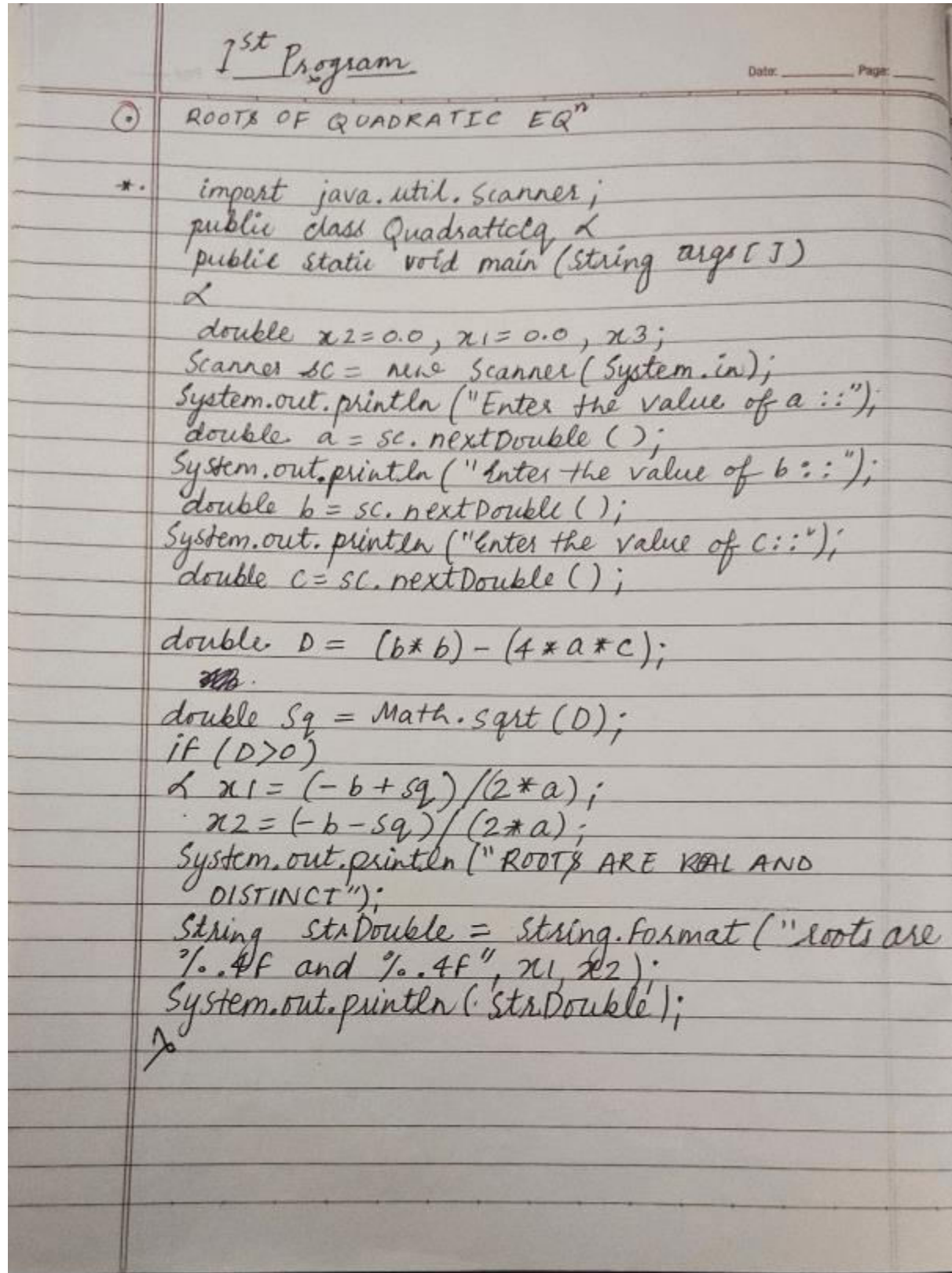


Q1. Develop a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$.

