

OOJ Lab Record-

Anitej prasad

1BM19CS194

3-D

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

Lab 1 Program 9-10-20

```
import java.util.Scanner;
class quadratic
{
    public static void main (String args []) {
        double a, b, c, disc;
        double r1, r2;
        Scanner inp = new Scanner ( System.in );
        System.out.println("Enter a value for a, b, c: ");
        a = inp.nextDouble();
        b = inp.nextDouble();
        c = inp.nextDouble();
        disc = ((b*b) - (4*a*c));
        if (disc > 0) {
            System.out.println("roots are real");
            r1 = (-b + Math.sqrt(disc)) / (2*a);
            r2 = (-b - Math.sqrt(disc)) / (2*a);
            System.out.println("r1 = " + r1 + " r2 = " + r2);
        }
        else if (disc == 0) {
            System.out.println("roots are real and equal");
            r1 = r2 = -b / (2*a);
        }
        else {
            System.out.println("real roots don't exist");
        }
    }
}
```

```
c:\workspace>javac quadratic.java

c:\workspace>java quadratic
Enter values for a,b,c:

1 -4 -10
roots are real
r1=5.741657386773941r2=-1.7416573867739413
c:\workspace>
```

```
c:\workspace>javac quadratic.java

c:\workspace>java quadratic
Enter values for a,b,c:

1 -3 -10
roots are real
r1=5.0r2=-2.0
c:\workspace>javac quadratic.java

c:\workspace>java quadratic
Enter values for a,b,c:

1 2 3
no real roots exist
```

LAB 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.

Anilaj Prasad
18M19C8194
Aid

Lab 2 program

```
import java.util.Scanner;
class Student
{
    String name;
    String usn;
    int marks[] = new int [5];
    int credits[] = new int [5];
    int i, n;
    int grade = 0;
    double total = 0;
    void get_details()
    {
        Scanner in = new Scanner (System.in);
        System.out.println ("Enter Student Name :");
        name = in.next();
        System.out.println ("Enter the USN:");
        usn = in.next();
        System.out.println ("Enter the no. of subjects :");
        n = in.nextInt();
        System.out.println ("Enter credits");
        for (i = 0; i < n; i++)
        {
            System.out.println ("Enter credits :");
            for (i = 0; i < n; i++)
            {
                credits[i] = in.nextInt();
            }
        }
        System.out.println ("Enter marks:");
        for (i = 0; i < n; i++)
```

```
System.out.println(" Marks in subject " + (i+1) + " : ");
marks[i] = in.nextInt();
```

```
}
```

```
}
```

```
void calculateScore()
```

```
{
```

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

```
{
```

```
if (marks[i] >= 90 && marks[i] <= 100)
```

```
grade = 10;
```

```
else if (marks[i] >= 80 && marks[i] < 90)
```

```
grade = 9;
```

```
else if (marks[i] >= 70 && marks[i] <= 80)
```

```
grade = 8;
```

```
else if (marks[i] >= 60 && marks[i] <= 70)
```

```
grade = 7;
```

```
else if (marks[i] >= 50 && marks[i] <= 60)
```

```
grade = 6;
```

```
else if (marks[i] >= 40 && marks[i] <= 50)
```

```
grade = 5;
```

```
else if (marks[i] >= 30 && marks[i] <= 40)
```

```
grade = 4;
```

```
else if (marks[i] >= 20 && marks[i] <= 30)
```

```
grade = 3;
```

```
else if (marks[i] >= 10 && marks[i] <= 20)
```

```
grade = 2;
```

```
else if (marks[i] >= 0 && marks[i] < 10)
```

```
else
```

```
System.out.println(" Invalid marks entered ");
```

```
total = total + (grade * credits[i]);
```

```
}
```

```

        total = total / 20 ;
        System.out.println("sgpa=" + total);
    }

    void student_information ()
    {
        System.out.println("Name: " + name);
        System.out.println("USN: " + usn);
        System.out.println("Marks & credits of all subjects:");
        for (i = 0; i < n; i++)
        {
            System.out.println("Subject: " + (i+1) + " : ");
            " (" + " Marks: " + marks[i];
            " (" + " Credits: " + credits[i];
        }

        calculate_sgpa();
    }

    public static void main(String args[])
    {
        Student s = new Student();
        s.get_details();
        s.calculate_sgpa();
        s.student_information();
    }
}

