

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.

```
import java.util.Scanner;

public class Main {

    public static void main(String[] Strings) {

        Scanner input = new Scanner(System.in);

        System.out.print("Input a: ");

        double a = input.nextDouble();

        System.out.print("Input b: ");

        double b = input.nextDouble();

        System.out.print("Input c: ");

        double c = input.nextDouble();

        double result = b * b - 4.0 * a * c;

        if (result > 0.0) {

            double r1 = (-b + Math.pow(result, 0.5)) / (2.0 * a);

            double r2 = (-b - Math.pow(result, 0.5)) / (2.0 * a);

            System.out.println("The roots are " + r1 + " and " + r2);

        } else if (result == 0.0) {

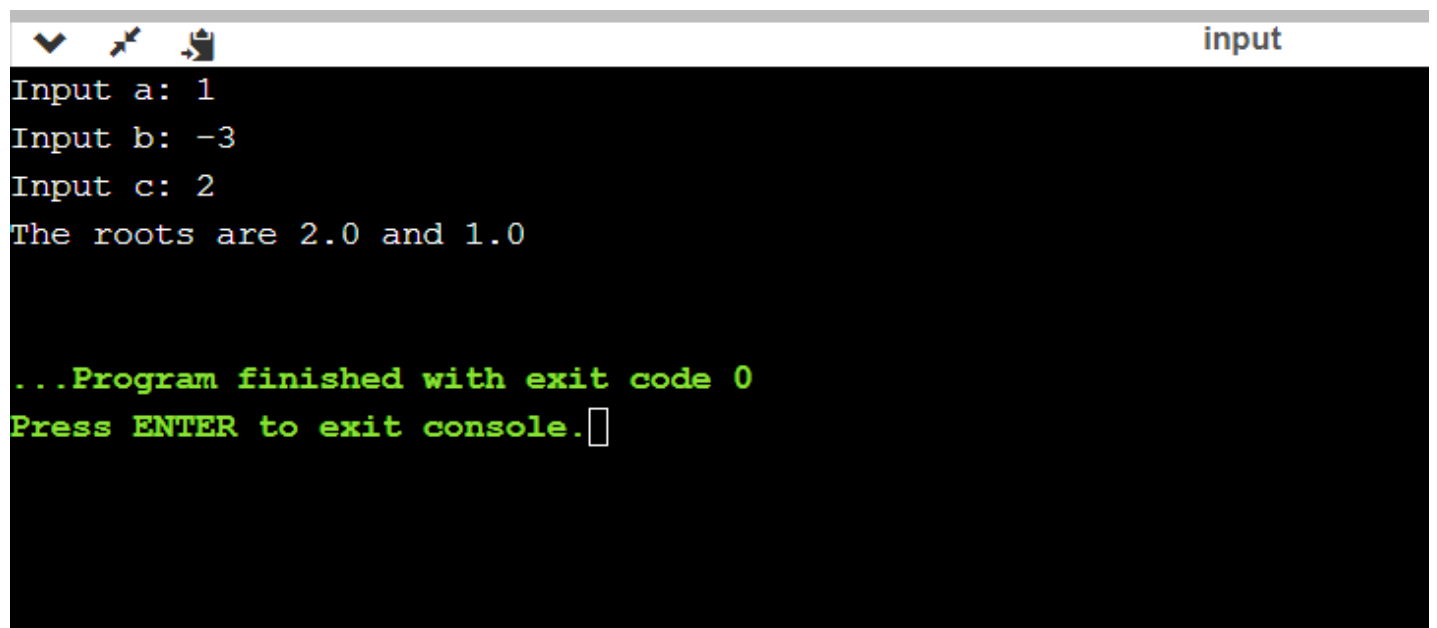
            double r1 = -b / (2.0 * a);

            System.out.println("The root is " + r1);

        } else {
```

```
        System.out.println("The equation has no real roots.");  
    }  
}  
}
```

OUTPUT:



```
Input a: 1  
Input b: -3  
Input c: 2  
The roots are 2.0 and 1.0  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

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.

