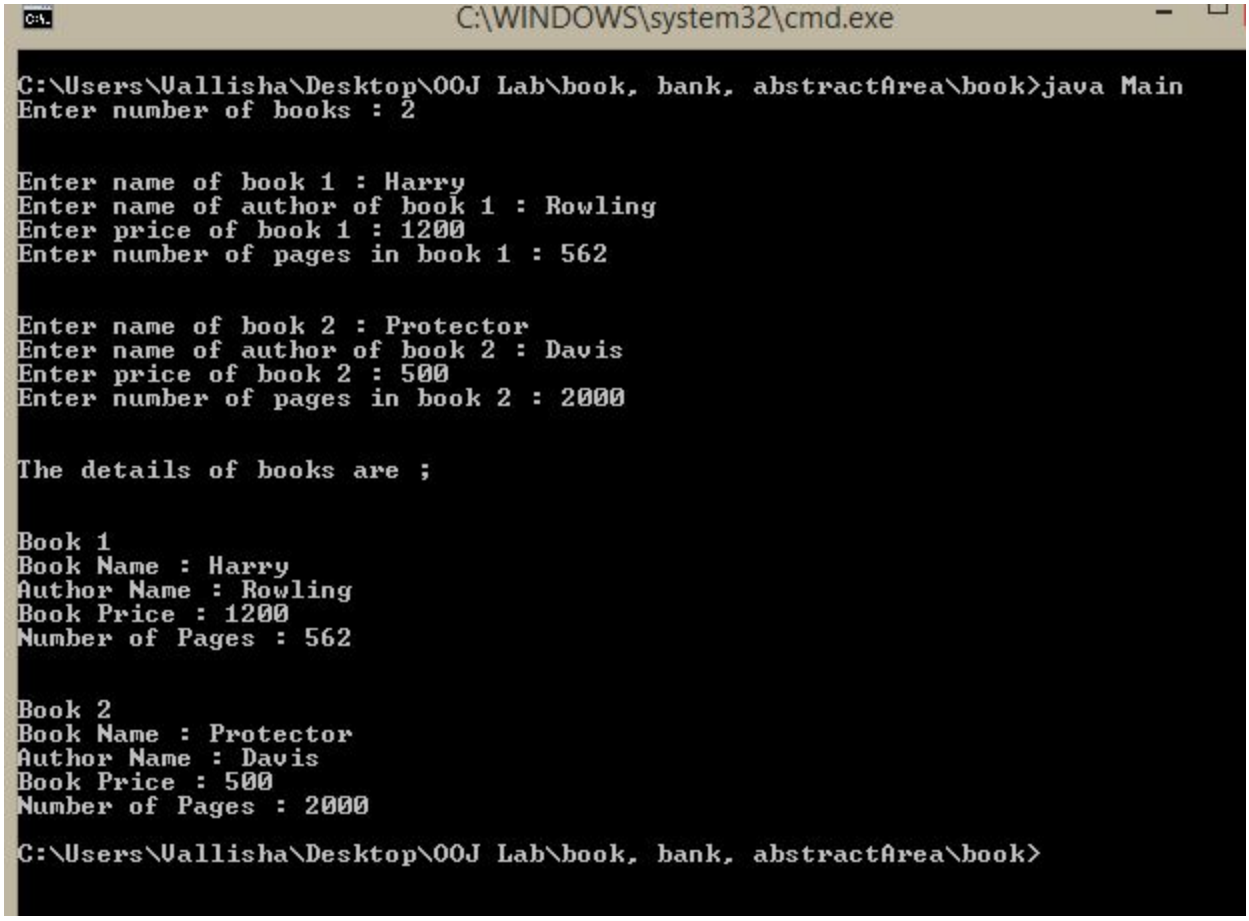
```
public class Book {

    private String name, author;
    private int num_pages, price;

    Book(){
        this.name = null;
        this.author = null;
        this.num_pages = 0;
        this.price = 0;
    }
    Book (String name, String author, int
          price, int num_pages)
    {
        this.author = author;
        this.name = name;
        this.price = price;
        this.num_pages = num_pages;
    }
    @Override
    public String toString(){

        return String.format("Book Name:
        "+name+"\nAuthor name : "+author
        +"\n Price : "+price+"\n Number
        Pages : "+num_pages);
    }
```

public class Main

```
import java.util.Scanner;
public class Main {

  public static void main(String args[]) {

    Scanner in = new Scanner(System.in);
    System.out.print("Enter number of
    books : ");
    int n = in.nextInt();
    Book books[] = new Book[n];
    String name, authorName;
    int num_pages, price;
    int i;
    for(i=0;i<n;i++){

      System.out.print("\n\nEnter
      name of book "+(i+1)+" : ");
      name = in.next();
      System.out.print("Enter name of
      author of book "+(i+1)+" : ");
      authorName = in.next();
      System.out.print("Enter price of
      book "+(i+1)+" : ");
      price = in.nextInt();
      System.out.print("Enter number
      of pages in book "+(i+1)+" : ");
      num_pages = in.nextInt();
```

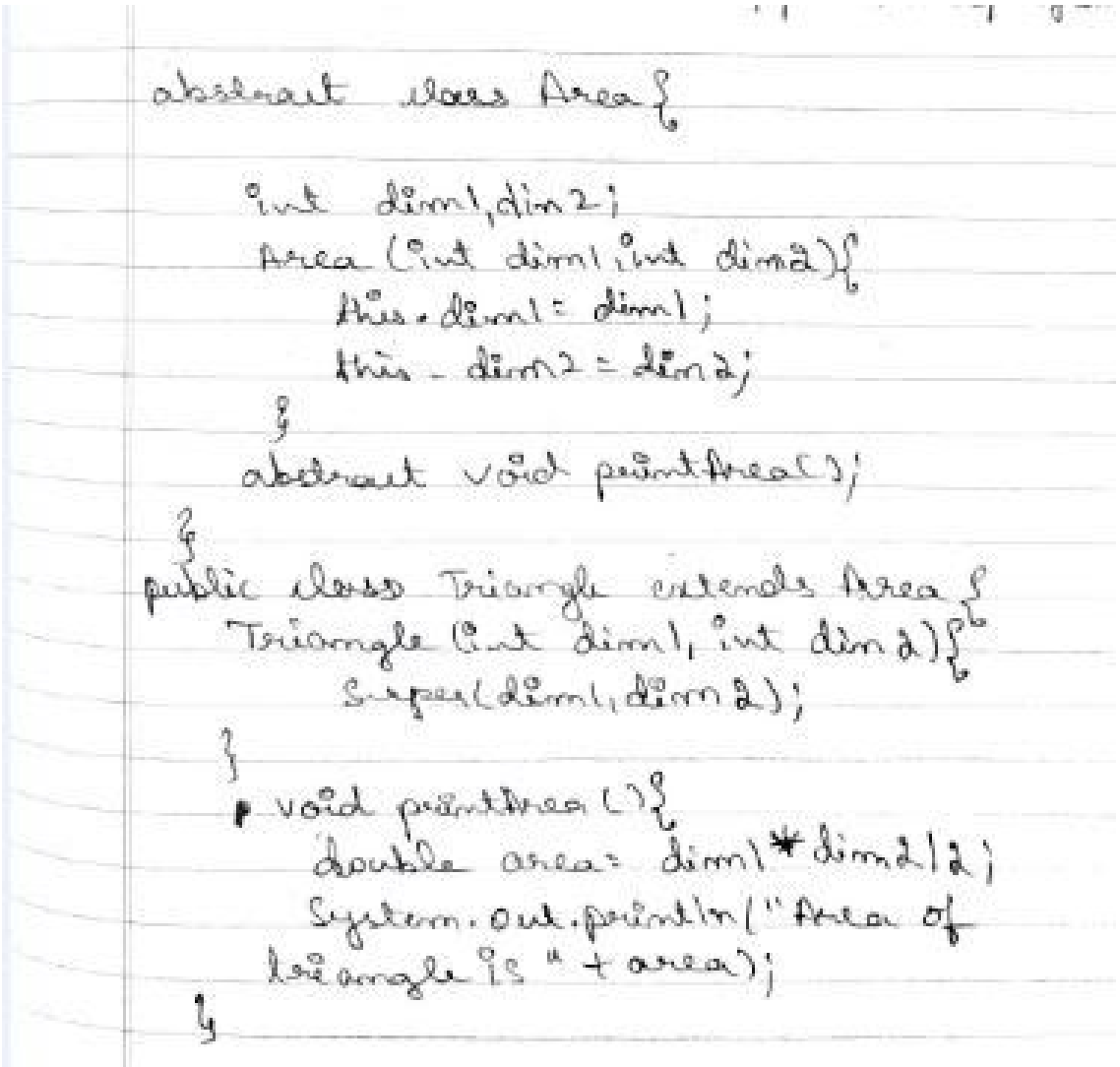
```
        books[i] = new Book (name, authorName,
                    price, numberOfPages);
    }
System.out.println("\n\n The details of
    books are; \n\n");
for (i=0; i<n; i++){
        System.out.println ("\n Book "+(i+1));
        System.out.println (books[i].
            toString());
    }
}
}

// --END--
```