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.



else if ($D == 0$)
 $x3 = (-b) / (2 * a)$;
 System.out.println("ROOTS ARE REAL AND EQUAL");
 String str1 = String.format("Roots are
%.4f and both are equal", $x3$);

System.out.println(str1);

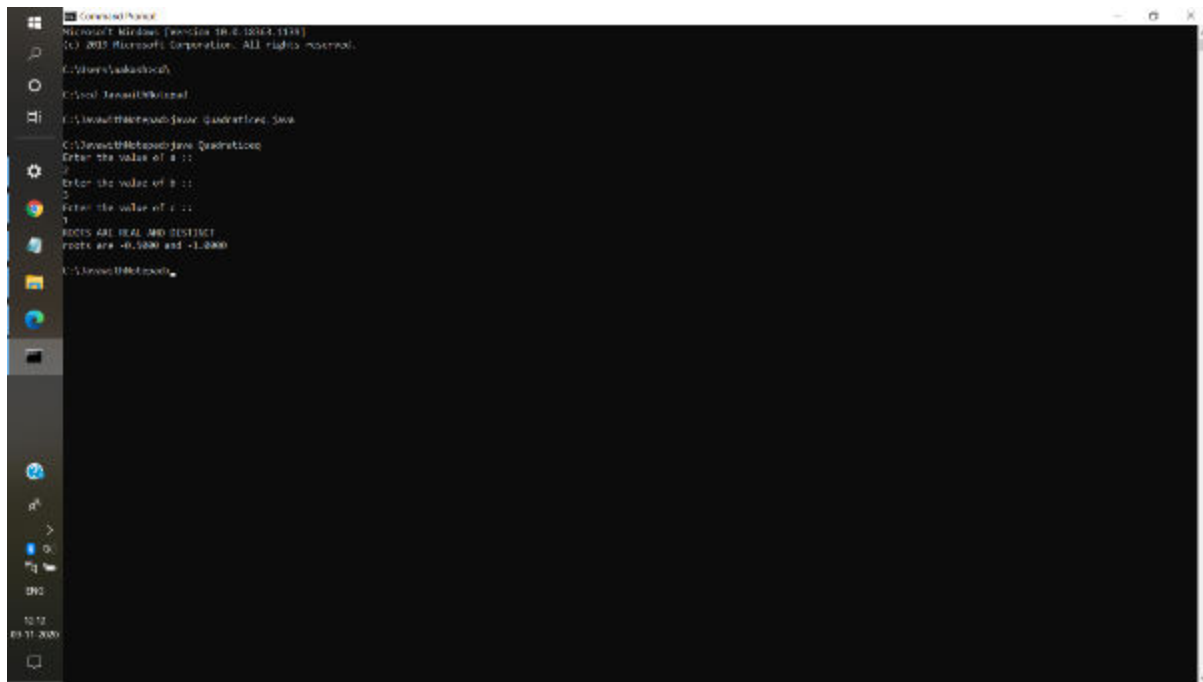
else if ($D < 0$)

 System.out.println("ROOTS ARE UNREAL");

}

}

}



The screenshot shows a Windows Command Prompt window titled "Command Prompt". The text inside the window is as follows:

```
Microsoft Windows [Version 10.0.18363.1118]  
(c) 2019 Microsoft Corporation. All rights reserved.  
C:\Users\jshah>  
C:\Users\jshah>  
C:\Users\jshah>java Quadratics.java  
C:\Users\jshah>java Quadratics  
Enter the value of a ::  
2  
Enter the value of b ::  
-1  
Enter the value of c ::  
1  
ROOTS ARE REAL AND DISTINCT  
roots are -0.5000 and +1.0000  
C:\Users\jshah>
```

The left sidebar of the Command Prompt shows various system icons and the date and time "10:12 09-11-2019".