```
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 the student");

        USN = ss.nextLine();

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

        name = ss.nextLine();

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

        n = ss.nextInt();

        int credits[] = new int[n];

        double marks[] = new double[n];

        System.out.println("Enter details of the subjects:");

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

        {

            System.out.println("Enter credits allotted to the subject "+(i+1));
```

```

credits[i] = ss.nextInt();

System.out.println("Enter marks in the subject "+(i+1));

marks[i] = ss.nextInt();

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

}

void Calculate(int credit,double mark,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 + (8*credit);

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

SGPA = SGPA + (7*credit);

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

SGPA = SGPA + (6*credit);

else if(mark>=40&&mark<=49)

SGPA = SGPA + (5*credit);

else

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

}

void Display()

{

System.out.println("Details of the Student");

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

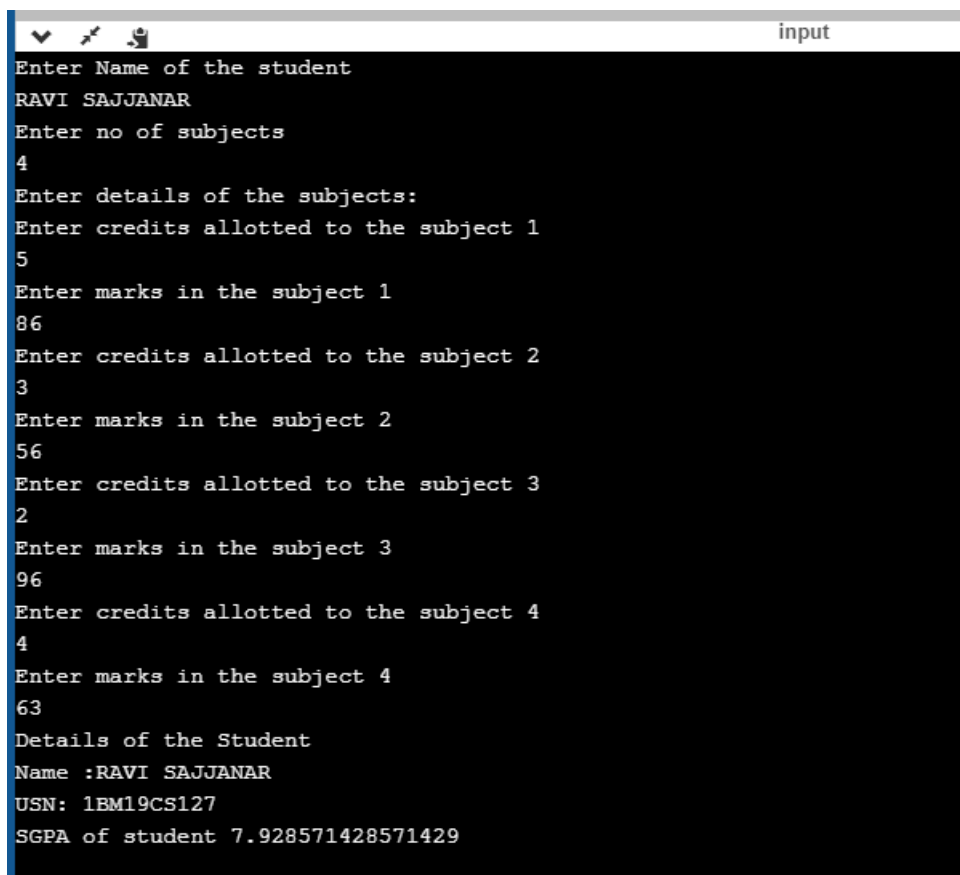
System.out.println("USN: "+USN);

```

```
System.out.println("SGPA of student "+(SGPA/totalCredits));  
  
}  
  
}
```

```
public class Lab2  
{  
  
    public static void main(String args[])  
    {  
  
        Student s1 = new Student();  
  
        s1.Details();  
  
        s1.Display();  
  
    }  
  
}
```

OUTPUT:



```
input  
Enter Name of the student  
RAVI SAJJANAR  
Enter no of subjects  
4  
Enter details of the subjects:  
Enter credits allotted to the subject 1  
5  
Enter marks in the subject 1  
86  
Enter credits allotted to the subject 2  
3  
Enter marks in the subject 2  
56  
Enter credits allotted to the subject 3  
2  
Enter marks in the subject 3  
96  
Enter credits allotted to the subject 4  
4  
Enter marks in the subject 4  
63  
Details of the Student  
Name :RAVI SAJJANAR  
USN: 1BM19CS127  
SGPA of student 7.928571428571429
```

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 n book objects.

PROGRAM:

```
import java.util.Scanner;

class Book
{
    private String book_name;
    private String author;
    private double price;
    private int no_of_pages;

