

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;

class Quadratic
{
    public static void main(String args[])
    {
        System.out.println("ANANYA AGARWAL");
        System.out.println("1BM22CS039\n");
        System.out.println("\n");

        float a,b,c,d;
        double x1,x2; //math.sqrt returns double
        Scanner obj =new Scanner(System.in);
        System.out.println("enter the value of a");
        a=obj.nextFloat();
        System.out.println("enter the value of b");
        b=obj.nextFloat();
        System.out.println("enter the value of c");
        c=obj.nextFloat();
        d=b*b-(4*a*c);
        if(a==0)
        {
            System.out.println("invalid input");
        }
        else
        {
            if(d==0)
            {
                System.out.println("roots are real and equal");
```

```

x1=x2=(-b)/(2*a);
System.out.println("roots are:"+x1 + "," +x2);
}
else if(d>0)
{
System.out.println("roots are real and unequal");
x1=(-b)+Math.sqrt(d))/(2*a);
x2=(-b)-Math.sqrt(d))/(2*a);
System.out.println("roots are:"+x1 + "," +x2);
}
else
{
System.out.println("roots are imaginary");
}
}
}
}
}

```

Output:

```

C:\Users\hp\Desktop\Ananya>dir
Volume in drive C has no label.
Volume Serial Number is CA5C-96F9

Directory of C:\Users\hp\Desktop\Ananya

29-02-2024  21:42    <DIR>          .
29-02-2024  21:42    <DIR>          ..
29-02-2024  21:42                938 Quadratic.java
               1 File(s)              938 bytes
               2 Dir(s)  43,317,960,704 bytes free

C:\Users\hp\Desktop\Ananya>javac Quadratic.java

C:\Users\hp\Desktop\Ananya>java Quadratic
Ananya Agarwal
1BM22CS039

enter the value of a
4
enter the value of b
5
enter the value of c
6
roots are imaginary

C:\Users\hp\Desktop\Ananya>

```

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
{
    String usn;
    String name;
    int[] credits;
    int[] marks;
    int number;
    float total_credits=0;
    float total_marks=0;
    float sgpa;
    public void info()
    {
        Scanner obj=new Scanner(System.in);
        System.out.println("enter name");
        name=obj.nextLine();
        System.out.println("enter usn");
        usn=obj.nextLine();
        System.out.println("enter number of subjects");
        number=obj.nextInt();
        credits=new int[number];
        marks=new int[number];
        System.out.println("enter the subject marks and its credit");
        for(int i=0;i<number;i++)
        {
            System.out.println("enter the marks of subject"+(i+1)+":");
            marks[i]=obj.nextInt();
            System.out.println("enter the credit of subject"+(i+1)+":");
```

```
credits[i]=obj.nextInt();  
}  
}
```

```
public int grade(int marks)  
{  
    if(marks>=90)  
    {  
        return 10;  
    }  
    else if(marks>=80)  
    {  
        return 9;  
    }  
    else if(marks>=70)  
    {  
        return 8;  
    }  
    else if(marks>=60)  
    {  
        return 7;  
    }  
    else if(marks>=50)  
    {  
        return 6;  
    }  
    else if(marks>=40)  
    {  
        return 5;  
    }  
    else if(marks>=30)  
    {
```

```
return 4;
}
else if(marks>=20)
{
return 3;
}
else if(marks>=10)
{
return 2;
}
else if(marks>0)
{
return 1;
}
else
{
return 0;
}
}

public void calculate()
{
for(int j=0;j<number;j++)
{
total_marks=total_marks+grade(marks[j])*credits[j];
total_credits=total_credits+credits[j];
}
sgpa=total_marks/total_credits;
}

public void student_details()
{
System.out.println("details of the student");
System.out.println("name:"+name);
```

```
System.out.println("usn:"+usn);
for(int k=0;k<number;k++){
System.out.println("subject" +(k+1)+": credits="+credits[k]+", marks=" +(marks[k]));
}
System.out.println("sgpa:"+sgpa);
}
}
class Main
{
public static void main(String args[])
{
System.out.println("ANANYA AGARWAL");
System.out.println("1BM22CS039\n");
Student obj2=new Student();
obj2.info();
obj2.calculate();
obj2.student_details();
}
}
```

