

VAGHDEVI PRAVEEN T
1BM19CS175
'3D' BATCH-2

OOJ LAB PROGRAMS

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

WRITEUP:-

NAME: T.VAGHDEVI PRAVEEN
U.S.N: 1BM19CS175
SIGNATURE: Vaghdevi

URBAN EDGE

```
import java.util.Scanner;
public class Lab1 {
    public static int det(int a, int b, int c) {
        int d = b * b - 4 * a * c;
        return d;
    }
    public static void main (String [] args) {
        double r1, r2, real, imag;
        Scanner x = new Scanner (System.in);
        System.out.println ("Enter the a,b,c values:");
        int a = x.nextInt();
        int b = x.nextInt();
        int c = x.nextInt();
        int d = det (a, b, c);
        if (d == 0) {
            r1 = -b + Math.sqrt (d * 1.0);
            r2 = -b - Math.sqrt (d * 1.0);
            System.out.println ("the roots are real and
equal :" + r1 + "," + r2);
        }
        if (d > 0) {
            r1 = -b + Math.sqrt (d * 1.0);
            r2 = -b - Math.sqrt (d * 1.0);
            System.out.println ("the roots are real but
not equal :" + r1 + "," + r2);
        }
        if (d < 0)
            real = -b;
            imag = d;
    }
}
```

The logo for Urban Edge, featuring the word "URBAN" stacked above "EDGE" in a bold, black, sans-serif font. The "E" in "EDGE" has a diagonal slash through it. To the left of the text is a stylized graphic element consisting of several curved, overlapping shapes in shades of blue and grey.

```
System.out.println("the roots are imaginary:");
    + (real) + " + (" + (+ 1.0 * imag) + 'i'), " + (real) +
    + (" + (- 1.0 * imag) + " i"));
```

3

3

OpenCT (v1.3) connects with both sites

$(\alpha_1 \cdot \alpha_2)^{1212} = \alpha_1^{1212} \cdot \alpha_2^{1212}$

3. (3) The following is true

~~(C) last name - first name
or (C) last name - first name~~

(C) 1977 DUNN & CO.

(9.1 b b) $\frac{1}{2} \pi^2 - 115 \cdot \frac{1}{2} \pi^2 = -114 \cdot \frac{1}{2} \pi^2$

~~and which results in~~

Digitized by srujanika@gmail.com

(e.g. F. "The Big Bang")

7. *On the other hand*, the *other* side of the *same* coin.

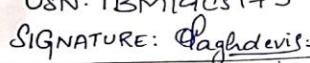
OUTPUT:-

```
E:\>cd java
E:\Java>javac lab1.java
E:\Java>java lab1
enter the a,b,c values:1
2
3
the roots are imaginary:-2.0+(-8.0i),-2.0+(8.0i)
E:\Java>java lab1
enter the a,b,c values:2
2
2
the roots are imaginary:-2.0+(-12.0i),-2.0+(12.0i)
E:\Java>lab1
'lab1' is not recognized as an internal or external command,
operable program or batch file.
E:\Java>java lab1
enter the a,b,c values:2
3
5
the roots are imaginary:-3.0+(-31.0i),-3.0+(31.0i)
```

LAB-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.

