

updated program.

paperism

Date:

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

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

```
class Quadratic
```

```
{
```

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

```
    double x1, x2, d;
```

```
    void getd()
```

```
{
```

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

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

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

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

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

```
}
```

```
void compute()
```

```
{
```

```
    while(a == 0)
```

```
{
```

```
    System.out.println("Not a quadratic eq.");
```

```
    System.out.println("Enter a non zero  
value for a:");
```

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

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

```
{
```

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

{ if ($d == 0$)

$$x_1 = (-b) / (2 * a);$$

System.out.println("Roots are real and equal");

{ System.out.println("Root1 = Root2 = " + x1);

{ else if ($d > 0$)

$$x_1 = ((-b) + \text{Math.sqrt}(d)) / (2 * a);$$

$$x_2 = ((-b) - \text{Math.sqrt}(d)) / (2 * a);$$

System.out.println("Roots are real & distinct");

{ System.out.println("Root1 = " + x1 + " Root2 = " + x2);

{ else if ($d < 0$)

System.out.println("Roots are imaginary");

$$x_1 = (-b) / (2 * a);$$

$$x_2 = \text{Math.sqrt}(-d) / (2 * a);$$

System.out.println("Root1 = " + x1 + " + i " + x2);

System.out.println("Root1 = " + x1 + " - i " + x2);

{ }

class QuadraticEq

{

public static void main (String args [])

Quadratic q = new Quadratic();

q.getd();

q.compute();

{}

{}

Output:

Enter the coefficients of a, b, c

1 4 4

Roots are real and equal

Root1 = Root2 = -2.0

Enter the coefficient of a, b, c

1 9 20

Roots are real and distinct

Root1 = -4.0 Root2 = -5.0

Enter the coefficient of a, b, c

1 2 30

Roots are imaginary

Root1 = -1.0 + i 5.38516

Root1 = -1.0 - i 5.38516

Enter the coefficients of a, b, c

0 0 0

Not a quadratic equation

Enter a non zero value for a:

2 3 4

Roots are real and equal

Root1 = Root2 = 0.0

```
Command Prompt + × - × Microsoft Windows [Version 10.0.22621.755]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Aisha Taffazul>cd C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs>
C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs>javac quadratic.java

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs>java QuadraticEq
Enter the coefficients of a,b,c
1 4 4
Roots are real and equal
Root1 = Root2 = -2.0

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs>java QuadraticEq
Enter the coefficients of a,b,c
1 9 20
Roots are real and distinct
Root1 = -4.0 Root2 = -5.0

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs>java QuadraticEq
Enter the coefficients of a,b,c
1 2 30
Roots are imaginary
Root1 = -1.0 + i5.385164807134504
Root1 = -1.0 - i5.385164807134504

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs>java QuadraticEq
Enter the coefficients of a,b,c
0 0 0
Not a quadratic equation
Enter a non zero value for a:
2 3 4
Roots are real and equal
Root1 = Root2 = 0.0

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs>
```

Q Develop a Java program to create a class Students with member USN, name, an array credits and an array marks. Include methods to accept and display details and to calculate SGPA of a student.

```
import java.util.Scanner;
class Student
{
    String name, usn;
    int marks[] = new int [5];
    int credits[] = new int [5];

    void input()
    {
        Scanner ss = new Scanner (System.in);
        System.out.println ("Enter your name :");
        name = ss.nextLine ();
        System.out.println ("Enter your usn :");
        usn = ss.next ();
        System.out.println ("Enter the marks of each subject :");
        for (int i=0; i< 5; i++)
        {
            marks [i] = ss.nextInt ();
        }
        System.out.println ("Enter the number of credits for each subject :");
        for (int i=0; i< 5; i++)
        {
            credits [i] = ss.nextInt ();
        }
    }

    void display()
    {
```

```
System.out.println("NAME: " + name);
System.out.println("USN: " + usn);
for (int i = 0; i < 5; i++)
    }
```

```
System.out.println("Marks of
each subject" + (i + 1) + " = " + mark);
System.out.println("Number of
credits for each subject" + (i + 1)
+ " = " + credits[i]);
    }
```

```
}
```

```
void calc()
    }
```

```
int gr_point[] = new int[5];
int sgpa = 0;
```

```
int sum = 0; float res;
for (int i = 0; i < 5; i++)
    }
```

```
if (marks[i] >= 90)
```

```
gr_point[i] = 10;
```

