

LAB 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.

Date \_\_\_\_\_

Expt. No. \_\_\_\_\_

Page No. \_\_\_\_\_

$\alpha \text{AB}$  (week 3).

→ ① Quadratic eqn program

public class Main {

    public static void main (String args[]) {

        double x1, x2;

        Scanner sc = new Scanner (System.in);

        System.out.println ("Enter a,b,c value : ");

        double a = sc.nextDouble();

        double b = sc.nextDouble();

        double c = sc.nextDouble();

        double d =  $b^2 - 4ac$ ;

        if (d < 0) {

            System.out.println ("Imaginary root");  
            System.out.println (" $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ ");

Teacher's Signature : \_\_\_\_\_

`System.out.println ("Imaginary root2: " +  
 (-b/2*a) + "j" + Math.sqrt((-D)/2*a))  
 + "i")j`

{ else }

$x_1 = (-b + \sqrt{D})/2*a - j$

$x_2 = (-b - \sqrt{D})/2*a - j$

`System.out.println ("real root 1: +x1)j`

`System.out.println ("real root2: +x2)j`

{}

{}

{}

## BASIC ALGO / LOGIC

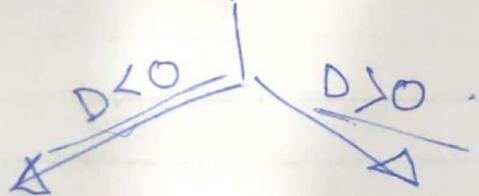
① Quadratic determinant :  $b^2 - 4ac$

where  $a, b, c$  are variables.

$$\underline{ax^2 + bx + c}$$

INPUT  $(a, b, c)$

$$b^2 - 4ac = D$$



$$\frac{-b \pm \sqrt{-D}}{2a} = \begin{cases} \text{(some form)} \\ \text{of fraction) } \end{cases} \frac{-b + \sqrt{D}}{2a} = \text{some value}$$

OUTPUT  $a \quad b \quad c$

$$\begin{aligned} \text{Imaginary root 1} &= -12.5 + 21.65 \\ \text{Imaginary root 2} &= -12.5 - 21.65 \end{aligned}$$

$$\left| \begin{array}{l} \frac{1}{4} \\ \text{real root 1: } -1.0 \\ \text{real root 2: } -4.0 \end{array} \right.$$

```
1 public class Main{  
2     public static void main(String args[]){  
3         double r1,r2;  
4         Scanner sc = new Scanner(System.in);  
5         System.out.println("Enter a,b,c value:\t");  
6         double a = sc.nextDouble();  
7         double b = sc.nextDouble();  
8         double c = sc.nextDouble();  
9         double D = b*b-4*a*c;  
10        if(D<0) {  
11            System.out.println("Imaginary root1: "+(-b/2*a)+"+"+(Math.sqrt(-D)/2*a)+"i");  
12            System.out.println("Imaginary root2: "+(-b/2*a)+"-"+(Math.sqrt(-D)/2*a)+"i");  
13        }else {  
14            r1 = (-b+Math.sqrt(D))/2*a;  
15            r2 = (-b-Math.sqrt(D))/2*a;  
16            System.out.println("real root1: "+r1);  
17            System.out.println("real root2: "+r2);  
18        }  
19    }  
20 }  
21 }  
22 }  
23 }  
24 }
```

input

```
Enter a,b,c value:  
5  
5  
5  
Imaginary root1: -12.5+ 21.65063509461097i  
Imaginary root2: -12.5- 21.65063509461097i
```

# LAB PROGRAM 2→

Develop a java program to create a class student with members usn, name and credits and array marks use methods to display and accept details.

Expt. No. ..... Date .....  
Page No. .....