Q2. 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.

lab-program - 2 Week

Date: _____ Page: _____

①

```
1. > import java.util.Scanner;
    class Student {
        Scanner xx = new Scanner(System.in);
        System.out.println("Enter Subject");
        int n = xx.nextInt();
        String usn;
        String name;
        int Credit [ ] = new int [n];
        double marks [ ] = new double [n];
        int i, total = 0;
        double total = 0.0;
        int q = 0;
        double p = 0.0, t = 0.0, w = 0.0;

        void StudentDetails ()
        {
            System.out.println("ENTER STUDENT DETAILS");
            Scanner xx = new Scanner(System.in);
            System.out.println("Enter usn");
            usn = xx.nextLine();
            System.out.println("Enter name:");
            name = xx.nextLine();
            System.out.println("Enter Credits and marks");
            for (i = 0; i < n; i++)
            {
                System.out.print("Enter marks of Subject " + (i+1) + " : ");
                marks[i] = xx.nextDouble();
                total = total + marks[i];
                System.out.print("Enter credits of Subject " + (i+1) + " : ");
                Credit[i] = xx.nextInt();
            }
        }
    }
}
```



```
total1 = total1 + credit[i];  
p = (double)(credit[i] * marks[i]) + p;
```

```
System.out.println("Total MARKS" + p);  
t = (p) / total1;  
w = t / 10.0;  
q = (int) t / 10;
```

```
void printDetails () {  
    System.out.println("*****");  
    System.out.println("STUDENT DATA");  
    System.out.println("USN:" + usn + "NAME:"  
        + name);  
}
```

```
void Grade ()
```

```
{  
    System.out.println("GRADE OF THE STUDENT  
        IS :");  
    if (q >= 9)  
        System.out.println("S");  
    else if (q >= 8 && q < 9)  
        System.out.println("A");  
    else if (q >= 7 && q < 8)  
        System.out.println("B");  
    else if (q >= 6 && q < 7)  
        System.out.println("C");  
    else if (q >= 5 && q < 6)  
        System.out.println("D");  
    else if (q >= 4 && q < 5)  
        System.out.println("E");  
}
```

```
else:  
    System.out.println("F");  
    System.out.println("SCPA:" + W);
```

```
public class StudentRunner {  
    public static void main(String args[])
```

```
{  
    Student s = new Student();
```

```
    s.StudentDetails();
```

```
    s.printDetails();
```

```
    s.Grade();  
}
```

```
C:\Users\Anand\Desktop> java StudentRunner.java
Enter Student Name:
Student Name
Enter marks
Enter marks
Enter marks and Name:
Enter Marks of Subject1:77
Enter Marks of Subject2:88
Enter Marks of Subject3:99
Enter Marks of Subject4:98
Enter Marks of Subject5:97
Enter Marks of Subject6:96
=====
STUDENT DATA
Name: Student Name Marks: 77 88 99 98 97
Average: 92.4
```

Q3. 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.

Lab Week 3

Date: _____ Page: _____

```
import java.util.Scanner;
```

```
class book
```

```
{ String name, author;
```

```
  int price, num-pages;
```

```
void book ()
```

```
{ name = "null";
```

```
  author = "null";
```

```
  price = 0;
```

```
  num-pages = 0;
```

```
}
```

```
void get ()
```

```
{ Scanner get = new Scanner (System.in);
```

```
  System.out.println ("\n Enter the name:");
```

```
  name = get.next ();
```

```
  System.out.println ("Enter the author:");
```

```
  author = get.next ();
```

```
  System.out.println ("Enter the no of pages:");
```

```
  price = get.nextInt ();
```

```
}
```

```
void p ()
```

```
{ System.out.println ("NAME: " + name);
```

```
  System.out.println ("AUTHOR: " + author);
```

```
  System.out.println ("PRICE: " + price);
```

```
  System.out.println ("PAGES: " + num-pages);
```

```
}
```

```
public String toString ()
```

```
{ return ("\nNAME: " + name + "\nAUTHOR: " + author +
```

```
  "\nPRICE: " + price + "\nPAGES: " + num-pages);
```


