

1) Quadratic

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

```
class quad
```

{

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

{

```
        int a; int b; int c;
```

```
        double d;
```

```
        double r1, r2;
```

```
        Scanner s1 = new Scanner (System . in);
```

```
        System . out . println ("Enter the value of a,b,c");
```

```
        a = s1 . nextInt ();
```

```
        b = s1 . nextInt ();
```

```
        c = s1 . nextInt ();
```

```
        d = b * b - (4 * a * c);
```

```
        if (a == 0)
```

{

```
            System . out . println ("The equation is not quadratic");
```

}

```
        else if (d == 0)
```

{

```
            System . out . println ("The roots are real and equal");
```

```
            r1 = - b / (2 * a);
```

```
            System . out . println (r1);
```

}

```
        else if (d > 0)
```

{

```
            System . out . println ("The roots are real and distinct");
```

```
            r1 = (- b + Math . sqrt (d)) / (2 * a);
```

```
            r2 = (- b - Math . sqrt (d)) / (2 * a);
```

System.out.println( $r_1$  + " " +  $r_2$ );  
}

else

{

System.out.println("The roots are imaginary.");  
 $r_1 = -b / (2 * a);$

$r_2 = \text{Math.sqrt}(\text{Math.abs}(d));$

System.out.println("The roots are " + " $+r_1 + i r_2$ " + " $+ -r_1 - i r_2$ ");  
+ Math.sqrt( $r_2$ ) + " and " +  $r_1 + -r_2$  + " \* " + Math.sqrt( $r_2$ ))

}

}

Command Prompt

```
C:\Users\nbrij>cd C:\Engg\3rd sem\JAVA lab
```

```
C:\Engg\3rd sem\JAVA lab>javac quad.java
```

```
C:\Engg\3rd sem\JAVA lab>java Quad
```

```
Enter the coefficients a,b,c
```

```
0
```

```
1
```

```
2
```

```
Invalid input for a
```

```
C:\Engg\3rd sem\JAVA lab>java Quad
```

```
Enter the coefficients a,b,c
```

```
1
```

```
2
```

```
3
```

```
The roots have no real solution and are imaginary
```

```
-1.0+i1.4142135623730951
```

```
1.4142135623730951-i1.4142135623730951
```

```
C:\Engg\3rd sem\JAVA lab>java Quad
```

```
Enter the coefficients a,b,c
```

```
1
```

```
2
```

```
1
```

```
The roots are real and equal
```

```
Roots are r1=r2=-1.0
```

```
C:\Engg\3rd sem\JAVA lab>java Quad
```

```
Enter the coefficients a,b,c
```

```
1
```

```
4
```

```
1
```

```
The roots are real and distinct
```

```
The roots are:-0.2679491924311228 -3.732050807568877
```

```
C:\Engg\3rd sem\JAVA lab>
```

## LAB - 2

PAGE:

DATE:

PAGE:

2)

```
import java.util.Scanner;
class Student {
    void display (String name , String USN)
    {
        System.out.println("USN of student" + USN);
        System.out.println ("Name of Student" + name);
    }
    void calculateSgpa (double[] marks , double[] credits,
                         int number)
    {
        double gradepoints [] = new double [number];
        double sgpa , sum=0 , lsum=0;
        for (int i=0 ; i < number , i++)
        {
            if (marks[i] >= 90)
                gradepoints[i] = 10;
            else if (marks[i] >= 80)
                grade_point[i] = 9;
            else if (marks[i] >= 70)
                grade_point[i] = 8;
            else if (marks[i] >= 60)
                grade_point[i] = 7;
            else if (marks[i] >= 50)
                grade_point[i] = 6;
            else if (marks[i] >= 40)
                grade_point[i] = 5;
            else
                gradepoints[i] = 0;
        }
    }
}
```

```
for (int i=0; i<number; i++)
```

```
{  
    sum += credit[i] * gradePoint[i];  
}
```

```
for (int i=0; i<number; i++)
```

```
{  
    tnum += credit[i];  
}
```

```
SGPA = sum / tnum;
```

```
System.out.println("SGPA is " + SGPA);
```

```
class SGPA {
```

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

```
    System.out.println("Enter name and USN of student");
```

```
    String name = s.next();
```

```
    String USN = s.next();
```

```
    Student s1 = new Student();
```