Q) SGPA Program -

```
import java.util.Scanner;
class Student
{
    private String USN;
    private String name;
    private int n;
    private double SGPA = 0;
    private int totalCredits = 0;

    Scanner ss = new Scanner(System.in);

    void Details()
    {
        System.out.println("Enter USN of student");
        USN = ss.nextLine();

        System.out.println("Enter name");
        name = ss.nextLine();

        System.out.println("Enter no of subjects");
        n = ss.nextInt();

        int credits[] = new int[n];
    }
}
```

Teacher's Signature : \_\_\_\_\_

double marks [] = new double [n];  
 System.out.println ("Enter details of sub");

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

{ System.out.println ("Enter credits allotted to sub" + (i+1));

marks[i] = ss.nextInt();

Calculate (credits[i], marks[i], i);

} }

void Calculate (int credits, double marks, int j)

{ totalCredits = totalCredits + credit;

if (mark >= 90 && mark <= 100) -

SGPA = SGPA + (10 \* credit);

else if (mark >= 80 && mark <= 89)

SGPA = SGPA + (9 \* credit);

else if (mark >= 70 && mark <= 79)

SGPA = SGPA + (6 \* credit);  
 Teacher's Signature : \_\_\_\_\_

else if (marks >= 60 && mark <= 69)

$$SGPA = SGPA + (7 * \text{credits})$$

else if (marks >= 50 && mark <= 59)

$$SGPA = SGPA + (6 * \text{credits})$$

else if (marks >= 40 && mark <= 39)

$$SGPA = SGPA + (5 * \text{credits})$$

else

System.out.println("Failed subject " + (j + 1));

}

void Display()

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

System.out.println("USN: " + USN);

System.out.println("CGPA of student " +  
(SGPA / total credits));

Teacher's Signature:

Date \_\_\_\_\_

Expt. No. \_\_\_\_\_

Page No. \_\_\_\_\_

Class Main

{ Public static void main ( String args [ ] )  
{

student s1 = new Student ();

s1. Details (); // similar to  
s1. Display () calling function

05  
J

}

X

Teacher's Signature : \_\_\_\_\_

Output

Enter USN of student  
151 -

Enter name of student

Shivanshu P

Enter no of subj

① - 3

Credits allotted to sub 1

5

Enter marks in sub 1.

80

Credits allotted to sub 2

4

Enter marks in sub 2

75

Enter ~~marks~~ credits allotted to sub 3

4 -

Enter marks in sub 3,

72

Details of student

~~Name~~ Name: Shivanshu P -

USM 8 151 -

SGPA 8 8.3846

basic logic for  
input

name —

VSN —

~~marks~~ subj — n

marks [ ] - - - marks [ $\leq n$ ]

credits [ ] - - - credits [ $\leq n$ ]

basic logic of program & marks [0+1]

↓  
marks [ $\leq n$ ]

total ↓ using if-else cond  
~~marks~~ SCIP A

$$= (\text{SCIP A} + 'x' * \text{credit})$$

↓

$$\text{SCIP A} = \frac{\text{total SCIP A}}{\text{total credits}}$$

'x' ⇒ Grade Point

10 - 5 or else fail.  
depending on if-else

C:\Users\shivanshu\Desktop> java lab

Enter USN of the student

151

Enter Name of the student

shivanshu p

Enter no of subjects

5

Enter details of the subjects:

Enter credits allotted to the subject 1

5

Enter marks in the subject 1

80

Enter credits allotted to the subject 2

4

Enter marks in the subject 2

75

Enter credits allotted to the subject 3

4

Enter marks in the subject 3

72

Enter credits allotted to the subject 4

3

Enter marks in the subject 4

71

Enter credits allotted to the subject 5

1

Enter marks in the subject 5

58

Enter credits allotted to the subject 6

3

Enter marks in the subject 6

88

Details of the Student

Name :shivanshu p

USN: 151

SGPA of student 8.523809523809524

C:\Users\shivanshu\Desktop>

LAB3 → 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.

```
import java.util.Scanner;  
class Main {  
    public static void main(String args[]) {  
        Scanner sc = new Scanner(System.in);  
        int n = sc.nextInt();  
        Book B[] = new Book[n];  
        for (int i=0; i<n; i++) {  
            System.out.println("Book " + B[i]);  
            System.out.print("\n");  
        }  
    }  
}
```

// Algo → create a class Book with constructor function to string() which shall return string  
class BookName Eg call the Book Class  
code → // contd.