Output:

```
C:\Users\hp\Desktop\Ananya>javac Main.java
C:\Users\hp\Desktop\Ananya>java Main
Ananya Agarwal
1BM22CS039
enter name
Ananya Agarwal
enter usn
1BM22CS039
enter number of subjects
8
enter the subject marks and its credit
enter the marks of subject1:
90
enter the credit of subject1:
4
enter the marks of subject2:
85
enter the credit of subject2:
4
enter the marks of subject3:
92
enter the credit of subject3:
3
enter the marks of subject4:
88
enter the credit of subject4:
3
enter the marks of subject5:
87
enter the credit of subject5:
3
enter the marks of subject6:
95
enter the credit of subject6:
1
enter the marks of subject7:
96
enter the credit of subject7:
1
enter the marks of subject8:
93
enter the credit of subject8:
1
details of the student
name:Ananya Agarwal
usn:1BM22CS039
subject1: credits=4, marks=90
subject2: credits=4, marks=85
subject3: credits=3, marks=92
subject4: credits=3, marks=88
subject5: credits=3, marks=87
subject6: credits=1, marks=95
subject7: credits=1, marks=96
subject8: credits=1, marks=93
sgpa:9.5
```

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.

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

```
public class Book {
```

```
    private String name;
```

```
    private String author;
```

```
    private double price;
```

```
    private int numPages;
```

```
    // Constructor
```

```
    public Book(String name, String author, double price, int numPages) {
```

```
        this.name = name;
```

```
        this.author = author;
```

```
        this.price = price;
```

```
        this.numPages = numPages;
```

```
    }
```

```
    // Getter methods
```

```
    public String getName() {
```

```
        return name;
```

```
    }
```

```
    public String getAuthor() {
```

```
        return author;
```

```
    }
```

```
    public double getPrice() {
```

```
        return price;
```



```
}
```

```
public int getNumPages() {  
    return numPages;  
}
```

```
// Setter methods
```

```
public void setName(String name) {  
    this.name = name;  
}
```

```
public void setAuthor(String author) {  
    this.author = author;  
}
```

```
public void setPrice(double price) {  
    this.price = price;  
}
```

```
public void setNumPages(int numPages) {  
    this.numPages = numPages;  
}
```

```
// toString method
```

```
public String toString() {  
    return "Book Details:\nName: " + name + "\nAuthor: " + author + "\nPrice: $" + price +  
"\nNumber of Pages: " + numPages;  
}
```

```
public static void main(String[] args) {  
    System.out.println("ANANYA AGARWAL");  
    System.out.println("1BM22CS039\n");  
    Scanner scanner = new Scanner(System.in);
```

```
// Asking for the number of books to create
System.out.print("Enter the number of books: ");
int n = scanner.nextInt();
scanner.nextLine(); // Consume newline

Book[] books = new Book[n];

// Getting details for each book from the user
for (int i = 0; i < n; i++) {
    System.out.println("\nEnter details for Book " + (i + 1) + ":");
    System.out.print("Name: ");
    String name = scanner.nextLine();
    System.out.print("Author: ");
    String author = scanner.nextLine();
    System.out.print("Price: $");
    double price = scanner.nextDouble();
    System.out.print("Number of Pages: ");
    int numPages = scanner.nextInt();
    scanner.nextLine(); // Consume newline

    // Creating the book object
    books[i] = new Book(name, author, price, numPages);
}

// Displaying details of the books
for (int i = 0; i < n; i++) {
    System.out.println("\nBook " + (i + 1) + ":\n" + books[i].toString() + "\n");
}

// Closing scanner
scanner.close();
```

```
}  
}
```

Output:

```
C:\Users\hp\Desktop\Ananya>javac Book.java  
  
C:\Users\hp\Desktop\Ananya>java Book  
Ananya Agarwal  
1BM22CS039  
  
Enter the number of books: 1  
  
Enter details for Book 1:  
Name: Mistborn  
Author: Brandon Sanderson  
Price: $1500  
Number of Pages: 800  
  
Book 1:  
Book Details:  
Name: Mistborn  
Author: Brandon Sanderson  
Price: $1500.0  
Number of Pages: 800
```

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.