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

```
    int number = s.nextInt();
```

```
    double credits[] = new double[number];
```

```
    double marks[] = new double[number];
```

```
    for (int i=0; i<number; i++)
```

```
{
```

```
        System.out.print("Credit of subject " + (i+1) + ": ");
```

```
        credits[i] = s.nextDouble();
```

```
        System.out.print("Marks of subject " + (i+1) + ": ");
```

```
        marks[i] = s.nextDouble();
```

```
}
```

```
s1.display(name, USN);
```

```
s1.calculateSGPA(marks, credits, number);
```

```
}
```

```
C:\Users\bmsce\Desktop\1BM21CS040>java Sgpa
enter your name:
Brijesh N
enter your usn:
40
enter the marks of each subject
66
77
88
56
87
enter the no of credits for each subject
4
3
3
1
3
name:Brijesh N
usn:40
marks of subject1=66
no of credits for the subj above=4
marks of subject2=77
no of credits for the subj above=3
marks of subject3=88
no of credits for the subj above=3
marks of subject4=56
no of credits for the subj above=1
marks of subject5=87
no of credits for the subj above=3
sgpa=8.0
```

3)

```
import java.util.*;
import java.lang.*;
class Book
{
    String name, author; int price, numPages;
    void getval()
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter book name");
        name = sc.nextLine();
        System.out.println("Enter author name");
        author = sc.nextLine();
        System.out.println("Enter price");
        price = sc.nextInt();
        System.out.println("Enter No. of pages");
        numPages = sc.nextInt();
    }
    public String toString()
    {
        return name + " " + author + " " + price + " " + numPages + "";
    }
    void display(Book o)
    {
        System.out.println(o);
    }
}
```

## Class BookList

{

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

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

```
System.out.println ("Enter the no. of Book objects")  
int n = in.nextInt();
```

```
Book [] ob = new Book [n];
```

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

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

```
for (int i = 0; i < n; i++)  
{ ob[i].getval (); }
```

```
for (int i = 0; i < n; i++)  
{ ob[i].display (ob[i]); }
```

```
}
```

Enter the number of books:

2

Enter the name of the book:

Power of thinking

Enter the author of the book:

James

Enter the price of the book:

450

Enter the number of pages of the book:

200

**BOOK DETAILS:**

NAME: Power of thinking

AUTHOR: James

PRICE: 450.0/-

NUMBER OF PAGES: 200

Enter the name of the book:

think like a monk

Enter the author of the book:

jay shetty

Enter the price of the book:

500

Enter the number of pages of the book:

450

**BOOK DETAILS:**

NAME: think like a monk

AUTHOR: jay shetty

PRICE: 500.0/-

NUMBER OF PAGES: 450

4)

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

```
abstract class Shape {
```

```
    double a, b;
```

```
    abstract void findArea();
```

```
}
```

```
class Triangle extends Shape {
```

```
    void getData(double x, double y) {
```

```
        a = x; b = y; }
```

```
    void printArea() {
```

```
}
```

```
    double area = 0.5 * a * b;
```

```
    System.out.println("Area of triangle = " + area);
```

```
}
```

```
    void getData(double x, double y) {
```

```
        a = x; b = y; }
```

```
    void printArea() {
```

```
        double area = a * b;
```

```
        System.out.println("Area of rectangle = " + area);
```

```
}
```

```
}
```

```
class Circle extends Shape {
```

```
    void getData(double x) {
```

```
        a = x; }
```

```

    void getdata
void getdata() ( )
{ a = x; }

```

```
void printArea()
{ double area = 3.14 * a * a;
```

```
System.out.println("Area of Circle = " + area);
```

```
}
```

(1)

```
Class abstractarea { public static void main (String args) }
```

```
{ int ch;
```

```
Shape si;
```

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

```
Rectangle r = new Rectangle();
```

```
Triangle t = new Triangle();
```

```
Circle c = new Circle();
```

```
SOP ("1. Area of Rectangle in 2. Area of Triangle in
```

```
3. Area of Circle in Enter your choice");
```

```
ch = sc.nextInt();
```

```
switch (ch) {
```

*(Case 1: System.out.println("Enter length and breadth")*

```
double a = sc.nextInt();
```

```
double b = sc.nextInt();
```

```
r.getData (l, b);
```

```
r.printArea();
```

```
System.out.println ("Area of Rectangle = " + r.area);
```

*(Case 2: System.out.println("Enter base and height")*

```
double b = sc.nextInt();
```

```
double h = sc.nextInt();
```

```
t.getData (b, h);
```

```
t.printArea();
```

```
System.out.println ("Area of Triangle = " + t.area);
```

