

Name:N.Akhilesh Kumar Dutt

USN:1BM19CS092

Section:3B

OOJ LAB OBSERVATION

Lab-Program-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.

## Java Program for Roots

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

class Roots
{
    public static void main (String args[])
    {
        double a, b, c, r1, r2, del;
        Scanner ss = new Scanner (System.in);
        System.out.println("\nEnter the values of  
a, b, c for a quadratic equation : \n");
        a = ss.nextDouble();
        b = ss.nextDouble();
        c = ss.nextDouble();
        del = (b*b - 4*a*c);
        if (a == 0)
        {
            System.out.println("a cannot be equal to  
zero \n");
            System.exit(0);
        }
        else
        {
            System.out.println("a is a real value \n");
        }
        if (del == 0)
        {
            System.out.println("Roots are real and equal \n");
            r1 = -(b/(2*a));
            r2 = -(b/(2*a));
            System.out.println("Roots are : \n" + r1 + " " + r2);
        }
    }
}
```

else if (del > 0)

{

system.out.println("Roots are real and unequal");

$r_1 = (-b + \text{Math.sqrt}(del)) / (2 * a);$

$r_2 = (-b - \text{Math.sqrt}(del)) / (2 * a);$

system.out.println("Roots are : " + r1 + " " + r2);

}

else

{

system.out.println("there are no Real solutions");

}

}

}

outPut

Enter the values of a, b, c for quadratic equation

: 1 -4 4

a is a real value.

Roots are real and equal.

~~Roots~~

Roots are 2.0, 2.0

Enter the values of a, b, c for quadratic equation

: 1 5 6

a is a real value.

Roots are real and unequal.

Roots are -3.0, -2.0

Enter the values of a, b, c for quadratic equation

: 2 1 3

a is a real value.

There are no real solutions.

```

C:\Users\DELL\Desktop\JAVA>javac Roots.java

C:\Users\DELL\Desktop\JAVA>java Roots

Enter the Values of a,b,c for a Quadratic equation :

1 -4 4
a is a real value

Roots are real and equal

Roots are :
2.0,2.0

C:\Users\DELL\Desktop\JAVA>java Roots

Enter the Values of a,b,c for a Quadratic equation :

1 6 8
a is a real value

Roots are real and unequal

Roots are :
-2.0,-4.0

C:\Users\DELL\Desktop\JAVA>java Roots

Enter the Values of a,b,c for a Quadratic equation :

3 1 2
a is a real value

There are no Real Solutions

C:\Users\DELL\Desktop\JAVA>java Roots

Enter the Values of a,b,c for a Quadratic equation :

0 1 2
a cannot be equal to zero

```

#### Lab-Program-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



### Lab - Program - 2

```
import java.util.Scanner;
class student
{
    Private int usn;
    Private String name;
    Private double semPA;
    Private int [] credits = new int[8];
    Private int [] marks = new int[8];
    void acceptDetails()
    {
        System.out.println("In Accept student details\n");
        Scanner s1 = new Scanner(System.in);
        System.out.println("Enter the credits of student for all subjects\n");
        for (int i=0; i<8; i++)
        {
            credits[i] = s1.nextInt();
        }
        System.out.println("Enter the credits of student for all subjects\n");
    }
}
```

```

for (int i=0; i<8; i++)
{
    credits[i] = si.nextInt();
}
system.out.println("\nEnter the Marks of
                    student for all subjects\n");
for (int i=0; i<8; i++)
{
    marks[i] = si.nextInt();
}
system.out.println("\nEnter student's USN\n");
usn = si.nextInt();
system.out.println("\nEnter student's Name\n");
name = si.nextInt();
}

void PrintDetails()
{
    system.out.println("\nThe credits of the student
                    for all subjects are : \n");
    for (int i=0; i<8; i++)
    {
        system.out.println(credits[i]);
    }
    system.out.println("\nThe Marks of the student
                    for all subjects are : \n");
    for (int i=0; i<8; i++)
    {
        system.out.println(marks[i]);
    }
    system.out.println("USN : " + usn);
    system.out.println("Name : " + name);
}

```