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

```
abstract class Shape {  
    protected int dimension1;  
    protected int dimension2;  
  
    public Shape(int dimension1, int dimension2) {  
        this.dimension1 = dimension1;  
        this.dimension2 = dimension2;  
    }  
  
    abstract void printArea();  
}
```

```
class Rectangle extends Shape {  
    public Rectangle(int length, int width) {  
        super(length, width);  
    }  
  
    void printArea() {  
        int area = dimension1 * dimension2;  
        System.out.println("Area of Rectangle: " + area);  
    }  
}
```

```
class Triangle extends Shape {
```

```
public Triangle(int base, int height) {  
    super(base, height);  
}
```

```
void printArea() {  
    double area = 0.5 * dimension1 * dimension2;  
    System.out.println("Area of Triangle: " + area);  
}  
}
```

```
class Circle extends Shape {  
    public Circle(int radius) {  
        super(radius, radius);  
    }  
}
```

```
void printArea() {  
    double area = Math.PI * dimension1 * dimension1;  
    System.out.println("Area of Circle: " + area);  
}  
}
```

```
class Main {  
    public static void main(String[] args) {  
        System.out.println("ANANYA AGARWAL");  
        System.out.println("1BM22CS039\n");  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.println("Enter dimensions for Rectangle (length width:");  
        int length = scanner.nextInt();  
        int width = scanner.nextInt();
```

```

Shape rectangle = new Rectangle(length, width);

System.out.println("Enter dimensions for Triangle (base height):");
int base = scanner.nextInt();
int height = scanner.nextInt();
Shape triangle = new Triangle(base, height);

System.out.println("Enter dimension for Circle (radius):");
int radius = scanner.nextInt();
Shape circle = new Circle(radius);

System.out.println("\nAreas:");
rectangle.printArea();
triangle.printArea();
circle.printArea();

scanner.close();
}
}

```

Output:

```

C:\Users\hp\Desktop\Ananya>javac Main.java

C:\Users\hp\Desktop\Ananya>java Main
Ananya Agarwal
1BM22CS039

Enter dimensions for Rectangle (length width):
5
2
Enter dimensions for Triangle (base height):
5
2
Enter dimension for Circle (radius):
2

Areas:
Area of Rectangle: 10
Area of Triangle: 5.0
Area of Circle: 12.566370614359172

```

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 Cur-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.

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

```
class Account {
```

```
    String customerName;
```

```
    int accountNumber;
```

```
    String accountType;
```

```
    double balance;
```

```
    public Account(String customerName, int accountNumber, String accountType, double balance) {
```

```
        this.customerName = customerName;
```

```
        this.accountNumber = accountNumber;
```

```
        this.accountType = accountType;
```

```
        this.balance = balance;
```

```
    }
```

```
    public void deposit(double amount) {
```

```
        balance += amount;
```

```
        System.out.println("Deposit successful. New balance: " + balance);
```

```
}
```

```
public void displayBalance() {  
    System.out.println("Current balance: " + balance);  
}
```

```
// Stub method for withdrawal
```

```
public void withdraw(double amount) {  
    System.out.println("Withdrawal method not implemented for this account type.");  
}  
}
```

```
class CurAcct extends Account {
```

```
    double minimumBalance;  
    double serviceCharge;
```

```
    public CurAcct(String customerName, int accountNumber, double balance, double  
minimumBalance, double serviceCharge) {  
        super(customerName, accountNumber, "Current", balance);  
        this.minimumBalance = minimumBalance;  
        this.serviceCharge = serviceCharge;  
    }
```

```
public void withdraw(double amount) {  
    if (balance - amount < minimumBalance) {  
        System.out.println("Insufficient balance. Withdrawal not permitted.");  
    } else {  
        balance -= amount;  
        System.out.println("Withdrawal successful. New balance: " + balance);  
        if (balance < minimumBalance) {  
            balance -= serviceCharge;
```



```

        System.out.println("Service charge of " + serviceCharge + " applied. New balance: "
+ balance);
    }
}
}
}

```

```

class SavAcct extends Account {
    double interestRate;

    public SavAcct(String customerName, int accountNumber, double balance, double
interestRate) {
        super(customerName, accountNumber, "Savings", balance);
        this.interestRate = interestRate;
    }
}

```

```

public void depositInterest() {
    double interest = balance * (interestRate / 100);
    balance += interest;
    System.out.println("Interest deposited. New balance: " + balance);
}