→ class Book {

int

String name, author; price, nPages;

Book()

2 Scanner sc = new Scanner(system.in);

System.out.println("Enter the name of book");

name = sc.nextLine();

System.out.println("Enter the author of book");

author = sc.nextLine();

System.out.println("Enter price of book");

price = sc.nextInt();

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

nPages = sc.nextInt();

}

public String to String() {

return ("Name of book : " + this.name +  
" In Author : " + author + " In Price : " + price  
+ " In Pgno : " + this.pgno + );

}

}

## Main.java

```
9 import java.util.Scanner;  
10
```

Enter the Number of Books:

2

Enter the Name of The book: da  
Enter the Author of book da: a  
Enter the Price of book da: 12  
Enter the Number of pages: 11

Enter the Name of The book:

Enter the Author of book : pas  
Enter the Price of book : 24  
Enter the Number of pages: 1

Name of Book : da

Author : a

Price : 12

Number of Pages: 11

Name of Book :

Author : pas

Price : 24

Number of Pages: 1

LAB4 DOWN → Program to create a abstract class shape with 2 intergers and method printArea() and 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.

## Area Abstract

Date : \_\_\_\_\_  
Page No. : \_\_\_\_\_

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

```
class Shape {
```

```
    int slenght;
```

```
    int shreath;
```

```
    void printArea() {
```

```
}
```

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

```
{
```

```
class Rectangle extends Shape {
```

```
    void printArea() {
```

```
        System.out.println("Enter the length of rect");
```

```
        Slenght = Sinp.nextInt();
```

```
        System.out.println("Enter breath of rect");
```

```
        Shreath = Sinp.nextInt();
```

```
        System.out.println("The area of Rectangle  
is :" + (Slenght * Shreath));
```

```
}
```



Class Triangle extends Shape {

void paintArea() {

System.out.println("Enter the height : ");

Slenght = Sinp.nextInt();

System.out.println("Enter the base : ");

Sbreadth = Sinp.nextInt();

"System.out.println("The Area of triangle  
is : " + (0.5 \* Sbreadth \* Slenght));

}

}

Class Circle extends Shape {

void paintArea() {

System.out.print("Enter the radius : ");

Slenght = Sinp.nextInt();

System.out.println("The area of circle  
is : " + (3.143 \* Slenght \* Slenght));

}

}



public class App {

public static void main (String [] args)

throws Exception {

    Rectangle R1 = new Rectangle ();  
    }

    Triangle T1 = new Triangle ();  
    }

    Circle C1 = new Circle ();  
    }

    R1.paintArea ();  
    }

    T1.paintArea ();  
    }

    C1.paintArea ();  
    }

}

3

o

—

OOJLABS - java1/src/ App.java - Eclipse IDE

Edit Source Refactor Navigate Search Project Run Window Help

Transactions.java App.java

```
1 package java1;
2
3
4
5 Problems Javadoc Declaration Console
6 terminated> App [Java Application] C:\Users\shivanshu.p2\pool\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (06-Nov-2020, 3:11:08 pm - 3:11:33 pm)
7 Enter the length of Rectangle
8 10
9 Enter the breadth of Rectangle
10 20
11 The AREA of RECTANGLE is : 200
12 Enter the Height :
13 11
14 Enter the Base :
15 10
16 The AREA of TRIANGLE is : 55.0
17 Enter the Radius :
18 7
19 The AREA of CIRCLE is : 154.00699999999998
```

## LAB 5 DOWN-→

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. -->

# BANK

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

```
class Bank
```

{

```
    int depositbal;
```

```
    int withdrawbal;
```

```
    String customername;
```

```
    String accountno;
```

```
    String accounttype;
```

```
    int Balenue = 27800;
```

```
    void accept()
```

{ Scanner s = new Scanner (System.in);

System.out.println ("Enter customer name\n");

```
customername = s.next();
```

System.out.println ("Enter acc number\n");

```
accountno = s.next();
```

System.out.println ("Enter acc type\n");

```
accounttype = s.next();
```

}



void display()