Output;


C:\WINDOWS\system32\cmd.exe
C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\book>java Main
Enter number of books : 2
Enter name of book 1 : Harry
Enter name of author of book 1 : Rowling
Enter price of book 1 : 1200
Enter number of pages in book 1 : 562
Enter name of book 2 : Protector
Enter name of author of book 2 : Davis
Enter price of book 2 : 500
Enter number of pages in book 2 : 2000
The details of books are ;
Book 1
Book Name : Harry
Author Name : Rowling
Book Price : 1200
Number of Pages : 562
Book 2
Book Name : Protector
Author Name : Davis
Book Price : 500
Number of Pages : 2000
C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\book>

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.

Writeup;



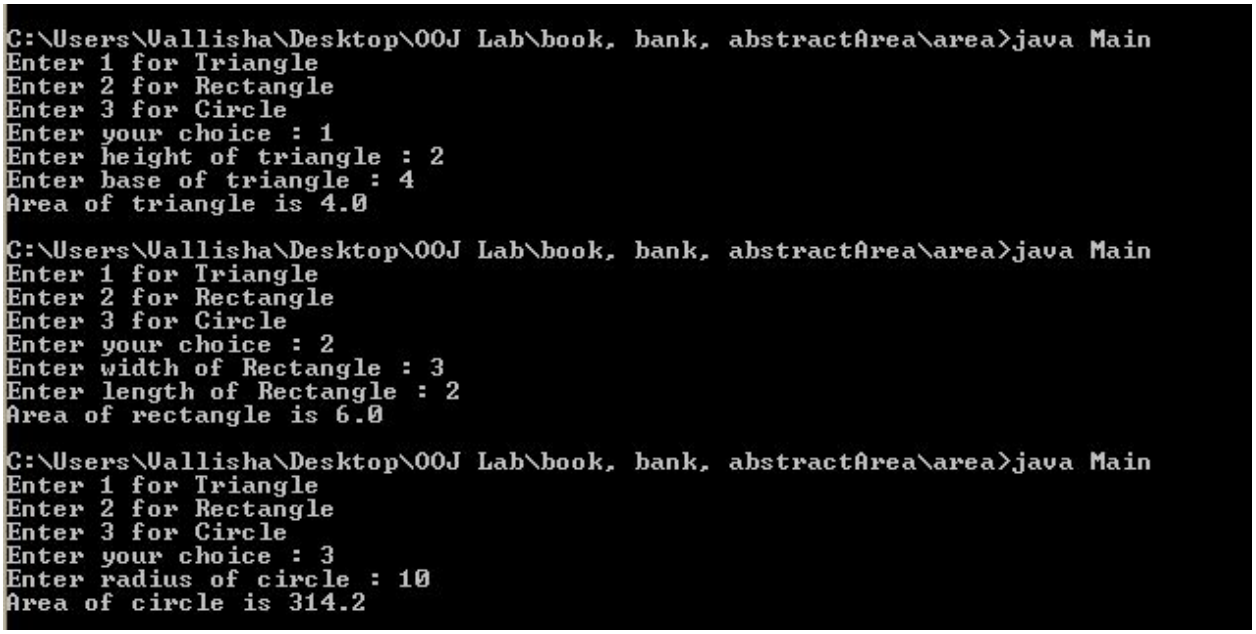
```
abstract class Area{

    int dim1,dim2;
    Area(int dim1,int dim2){
        this.dim1 = dim1;
        this.dim2 = dim2;
    }
    abstract void printArea();
}
public class Triangle extends Area{
    Triangle(int dim1, int dim2){
        super(dim1,dim2);
    }
    void printArea(){
        double area = dim1*dim2/2;
        System.out.println("Area of
    triangle is " + area);
}
```

```
public class Rectangle extends Area{
    Rectangle (int dim1, int dim2){
        super (dim1, dim2);
    }
    void printArea(){
    double area = dim1 * dim2;
    System.out.println("Area of rectangle
    is "+area);
    }
}
public class Circle extends Area{
    Circle (int dim){
        super(dim, dim);
    }
    void printArea(){
        double area = 3.142 * dim1 * dim2;
    System.out.println("Area of circle is
    "+area);
    }
}
import java.util.Scanner;
public class Main{
    public static void main(String args[])
    {
        Scanner sc = new Scanner (System
                                  .in);
    System.out.println("Enter 1 for Triangle");
    System.out.println("Enter 2 for Rectangle");
```

```
System.out.println("Enter 3 for circle : ");
int choice = sc.nextInt();
switch(choice){
    case 1:{
        System.out.println("Enter height of
        triangle : ");
        dim1 = in.nextInt();
        System.out.print("Enter base of
        triangle : ");
        dim2 = in.nextInt();
        Triangle triangle = new Triangle(dim1,
        dim2);
        triangle.printArea();
        break;
}
 case 2:{
 System.out.println("Enter width of
 rectangle : ");
 dim1 = sc.nextInt();
 System.out.println("Enter length of
 rectangle : ");
 dim2 = sc.nextInt();
 Rectangle rectangle = new Rectangle
 (dim1, dim2);
 rectangle.printArea();
 break;
}
```

```
case 3:{
    System.out.println("Enter radius of
    circle :");
    dim1 = in.nextInt();
    Circle circle = new Circle(dim1);
    circle.printArea();
    break;
  }
default:{
    System.out.println("Wrong Input");
    break;
  }
 }sc.close();
}
}
```

Output;



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 **Curr-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.