Date _____ Page _____

```
}  
}  
class BookMain  
{ public static void main (string args [])  
  { Scanner get = new Scanner (System.in);  
    int n, ch;  
    System.out.println ("Enter the no of books to be entered :");  
    n = get.nextInt ();  
    book b[] = new book [n];  
    for (int i = 0; i < n; i++)  
    { b[i] = new book ();  
      b[i].get ();  
      System.out.println (b[i]);  
    }  
}
```

```
Microsoft Windows [Version 10.0.10240.17134]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\shashank>
C:\Users\shashank>
C:\Users\shashank> cd C:\Users\shashank> javac BookMain.java
C:\Users\shashank> java BookMain
Enter the no. of books to be entered:
3
Enter the name:
RD Vaidya
Enter the author:
Jyoti Bhat
Enter the price:
400
Enter the no. of pages:
300
SAVE: RD
VOLUME: Vaidya
PRICE: 400
PAGES: 300
Enter the name:
PINKY
Enter the author:
Jyoti Bhat
Enter the price:
500
Enter the no. of pages:
400
SAVE: PINKY
VOLUME: Jyoti Bhat
PRICE: 500
PAGES: 400
C:\Users\shashank>
```

Q4. 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.

LAB - 4

Date _____

Page _____

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

```
abstract class shape {
```

```
    int l, b, r;
```

```
    Scanner input = new Scanner (System.in);
```

```
    abstract void print Area();
```

```
}
```

```
class Rectangle extends shape {
```

```
    void print Area () {
```

```
        System.out.println ("FOR RECTANGLE");
```

```
        System.out.print ("Enter length and breadth:");
```

```
        l = input.nextInt();
```

```
        b = input.nextInt();
```

```
        System.out.println ("The area of rectangle is: "+ l*b);
```

```
}
```

```
}
```

```
class Triangle extends shape {
```

```
    void Print Area () {
```

```
        System.out.println ("FOR TRIANGLE");
```

```
        System.out.print ("Enter Base And Height:");
```

```
        l = input.nextInt();
```

```
        b = input.nextInt();
```

```
        System.out.println ("The area of Triangle is: " + (l*b / 2));
```

```
}
```

```
}
```

```
class Circle extends shape {
```

```
    void print Area () {
```

```
        System.out.println ("FOR CIRCLE");
```



```

    System.out.print("Enter Radius :");
    r = input.nextInt();
    System.out.println("The area of circle is : " + 3.14f
        * r * r);
}

```

```

}
public class AreaRunner {
    public static void main (String[] args) {
        Rectangle r = new Rectangle();
        r.printArea();
        Triangle t = new Triangle();
        t.printArea();
        Circle c = new Circle();
        c.printArea();
    }
}

```

— X —

The screenshot shows a Windows 10 desktop environment. A terminal window titled "Command Prompt" is open, displaying the execution of a Java program. The program prompts the user to enter dimensions for a rectangle, triangle, and circle, and then calculates their respective areas. The output of the program is as follows:

```

C:\Users\ankush> java AreaRunner.java
FOR RECTANGLE
enter Length and breadth: 30
40
The area of Rectangle is: 500
FOR TRIANGLE
enter base and height: 60
14
The area of Triangle is: 1500
FOR CIRCLE
enter Radius: 87
The area of Circle is: 23796.682
C:\Users\ankush>

```

The desktop background is dark, and the taskbar at the bottom shows the Start button, a search icon, and several application icons including a web browser, a file explorer, and a terminal. The system tray at the bottom right indicates the date and time as 12:38 on 03-11-2020.