```



```
Enter Student Name:
ANITEJ
Enter the USN:
1BM19CS194
Enter the no. of subjects:
5
Enter Subject credits:
Credits for subject1:
3
Credits for subject2:
2
Credits for subject3:
4
Credits for subject4:
3
Credits for subject5:
2
Marks in subject1:
56
Marks in subject2:
65
Marks in subject3:
76
Marks in subject4:
57
Marks in subject5:
90
```

```
Name:ANITEJ
USN:1BM19CS194
Marks & Credits of all subjects:
subject:1:
Marks:56
Credits:3
subject:2:
Marks:65
Credits:2
subject:3:
Marks:76
Credits:4
subject:4:
Marks:57
Credits:3
subject:5:
Marks:90
Credits:2
```

```
Sgpa=5.25
```

```
-----
(program exited with code: 0)
```

```
Press any key to continue . . .
```

LAB 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.

```
Lab 3 program Aniraj Prasad  
18M19CS194  
  
import java.util.*;  
import java.lang.*;  
class Book {  
    String name, author;  
    double price;  
    int num_pages;  
    Scanner in = new  
    Scanner(System.in);  
  
    Book() {  
        System.out.println("Enter name of the Book:");  
        name = in.nextLine();  
        System.out.println("Enter price name of author:");  
        author = in.nextLine();  
        System.out.println("Enter price of book in Rs:");  
        price = in.nextDouble();  
        System.out.println("Enter number of pages in the book:");  
        num_pages = in.nextInt();  
    }  
  
    void show() {  
        System.out.println("Name: " + name);  
        System.out.println("Author: " + author);  
        System.out.println("Price: " + price);  
        System.out.println("Number of pages: " + num_pages);  
    }  
  
    public String toString() {  
        return name + ", By " + author + " for Rs. " + price + " and  
        has " + num_pages + " pages";  
    }  
}
```



```

public static void main (String [] args)
{
    Scanner in = new
    Scanner (System.in);

    int n, x;

    System.out.println ("Enter number of books to be created :");
    n = in.nextInt();

    Book B[] = new Book[n];
    for (int i = 0; i < n; i++) {
        System.out.println ("Book" + (i+1));
        B[i] = new Book();
        System.out.println ();
    }

    for (int i = 0; i < n; i++) {
        System.out.println ("Book" + (i+1));
        B[i] = new Book();
        System.out.println ();
    }

    for (int i = 0; i < n; i++) {
        System.out.println ("Book" + (i+1));
        System.out.println (B[i]);
        System.out.println ();
    }

    do {
        System.out.println ("Enter the Book number whose details you
        want to display :");
    }
}

```

```

x = in.nextInt();
while (x < 1 || x > n);
{ x-17. show();
}
}

```

if (x is not in the range of 1 to n) return false;

```

if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;

```

```

if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;

```

```

if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;
if (x is not in the range of 1 to n) return false;

```

```
Enter number of books:
2
book 1
Name of book:
the adventure
Name of author:
william
Price of book in Rs:
300
Number of pages in the book:
678

book 2
Name of book:
the oath
Name of author:
wordsworth
Price of book in Rs:
450
Number of pages in the book:
456
```

```
book 1
the adventure, By william for Rs.300.0 and has 678 pages

book 2
the oath, By wordsworth for Rs.450.0 and has 456 pages

Enter the book whose deatils are to be shown:
2
Name: the oath
Author: wordsworth
Price: 450.0
Number of pages: 456

-----
(program exited with code: 0)

Press any key to continue . . .
```

LAB 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.

Lab 4 - program

Anilj Prasad
18M19C5194

```
import java.util.*;  
import java.lang.*;  
  
abstract class Shape {  
    Scanner in = new  
    Scanner(System.in);  
    int a1, a2;  
    Shape () {  
        System.out.println("Input 2 integer values:");  
        a1 = in.nextInt();  
        a2 = in.nextInt();  
    }  
    abstract void printArea();  
}  
  
class Rectangle extends Shape {  
    void printArea() {  
        System.out.println("Rectangle: " + a1 * a2);  
    }  
}  
  
class Circle extends Shape {  
    void printArea() {  
        System.out.println("Circle 1: " + (3.14 * a1 * a1));  
        System.out.println("Circle 2: " + (3.14 * a2 * a2));  
    }  
}
```

```

class testAbstract {
    public static void main (String [] args)
    {
        Shape s;
        s = new Rectangle();
        s.printArea();

        s = new TriangleRectangle();
        s.printArea();

        s = new Circle();
        s.printArea();
    }
}

```


Input 2 integer values:

1 2

Area of Rectangle : 2

Input 2 integer values:

3 4

Area of Triangle : 6

Input 2 integer values:

5 6

Area of Circle : 78.5

(program exited with code: 0)

Press any key to continue . . .