Writeup;

Vallisha M
1BM19CS177

Bank Program

```
public class Bank {

    private String customerName;
    private String accountType
    private int accountNumber;
    Bank (){
       customerName = null;
       accountType = null;
       accountNumber = 0;
    }
   Bank(String customerName, int
         accountNumber, String accountType){
   this.customerName = customerName;
   this.accountNumber = accountNumber;
   this.accountType = accountType;
}
@Override
public String toString (){

 return String.format("Customer Name:
  " + customerName + "\n Account number:
  " + accountNumber + "\n Account Type: "
   + accountType);
}
```

```
public class CurrentAccount extends Bank{

    private double balance;

    CurrentAccount(){
        balance=0.0D;
    }
    public void credit(double amount){
            balance = balance+amount;
            this.minimumBalance();
    }
    public int debit(double amount){
        if(balance>100 && balance-amount>=0){
            this.balance = balance -amount;
            this.minimumBalance();
            return 0;
        }
        return -1;
}
public void minimumBalance(){
        int fine =0;
        if(balance<1000)
            fine=100;
        balance = balance-fine;
}
public double getBalance(){
        return this.balance;
}}}
```

```
public class SavingsAccount extends
Bank {

    private double balance, rate;
    SavingsAccount(){
        rate = 4.567;
        balance = 0.0D;
    }
    public double getBalance(){
        return balance;
    }
    public void credit(double amount){
        balance = balance + amount;
        this.currentBalance();
    }
    public int debit(double amount){
        if(balance<100)

        if(balance - amount >=0){
            balance = balance - amount;
            return 0;
        }
        return -1;
    }
    public void calculateInterest(double time)
    {
        rate = rate/100;
        if(time> 5)  rate = 4.9/100;
        if(time>8)   rate = 5.00/100;
```

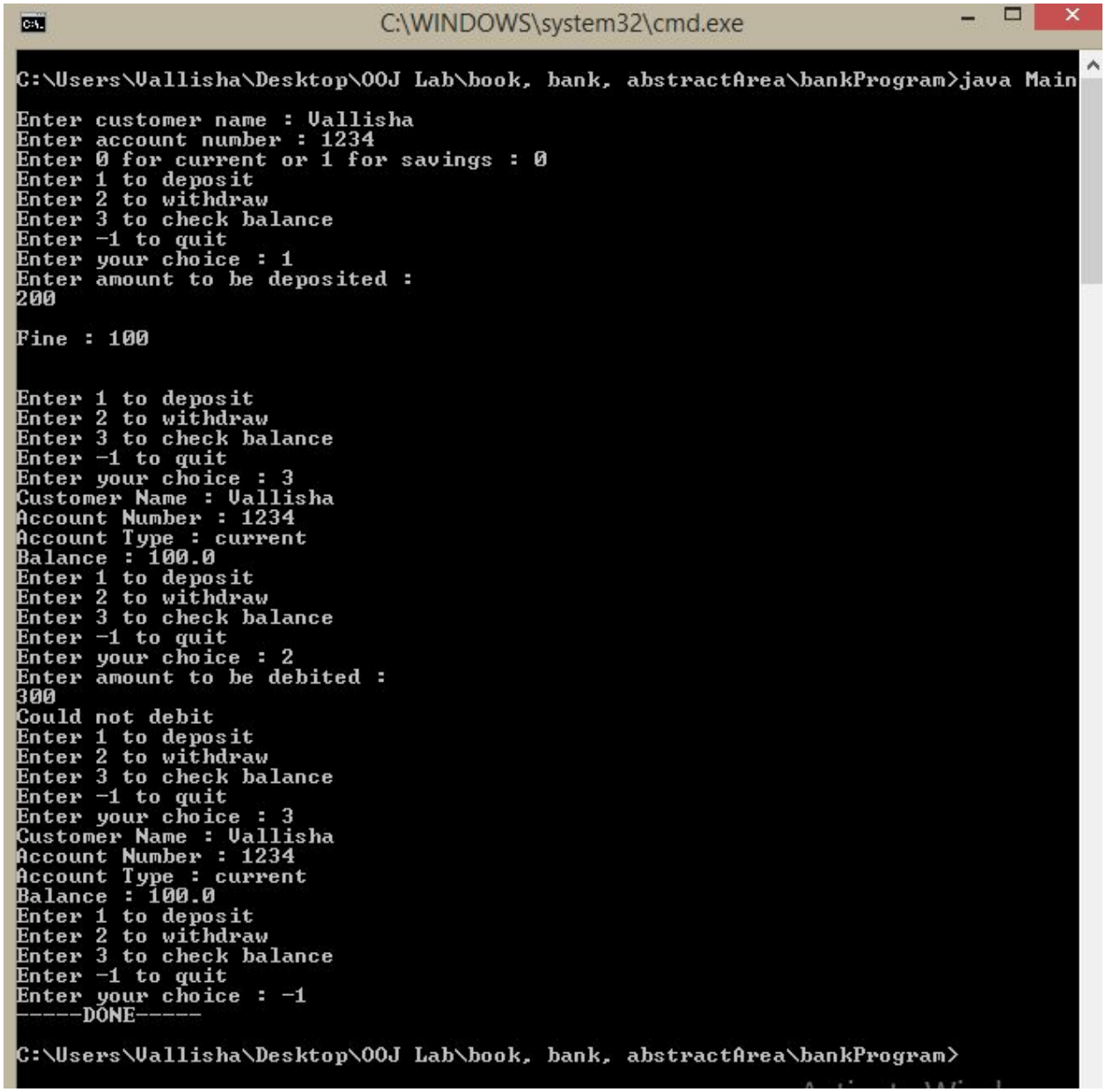
```
        balance = balance * Math.pow(
                (1+rate), time);

    }
}
import java.util.Scanner;
public class Main{

    public static void main(String args[]){

        Scanner in = new Scanner(System.in);
        Bank bank;
        String cn, ct;
        int ano, choice;
        double amt;
        System.out.print("Enter customer
        name : ");
        cn = in.next();
        System.out.print("Enter account
        number : ");
        ano = in.nextInt();
        CurrentAccount account = new
        CurrentAccount();
        SavingsAccount account = new
        SavingsAccount();
        System.out.print("Enter 0 for
        current account or 1 for
        savings account : ");
boolean flag = in.nextInt()==0?true:false;
```

```
if (flag)
    at = "current";
else
    at = "savings"
bank = new Bank (cn, ano, at);

while (true){

    System.out.println("Enter 1 to deposit");
    System.out.println("Enter 2 to withdraw");
    );
    System.out.println("Enter 3 to check
    balance");
    ~~if(at~~
    if(!flag)
        System.out.println("Enter 4 to
        calculate Interest");
    System.out.println("Enter -1 to quit");
    System.out.print("Enter your choice :");
    choice = in.nextInt();
    if(choice == -1)
        break;
    if(choice == 1){
        System.out.println("Enter amount to
        be credited : ");
        amt = Math.abs(in.nextDouble());
        if(flag)  account.credit(amt);
        else account1.credit(amt);
}
```