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

```
gr_point[i] = 9;
```

```
(+) else if (marks[i] >= 70)
```

```
gr_point[i] = 8;
```

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

```
gr_point[i] = 7;
```

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

```
gr_point[i] = 6;
```

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

```
gr_point[i] = 5;
```

```
else if (marks[i] >= 35)
```

```
gr_point[4];
```

```
else if (marks[i] < 35 && marks > 0)
```

```
gr_point = 0;
```

else

System.out.println (" Invalid marks

for subject " + (i+1) + "

entered. Try Again ");

sgpa += (gr_point[i] * credits[i]);

sum += (credits[i]);

}

res = (float) sgpa / sum;

System.out.println (" SGPA = " + res);

}

}

{

public static void main (String xx [])

{

student s1 = new student();

s1. input();

s1. display();

s1. calc();

{

.

Output:

Enter your USN:

IBM21C5010

Enter the marks of each subject:

90

91

92

93

94

Enter the number of credits for each subject:

3

4

3

2

2

NAME: aisha

USN: IBM21CS010

Marks of subject 1 = 80

No. of credits for the subject 1 = 3

Marks of subject 2 = 91

No. of credits for the subject 2 = 4

Marks of subject 3 = 92

No. of credits for the subject 3 = 3

Marks of subject 4 = 93

No. of credits for the subject 4 = 2

Marks of subject 5 = 94

No. of credits for the subject 5 = 2

SGPA = 10.0

```
Command Prompt + × - × Microsoft Windows [Version 10.0.22621.755]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Aisha Taffazul>cd C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\sgpa

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\sgpa>javac lab_prog_2.java

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\sgpa>java sgpa
Enter your name:
aisha
Enter your USN:
1BM21CS010
Enter the marks of each subject:
90
91
92
93
94
Enter the number of credits for each subject:
3
4
3
2
2
NAME:aisha
USN:1BM21CS010
Marks of subject 1 = 90
Number of credits for subject 1 = 3
Marks of subject 2 = 91
Number of credits for subject 2 = 4
Marks of subject 3 = 92
Number of credits for subject 3 = 3
Marks of subject 4 = 93
Number of credits for subject 4 = 2
Marks of subject 5 = 94
Number of credits for subject 5 = 2
SGPA= 10.0

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\sgpa>
```

- Q Create a class Book which contains four members: name, author, price, numPages. 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;
```

```
class Books
```

```
{
```

```
    String name;
```

```
    String author;
```

```
    int price;
```

```
    int numPages;
```

```
Books()
```

```
{}
```

```
Books (String name, String author, int  
        price, int numPages).
```

```
{
```

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

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

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

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

```
}
```

```
public String toString()
```

```
{
```

```
    String name, author, price, numPages;
```

```
    name = "Book name: " + this.name + "\n";
```

```
    price = "Price: " + this.price + "\n";
```

```
    numPages = "Number of pages: " + this.numPages;
```

paperism
Date.

"\n";
} return name + author + price + numPages;
}

class Main

{ public static void main (String args [])

: Scanner s = new Scanner (System.in);
int n;

String name;

String author;

int price;

int numPages;

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

n = s.nextInt();

Books b [];

b = new Books [n];

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

System.out.println ("Book" + (i+1) + ":");

System.out.println ("Enter the ^{name} author of the
book :");

name = s.next();

System.out.println ("Enter the author of the book :");

author = s.next();

System.out.println ("Enter the price of the book :");

price = s.nextInt();

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

of the book : ");

numPages = s.nextInt();

b[i] = new Books (name, author, price,
numPages);

} System.out.println();

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

{ System.out.println ("Book" + (i+1) + ":" +
+ b[i]); }

Output:

Enter the number of books : 2

Book 1:

Enter the name of the book : happy

Enter the author of the book : William

Enter the price of the book : 400

Enter the number of pages of the book : 200

Book 2:

Enter the name of the book : Oracle

Enter the author of the book : Herbert Schildt

Enter the price of the book : 1,500

Enter the number of pages of the book : 1912