Enter height and width of rectangle

2

3

Area of Rectangle is 6

Enter height and base of rectangle

3

6

Area of Trianle is 9.0

Enter radius of Circle

4

Area of Circle is 50.24

C:\Engg\3rd sem\JAVA lab\p4>

Q)

```
Import java.util.*;
```

```
class Bank
```

```
    public String name;  
    public int acc-no;  
    public float bal;  
    public float si;  
    public void acceptance()
```

```
Scanner s = new Scanner (System.in)
```

```
System.out.print("Enter the account welder number : ");
```

```
System.out.print("Enter the account number : ");
```

```
acc-no = s.nextInt();
```

```
System.out.print("Enter account balance : ");
```

```
bal = s.nextFloat();
```

```
public void display()
```

```
SOP ("Details");
```

```
SOP ("Name : " + Name + " Account number : ", +  
      acc-no + "\n Balance : " + bal);
```

public void simple\_interest()

{

System.out.println("With Rate of Interest = 8%");  

$$SI = (bal * 8) / 100;$$

{

System.out.print("Simple interest for one year is Rs" + SI);

Class Savings extends Bank

{

public void cheque()

{

System.out.println("No Cheque Services");

{

public void withdrawal()

{

float amount;

Scanner a = new Scanner(System.in);

System.out.println("No minimum Balance Required");

System.out.print("Enter amount to be withdrawn: ");

amount = a.nextFloat();

if (amount > supr.bal)

{

SOP("Balance is insufficient");

{

else

{

supr.bal = supr.bal - amount;

SOP("Amount " + "withdrawn " )

SOP("Available balance = " + supr.bal);

{

{

{

class Current extends Bank

public void deposit()

}

SOP("Cheque Service available");

}

public void withdrawal()

{

float amount;

Scanner a = new Scanner (System.in)

SOP("Minimum Balance = Rs 1000.00");

if (Super.Bal < 1000)

{

SOP("Balance is sufficient to withdraw");

float Service\_charge;

Service\_charge = (1 \* Super.Bal) / 100;

Super.Bal = Super.Bal - Service\_charge;

~~System.SOP("Service charge of Rs " + Service\_charge + " added");~~

SOP("Available Balance = Rs " + Super.Bal);

}

else

{

SOP("Enter amount to be withdrawn : ");

amount = a.nextFloat();

if (amount > (Super.Bal - 1000))

{

SOP("Balance is insufficient");

}

else

{

~~SOP(Super.Bal = Super.Bal - amount);~~

SOP ("amount + " withdrawal");

SOP ("Available balance = " + Super. bal);

{}

{}

{}

class main {

PSUM (Savings obj [ ])

{}

Savings obj [ ] = new Savings [ ];

current obj [ ] = new Current [ ];

SOP ("Enter the no. of accounts");

int n = n. nextInt();

int a[ i = 0 ];

int j = 0;

int k = 0;

while ( i &lt; n )

{}

SOP ("New account" + ( i + 1 ));

SOP ( "1 " + Saving in "Current" );

SOP ("Enter type of account");

int ch = n. nextInt();

if ( ch == 1 )

{}

obj [ j ]. newSavings();

obj [ j ]. accept();

obj [ j ]. display();

obj [ j ]. cheque();

obj [ j ]. simpleInterest();

obj [ j ]. withdraw();

j ++;

{}

else

2

```
obj2[k] = new Current();
obj2[k].accept();
obj2[k].display();
obj2[k].cheque();
obj2[k].withdraw();
k++;
}
```

3

ist,

3

4

}

### Output :-

Enter no number of account : 1

Account :

1> Saving

2> Current

Enter type of account : 1

Enter name of acc holder : ee

Enter account no. : 28845

Enter account balance : 10000

& Details \*

Name : ee

A/c no. : 28845

Balance : 70000

No Cheque Service

Rate of interest = 8%

Simple interest (for one year) = Rs 560.0

No min bal required

```
Enter your account type:  
1. Savings account  
2. Current account  
  
Cheque Facility not available  
Enter customer name:  
E  
Enter ee's account number  
2  
Enter balance amount  
0000  
Customer Name:ee  
Your account number:22  
Your Account Balance:10000.0  
Press 1 to deposit  
  
invalid Input  
Enter rate of interest  
  
Enter number of times interest applied per time period  
Enter number of time periods  
  
Interest amount=11576.250000000002  
Balance amount without interest is:10000.0  
Available balance after updating is:11576.250000000002  
Press 1 to withdraw amount  
  
Enter the amount to be withdrawn  
500  
Available Balance:9500.0  
C:\Users\bmsce\Desktop>java Lab5
```