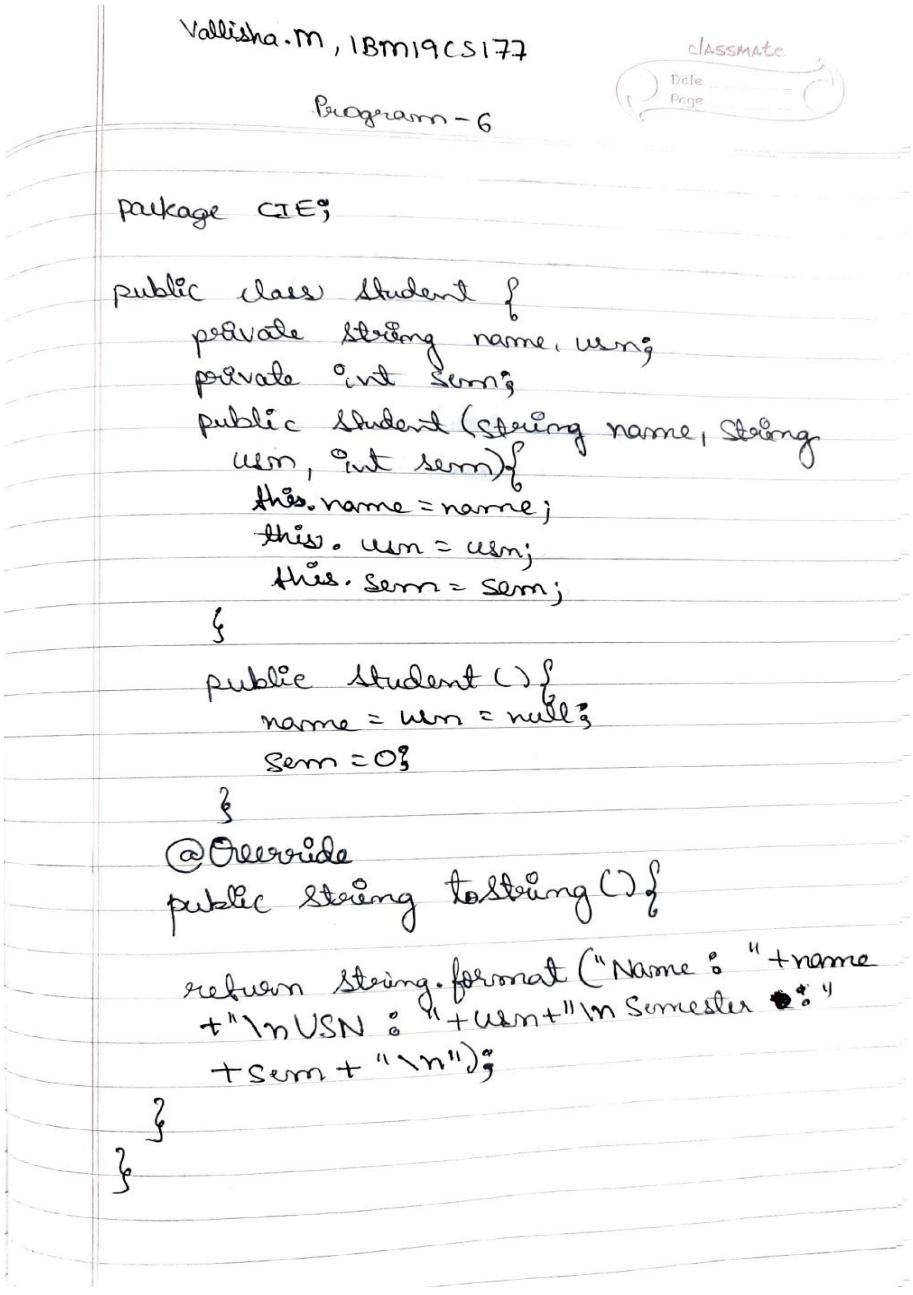
```
if (choice == 2){
  System.out.println("Enter amount to
  be debited : ");
  amt = Math.abs(in.nextDouble());
  int status = 0;
  if (flag) status = account.debit(amt);
  else status = account1.debit(amt);
  if (status == -1)
     System.out.println("Could not
     debit");
}
else if(choice == 3){
    System.out.println(bank.toString());
    if(flag)
    System.out.println("Balance : "
    +account.getBalance());
    else
     System.out.println("Balance : "
     +account1.getBalance());
}
else if(choice == 4 && flag){
 System.out.println("Enter number of
 years : ");
 amt = Math.abs(in.nextDouble());
 System.out.println("Balance before
 interest : "+account1.getBalance());
 account1.calculateInterest(amt);
 System.out.println("Balance after
 interest : "+account1.getBalance());
}
```

```
else if (choice > 5 || choice == 0 || choice < -1
    || ( flag && choice == 4)
    System.out.println("Invalid input");
}
System.out.println("----- DONE -----");
in.close();
}
}
```

Output;


C:\WINDOWS\system32\cmd.exe
C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\bankProgram>java Main
Enter customer name : Vallisha
Enter account number : 1234
Enter 0 for current or 1 for savings : 0
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : 1
Enter amount to be deposited :
200
Fine : 100
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : 3
Customer Name : Vallisha
Account Number : 1234
Account Type : current
Balance : 100.0
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : 2
Enter amount to be debited :
300
Could not debit
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : 3
Customer Name : Vallisha
Account Number : 1234
Account Type : current
Balance : 100.0
Enter 1 to deposit
Enter 2 to withdraw
Enter 3 to check balance
Enter -1 to quit
Enter your choice : -1
-----DONE-----
C:\Users\Vallisha\Desktop\OOJ Lab\book, bank, abstractArea\bankProgram>

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.



Vallisha.M, 1BM19CS177

Program - 6

```
package CIE;

public class Student {
    private String name, usn;
    private int sem;
    public Student(String name, String
      usn, int sem){
       this.name = name;
       this.usn = usn;
       this.sem = sem;
    }
    public Student(){
       name = usn = null;
       sem = 0;
    }
@Override
public String toString(){

 return String.format("Name : "+name
 +"\nUSN : "+usn+"\nSemester : "
 +sem+"\n");
  }
}
```

```
package CIE;

public class Internals {
    private int marks[];
    public Internals(int arr[]){
        marks = new int[5];
        for(int i=0;i<5;i++)
            marks[i]=arr[i];
    }
    public int getMark(int i){
        return marks[i];
    }
}
```

```
package SEE;
~~import SEE.Student;~~ import CIE.Student;
public class External extends Student{
    private int marks[];
public External(int arr[]){
        marks = new int[5];
        for(int i=0; i<5; i++)
            marks[i] =arr[i];
    }
    public int getMark(int i){
        return marks[i];
    }
```

```
import java.util.Scanner;
import CIE.*;
import SEE.*;

public class Main{

  public static void main(String args[]){

Scanner in = new Scanner(System.in);
int sem, n, i, j;
String name, usn;
System.out.print("Enter number of students : ");
n = in.nextInt();
int marks[][] = new int[n][5];
int arr[] = new int[5];
Student stu[] = new Student[n];
Internals internal[] = new Internals[n];
External ext[] = new External[n];
for(i=0; i<n; i++){

System.out.println("---- Student "+(i+1)+
" ---- \n");
System.out.print("Enter name : ");
name = in.next();
System.out.print("Enter USN : ");
usn = in.next();
```

```
System.out.print("Enter semester : ");
sem = in.nextInt();
stu[i] = new Student(name, usn, sem);
for (j=0; j<5; j++){
    System.out.print("Enter CIE marks
    in course "+(j+1)+" : ");
    arr[j] = in.nextInt();
}
System.out.println("\n\n");
internal[i] = new Internals(arr);
for (j=0; j<5; j++){
   System.out.print("Enter SEE marks
in course "+(j+1)+" : ");
arr[j] = in.nextInt();
}
ext[i] = new External(arr);
}
in.close();
for (i=0; i<n; i++)
      for (j=0; j<5; j++)
        marks[i][j] = internal.getMark(j)
  +((ext[i].getMark(j))/2);
System.out.println("-----STUDENT
DETAILS -----\n\n");
for (i=0; i<n; i++){
    System.out.println("------ Student
      "+(i+1)+" ------\n\n");
    System.out.println(stu[i].toString
      ());
```