```

void calculate GPA()
{
    int grade_Points = 0;
    int sum = 0;
    int deno = 0;
    for (int i = 0; i < 8; i++)
    {
        if (marks[i] >= 90)
        {
            grade_Points = 10;
        }
        else if (marks[i] >= 80 && marks[i] < 90)
        {
            grade_Points = 9;
        }
        else if (marks[i] >= 70 && marks[i] < 80)
        {
            grade_Points = 8;
        }
        else if (marks[i] >= 60 && marks[i] < 70)
        {
            grade_Points = 7;
        }
        else if (marks[i] >= 50 && marks[i] < 60)
        {
            grade_Points = 6;
        }
        else if (marks[i] >= 40 && marks[i] < 50)
        {
            grade_Points = 5;
        }
    }
}

```



```

else
{
    grade_points = 0;
}
sum += grade_points * credits[i];
deno += credits[i];
}
SGPA = sum/deno;
System.out.println("SGPA : " + SGPA);
}
}

class student {
    public static void main (String args[])
    {
        student s1 = new student();
        s1.acceptDetails();
        s1.printDetails();
        s1.calculateSGPA();
    }
}

```

### outPut

Accept student Details

Enter the Credits of student for all subjects  
4 4 4 4 3 3 2 2

Enter the Marks of student for all subjects  
90 88 92 75 65 99 88 95

Enter Student's USN  
12

Enter Student's Name  
Rahul

The credits of the student for all subjects are :

4  
4  
4  
4  
3  
3  
2  
2

The marks of the student for all subjects are :

90  
85  
92  
75  
65  
99  
85  
95

UGN : 12

Name : Rahul

SGPA : 9.0



C:\WINDOWS\SYSTEM32\cmd.exe

Accept Student Details

Enter the Credits of Student for all Subjects

4 4 4 4 3 3 2 2

Enter the Marks of Student for all Subjects

90 85 92 75 65 99 85 95

Enter Student's USN

12

Enter Student's Name

Rahul\_

```
C:\WINDOWS\SYSTEM32\cmd.exe

The Credits of the Student for all Subjects are :

4
4
4
4
3
3
2
2

The Marks of the Student for all Subjects are :

90
85
92
75
65
99
85
95
USN :12
Name :Rahul
SGPA :9.0
```

### Lab-Program-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 a book object

### Lab-Program-3

```
import java.util.Scanner;  
import java.lang.*;  
class Book
```

```
{
```

```
    String name;
```

```
    String author;
```

```
    String double Price;
```

```
    double num_Pages;
```

```
    Book()
```

```
{
```

```
    name = " ";
```

```
    author = " ";
```

```
    Price = 0.0;
```

```
    num_Pages = 0;
```

```
}
```



```
Public void getDetails()
```

```
{
```

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

```
System.out.println("Enter the Book Details");
```

```
System.out.println("Enter Book Name:");
```

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

```
System.out.println("Enter the Name of the Author");
```

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

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

```
Price = ss.nextDouble();
```

```
System.out.println("Enter the number of Pages  
in the book");
```

```
num_Pages = ss.nextDouble();
```

```
}
```

```
Public String toString()
```

```
{
```

```
return("In the name of the book: "+name+"  
In the author of the book: "+author+"  
In the Price of the book: "+Price+"In the  
number of Pages in book: "+num_Pages);
```

```
}
```

```
class Bookmain {
```

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

```
{
```

```
int i, n;
```

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

```
System.out.println("Enter the number of books");
```

```
n = ss.nextInt();
```

```
Book b[] = new Book();
```

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

```
{
```

```
b[i] = new Book();
```

```
b[i].getDetails();
```

```
}
```

```
}
```

```

for (i=0; i<n; i++)
{
    system.out.println("DETAILS OF BOOK"+(i+1));
    system.out.println(b[i]);
}
}
}

```

### Output

Enter the number of books

2

Enter the Book Details

Enter Book Name:

ooj

Enter the Name of the Author

Rahul

Enter the Price of the book

300

Enter the number of pages in the book

100

Enter the Book Details

Enter Book Name:

COA

Enter the Name of the Author

Vijay

Enter the Price of the book

400

Enter the number of pages in the book

200

### DETAILS OF BOOK 1

The name of the book : ood

The author of the book : Rahul.

The Price of the book : 300.0

~~The~~ Number of Pages in book : 100.0.

### DETAILS of Book 2

The name of the book : COA.

The author of the book : Vijay.

The Price of the book : 400.0.

Number of Pages in book : 200.0.

e C:\WINDOWS\SYSTEM32\cmd.exe

Enter the number of books

2

Enter the Book Details

Enter Book Name:

ooj

Enter the Name of the Author

Rahul

Enter the Price of the book

300

Enter the number of pages in the book

100

Enter the Book Details

Enter Book Name:

```
C:\WINDOWS\SYSTEM32\cmd.exe

300
Enter the number of pages in the book
100
Enter the Book Details
Enter Book Name:
COA
Enter the Name of the Author
vijay
Enter the Price of the book
400
Enter the number of pages in the book
200
DETAILS OF BOOK1

The name of the book : ooj
The author of the book :Rahul
The price of the book: 300.0
Number of pages in book :100.0
DETAILS OF BOOK2

The name of the book : COA
The author of the book :vijay
The price of the book: 400.0
Number of pages in book :200.0
```

#### Lab-Program-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-Program-4 :-

```
* import java.util.Scanner;
abstract class shape
{
    int a1;
    int a2;
    shape(int a, int b)
    {
        a1 = a;
        a2 = b;
    }
    abstract double Printarea();
}
class Rectangle extends shape
{
    double area;
    Rectangle(int a, int b)
    {
        super(a, b);
    }
    double Printarea()
    {
        area = a1 * a2;
        return (area);
    }
}
```

```

class Triangle extends shape
{
    double area;
    Triangle(int a, int b)
    {
        super(a, b);
    }
    double Printarea()
    {
        area = (double) (a1 * a2) / 2;
        return (area);
    }
}

class circle extends shape
{
    double area;
    Circle(int a, int b)
    {
        super(a, b);
    }
    double Printarea()
    {
        area = (double) 3.14 * a1 * a1;
        return (area);
    }
}

```

class areas

{

Public static void main(String args[])

{

Rectangle r = new Rectangle(11, 18);

Triangle t = new Triangle(9, 7);

Circle c = new Circle(9, 9);

System.out.println("Area of the rectangle: " + r.PrintArea());

System.out.println("Area of the triangle: " + t.PrintArea());

System.out.println("Area of the circle: " + c.PrintArea());

}

}

```
C:\Users\DELL\Desktop\JAVA>javac areas.java

C:\Users\DELL\Desktop\JAVA>java areas
Area of the rectangle :165.0
Area of Triangle :31.5
Area of the circle :254.34

C:\Users\DELL\Desktop\JAVA>
```

#### Lab-Program-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:

- 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 - Program - 5 :-

```
* import java.util.Scanner;
abstract class Account
{
    String cust_name;
    long acc_no;
    String acc_type;
    double balance;
    double min_bal = 1000.0;
    Account(String cust_name, long no, String acc_type,
            double balance)
    {
        this.cust_name = cust_name;
        this.acc_no = acc_no;
        this.acc_type = acc_type;
        this.balance = balance;
    }
    abstract void deposit(double amount);
    abstract void display();
    abstract void withdrawal(double amount);
}
```



```

class curr_acct extends Account
{
    double Penalty = 100.0;
    curr_acct (String cust_name, long acc_no,
               String acc_type, double balance)
    {
        super (cust_name, acc_no, acc_type, balance);
        system.out.println("name of the customer:"
                           + cust_name);
        system.out.println("Account Number:" + acc_no);
        system.out.println("Account type:" + acc_type);
        system.out.println("Balance:" + balance);
    }

    void deposit (double amount)
    {
        this.balance = this.balance + amount;
    }

    void display ()
    {
        system.out.println("balance is:" + this.balance);
    }
}

```

```

void withdraw(double amount)
{
    this.balance = this.balance - amount;
    imposePenalty();
}

void imposePenalty()
{
    if (this.balance < min-bal)
    {
        this.balance = this.balance - Penalty;
    }
}

class sav-acct extends AccAccount
{
    sav-acct (String cust_name, long acc_no,
              String acc_type, double balance)
    {
        super (cust_name, acc_no, acc_type, balance);
        System.out.println("Name of the customer:"
                           + cust_name);
        System.out.println("Account Number:"
                           + acc_no);
    }
}

```

```
system.out.println("Account type:" + acc_type);  
system.out.println("Balance:" + balance);
```