```
Command Prompt X + Microsoft Windows [Version 10.0.22621.755]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Aisha Taffazul>cd C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\book

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\book>javac Lab_program_3.java

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\book>java Main
Enter the number of books
1
Book1:
Enter the name of the book:
happy
Enter the authors name
willam
Enter the price of the book:
200
Enter the number of pages of the book:
400

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\book>
```

Lab Program 4.

- Q 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.

abstract class Shape

{

 double length;

 double breadth;

 double radius;

Shape (double l, double b)

{

 length = l;

 breadth = b;

}

Shape (double r)

{

 radius = r;

}

abstract double area();

}

Class Rectangle extends Shape

{

Rectangle (double l, double b)

{

 super (l, b);

}

double area()

{

return length * breadth;

{

class triangle extends Shape

{

triangle (double l, double b)

{

super (l, b);

{

double area()

{

return (length * breadth) / 2;

{

class circle extends Shape

{

circle (double rad)

{

super (rad);

{

double area()

{

return 3.14 * radius * radius;

{

class areas {

public static void main (String args)

{

Rectangle r = new Rectangle(2, 2);

triangle t = new triangle(3, 3);

circle c = new circle(2);

System.out.println("area of rectangle : "+
"x.area ());

System.out.println("area of triangle : " + t.area());

System.out.println("area of circle : " + e.area());

OUTPUT:

area of rectangle : 4.0

area of triangle : 4.5

area of circle : 12.56.

N
9/12/22

Command Prompt

X + ▾

Microsoft Windows [Version 10.0.22621.755]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Aisha Taffazul>cd C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\shapes

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\shapes>javac lab_program_4.java

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\shapes>java areas

area of rectangle :4.0

area of triangle :4.5

area of circle : 12.56

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\shapes>

paperism

Date.

Lab Program 5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called saving 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 req. 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 balance

Check for the minimum balance, impose penalty if necessary and update the balance.

import java.util.Scanner;

import java.lang.Math;

class Account

{

String name, acc_type;

int acc_no;

double bal, dep;

Scanner ss = new Scanner(System.in);

void setd()

```
{  
    System.out.println ("Enter your  
    name");  
    name = ss.next();  
    System.out.println ("Enter your  
    account no.");  
    acc_no = ss.nextInt();  
    System.out.println ("Enter your  
    account type (Saving / Current).");  
    acc_type = ss.next();  
    System.out.println ("Enter the  
    account balance.");  
    acc_bal = ss.nextDouble();  
}
```

```
void disp () {  
    System.out.println ("Name: " + name);  
    System.out.println ("Accountno " + acc_no);  
    System.out.println ("Accounttype " + acc_type);  
    System.out.println ("Account balance " +  
        acc_bal);  
}
```

```
void deposite () {  
    System.out.println ("Enter the  
    amount which has to be deposited.");  
    dep = ss.nextDouble();  
    bal = dep;  
    System.out.println ("Balance amount "  
        + bal - acc_bal);  
}
```

{

boolean acc (String acc_type)

{ if (acc_type == "saving")

return true;

else if (acc_type == "current")

return false;

else

return true;

}

class cur_acct extends account

{ int penalty()

double min_bal;

System.out.println ("Enter minimum
balance");

min = ss.nextDouble();

pen = min * 0.05;

{

bal -= pen;

System.out.println ("Penalty imposed
for having insufficient balance"); return;

else

{ System.out.println ("No penalty");

return; }

void withdrawal ()

{

double amt;

System.out.println ("Enter the amount
to be withdrawn");

amt = ss.nextDouble();

int a = penal();

if (a == 1)

```
{  
    if (bal >= amt)  
        { bal = bal - amt;  
        System.out.println("Your account balance  
after withdrawal is :" + bal); }  
    }  
    else  
        System.out.println("The amount  
can't be withdrawn");  
}
```