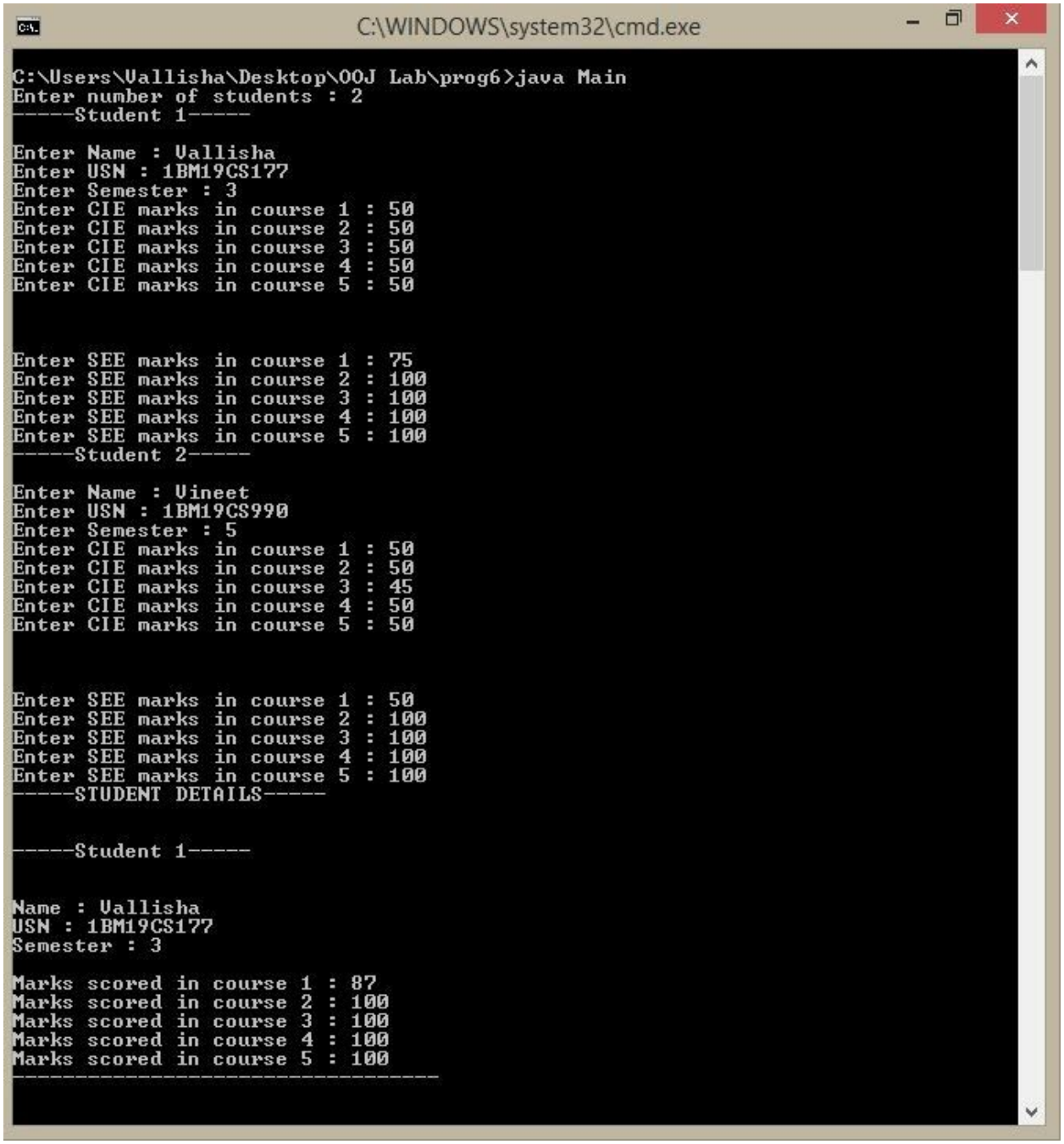
```
        for (j=0; j<5; j++){
            System.out.println("Marks scored
in      course " +(j+1) +" : " + marks[i][j]);
        }
        System.out.println("-----\n\n");
    }
}
```

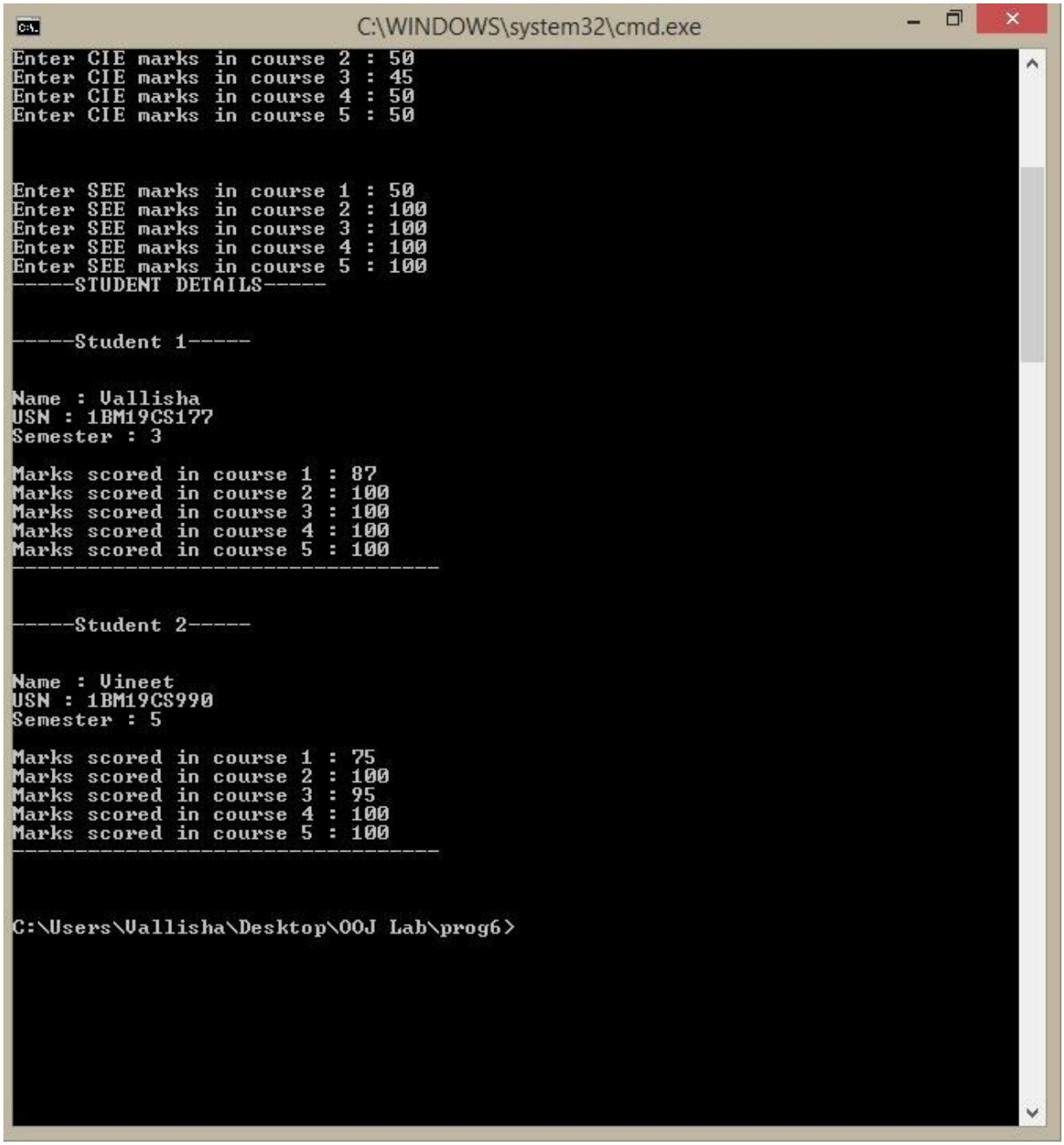
FOLDER STRUCTURE;