LAB 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.

Lab5- program

Amity Prasad
18m19CS194
Apt

```
import java.util.*;  
import java.lang.*;  
  
class Account {  
    String name, abc;  
    int accNo;  
    char accType;  
    double deposit;  
    Scanner in = new  
    Scanner(System.in);  
  
    void input_data() {  
        System.out.println("Enter your account type (S/C):");  
        abc = in.nextLine();  
        accType = abc.charAt(0);  
    }  
  
    void deposit() {  
        System.out.println("Enter an amount to deposit:");  
        deposit = in.nextDouble();  
        bal += deposit;  
        System.out.println("Balance has been updated.");  
    }  
  
    void view_balance() {  
        System.out.println("Balance = " + bal);  
    }  
}
```

```

public static void main (String [] args )
{
    Scanner s = new
    Scanner (System.in);
    int x;
    Account a1 = new Account ();
    a1.input_data ();
    if (a1.acctype == 'c' || a1.acctype == 'c') {
        Current a2 = new Current ();
        do {
            System.out.println ("Welcome to your current Account ");
            System.out.println ("1. Deposit ");
            System.out.println ("2. Check Balance ");
            System.out.println ("3. Issue Cheque ");
            System.out.println ("4. Exit ");
            System.out.println ("Enter your choice ");
            x = s.nextInt ();
            switch (x) {
                case 1: a2.deposit ();
                    break;
                case 2: a2.check_balance ();
                    break;
                case 3: a2.issue_cheque ();
                    break;
                case 4: System.exit (0);
                    break;
                default;
                    System.out.println ("Error, Invalid choice ");
            }
        } while (x <= 4 && x >= 1);
    }
}

```

```

    }
    else if (a1. accType == 'S' || a1. accType == 's') {
        Savings a3 = new Savings(1);
        do {
            System.out.println("Welcome to your Savings Account");
            System.out.println("1. Deposit");
            System.out.println("2. View balance");
            System.out.println("3. Withdraw");
            System.out.println("4. Calculate compound interest");
            System.out.println("5. Exit");
            System.out.println("Enter your choice:");
            x = s.nextInt();
            switch (x) {
                case 1: a3.deposit(1);
                    break;
                case 2: a3.viewcheckBalance();
                    break;
                case 3: a3.withdraw balancewithdrawCheque();
                    break;
                case 4: System.exit(0); a3.computeCI();
                    break;
                default: System.out.println("Error. Invalid choice");
                case 5: System.exit(0);
                    break;
                default: System.out.println("Error. Invalid choice");
            }
        } while (x <= 5 && x >= 1);
    }
    else System.out.println("Invalid Account type");
}
}

```

```

class Current extends Account {
    Current () {
        System.out.println("Enter your name:");
        name = in.nextLine();
        System.out.println("Enter your Account number:");
        accNo = in.nextInt();
        deposit();
    }

    double chq_amount;
    void issue_cheque () {
        System.out.println("Enter amount for which cheque is to be issued.");
        cheque chq_amount = in.nextDouble();
        if (chq_amount > bal) {
            System.out.println("Error! Insufficient balance in account.");
        }
        else {
            bal = chq_amount;
            System.out.println("Cheque has been issued successfully");
        }
    }

    void check_balance () {
        if (bal < 1000) {
            System.out.println("Current available balance is lesser than minimum required balance");
            bal = 100;
            System.out.println("Service charge of Rs. 100 has been deducted from your phone balance");
        }
        view_balance();
    }
}

```



```

class Savings extends Account {
    double c1, withdrawal_amount, time;
    Savings() {
        System.out.println("Enter your name: ");
        name = in.nextLine();
        System.out.println("Enter your account number: ");
        accNo = in.nextInt();
        deposit();
    }
    void compute_c1() {
        System.out.println("Enter time period: ");
        time = in.nextInt();
        c1 = bal * Math.pow(1 + (0.08 / 12), 12 * time) - bal;
        System.out.println("C1 = " + c1);
        bal += c1;
        System.out.println("C1 has been deposited");
    }
    void withdraw_balance() {
        System.out.println("Enter the amount you want to withdraw: ");
        withdrawal_amount = in.nextDouble();
        if (withdrawal_amount > bal) {
            System.out.println("Error! The entered amount is greater than the available balance");
        }
        else {
            bal = withdrawal_amount;
            System.out.println("Amount has been successfully withdrawn");
        }
    }
}