Q5. 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: • Accept deposit from customer and update the balance. • Display the balance. • Compute and deposit interest • Permit withdrawal and update the balance • Check for the minimum balance, impose penalty if necessary and update the balance

LAB-45

Date _____ Page _____

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

```
class Account  
{String name, type, accno;  
double balance;
```

```
void deposit()  
{Scanner get = new Scanner(System.in);  
double depo;  
System.out.println("Enter the deposit :");  
depo = get.nextDouble();  
balance = balance + depo;
```

```
void withdraw()  
{Scanner get = new Scanner(System.in);  
double withdraw;  
System.out.println("Enter the amount to withdraw: {<"+balance+"  
?}");  
withdraw = get.nextDouble();  
balance = balance - withdraw;  
System.out.println("Balance: "+balance);  
}  
}
```

```
class Curr_acct extends Account  
{int intr=6;  
boolean cheque=true;  
void dispinc()  
{System.out.println("Balance: "+balance);
```

```

}
void create ()
{ Scanner get = new Scanner (System.in);
  System.out.println ("Name:");
  name = get.next ();
  accno = "current";
  System.out.println ("Account No:");
  acco = get.next ();
  System.out.println ("Balance:");
  Balance = get.next Double ();
}

```

```

void check ()
{ System.out.println ("\nminimum Balance: (" + 5000);
  IF (Balance (5000)
  { System.out.println ("Penalty is imposed please deposit
  minimum" = (5000 - balance + 200) + "Rs\nRs 200 Service charge
  deposit ();
    balance = balance - 200;
  }
  else.
  { System.out.println ("Balance: " + balance + "safe ");
  }
}

```



```
else  
(System.out.println("Balance : "+balance + "Safe");)
```

```
}
```

```
{
```

```
class sav-Acct extends Account
```

```
{double intr = 4;
```

```
boolean cheque = false;
```

```
void dispbIn()
```

```
{System.out.println("Name:");
```

```
name = get.next();
```

```
accno = "~Savings";
```

```
System.out.println("Account No:");
```

```
accno = get.next();
```

```
System.out.println("Balance:");
```

```
balance = get.nextDouble();
```

```
}
```

```
void calcInt()
```

```
{double interest;
```

```
Scanner.get = new Scanner(System.in);
```

```
System.out.println("Enter time:");
```

```
int time;
```

```
time = get.nextInt();
```

```
interest = balance * Math.pow(1 + intr / 100, time) - balance;
```

```
System.out.println("Interest : " + interest);
```

```
balance = balance + interest;
```

```
System.out.println("Balance : " + balance);
```

```
}
```

```
}
```



```

class Bank
{
    public static void main (String args [])
    {
        Scanner get = new Scanner (System.in);
        String type;
        Sav_acct accr = new Curr_acct ();
        System.out.println ("Enter type of account: (Current/
        Savings)");

        type = get.next ();
        if (type.equals ("Savings"))
    
```

```

        accr.create ();
    }
    else if (type.equals ("current"))
        accr.create ();
    }
}

```

```

int ch;
do

```

```

{
    System.out.print ("1. Deposit \n 2. Display Balance \n 3.
    Deposit Interest \n 4. Withdraw \n 5. Check
    \n 6. Cheque Book (under Development) \n 7. Exit");
}

```

```

ch = get.next ();

```

```

switch (ch)

```

```

{
    case 1: if (type.equals ("Savings"))
    
```

```

        accr.deposit ();
    }
    else
    
```

```

        accr.deposit ();
    }
    break
}

```

```

case 2: if (type.equals ("Savings"))
    
```

```

        accr.display ();
    }
}

```

```

else
    accr.deposit();
break;
Case 2: if (type.equals("Savings"))
    accs.dispinc();
else
    accr.dispinc();

break;
Case 3: if (type.equals("Savings"))
    accs.calcint();
else
    System.out.println("This account does not have this provision");
break;
Case 4: if (type.equals("Savings"))
    accs.withdraw();
break;
Case 5: if (type.equals("Savings"))
    System.out.println("This account does not have this provision");
else
    accr.check();
break;
Case 6: if (type.equals("Savings"))
    System.out.println("This account does have this provision");
else
    System.out.println("This account does have this provision");
break;
default: if (ch == 7)
    System.out.println("Enter valid option");
} while (ch != 7);