prog6

- CIE
  - Student.java
  - Internals.java
- SEE
  - External.java
- main.java

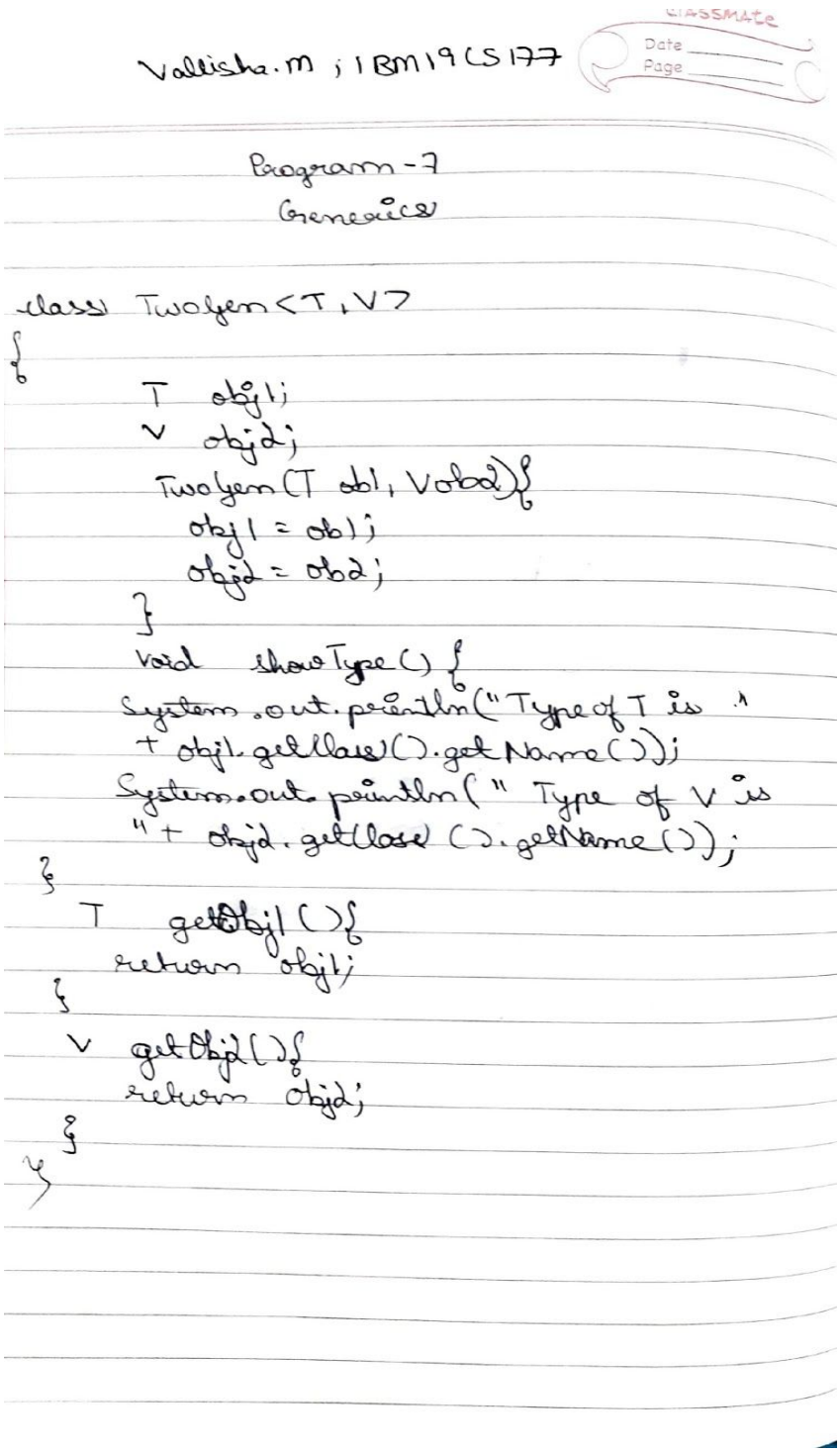
Output;


C:\WINDOWS\system32\cmd.exe
C:\Users\Vallisha\Desktop\OOJ Lab\prog6>java Main
Enter number of students : 2
-----Student 1-----
Enter Name : Vallisha
Enter USN : 1BM19CS177
Enter Semester : 3
Enter CIE marks in course 1 : 50
Enter CIE marks in course 2 : 50
Enter CIE marks in course 3 : 50
Enter CIE marks in course 4 : 50
Enter CIE marks in course 5 : 50
Enter SEE marks in course 1 : 75
Enter SEE marks in course 2 : 100
Enter SEE marks in course 3 : 100
Enter SEE marks in course 4 : 100
Enter SEE marks in course 5 : 100
-----Student 2-----
Enter Name : Vineet
Enter USN : 1BM19CS990
Enter Semester : 5
Enter CIE marks in course 1 : 50
Enter CIE marks in course 2 : 50
Enter CIE marks in course 3 : 45
Enter CIE marks in course 4 : 50
Enter CIE marks in course 5 : 50
Enter SEE marks in course 1 : 50
Enter SEE marks in course 2 : 100
Enter SEE marks in course 3 : 100
Enter SEE marks in course 4 : 100
Enter SEE marks in course 5 : 100
-----STUDENT DETAILS-----
-----Student 1-----
Name : Vallisha
USN : 1BM19CS177
Semester : 3
Marks scored in course 1 : 87
Marks scored in course 2 : 100
Marks scored in course 3 : 100
Marks scored in course 4 : 100
Marks scored in course 5 : 100


C:\WINDOWS\system32\cmd.exe
Enter CIE marks in course 2 : 50
Enter CIE marks in course 3 : 45
Enter CIE marks in course 4 : 50
Enter CIE marks in course 5 : 50
Enter SEE marks in course 1 : 50
Enter SEE marks in course 2 : 100
Enter SEE marks in course 3 : 100
Enter SEE marks in course 4 : 100
Enter SEE marks in course 5 : 100
-----STUDENT DETAILS-----
-----Student 1-----
Name : Vallisha
USN : 1BM19CS177
Semester : 3
Marks scored in course 1 : 87
Marks scored in course 2 : 100
Marks scored in course 3 : 100
Marks scored in course 4 : 100
Marks scored in course 5 : 100
-----Student 2-----
Name : Vineet
USN : 1BM19CS990
Semester : 5
Marks scored in course 1 : 75
Marks scored in course 2 : 100
Marks scored in course 3 : 95
Marks scored in course 4 : 100
Marks scored in course 5 : 100
C:\Users\Vallisha\Desktop\OOJ Lab\prog6>

7.Write a program to demonstrate generics with multiple object parameters

Writeup;



Vallisha.m ; 1BM19CS177

Program-7
Generics

```
class TwoGen<T,V>
{
    T obj1;
    V obj2;
    TwoGen(T ob1, V ob2){
      obj1 = ob1;
      obj2 = ob2;
    }
    void showType() {
    System.out.println("Type of T is "
    + obj1.getClass().getName());
    System.out.println(" Type of V is
    " + obj2.getClass().getName());
  }
   T getObj1(){
    return obj1;
  }
   V getObj2(){
    return obj2;
  }
}
```

```
class Main {

  public static void main (String args[]) {

    TwoGen<Integer, String> obj = new
     TwoGen<Integer, String> (99, "Java");

     obj.showTypes();
     int v = obj.getObj1();
     System.out.println("value of integer = "
       + v);
     String str = obj.getObj2();
     System.out.println(" value of String : "
      + str);

    }
}
```

Output;

```
C:\Users\Vallisha\Desktop\OOJ Lab\program_7>java Main
Type of T is java.lang.Integer
Type of V is java.lang.String
Integer value : 99
String value : Java
```

8.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 Wrong Age( ) 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.