{ System.out.println ("CUSTOMER NAME : " + customer  
- name + "\n"); }

System.out.println ("CUSTOMER Acc NO : " + accountno)

System.out.println ("CUSTOMER Acc TYPE : " + acc\_type);

System.out.println ("BALANCE : " + balance);

}

} // end of class .i

Class current extends Bank {

int updatedbalance;

int AfterWithdrawing;

int updatedlostbalance;

int cdeposit() {

updatedbalance = Balance + depositbalance;

return updatedbalance;

}



int c.withdraw() {

Afterwithdraw = ((updated balance) - withdrawn  
balance) ;

}

int minimum ()

{ if (After withdraw) <= (2000))

{ updatedlostbalance = ((After withdraw) - (200));

System.out.println ("You have minimum but  
below 2000 so u have lost 200Rs");

return updatedlostbalance; }

else

return Afterwithdraw;

}

class Saurav extends Bank {

int updatedbalance;

int Afterwithdraw;

int updatedlostbalance;

scop - bas();



subtotal balance = Balance + deposit balance

return subtotal balance;

{

int interest();

{ double r = 0.08; }

int n = 12;

int t = 5;

compound interest = (int) ↗

((subtotal balance) \* (Math · pow ((1+r/n)  
+ (n\*t)))) ;

return compound interest;

{

int Smithba(); {

- Afterwithdraws = (compoundinterest) - (balance);

return Afterwithdraws;

int minimum();

{ if ((Afterwithdraws) < -(1000))

$S \cdot \text{updatedlostbalance} = ((\text{After withdraw}) - 100);$

return updatedlostbalance;

}

else

return Afterwithdraw;

Class Transactions {

public static void main (String args ()) {

Scanner s = new Scanner (System.in);

current CA = new current();

CA.accept();

System.out.println ("Enter the money you  
want to deposit in current acc");

CA.depositbalance = s.nextInt();

System.out.println ("After your deposit,  
of " + CA.depositbalance + " New Balance is:  
+ CA.cdeposit());

System.out.println ("Enter the money you  
want to withdraw");

CA.withdrawbalance = s.nextInt();



System.out.println ("After your withdrawal of "+  
CA.withdrawBalance + "\n Now your total balance  
is Rs - " + CA.cWithBal() );

System.out.println ("After checking if you have  
minimum balance is Rs " + CA.minimum() );

SavAcc SA = new SavAcc();  
SA.accept();

System.out.println ("Enter money u want to  
deposit in Saving acc");

SA.depositBalance = sc.nextInt();

SA.display();

System.out.println ("After your deposition of " + SA.  
"\n Now your total bal is Rs " + SA.depositBal());

System.out.println ("After interest your total "  
+ SA.interest());

System.out.println ("After checking if u have  
minimum balance you are not updated  
total balance hence is Rs - " + SA.minimum());

3

3

```
File Edit Source Refactor Navigate Search Project Run Window Help
Problems Javadoc Declaration Console
terminated Transactions [Java Application] C:\Users\shivanshu\p2\pool\plugins\org.eclipse.jdt.openjdkhotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre
Enter the customer name
shivanshu
Enter the Account Number
911
Enter the Account type
current
Enter the money u want to deposit in current account in rupees
1000
CUSTOMER NAME : shivanshu
ACCOUNT NUMBER : 911
ACCOUNT TYPE : current
After your deposition of 1000
Now your total balance is RS-28800
Enter the money you want to withdraw in rupees
800
After your withdrawal of 800
Now your total balance is RS-28000
After checking if u have minimum balance are not your updated total balance is RS-28000
Enter the customer name
shivanshu
Enter the Account Number
911
Enter the Account type
savings
Enter the money u want to deposit in Saving account
10000
CUSTOMER NAME : shivanshu
ACCOUNT NUMBER : 911
ACCOUNT TYPE : savings
After your deposition of 10000
Now your total balance is RS-37800
After interest ur updated balance is RS-56316|
Enter the money you want to withdraw in Saving account
316
After your withdrawal of RS-316
Now your total balance is RS-56000
After checking if u have minimum balance are not your updated total balance is RS-56000
<
```

```
ACCOUNT TYPE : savings
After your deposition of 10000
Now your total balance is RS-37800
After interest ur updated balance is RS-56316|
Enter the money you want to withdraw in Saving account
316
After your withdrawal of RS-316
Now your total balance is RS-56000
After checking if u have minimum balance are not your updated total balance is RS-56000
<
```







LAB 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.