```

```

C:\Users\UMM\OneDrive\Bank
Enter type of account: (savings/savings)
savings
Enter :
Amount to :
70000
Balance :
5000
1.Deposit
2.Display Balance
3.Deposit Interest
4.Withdrawal
5.Check
6.Cheque Book(under development)
7.Exit
Enter the deposit :
50000
1.Deposit
2.Display Balance
3.Deposit Interest
4.Withdrawal
5.Check
6.Cheque Book(under development)
7.Exit
Balance : 5000.0
1.Deposit
2.Display Balance
3.Deposit Interest
4.Withdrawal
5.Check
6.Cheque Book(under development)
7.Exit
This account does not have this provision
1.Deposit
2.Display Balance
3.Deposit Interest
4.Withdrawal
5.Check
6.Cheque Book(under development)
7.Exit
Enter the amount to withdraw: (50000.0)
50000
Balance : 5000.0
1.Deposit
2.Display Balance
3.Deposit Interest
4.Withdrawal
5.Check
6.Cheque Book(under development)
7.Exit

```

```

6.Cheque Book(under development)
7.Exit
Enter the amount to withdraw: (50000.0)
50000
Balance : 5000.0
1.Deposit
2.Display Balance
3.Deposit Interest
4.Withdrawal
5.Check
6.Cheque Book(under development)
7.Exit
Minimum Balance : 5000
Balance : 5000.0000
1.Deposit
2.Display Balance
3.Deposit Interest
4.Withdrawal
5.Check
6.Cheque Book(under development)
7.Exit
C:\Users\UMM\OneDrive\

```

Q6. 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.

Cie

```
package cie;  
import java.util.*;  
public class Student {  
    public String uen;  
    public String name;  
    public int sem;
```

```
    public void read()  
    {
```

```
        Scanner sc = new Scanner(System.in);  
        System.out.println("Enter the uen of the  
        student");  
        uen = sc.next();  
        System.out.println("Enter the name of the  
        student");  
        name = sc.next();  
        System.out.println("Enter the sem of the  
        student");  
        sem = sc.nextInt();
```

```
    }  
}
```

package cie;
import java.util.*;

public class internals extends Student *

public int[] Ta = new int[5];
public void read()

✕

super.read();

Scanner sc = new Scanner(System.in);

System.out.println("Enter the cie ~~student~~
marks of 5 courses");

for (int i = 0; i < 5; i++)

✕

System.out.println("Enter marks of the
course " + (i + 1));

a[i] = sc.nextInt();

✕

✕

public void display()

✕

System.out.println("USN of the student is"
+ usn);

System.out.println("Name of the student is"
+ name);

System.out.println("Semester of the student
is " + sem);

✕

✕

```
package sel;  
import java.util.*;  
import java.io.*;  
import java.lang.*;
```

```
public class external extends cie.student
```

```
{  
    public int [] b = new int [5];  
    public int [] mark;  
    public void read () {
```

```
Scanner sc = new Scanner (System.in);  
for (int i = 0; i < 5; i++)
```

```
{  
    System.out.println ("Enter the sel marks of  
the course" + (i+1));  
    b[i] = sc.nextInt();
```

```
}
```



```
import java.util.*;  
import java.io.*;  
import java.lang.*;  
import cil.*;  
import sel.*;
```

```
public class student_end {  
    public static void main (String [] args)
```

{

```
    int n;  
    Scanner sc = new Scanner (System.in);  
    int final-mark;  
    System.out.println ("Enter the number of Students");  
    n = sc.nextInt ();  
    internals [] in = new internals [n];  
    externals [] ex = new externals [n];  
    internals ob1 = new internals ();  
    externals ob2 = new externals ();  
    ob2.mar = new int [n];
```

```
    for (int i = 0; i < n; i++)  
        {  
            System.out.println ("Enter the details of  
            student " + (i+1));  
            in[i] = new internals ();  
            in[i].read ();  
            ex[i] = new externals ();  
            ex[i].read ();
```