Enter your account type:

1. Savings account

2. Current account

2

Cheque Facility available

Enter customer name

er

Enter er's account number

23

Enter balance amount

50000

Customer Name:er

Your account number:23

Your Account Balance:50000.0

Press 1 to deposit

1

Enter amount to be deposited

4500

Press 1 to withdraw amount

1

Enter the amount to be withdrawn

2500

## LAB - 7

DATE:

PAGE:

- Q) 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 Bms implements Runnable {

Thread t1;

Bms () {

t1 = new Thread(this, "Bms");

}

public void run() {

try {

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

S. O. P ("BMS College of Engineering");

Thread.sleep(10000);

}

}

catch (InterruptedException e) {

S. O. P ("BMS interrupted");

}

} S. O. P ("Exiting;" + t1);

}

}

Class Cse implements Runnable {

Thread t2;

Cse ()

{

t2 = new Thread(this, "Cse");

}

public void run() {

try {

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

SOP ("CSE");

Thread.Sleep(2000);

}

Catch (Uncaught Exception e) {

SOP ("CSE interrupted");

}

SOP ("Ending : " + t2);

}

Class Thread has {

PSVM (Starting args[])

{

BMS obj1 = new BMS();

CS2 obj2 = new CS2();

obj1.t1.start();

obj2.t2.start();

}

}

Output :

BMS

CSE

CSE

CSE

CSE

BMS

Error : thread [CS2, 5, main]

BMS

BMS

Error : thread [BMS, 5, main]

C:\Engg\3rd sem\JAVA lab>jav  
BMS College of Enginnering  
CSE  
CSE  
CSE  
CSE  
CSE  
CSE  
BMS College of Enginnering  
CSE  
CSE  
CSE  
CSE  
C:\Engg\3rd sem\JAVA lab>

### Lab program - 6

- a) A program demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class "Son" which extends base class. In Father class implement a constructor which takes age and throws the exception wrongAge() when the input is 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.

```
import java.util.*;  
class FatherAgeException extends Exception  
{  
    public String toString()  
    {  
        return ("Father's age less than 0");  
    }  
}
```

```
class SonAgeException extends Exception  
{  
    int a;  
}
```

```
SonAgeException (int age)  
{  
    a = age;  
}
```

```
public String toString()  
{  
    if (a < 0)  
}
```

```
.return ("Son's age is less than 0");  
else
```

```
return ("Son's age is more than Father's  
age");  
}
```

class Father {

    public int age1;

    Scanner S = new Scanner(System.in)

    } } }

System.out.print("Enter Father age: ");

age1 = S.nextInt();

}

void ex1() throws FatherAgeException {

    if (age1 < 0)

        throw new FatherAgeException();

}

}

Class Son extends Father {

    public int age2;

    Scanner S;

    S = new Scanner(System.out);

    age2 = S.nextInt();

void ex2() throws SonAgeException {

    if (age2 < 0 || age2 > Super.age1)

        throw new SonAgeException(age2);

S

S

Class FatherSon {

    PSVM (String args[])

        fatherf = new Father();

    } } }

key {

    SI : on1();

    ff.e ent();

ff

Catch (FatherAgeException())

    { System.out.println(e); }

try {  
 System.out.println("Hello World");  
}

Catch (Son age exception o) {  
 System.out.println("Exception caught");  
}

{

Output

Enter father's age : -2

Enter son's age : 5

Father age is less than 0

Son's age is more than father's age

Enter father's age : 40

Enter son's age : 50

Son's age is more than father's age

Enter father's age : 40

Enter son's age : 15

```
: \Users\nbrij>cd C:\Engg\3rd sem\JAVA lab\p4  
C:\Engg\3rd sem\JAVA lab\p4>javac p4.java  
C:\Engg\3rd sem\JAVA lab\p4>java A_main  
Enter the father's age:  
Enter the age of son:  
2  
invalid input. Father's age can not be lesser than 0  
age of father less than son  
C:\Engg\3rd sem\JAVA lab\p4>java A_main  
Enter the father's age:  
.3  
Enter the age of son:  
2  
FATHER'S AGE:23  
SON'S AGE:2  
C:\Engg\3rd sem\JAVA lab\p4>java A_main  
Enter the father's age:  
22  
Enter the age of son:  
25  
age of father less than son  
C:\Engg\3rd sem\JAVA lab\p4>
```