```

```
Enter your account type (Savings/Current):
savings
Enter your name:
anitej
Enter your account number:
12345
Enter an amount to deposit:
200
Balance has been updated.
WELCOME TO YOUR SAVINGS ACCOUNT
1. Deposit
2. View Balance
3. Withdraw
4. Calculate compound interest
5. Exit
Enter your choice:
1
Enter an amount to deposit:
200
Balance has been updated.
WELCOME TO YOUR SAVINGS ACCOUNT
1. Deposit
2. View Balance
3. Withdraw
4. Calculate compound interest
5. Exit
Enter your choice:
exit
```

```
Enter your account type (Savings/Current):
current
Enter your name:
anitej
Enter your account number:
12345
Enter an amount to deposit:
300
Balance has been updated.
WELCOME TO YOUR CURRENT ACCOUNT
1. Deposit
2. Check Balance
3. Issue Cheque
4. Exit
Enter your choice:
3
Enter amount for which cheque is to be issued.
240
Cheque has been issued SUCCESSFULLY
WELCOME TO YOUR CURRENT ACCOUNT
1. Deposit
2. Check Balance
3. Issue Cheque
4. Exit
Enter your choice:
```

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.

20-11-20
Aniraj Prasad
18M1A15194

Lab Program-6

```
package CIE;
import java.util.*;
public class Personal {
    public String name;
    public int semester;
    public String usn;

    public void read() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the name:");
        name = sc.next();
        System.out.println("Enter Semester:");
        semester = sc.nextInt();
        System.out.println("USN:");
        usn = sc.next();
    }

    public void display() {
        System.out.println("Student details");
        System.out.println("Name: " + name + "\n USN: " + usn + "\n Semester: " + semester);
    }
}

package CIE;
import java.util.*;
public class Internals extends Personal
```

```
public double cie[];  
public void accept()
```

```
{  
    cie = new double [5];  
    Scanner sc = new Scanner (System.in);  
    for (int i = 0; i < 5; i++)
```

```
{  
        System.out.println ("CIE mark for course " + (i+1) + " : ");  
        cie[i] = sc.nextDouble();  
    }
```

```
}  
  
package SEE;  
import java.util.*;  
import CIE.*;  
public class external extends personal  
{
```

```
    public double see[];  
    public void get()
```

```
{  
        see = new double [5];  
        Scanner sc = new Scanner (System.in);  
        for (int i = 0; i < 5; i++)
```

```
{  
            System.out.println ("SEE mark for course " + (i+1) + " : ");  
            see[i] = sc.nextDouble();  
        }
```

```

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

```

```

class Main
{

```

```

    public static void main (String args[])
    {

```

```

        Scanner sx = new Scanner (System.in);
        System.out.println ("Enter the number of students");
        int n = sx.nextInt ();
        CIE .internals in [] = new CIE .internals [n];
        SEE .externals en [] = new SEE .externals [n];
        int i, j;
        for (i=0; i<n; i++)

```

```

        {
            System.out.println ("Student " + (i+1));
            in [i] = new CIE .internals ();
            en [i] = new SEE .externals ();
            in [i].read ();

```

```

        }
        System.out.println ("CIE MARKS:");
        in [i].accept ();
        System.out.println ("SEE MARKS:");
        en [i].get ();
        System.out.println ();
        in [i].display ();
        for (j=0; j<5; j++)

```

```

        {
            System.out.println ("Total Marks for course" + (j+1) + " : " + (in [i].cie [j]
            + (en [i].see [j]/2));
        }
    }
}

```


LAB7:

Write a program to demonstrate generics with multiple object parameters.