}


```
system.out.println();
```

```
for(int i=0; i<n; i++)
```

```
{  
    System.out.println("x Details of the  
    students " + (i+1));
```

```
    System.out.println("USN of the Student is "  
    + in[i].usn);
```

```
    System.out.println("Name of the Student is "  
    + in[i].name);
```

```
    System.out.println("Semester of the Student is "  
    + in[i].sem);
```

```
    for(int j=0; j<5; j++)
```

```
{  
    x Final mark = in[i].a[j] + ((ex[i].b[j])/2);  
    System.out.println("Final marks of the  
    student " + (i+1) + " " + "in course " +  
    (j+1) + " " + final mark);
```

```
}  
    System.out.println();
```

```
}
```

```
}
```

```
}
```

Q7. Write a program to demonstrate generics with multiple object parameters

LAB-7

Date _____ Page _____

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

```
class gen<T>
```

```
{
```

```
    Tob;
```

```
    gen (To)
```

```
{
```

```
    ob = 0;
```

```
}
```

```
T getob ()
```

```
{
```

```
    return ob;
```

```
}
```

```
void showtype ()
```

```
{
```

```
    System.out.println ("Type of T is"
```

```
+ ob.get class ().get Name ());
```

```
}
```

```
}
```

```
class generic
```

```
{
```

```
Public static void main (String[]
```

```
args)
```

```
{
```

```
String n;
```

```
Scanner sc = new
```

```
Scanner (System.in);
```

```
System.out.println ("Enter the Integer Number to Be Displayed
```


using the generic style");

n = sc.next();

gen < Integer > ob1 = new

gen < Integer > (Integer.parseInt(n));

ob1.showtype();

int val = ob1.gettob();

System.out.println("Value is:" + val);

System.out.println();

System.out.println("Enter the string to Be Displayed using the
generic style");

n = sc.next();

gen < String > ob2 = new

gen < String > (n);

ob2.showtype();

String x = ob2.gettob();

System.out.println("Value:" + x);

System.out.println();

System.out.println("Enter the Double Number to Be Displayed
using the generic style");

n = sc.next();

gen < Double > ob3 = new

gen < Double > (Double.parseDouble(n));

ob3.showtype();

double ans = ob3.gettob();

System.out.println("Value:" + ans);

}

}

LAB-8

Date _____ Page _____

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

```
class Wrongage extends Exception
```

```
{
```

```
    public int a;
```

```
    Wrongage (int x)
```

```
{
```

```
        a = x;
```

```
}
```

```
    public String toString ()
```

```
{
```

```
        return "Wrongage [" + a + "];
```

```
}
```

```
}
```

```
class Father
```

```
{
```

```
    public int age;
```

```
    Father (int a)
```

```
{
```

```
        age = a;
```

```
}
```

```
    public void check () throws Wrongage
```

```
{
```

```
        System.out.println ("Checking the age of the Father");
```

```
        System.out.println ();
```

```
        if (age < 0)
```

```
            throw new Wrongage (age);
```

```
        System.out.println ("Correct Age");
```



```
}  
}  
Class Son extends Father  
{  
    Public int son_age;  
    Son (int Fa_age, int i)  
    {  
        Super (Fa_age);  
        son_age = i;  
    }  
    Public void check () throws Wrongage  
    {  
        Super (Fa_age);  
        son_age = i;  
    }  
    Public void check () throws Wrongage  
    {  
        Super.check ();  
        System.out.println ();  
        System.out.println ("checking the age of the son");  
        System.out.println ();  
        if (son_age < 0 || son_age > age)  
            throw new Wrongage (son_age);  
        System.out.println ("correct Age");  
    }  
}  
Public class errortest  
{  
    Public static void main (String [] args)
```

```
{
int So_age, Father_age;
Scanner Sc = new
Scanner (System.in);
System.out.println ("Enter the Age of the Father");
Father_age = Sc.nextInt();
System.out.println ("Enter the age of the son");
    So_age = Sc.nextInt();
    Son S = new
    Son (Father_age, So_age);
    try
    {
        S.check();
    } catch (Wrongage W)
    {
        System.out.println ("Exception: " + W);
    }
}
}
```