Expt. No.....	Page No.....
<p>CIE Package (Student Class) -</p> <p>package cie;</p> <p>public class Student {</p> <p>    public int usn;          public string name;          public int sem;</p> <p>    public Student(int usn; string name; int sem)</p> <p>        // { this.usn = usn;              this.name = name;              this.sem = sem;} }  }</p> <p>package cie;</p> <p>public class Internals extends Student {</p> <p>    public int [ ] ie_marks = new int [5];</p> <p>    public Internals (int usn, string name, int sem, int [ ] ie_marks) {</p> <p>        super(usn, name, sem);              Teacher's Signature :              this.ie_marks = ie_marks; } }</p>	

Date .....

Expt. No. ....

Page No. ....

SEE package

package see;  
import java.\*;  
public class External extends Student {

int [] marks = new int [5];

public External (int vno, String name, int  
sum, int [] marks) {

super (vno, name, sum);

this. marks = marks;

}

}

Class main

```
Enter the Number of students : 10
Enter the details of the student 1:
Enter usn of the student : 141
Enter name of the student : SAM
Enter semester of the student : 3
Enter the CIE marks :
Enter marks of the course 1: 12
Enter marks of the course 2: 13
Enter marks of the course 3: 14
Enter the SEE marks :
Enter the SEE marks of the course 1: 60
Enter the SEE marks of the course 2: 45
Enter the SEE marks of the course 3: 67
Enter the details of the student 2:
Enter usn of the student : 120
Enter name of the student : shivanshu
Enter semester of the student : 3
Enter the CIE marks :
Enter marks of the course 1: 15
Enter marks of the course 2: 20
Enter marks of the course 3: 38
Enter the SEE marks :
Enter the SEE marks of the course 1: 89
Enter the SEE marks of the course 2: 78
Enter the SEE marks of the course 3: 90
```

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

papergrid

Date: / /

Lab-7

import java.util.Scanner

class Generic < T >

T val ;

void generic (T gen) {

val = gen ; }

T display () {

return val ; }

public class Transaction {

public static void main (String args []) {

Scanner in = new Scanner (System.in);

Generic<Integer> Roll no = new Generic<Integer>();

Generic<String> Name = new Generic<String>();

System.out.println ("Enter name : ");

Name name = in.nextLine();

Name Generic<Name> ;

```
System.out.println("Enter roll no:");
```

int valno = inf.nesilt();

~~B~~ Roll no. → Genrich (roll no.) ;

```
System.out.println("The name of roll no is "+  
+ Name.display() + " " + RollNo.display());
```

inf.close(); } }

OOJLAB - java1/src/java1/Transactions.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Transactions.java

```
4 import java.util.*;
5
6 class Genrics<T>{
7     T var1;
8
9     void Genrics(T gvar){
10         var1 = gvar;
11     }
12
13     T Gdisplay(){
14         return var1;
15     }
16 }
17
18
19 public class Transactions {
20     public static void main(String[] args) throws Exception {
21         System.out.println("Hello, World!");
22
23         Scanner Minp = new Scanner(System.in);
24
25         Genrics<Integer> Rollno= new Genrics<Integer>();
26         Genrics<String> Name = new Genrics<String>();
27     }
28 }
```

Problems Javadoc Declaration Console

```
<terminated> Transactions [Java Application] C:\Users\shivanshu.p2\pool\plugins\org.eclipse.jdt\openjdkhotspot\re\full\win32\x86_64_14.0.2.v20200815-0932\re\bin\javaw.exe (27-Nov-2020, 12:50:44 pm - 12:51:33 pm)
Hello, World!
Enter Name of Student
shivanshu
Enter USN of Student
151
The student details are :
Name : shivanshu
USN : 151
```

LAB 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=father's age.