    Scanner s = new Scanner(System.in);

    public Book() {
        this.book_name = "";
        this.author = "";
        this.price = 0;
        this.no_of_pages = 0;
    }

    void getdetails(){
        System.out.print("\nEnter book name:");
        book_name = s.nextLine();

        System.out.print("Enter author's name:");
        author = s.nextLine();

        System.out.print("Enter price of book:");
```

```

        price = s.nextDouble();

        System.out.print("Enter the number of pages of book:");

        no_of_pages = s.nextInt();

    }

    public String toString()

    {

        return ("-----BOOK DETAILS-----\nBook name:"+ book_name
+" \n"+"Author:"+author+" \n"+ "Price:"+price+" \n"+ "Number of Pages:"+no_of_pages+" \n");

    }

}

```

```

public class Main {

    public static void main(String args[]) {

        int n,i=0;

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the number of books:");

        n = sc.nextInt();

        sc.nextLine();

        Book [] b = new Book[n];

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

            System.out.print("Enter the details of book "+(i+1));

            b[i] = new Book();

            b[i].getdetails();

        }

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

            System.out.println(b[i]);

        }

    }

}

```

OUTPUT:

```
Enter the number of books:2
Enter the details of book 1
Enter book name:DS
Enter author's name:SHAMSUNDAR
Enter price of book:236
Enter the number of pages of book:1245
Enter the details of book 2
Enter book name:OOJ
Enter author's name:KALPATARU
Enter price of book:562
Enter the number of pages of book:889
-----BOOK DETAILS-----
Book name:DS
Author:SHAMSUNDAR
Price:236.0
Number of Pages:1245

-----BOOK DETAILS-----
Book name:OOJ
Author:KALPATARU
Price:562.0
Number of Pages:889
```


Lab program4

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.

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

```
abstract class Shape{
```

```
    int a;
```

```
    int b;
```

```
    Shape(int a,int b){
```

```
        this.a=a;
```

```
        this.b=b;
```

```
    }
```

```
    Shape(int a){
```

```
        this.a=a;
```

```
    }
```

```
    abstract double printArea();
```

```
}
```

```
class Rectangle extends Shape{
```

```
    Rectangle(int a,int b){
```

```
        super(a,b);
```

```
    }
```

```
    double printArea(){
```

```
        return a*b;
    }
}
```

```
class Triangle extends Shape{
    Triangle(int a,int b){
        super(a,b);
    }

    double printArea(){
        return a*b/2;
    }
}
```

```
class Circle extends Shape{
    Circle(int a){
        super(a);
    }

    double printArea(){
        return Math.PI*a*a;
    }
}
```

```
/* public class AbstractShape{
```

```

public static void main(String args[]){

    Rectangle p=new Rectangle(5,6);

    Triangle q=new Triangle(4,8);

    Circle r=new Circle(3);

    */

public class AbstractShape{

public static void main(String args[]){

    Scanner input=new Scanner(System.in);

    System.out.println("Enter the Dimensions of Rectangle");

    Rectangle p=new Rectangle(input.nextInt(),input.nextInt());

    System.out.println("Enter the Dimensions of Triangle");

    Triangle q=new Triangle(input.nextInt(),input.nextInt());

    System.out.println("Enter the Dimensions of Circle");

    Circle r=new Circle(input.nextInt());


Shape figref;

figref=p;

System.out.println("Area of Rectangle is: "+p.printArea());

figref=q;

System.out.println("Area of Triangle is: "+q.printArea());

figref=r;

System.out.println("Area of Circle is: "+r.printArea());

}

}

```

Result

CPU Time: 0.27 sec(s), Memory: 37136 kilobyte(s)

compiled and executed in 0.936 sec(s)

```
Enter the Dimensions of Rectangle
Enter the Dimensions of Triangle
Enter the Dimensions of Circle
Area of Rectangle is: 24.0
Area of Triangle is: 12.0
Area of Circle is: 50.26548245743669
```

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

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

```
class Account{  
    String name;  
    int accountNo;  
    String accountType;  
    double balance;  
    Account(String name,int accountNo,String accountType,double balance){  
        this.name = name;  
        this.accountNo = accountNo;  
        this.accountType = accountType;  
        this.balance = balance;  
    }  
    void DisplayStatus() {  
        System.out.println("***"+this.accountType+"***");  
        System.out.println("Name: "+this.name);  
    }  
}
```

```

        System.out.println("Account no.: "+this.accountNo);