WRITEUP:-

NAME : T. VAGHDEUL PRAVEEN
USN: IBM19CS175
SIGNATURE:  DOJ
LAB - 2
URBAN EDGE

```
import java.util.*;  
public class Student {  
    public static void main (String args []) {  
        int usn, n;  
        int sgpa, sum = 0, meum = 0;  
        int [] credits;  
        int [] marks;  
        String name;  
        Scanner sc = new Scanner (System.in);  
        System.out.println ("Enter the number of subjects");  
        n = sc.nextInt();  
        credits = new int [n];  
        marks = new int [n];  
  
        System.out.println ("Enter the name of the student");  
        name = sc.nextLine();  
        System.out.println ("Enter the usn of the student");  
        usn = sc.nextInt();  
        for (int i = 0; i < n; i++)  
        {  
            System.out.println ("Enter the Credits and marks of  
the subject " + (i + 1));  
            credits [i] = sc.nextInt();  
            marks [i] = sc.nextInt();  
        }  
  
        Student s1 = new Student ();  
        for (int x : credits) {  
            sum += x; }  
        for (int y : marks) {  
            meum += y; }  
        sgpa = sum / meum;
```

```
S1. accept (usn, credits, marks, name);  
Sgpa = S1. Find Sgpa (sum);  
S1. display (meum, Sgpa);  
}  
}  
}
```

```
Class Student {  
    int usn;  
    int [] credits = new int [100];  
    int [] marks = new int [100];  
    String name;  
    void accept (int usn, int [] credits, int [] marks, String name)  
    {  
        this.usn = usn;  
        this.credits = credits;  
        this.marks = marks;  
        this.name = name;  
    }  
    void display (double tot, int Sgpa)  
    {  
        System.out.println ("Name: " + name + " USN: " + usn +  
            "\n Total Marks: " + tot + "\n Sgpa: " + Sgpa);  
    }  
    int Find Sgpa (int vsum)  
    {  
        int Sgpa, int sum=0, v=0;  
        for (int x : marks)  
        {  
            sum += (credits[v++]*x);  
        }  
        Sgpa = sum / (vsum * 10);  
        return (Sgpa);  
    }  
}
```

OUTPUT:-

```
E:\>cd java
E:\Java>java Student
Enter the Number Of Subjects
2
Enter the name of the Student
VAGHDEVI
Enter the USN of The Student
175
Enter the Credits And Marks Of The Subject1
3
100
Enter the Credits And Marks Of The Subject2
3
100
Name: VAGHDEVIUSN: 175
Total Marks: 200.0
Sgpa: 10
E:\Java>
```

LAB-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.

WRITEUP:-

NAME: T.VAGHDEVI PRAVEEN
USN: IBM19CS175
SIGNATURE: Vaghdevi OOT LAB-3 URBAN EDGE

```
import java.util.*;
class book {
    String name;
    String author;
    double price;
    int num_pages;
    public book() {
        name = "java the complete reference";
        author = "herbert Schildt";
        price = 1000.0;
        num_pages = 1882;
    }
    public book(String name, String author, double price,
               int num_pages) {
        this.name = name;
        this.author = author;
        this.price = price;
        this.num_pages = num_pages;
    }
    void setDetails() {
        Scanner x = new Scanner (System.in);
        name = x.nextLine();
        author = x.nextLine();
        price = x.nextDouble();
        num_pages = x.nextInt();
    }
    public String toString() {
        return (name + ", " + author + ", " + price + ", " + num_pages);
    }
}
```

```
public class Main {  
    public static void main (String [] args) {  
        int n = 0;  
        Scanner x = new Scanner (System.in);  
        System.out.println ("enter the value for n:");  
        n = x.nextInt();  
        book [] b = new book [n];  
        System.out.println ("enter the details of book(name,"  
                           "author,price,num pages"));  
        for (int i=0; i<n; i++) {  
            b[i] = new book();  
            System.out.println ("details of "+(i+1)+" book:");  
            b[i].setDetails();  
        }  
        System.out.println ("the details of the book are:");  
        for (int i=0; i<n; i++) {  
            System.out.println ((i+1)+" book:");  
            System.out.println (b[i]);  
        }  
    }  
}
```