Writeup;

Vallisha.M 1BM19CS177

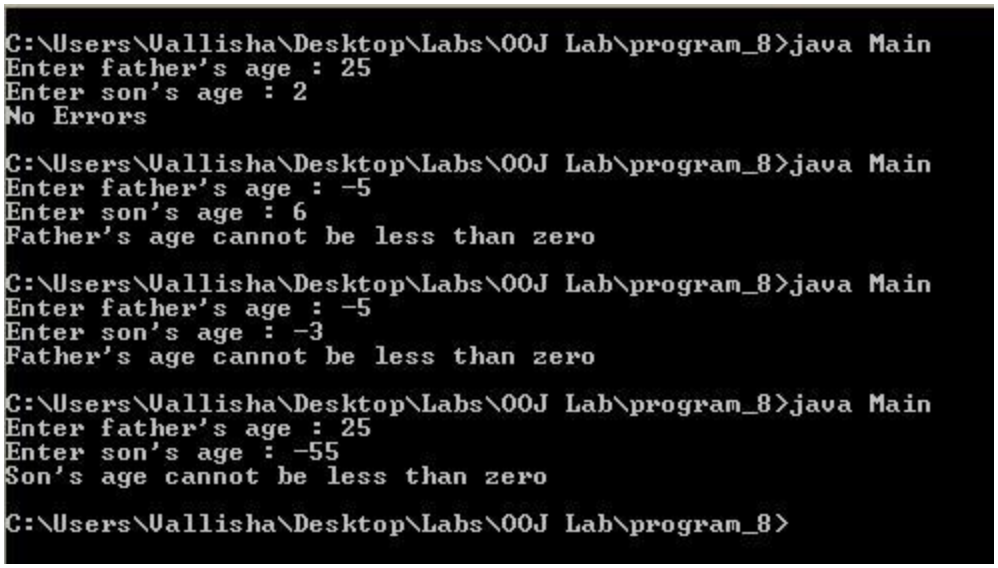
Program-8 -Exception

```
import java.util.Scanner;

class Main {
  public static void main (string args[]){
      Scanner in = new Scanner(System.in);
      System.out.print("Enter father's age: ");
      int fatherAge = in.nextInt();
      System.out.print("Enter son's age: ");
      int sonAge = in.nextInt();
      in.close();
```

```
Son son;
try {
    son = new Son (sonAge, fatherAge);
    System.out.println("No errors");
}
catch (AgeLessThanZeroException exception){
    System.out.println(exception.getMessage());
}
catch (SonOlderThanFatherException exception){
    System.out.println(exception.getMessage());
}
}
}

public class AgeLessThanZeroException extends
  Exception{
    AgeLessThanZeroException(String message){
        super(message);
    }
}
public class SonOlderThanFatherException extends
Exception{
    SonOlderThanFatherException(String message){
        super(message);
    }
}
```

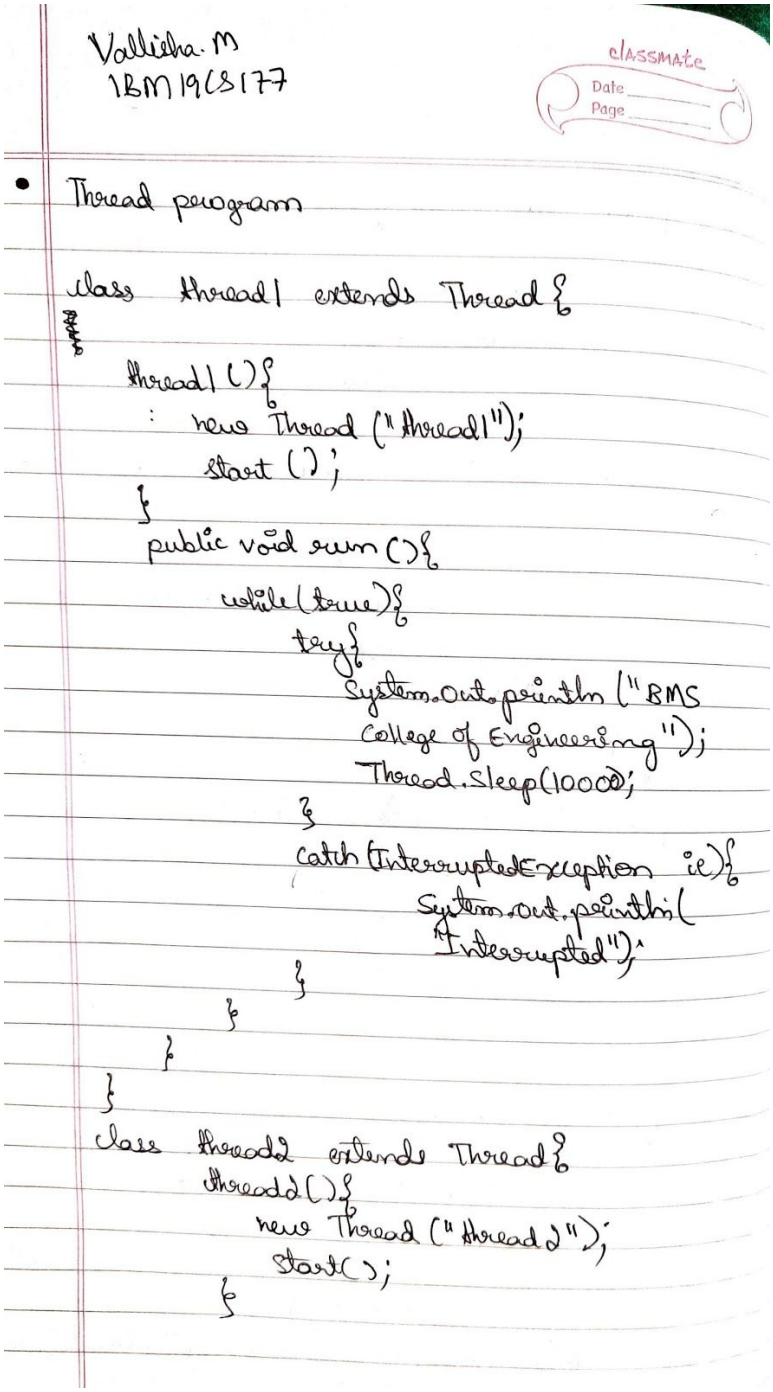
```
public class Father{
    int age=0;
    message = "Father's Age cannot be less
                than zero";
    Father(){}
    Father (int age)throwes AgeLessThanZeroException
    {
        if(age<0)
            throw new AgeLessThanZeroException(
            message);
        this.age = age;
    }
}
public class Son extends Father{
    String message1 = "son's age cannot be greater
     than father's age";
    string message2 = "Son's age cannot be
     less than zero";
     int age;
     Son (int sonAge, int FatherAge) throwes
AgeLessThanZeroException, SonOlderThanFatherException
{
        super (fatherAge);
        if (sonAge<0)
          throw new AgeLessThanZeroException(message2);
        if(sonAge>=fatherAge)
        throw new SonOlderThanFatherException(message1);
         age = sonAge;
}}
```

Output;


C:\Users\Vallisha\Desktop\Labs\OOJ Lab\program_8>java Main
Enter father's age : 25
Enter son's age : 2
No Errors
C:\Users\Vallisha\Desktop\Labs\OOJ Lab\program_8>java Main
Enter father's age : -5
Enter son's age : 6
Father's age cannot be less than zero
C:\Users\Vallisha\Desktop\Labs\OOJ Lab\program_8>java Main
Enter father's age : -5
Enter son's age : -3
Father's age cannot be less than zero
C:\Users\Vallisha\Desktop\Labs\OOJ Lab\program_8>java Main
Enter father's age : 25
Enter son's age : -55
Son's age cannot be less than zero
C:\Users\Vallisha\Desktop\Labs\OOJ Lab\program_8>

9.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.