LAB-9

Date _____ Page _____

```
class T implements Runnable
```

```
{
```

```
    Thread t;
```

```
    String name;
```

```
    int time;
```

```
    T (String nm, int tm)
```

```
{
```

```
        name = nm;
```

```
        time = tm;
```

```
        t = new Thread (this, nm);
```

```
}
```

```
    public void run ()
```

```
{
```

```
        try
```

```
{
```

```
            for (int n = 10; n > 0; n--)
```

```
{
```

```
                System.out.println (name);
```

```
                Thread.sleep (time);
```

```
}
```

```
}
```

```
        catch (InterruptedException ie)
```

```
{
```

```
            System.out.println ("Interruption occur");
```

```
}
```

```
}
```

```
}
```

```
class H Thread
```

```
{
```

```
    public static void main (String ss[])
```

```
{  
T bms = new T ("BMS College of Engineering", 10000);  
T CSE = new T ("CSE", 2000);  
bms.t.start ();  
CSE.t.start ();  
}
```

```
}
```

```
}
```



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

```
class SampleDialog extends Dialog implements  
ActionListener &
```

```
Diside AWT bld;
```

```
& SampleDialog(Frame parent, String title.)
```

```
super (parent, title, false);
```

```
bld = (Diside AWT) parent;
```

```
setLayout (new FlowLayout ());
```

```
setSize (500, 300);
```

```
Button b;
```

```
add (b = new Button ("OK"));
```

```
b.addActionListener (this);
```

```
>
```

```
public void actionPerformed (ActionEvent ae)
```

```
& dispose ();
```

```
>
```

```
public void paint (Graphics g) &
```

```
color c = new Color (0, 0, 0);
```

```
g.setColor (c);
```

```
g.drawString (bld.msg, 5, 50);
```

```
>
```

```
>
```

class DivideAWT extends Frame implements
ActionListener {

Button b1;
TextField t1, t2;
String click = "", msg = "";

public DivideAWT() {
 setLayout(new FlowLayout());
 t1 = new TextField(12);
 add(t1);
 b1 = new Button("/");
 add(b1);
 t2 = new TextField(12);
 add(t2);
 b1.addActionListener(this);
 addWindowListener(new MyWindowAdapter()); }
}

public void actionPerformed(ActionEvent ae)
{
 if (ae.getActionCommand().equals("/"))
 {
 click = ae.getActionCommand();
 repaint();
 }
}

```
public void paint(Graphics g)
{
    int a, b;
    if (t1.getText().equals("") || t2.getText().equals(""))
    {
        msg = "Fields cannot be empty";
    }
    else
    {
        try
        {
            a = Integer.parseInt(t1.getText());
            b = Integer.parseInt(t2.getText());
            int div = a/b;
            if (click.equals("/"))
            {
                msg = "Result of: " + a + "/" + b + " = " + div;
            }
        }
        catch (NumberFormatException ex1)
        {
            msg = "Enter valid nos";
        }
        catch (ArithmeticException ex2)
        {
            msg = "Division by zero not possible";
        }
        SampleDialog d = new SampleDialog(this, "Dialog");
        d.setVisible(true);
    }
}
```


public static void main (String args [])

↳
JFrame t = new JFrame();
t.setSize(new Dimension(1000,1000));
t.setTitle("DIVISION OF NUMBERS");
t.setVisible(true);
↳

↳
class MyWindowAdapter extends WindowAdapter

↳
public void WindowClosing(WindowEvent we)

↳
System.exit(0);
↳

XXXXXXXXXXXXXXXXXX