```

```

public void withdraw(double amount) {
    if (balance - amount < 0) {
        System.out.println("Insufficient balance. Withdrawal not permitted.");
    } else {
        balance -= amount;
        System.out.println("Withdrawal successful. New balance: " + balance);
    }
}
}
}

```

```
class Bank {  
    public static void main(String[] args) {  
System.out.println("ANANYA AGARWAL");  
System.out.println("1BM22CS039\n");  
        Scanner scanner = new Scanner(System.in);  
  
        System.out.println("Enter customer name:");  
        String customerName = scanner.nextLine();  
  
        System.out.println("Enter account number:");  
        int accountNumber = scanner.nextInt();  
  
        System.out.println("Enter initial balance:");  
        double initialBalance = scanner.nextDouble();  
  
        System.out.println("Enter account type (1 for Current, 2 for Savings):");  
        int accountTypeChoice = scanner.nextInt();  
  
        Account account;  
        if (accountTypeChoice == 1) {  
            System.out.println("Enter minimum balance:");  
            double minimumBalance = scanner.nextDouble();  
  
            System.out.println("Enter service charge:");  
            double serviceCharge = scanner.nextDouble();  
  
            account = new CurAcct(customerName, accountNumber, initialBalance,  
minimumBalance, serviceCharge);  
        } else {  
            System.out.println("Enter interest rate:");  
            double interestRate = scanner.nextDouble();  

```

```
        account = new SavAcct(customerName, accountNumber, initialBalance,
interestRate);
    }
```

```
    System.out.println("Enter deposit amount:");
    double depositAmount = scanner.nextDouble();
    account.deposit(depositAmount);
```

```
    System.out.println("Enter withdrawal amount:");
    double withdrawalAmount = scanner.nextDouble();
    account.withdraw(withdrawalAmount);
```

```
    account.displayBalance();
```

```
    if (account instanceof SavAcct) {
        ((SavAcct) account).depositInterest();
    }
```

```
    account.displayBalance();
```

```
    scanner.close();
}
}
```

Output:

```
C:\Users\hp\Desktop\Ananya>javac Bank.java

C:\Users\hp\Desktop\Ananya>java Bank
Ananya Agarwal
1BM22CS039

Enter customer name:
Ananya Agarwal
Enter account number:
2004
Enter initial balance:
5000
Enter account type (1 for Current, 2 for Savings):
1
Enter minimum balance:
500
Enter service charge:
50
Enter deposit amount:
50
Deposit successful. New balance: 5050.0
Enter withdrawal amount:
550
Withdrawal successful. New balance: 4500.0
Current balance: 4500.0
Current balance: 4500.0
```

Lab Program 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.

```
package CIE;
```

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

```
public class Student{  
    public String name;  
    public String usn;  
    public int sem;  
  
    public void display(){  
        Scanner sc = new Scanner(System.in);  
        System.out.println("Name:");  
        name = sc.next();  
        System.out.println("USN:");  
        usn = sc.next();  
        System.out.println("Sem:");  
        sem = sc.nextInt();  
    }  
}
```

```
package CIE;
```

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

```
public class Internals extends Student{  
    public double ciem[];  
  
    public void display(){
```

```

        ciem = new double[5];

        Scanner c = new Scanner(System.in);

        System.out.println("Enter marks out of 50:");

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

            ciem[i] = c.nextDouble();

        }

    }

}

```

```

package SEE;

```

```

import CIE.*;

```

```

import java.util.*;

```

```

public class Externals extends CIE.Student{

```

```

    public double seem[];

```

```

    public void display(){

```

```

        seem = new double[5];

```

```

        Scanner s = new Scanner(System.in);

```

```

        System.out.println("SEE marks for 5 subjects out of 100:");

```

```

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

```

```

            seem[i]=s.nextDouble();

```

```

        }

```

```

    }

```

```

}

```

```

import CIE.*;

```

```

import SEE.*;

```

```

import java.util.*;

```

```

public class Main{

```

```

    System.out.println("ANANYA AGARWAL");

```

```

System.out.println("1BM22CS039\n");

    public static void main(String[] args){

        int n;

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter no. of students:");

        n = sc.nextInt();

        CIE.Student st[] = new CIE.Student[n];

        CIE.Internals in[] = new CIE.Internals[n];

        SEE.Externals ex[] = new SEE.Externals[n];


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

            st[i] = new CIE.Student();

            in[i] = new CIE.Internals();

            ex[i] = new SEE.Externals();


            st[i].display();

            in[i].display();

            ex[i].display();

            System.out.println("Total Marks of "+st[i].name+"\n");

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

                System.out.println(in[i].ciem[j]+ex[i].seem[j]/2);

            }

        }

    }

}