Writeup;



Vallisha. M
1BM19CS177

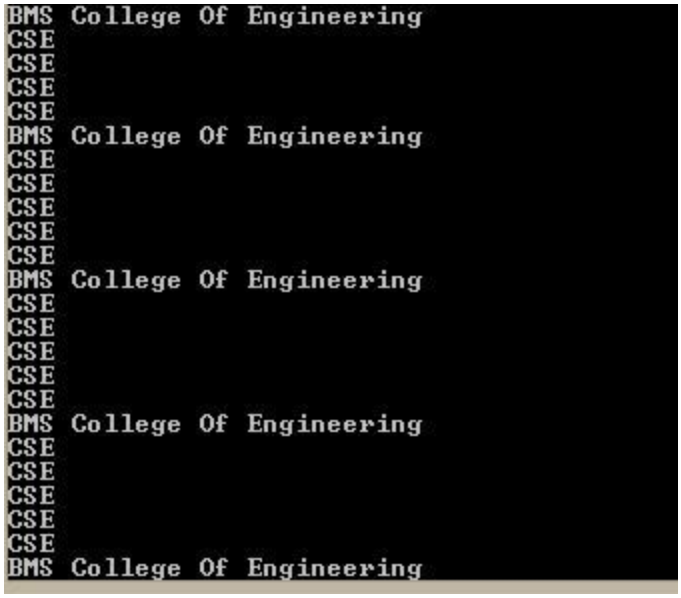
classmate
Date
Page

- Thread program

```
class thread1 extends Thread{

    thread1(){
        new Thread("thread1");
        start();
    }
    public void run(){
        while(true){
            try{
                System.out.println("BMS
                College of Engineering");
                Thread.sleep(10000);
            }
            catch(InterruptedException ie){
                    System.out.println(
                    "Interrupted");
            }
        }
    }
}
class thread2 extends Thread{
        thread2(){
            new Thread("thread2");
            start();
        }
```

```
public void run()
{
    while(true)
    {
        try{
            System.out.println("ESE");
            Thread.sleep(2000);
        }
        catch (InterruptedException e){
            System.out.println("Interrupted");
        }
    }
}
class Main{
    public static void main(String args[]){
        System.out.println("Enter ctrl+c to stop");
        Thread1 t1 = new Thread1();
        Thread2 t2 = new Thread2();
    }
}
```

Output;


BMS College Of Engineering
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College Of Engineering

10.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.

VALLISHA.M
1BM19CS177

PROGRAM-10

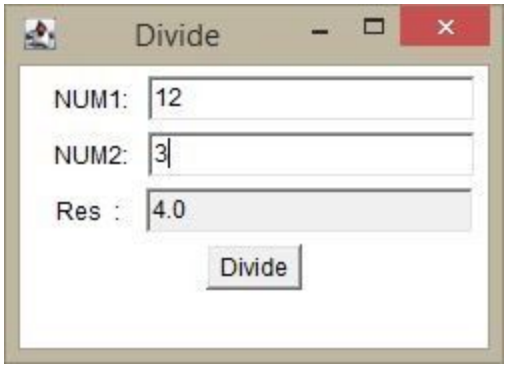
```
import java.awt.event.*;
import java.awt.*;

public class Main extends Frame implements
ActionListener{

    String msg = "";
    Button div;
    TextField n1, n2, res;

    public Main() {
      setLayout(new FlowLayout());
      Label num1 = new Label("NUM1 : ", Label.RIGHT);
      Label num2 = new Label("NUM2 : ", Label.RIGHT);
      Label r = new Label("Res : ", Label.RIGHT);
      n1 = new TextField(20);
      n2 = new TextField(20);
      res = new TextField(20);
      res.setEditable(false);
      add(num1); add(n1);
      add(num2); add(n2);
      add(r); add(res);
      n1.addActionListener(this);
      n2.addActionListener(this);
      Button divide = new Button("divide");
      div = (Button) add(divide);
      div.addActionListener(this);
```

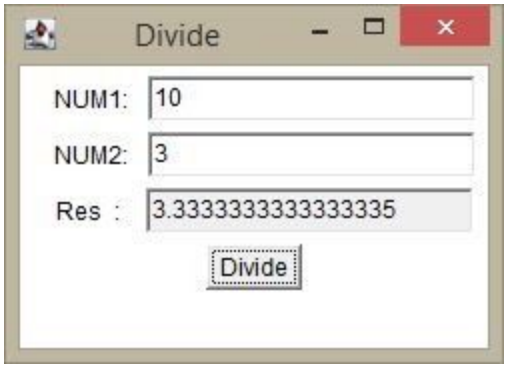
```
addWindowListener(new MyWindowAdapter());
}
public void actionPerformed(ActionEvent ae){
    if(ae.getSource()==div);
    String a=n1.getText();
    String b = n2.getText();
    int i,j;
    double ans = 0.0D;
    boolean flag=false;
    boolean flag1 = false;
    try{
        i = Integer.parseInt(a);
        j = Integer.parseInt(b)
        if(j==0.0){ flag1=true;
          throw new ArithmeticException
        ("Division by zero not Possible");
        }
        ans = (double)(i/j);
        catch(E
    }
    Catch(Exception e){
        flag=true;
        msg = flag1?e.getMessage():
              e.toString();
        DialogBox d= new DialogBox(this,
          "ERROR");
        d.setVisible(true);
    }
```

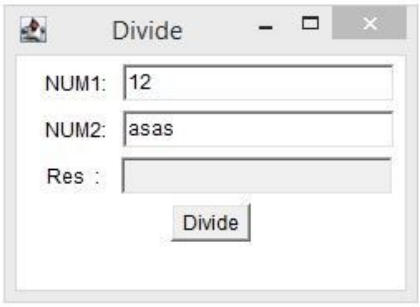
```
    if(!flag) //No exception
        res.setText(String.valueOf(ans));
  }
  public static void main(String args[]){

      Main main = new Main();
      main.setSize(new Dimension(250,180));
      main.setTitle("Divide");
      main.setVisible(true);
  }
}

import java.awt.event.*;
import java.awt.*;

class Dialog
public class DialogBox extends Dialog implements
ActionListener {
      Main bld;
public DialogBox(Frame Parent, String title){
      super(parent, this, false);
      bld = (Main) parent;
      setLayout(new FlowLayout());
      setSize(600,100);
      add(new Label(bld.msg));
      Button b;
      add(b = new Button("Close"));
      b.addActionListener(this);
```

```
addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent ae)
        {
                dispose();
        } });
}
public void actionPerformed(ActionEvent ae){
        dispose();
}
}
class MyWindowAdapter extends
WindowAdapter {
    public void windowClosing(Window
     WindowEvent we) {
        System.exit(0);
    }
}
```

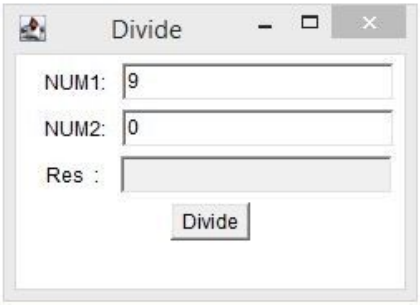
Output;


Divide
NUM1:
12
NUM2:
3
Res :
4.0
Divide


Divide
NUM1:
10
NUM2:
3
Res :
3.3333333333333335
Divide


Divide
NUM1:
12
NUM2:
asas
Res :
Divide


ERROR
java.lang.NumberFormatException: For input string: "asas"
Close


Divide
NUM1:
9
NUM2:
0
Res :
Divide


ERROR
Division by zero not possible.
Close