```
}  
class Sav_acct extends Account
```

```
{  
    void calc_interest()  
    {  
        System.out.println("Enter time in years  
and rate of interest");  
        double t = scan.nextDouble();  
        double r = scan.nextDouble();  
        double CI = bal * Math.pow((1+r/100),t);  
        System.out.println("Account balance &  
compound interest :" + CI);  
    }  
}
```

```
void withdrawal()
```

```
{  
    double amt;  
    System.out.println("Enter amount to  
be withdrawn:");  
    amt = scan.nextInt();  
    if (bal >= amt)  
        { bal = bal - amt;  
        }
```

```
System.out.println("Account Balance  
after withdrawal is :" + bal); }  
else
```

System.out.println("The amount can't be withdrawn");

}

}
class Bank

public static void main (String arg[])

Scanner ss = new Scanner (System.in);

Account bl = new Account();

bl.setd();

if (bl.acc-type.equals("Savings"))

{

Sav-Acc sl = new Sav-Acc();

sl.name = bl.name; sl.acc.no = bl.accno;

sl.acc-type = bl.acc-type; sl.bal = bl.bal;

while (true)

{

System.out.println("Enter your choice:\n 1. Deposit

2. Calculate interest\n 3. Withdraw

4. Display\n 5. Exit");

int choice = ss.nextInt();

switch (choice)

{

case 1: sl.deposit(); break;

case 2: sl.calc-interest(); break;

case 3: sl.withdrawal(); break;

case 4: sl.disp(); break;

case 5: System.exit(0);

default: System.out.println("Invalid input");

}

}

}

else if (bl.acc_type.equals("Current"))

{
 cur_acct cl = new Cur_acct();

 cl.name = bl.name;

 cl.acc_no = bl.acc_no;

 cl.acc_type = bl.acc_type;

 cl.bal = bl.bal;

 while (true)

{

 System.out.println("Enter your choice:

 1. Deposite \n 2. Penalty Check \n 3. Withdraw

 4. Display \n 5. Exit");

 int choice = ss.nextInt();

 switch (choice)

{

 case 1: cl.deposite(); break;

 case 2: cl.penalty(); break;

 case 3: cl.withdrawl(); break;

 case 4: cl.disp(); break;

 case 5: System.exit(0);

 default: System.out.println("Invalid
 input");

}

{

 else

 System.out.println("Invalid Account
 Type");

2.

Output

Enter your Name:

Aisha.

Enter account number:

12345678

Enter your account type : (saving / current)
Savings

Enter the Bank Balance:

123456

Enter your choice:

1. Deposite
2. Calculate interest
3. Withdrawal
4. Display
5. Exit

1

Enter the amount to be deposited

1000

balance amount : 124456.0.

Enter your choice:

1. Deposit
2. Calculate interest
3. Withdraw
4. Display
5. Exit

2

Enter Time in years and Rate of interest

23

2

ACCOUNT BALANCE and compound interest 124456

Enter your choice:

1. Deposite
2. Calculate interest
3. Withdrawal
4. Display
5. Exit

4

Name : Aisha

Account number : 12345678

Account Type : Savings

Current balance is 124456.0

Enter your choice :

1. Deposit
2. Calculate interest
3. Withdraw
4. Display
5. Exit
- 5.

✓ N
16/12/22

30/12/22

paperism
Date:

lab program 6

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 eases both father and son's age and throws an exception if son's age is \geq father's age.

```
import java.util.Scanner;  
class WrongAge extends Exception  
{}
```

```
String msg = new String();  
WrongAge()  
{ msg = "age error"; }  
WrongAge(String s)  
{ msg = s; }  
public String toString()  
{ return msg; }
```

```
class Father
```

```
{
```

```
int fage;  
Father() throws WrongAge  
{}
```

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