```

Output:

```
Ananya Agarwal
1BM22CS039
Enter n:
1
Enter details 1
Enter U, N, S:

039
Ananya Agarwal
1
Enter im and sm of sub 1
48
45
Enter im and sm of sub 2
38
49
Enter im and sm of sub 3
32
42
Enter im and sm of sub 4
50
50
Enter im and sm of sub 5
39
49
Final marks of Ananya Agarwal
Course 1 = 93
Course 2 = 87
Course 3 = 74
Course 4 = 100
Course 5 = 88
```


Lab Program 7:

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=father's age.

```
// Custom exception class WrongAge
```

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

```
// Custom exception class WrongAge
```

```
class WrongAge extends Exception {
```

```
    public WrongAge(String message) {
```

```
        super(message);
```

```
    }
```

```
}
```

```
// Base class Father
```

```
class Father {
```

```
    private int age;
```

```
// Constructor for Father class
```

```
public Father(int age) throws WrongAge {
```

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

```
        throw new WrongAge("Age cannot be negative");
```

```
    }
```

```
    this.age = age;
```

```
}
```

```
// Getter method for age
```

```
public int getAge() {
```

```
    return age;
```

```
}
```

```
}
```

```
// Derived class Son
class Son extends Father {
    private int sonAge;

    // Constructor for Son class
    public Son(int fatherAge, int sonAge) throws WrongAge {
        super(fatherAge);
        if (sonAge >= fatherAge) {
            throw new WrongAge("Son's age should be less than Father's age");
        }
        this.sonAge = sonAge;
    }

    // Getter method for son's age
    public int getSonAge() {
        return sonAge;
    }
}

class Main {
    public static void main(String[] args) {
        System.out.println("ANANYA AGARWAL");
        System.out.println("1BM22CS039\n");
        Scanner scanner = new Scanner(System.in);

        try {
            // Taking input for father's age
            System.out.println("Enter Father's age:");
            int fatherAge = scanner.nextInt();

            // Creating a Father object
```

```

    Father father = new Father(fatherAge);

    System.out.println("Father's age: " + father.getAge());

    // Taking input for son's age
    System.out.println("Enter Son's age:");
    int sonAge = scanner.nextInt();

    // Creating a Son object
    Son son = new Son(fatherAge, sonAge);

    System.out.println("Son's age: " + son.getSonAge());
} catch (WrongAge e) {
    System.out.println("Exception caught: " + e.getMessage());
}

scanner.close();
}
}

```

Output:

```

C:\Users\hp\Desktop\Ananya>javac Main.java

C:\Users\hp\Desktop\Ananya>java Main
Ananya Agarwal
1BM22CS039

Enter Father's age:
50
Father's age: 50
Enter Son's age:
20
Son's age: 20

```

Lab Program 8:

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.

```
class BMSCEThread extends Thread {  
    public void run() {  
        while(true) {  
            System.out.println("BMS College of Engineering");  
            try {  
                Thread.sleep(10000); // Sleep for 10 seconds  
            } catch (InterruptedException e) {  
                e.printStackTrace();  
            }  
        }  
    }  
}
```

```
class CSEThread extends Thread {  
    public void run() {  
        while(true) {  
            System.out.println("CSE");  
            try {  
                Thread.sleep(2000); // Sleep for 2 seconds  
            } catch (InterruptedException e) {  
                e.printStackTrace();  
            }  
        }  
    }  
}
```

```
class Main {  
    public static void main(String[] args) {  
        System.out.println("ANANYA AGARWAL");  
    }  
}
```

```
System.out.println("1BM22CS039\n");

    BMSCEThread bmsceThread = new BMSCEThread();
    CSEThread cseThread = new CSEThread();

    bmsceThread.start();
    cseThread.start();
}
}
```

Output:

```
C:\Users\hp\Desktop\Ananya>javac Main.java

C:\Users\hp\Desktop\Ananya>java Main
Ananya Agarwal
1BM22CS039

BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
```