        System.out.println("Account Type: "+this.accountType);

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

    }

}

class SavAcct extends Account{

    double depositAmount;

    double Withdrawmount;

    SavAcct(String name,int accountNo,String accountType,double balance){

        super(name,accountNo,accountType,balance);

    }

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

    private void checkBalance() {

        if(balance<0) {

            System.out.println("Transaction is not possible. Balance becomes less
than zero");

            balance+=Withdrawmount;

            Withdrawmount=0;

            Withdraw();

        }

    }

    void CallInterest() {

        System.out.println("Interest To Be added");

        System.out.println("Annual rate of interest: 4%");

        System.out.println("Enter the tenure in terms of year");
    }
}

```

```

        int tenure = input.nextInt();

        balance = balance*Math.pow(1.04, tenure);
    }

    void Deposit() {

        System.out.println("Enter the Deposit amount");

        depositAmount = input.nextDouble();

        balance+=depositAmount;
    }

    void Withdraw() {

        System.out.println("Enter the Withdrawal amount");

        Withdrawmount = input.nextDouble();

        balance-=Withdrawmount;

        checkBalance();

        System.out.println("Withdraw amount = "+Withdrawmount);
    }
}

```

```

class CurrAcct extends Account{

    double minBalance = 1000;

    double depositAmount;

    double Withdrawmount;

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

    CurrAcct(String name,int accountNo,String accountType,double balance){

        super(name,accountNo,accountType,balance);
    }

    private void checkBalance() {

```

```

        if(balance<minBalance) {

            System.out.println("Transaction is not possible. Balance becomes less
than minimum balance.");

            balance+=Withdrawmount;

            System.out.println("Do u still want to do the transaction with added
service charges");

            String ans = input.next();

            if(ans.toLowerCase().equals("yes")) {

                balance-=(Withdrawmount+(0.05*Withdrawmount)+1000);

                System.out.println("ALERT: Negative balance.\nService Charge
added: "+(0.05*Withdrawmount));

            }else {

                Withdrawmount = 0;

            }

        }

    }

    void Deposit() {

        System.out.println("Enter the Deposit amount");

        depositAmount = input.nextDouble();

        balance+=depositAmount;

    }

    void Withdraw() {

        System.out.println("Enter the Withdrawal amount");

        Withdrawmount = input.nextDouble();

        balance-=Withdrawmount;

        checkBalance();

        System.out.println("withdraw amount = "+Withdrawmount);

```



```

    }
}

public class Bank {

    public static void main(String[] args) {

        Scanner in = new Scanner(System.in);

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

        String name = in.next();

        System.out.println("Enter the account no.");

        int num = in.nextInt();

        int i=0;

        while(i<2) {

            System.out.println("Enter the account type\ncurr-current acc.\nsav-savings
acct.");

            String type = in.next();

            if(type.equals("curr")) {

                double bal = in.nextInt();

                CurrAcct c1 = new CurrAcct(name,num,"Current Account",bal);

                c1.DisplayStatus();

                c1.Deposit();

                c1.DisplayStatus();

                c1.Withdraw();

                c1.DisplayStatus();

            }else if(type.toLowerCase().equals("sav")) {

                double bal = in.nextInt();

```

```

        SavAcct s1 = new SavAcct(name,num,"Savings Account",bal);

        s1.DisplayStatus();

        s1.Deposit();

        s1.DisplayStatus();

        s1.Withdraw();

        s1.DisplayStatus();

        s1.CallInterest();

        s1.DisplayStatus();

    }

    i++;

}

in.close();

}

}

```

Result

compiled and executed in 120.568 sec(s)

```

Enter the name
Ravi
Enter the account no.
123456789
Enter the account type
curr-current acc.
sav-savings acct.
curr
12302
**Current Account***

Name: RaviAccount no.: 123456789
Account Type: Current Account
Balance: 12302.0
Enter the Deposit amount
12345
**Current Account***
Name: Ravi
Account no.: 123456789
Account Type: Current Account
Balance: 135758.0
Enter the Withdrawal amount
456892
Transaction is not possible. Balance becomes less than minimum balance.
Do u still want to do the transaction with added service charges
yes
ALERT: Negative balance.
Service Charge added: 22844.600000000002
withdraw amount = 456892.0
**Current Account***
Name: Ravi
Account no.: 123456789
Account Type: Current Account
Balance: -344978.6
Enter the account type
curr-current acc.
sav-savings acct.

```