```
Scanner ss = new Scanner(System.in);  
if (fage < 0)
```

```
throws new WrongAge ("Invalid  
Father's age cannot be lesser than 0");
```

{

Father (int a) throws WrongAge.

{

fage = a;

{ if (fage < 0)

throw new WrongAge ("Input is invalid,
Father's age cannot be lesser than 0");

{

Father (int a) throws WrongAge

{

fage = a;

{ if (fage < 0)

throw new WrongAge ("Invalid Father's
age cannot be lesser than 0");

{

public String toString()

{

return "Father's age:" + fage;

{

✓ Class Son extends Father

{

int sage;

Son () throws WrongAge.

{

Scanner ss = new Scanner (System.in);

System.out.println ("Enter Son's age:");

int sage = ss.nextInt();

if (sage <= 0) { throw new WrongAge ("Invalid input. Son's age can be

{ if (sage >= fage)

{

ss. close();
throw new WrongAge(" Sons age cannot
be greater than father's
age");

}

{ ss. close();

}

public String toString()

{

return " Father's age : " + fage + "\n Son's
age : " + sage;

}

class Main

{

public static void main (String args[])

{

try

{

Father f = new Father();

Son s = new Son();

System.out.println (f);

System.out.println (s);

{

catch (WrongAge wa)

{ System.out.println (wa); }

catch (Exception e)

{ System.out.println (" Exception
encountered. Try again! "); }

{

{

Output:

Enter Father's age:

28

Enter son's age:

2

Enter father's age

2

equals to

Son's age cannot be greater than father's age.

Enter father's age:

23

Enter son's age:

30

Son's age cannot be greater than father's

Enter Father's age:

-12

Invalid input. Father's age cannot be lesser than 0

Enter Father's age

34

Enter Father's age

34

Enter son's age:

45

Son age can not be greater than father's age.

N
1/23

Enter Father's age:

32

Enter Father's age:

32

Enter son's age:

-12

Invalid input. Son's age cannot be lesser than 0.

Command Prompt

```
C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\father son program>java Main
Enter Father's age:
23
Enter Father's age:
23
Enter Son's age:
23
Sons age can not be greater than father's age

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\father son program>java Main
Enter Father's age:
-12
Invalid input. Father's age can not be lesser than 0

C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\father son program>java Main
Enter Father's age:
23
Enter Father's age:
30
Enter Son's age:
3
Father's age: 23
Father's age: 30
```

```
C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\father son program>java Main  
Enter Father's age:  
34  
Enter Father's age:  
34  
Enter Son's age:  
45  
Sons age can not be greater than father's age
```

```
C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\father son program>
```

Lab program 7

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 Point {  
    static void print(boolean flag)  
    {  
        if (flag)  
        {  
            System.out.println("BMS college  
of Engineering");  
        }  
        else  
        {  
            System.out.println("CSE");  
        }  
    }  
  
    class BMS extends Thread  
    {  
        public void run()  
        {  
            try {  
                for (int i = 0; i < 5; i++)  
                {  
                    Point.print(true);  
                    Thread.sleep(1000);  
                }  
            } catch (Exception e)  
            {  
                System.out.println("Error  
occurred");  
            }  
        }  
    }  
}
```

```
class CSE extends Thread  
{  
    public void run()  
    {  
        for (int i = 0; i < 10; i++)  
        {  
            System.out.print("CSE ");  
            Thread.sleep(2000);  
        }  
    }  
    catch (Exception e)  
    {  
        System.out.println("Error occurred");  
    }  
}  
  
class Main {  
    public static void main (String args[])  
    {  
        BMS thread1 = new BMS();  
        CSE thread2 = new CSE();  
  
        thread1.start();  
        thread2.start();  
    }  
}
```

Output.

BMS college of Engineering

CSE

CSE

CSE

CSE

CSE

paperism

Date: 6 Ja

BMS college of Engineering

CSE

CSE

CSE

CSE

CSE

BMS college of Engineering

BMS college of Engineering

BMS college of Engineering.

Dual
VLSI

(32MA mon + 16MBit 2MA)

(192) mon + 5 board 32A

(1) 16Kb + 16MBit

(1) 16Kb + 16MBit

```
C:\Windows\System32\cmd.e + - X Microsoft Windows [Version 10.0.22621.963] (c) Microsoft Corporation. All rights reserved. C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\8. thread>javac lab_program8.java C:\Users\Aisha Taffazul\Desktop\notes\3rd sem\java programs\8. thread>java Main BMS college of Engineering CSE CSE CSE CSE CSE CSE BMS college of Engineering CSE CSE CSE CSE CSE CSE BMS college of Engineering BMS college of Engineering |
```