27-11-20
Aniraj Prasad
18M19CS194
Aniraj

```
Lab Program - 7

class G FourGen <T, V, W, X> {
    T ob1;
    V ob2;
    W ob3;
    X ob4;
    FourGen (T o1, V o2, W o3, X o4) {
        ob1 = o1;
        ob2 = o2;
        ob3 = o3;
        ob4 = o4;
    }
    void showTypes() {
        System.out.println("Type of T is " + ob1.getClass().getName());
        System.out.println("Type of V is " + ob2.getClass().getName());
        System.out.println("Type of W is " + ob3.getClass().getName());
        System.out.println("Type of X is " + ob4.getClass().getName());
    }
    T getOb1() {
        return ob1;
    }
    V getOb2() {
        return ob2;
    }
    W getOb3() {
        return ob3;
    }
    X getOb4() {
        return ob4;
    }
}
```

```
class & Simp Gen {
```

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

```
        FourGen < Integer, String, IntegerDouble, StringLong > tgObj =
```

```
        new FourGen < Integer, String, IntegerDouble, StringLong > (19, "Anitej", 2026.876, "Anitej"375666L);
```

```
        tgObj.showTypes();
```

```
        int v = tgObj.getob1();
```

```
        System.out.println("Value: " + v);
```

```
        String str = tgObj.getob2();
```

```
        System.out.println("Value: " + str);
```

```
int x = tgObj.getob3();
```

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

```
String str = tgObj.getob2();
```

```
System.out.println("Value: " + str);
```

```
        Double dbl = tgObj.getob3();
```

```
        System.out.println("Value: " + dbl);
```

```
        Long dbl = tgObj.getob4();
```

```
        System.out.println("Value: " + lng);
```

```
    }
```

```
C:\windows\SYSTEM32\cmd.exe
Type of T is java.lang.Integer
Type of V is java.lang.String
Type of W is java.lang.Double
Type of X is java.lang.Long
value: 19
value: Anitej
value:23.8765
value:327666

-----
(program exited with code: 0)
Press any key to continue . . .
```

LAB8:

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 takes both father and son's age and throws an exception if son's age is >= father's age.

LAB8- Father Son Program

```
import java.util.Scanner.*;
class WrongAge extends Exception
{
    private int a, b;
    WrongAge(int x, int y)
    {
        a = x;
        b = y;
    }

    public String toString()
    {
        if (a < 0 || b < 0)
            return "age cannot be less than 0";
        else if (a <= b)
            return "valid ages have not been entered";
        return "";
    }
}
```

Class Father

```
{
    int fAge, sAge;
    Scanner sc = new Scanner(System.in);
    Father() throws WrongAge {
        System.out.println("age of father");
        fAge = sc.nextInt();
        System.out.println("age of son");
        sAge = sc.nextInt();
        if (fAge < 0 || sAge < 0)
            throw new WrongAge(fAge, sAge);
    }
}
```

27-11-20

Anil Prasad

18M19C094

ajal

```

        throw new WrongAge (Page, sage);
    }
}

class son extends Father
{
    son() throws WrongAge
    {
        if (sage >= Page)
            throw new WrongAge (Page, sage);
    }
    else
        System.out.println ("Valid ages have been entered");
}

class FatherSon
{
    public static void main (String args[])
    {
        try {
            son s = new son();
        }
        catch (WrongAge e) {
            System.out.println ("error" + e);
        }
    }
}

```



```
C:\windows\SYSTEM32\cmd.exe
age of father
45
age of son
20
valid ages have been entered

-----
(program exited with code: 0)
Press any key to continue . . .
```

LAB9:

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.

11-12-20
Anil Prasad
18M19CS194
adil

Lab-9

```
class Thread_test implements Runnable {  
    String name;  
    Thread t;  
    int time;  
    Thread_test(String Thread_name, int time) {  
        name = Thread_name;  
        this.time = time;  
        t = new Thread(this, name);  
        System.out.println("Thread: " + t);  
        t.start();  
    }  
    public void run() {  
        try {  
            for (int i = 5; i > 0; i--) {  
                System.out.println(name);  
                Thread.sleep(time);  
            }  
        } catch (InterruptedException e) {  
            System.out.println(name + " Interrupt");  
        }  
        System.out.println(name + " terminate");  
    }  
}  
class Threadmain {  
    public static void main (String args[]) {  
        Thread_test t1 = new Thread_test("BMS College of Engg", 10000);  
        Thread_test t2 = new Thread_test("CSE", 2000);  
    }  
}
```

```
C:\windows\SYSTEM32\cmd.exe
thread:Thread[BMS College of Engineering,5,main]
thread:Thread[CSE,5,main]
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE terminate
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering
BMS College of Engineering terminate

-----
(program exited with code: 0)
Press any key to continue . . .
```