```
3  
void deposit(double amount)
```

```
{  
    this.balance = this.balance + amount;  
    interest();
```

```
3  
void interest()
```

```
{  
    int rate = 10, time = 1;  
    float ci = (float)(this.balance * Math.Pow  
        (1 + rate/100.0, time) - this.balance);
```

```
    this.balance = this.balance + ci;
```

```
3  
void display()
```

```
{  
    system.out.println("Balance is:" + this.balance);
```

```
3
```

```
void withdrawal (double amount)
```

```
{
```

```
    this.balance = this.balance - amount;
```

```
}
```

```
class bank
```

```
{
```

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

```
{
```

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

```
        double amount;
```

```
        int flag = 0;
```

```
        while (flag == 0)
```

```
{
```

```
            System.out.println ("Enter the type of
```

```
Account : \n 1: Current account \n 2: Savings
```

```
account");
```

```
            int choice = ss.nextInt();
```

```
            switch (choice)
```

```
{
```

```
                case 1: System.out.println ("Current account
```

```
\n");
```

```
                curr_acct c = new Curr_acct ("Rahul", 15768912,
```

```
"Current", 30000.00);
```

```

int flag1 = 0;
while (flag1 == 0)
{
System.out.println("In current account:");
system.out.println("Enter your choice\n1: deposit\n2: Display Balance\n3: withdraw");
int choice1 = ss.nextInt();
switch (choice1)
{
case 1:
system.out.println("Enter amount to be deposited:");
amount = ss.nextDouble();
c.deposit(amount);
break;
case 2:
c.display();
break;
case 3:
system.out.println("Enter amount you want to withdraw:");
amount = ss.nextDouble();
c.withdrawal(amount);
break;
default:
flag1 = 1;
}
}
}

```



```
break;
```

```
case 2: system.out.println("In savings account  
In");
```

```
sav_acct s = new sav_acct("vijay", 68432179,  
"Savings", 6000.00);
```

```
int flag2 = 0;
```

```
while(flag2 == 0)
```

```
{
```

```
system.out.println("Enter your choice
```

```
1: Deposit amount 2: Display Balance 3: withdraw");
```

```
int choice2 = ss.nextInt();
```

```
switch(choice2)
```

```
{
```

```
case 1: system.out.println("Enter the  
amount to be deposited: ");
```

```
amount = ss.nextDouble();
```

```
s.deposit(amount);
```

```
break;
```

```
case 2:
```

```
s.display();
```

```
break;
```

case 3:

system.out.println("Enter the Amount you  
want to withdraw:");

amount = ss.nextDouble();

break;

default:

flag2 = 1;

}

}

break;

default: flag = 1;

}

}

}

}

Command Prompt - java bank

1:Current account

2:Savings account

1

Current account:

Name of the customer: Rahul

Account Number : 15768912

Account type: current

Balance: 30000.0

Enter your choice

1:Deposit amount

2:DisplayBalance

3:Withdraw

1

Enter amount to be deposited:

100

Enter your choice

1:Deposit amount

2:DisplayBalance

3:Withdraw

3

Enter amount you want to withdraw:

10000

Enter your choice

1:Deposit amount

2:DisplayBalance

3:Withdraw

2

Balance is: 20100.0

Enter your choice

1:Deposit amount

2:DisplayBalance

3:Withdraw

1

Enter amount to be deposited:

200

Enter your choice

1:Deposit amount

2:DisplayBalance

3:Withdraw

2

Balance is: 20300.0

Enter your choice

1:Deposit amount

2:DisplayBalance

3:Withdraw

4

Enter the type of Account:

1:Current account

2:Savings account

GA Command Prompt - java bank

```
2:DisplayBalance
3:Withdraw
4
Enter the type of Account:
1:Current account
2:Savings account
2

Savings account:

Name of the customer: Vijay
Account Number : 68432179
Account type: Savings
Balance: 6000.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
1
Enter the Amount to be Deposited:
200
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
2
Balance is: 6820.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
3
Enter the Amount you want to Withdraw:
6000
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
2
Balance is: 820.0
Enter your choice
1:Deposit amount
2:DisplayBalance
3:Withdraw
4
Enter the type of Account:
1:Current account
2:Savings account
```