Expt. No. ....

Page No. ....

CIE Package (Student Class) -

package cie;

public class Student {

public int usn;  
public String name;  
public int sem;

public Student(int usn; String name; int sem)

// {  
//     this.usn = usn;  
//     this.name = name;  
//     this.sem = sem;}  
// }  
// }

package cie;

public class Internals extends Student {

public int IT\_marks = new int [5];

public Internals (int usn, String name, int sem, int [ ] IT\_marks) {

super(usn, name, sem);

Teacher's Signature : \_\_\_\_\_

This.Cmarks = IT\_marks; }

} (  $f \leq 0 \text{ || } s \leq 0$  ) {

throws new MyException( ) ; }

else if (  $s > f$  ) {

throws new MyException( s, f );

}

else if (  $s > f$  ) {

throws new MyException(

else {

System.out.println ( "Failure Age : " + f );  
System.out.println ( "Success Age : " + s );

}

}

}

0

```
Enter father's age
2
Enter son's age
19
Father's age is :2
Exception in thread "main" java.lang.ArithmetcException: Son's age cannot be greater than father's age
    at java/lab6main.Son.show(LAB8.java:39)
    at java/lab6main.LAB8.main(LAB8.java:14)
```

```
Enter father's age
57
Enter son's age
27
Father's age is :57
Father's age is :57 and Son age is 27
```

LAB 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.

Date .....

Expt. No. ....

Page No. ....

class NewThread implements Runnable {

    Thread t;

    NewThread() {

        t = new Thread(this, "Demo Thread");

        t.start();

    public void run() {

        try {

            for (int i=25; i>0; i--) {

                System.out.println("CSE");

                Thread.sleep(2000);

    }

    catch (InterruptedException e) {

        System.out.println("Thread 2 interrupted");

}

Teacher's Signature : \_\_\_\_\_

`System.out.println("Exiting thread 2.");`

`}`

`{`

`class Thread1 {`

`public static void main (String args []) {`  
`new NewThread();`

`try {`

`for (int i = 5; i > 0; i--) {`

`System.out.println ("BMS College of engineering")`

`Thread.sleep (10000); }`

`catch (InterruptedException e) {`

`System.out.println ("Main thread interrupted");`

`}`

`System.out.println ("Main thread exiting.");`

Teacher's Signature : \_\_\_\_\_

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

LAB10:- 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 ArithmeticException. Display the exception in a message dialog box.

papergrid

Date: / /

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

public class lab10 extends Frame implements ActionListener {
    JTextField num1, num2;
    Label ob;
    Button n;

    lab10() {
        num1 = new JTextField();
        num1.setBounds(50, 100, 200, 25);

        num2 = new JTextField();
        num2.setBounds(50, 150, 200, 25);

        ob = new Label();
        ob.setBounds(50, 150, 200, 50);

        n = new Button("Divide");
        n.setBounds(50, 200, 100, 50);
        n.addActionListener(this);

        add(n);
        add(num1);
        add(num2);
        add(ob);
        setSize(800, 800);

        setLayout(null);
        setVisible(true);
    }

    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == n) {
            try {
                int a = Integer.parseInt(num1.getText());
                int b = Integer.parseInt(num2.getText());
                ob.setText("Result: " + (a / b));
            } catch (NumberFormatException ex) {
                JOptionPane.showMessageDialog(this, "Please enter integers!");
            }
        }
    }
}
```

3

public void calculatePerformed( ActionEvent ) {

try {

String n1 = num1.getText();  
String n2 = num2.getText();

ob.setText ("Quotient is " + Integer.parseInt(n1) /  
Integer.parseInt(n2));

}

Catch {

ob.setText ("Cannot divide a non integer");

}

Catch ("ArithmeticException") {

ob.setText ("Cannot divide");

}

} }

public static void main (String[] args) {

new lab10(); }

}



Applet Viewer: ...



Applet

Dividend

30

Divisor

0

Result

Can't be divided by Zero



Applet Viewer: ...



Applet

Dividend

30

Divisor

10

Result

3