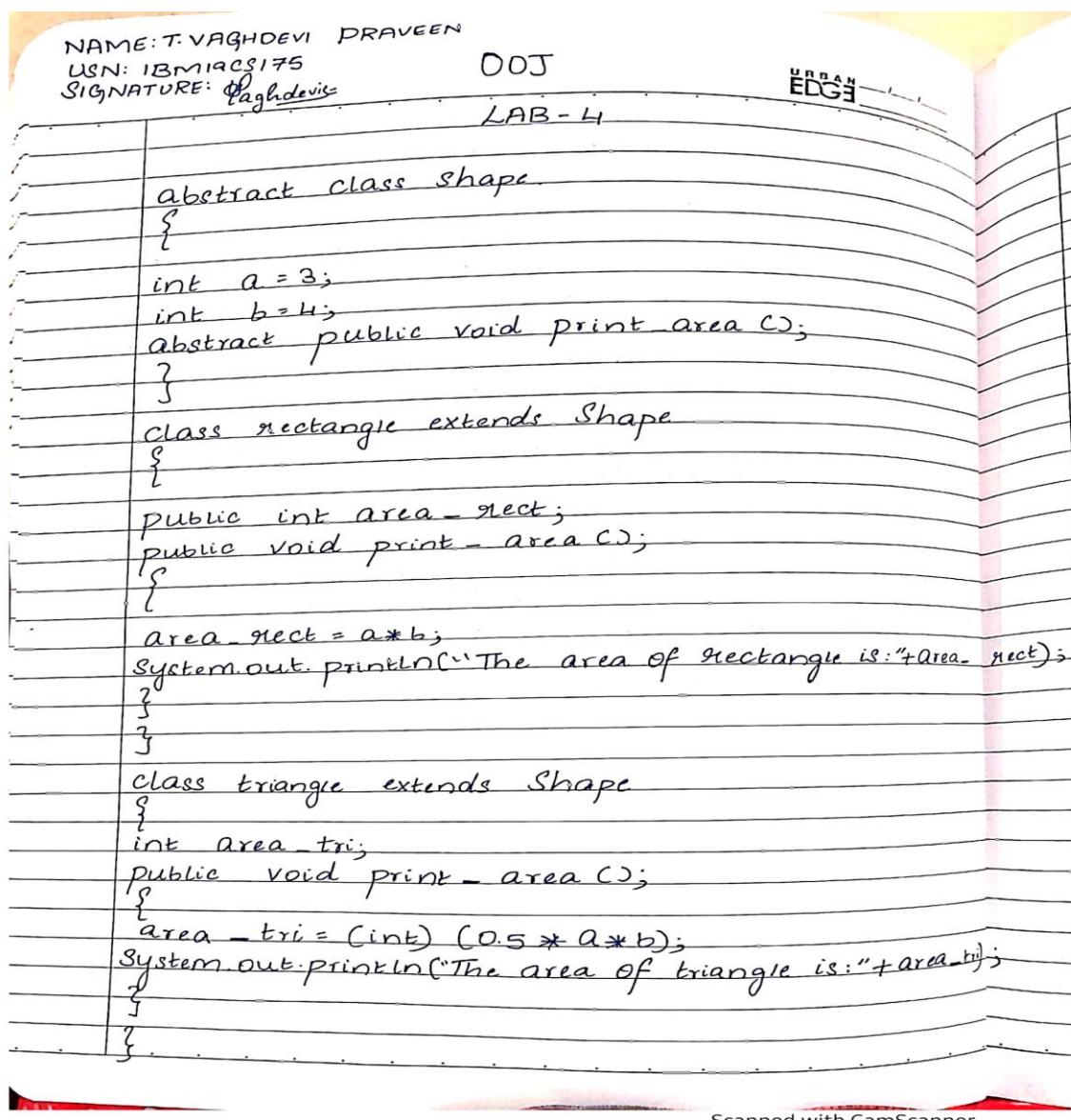
OUTPUT:-

```
E:\>cd java
E:\Java>javac Main.java
E:\Java>java Main
enter the value for n:2
enter the details of book(name,author,price,num_pages)
details of 1 book:
java the complete reference
herbert schildt
1000
1882
details of 2 book:
introduction to java programming
y daniel liang
999
1446
the details of the books are:
1 book:
java the complete reference,herbert schildt,1000.0,1882
2 book:
introduction to java programming,y daniel liang,999.0,1446
```

LAB-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.

WRITEUP:-



```
class circle extends shape
{
    int area_circle;
    public void print_area()
    {
        area_circle = (int) (3.14 * a*a);
        System.out.println ("The area of circle is:" + area_
circle);
    }
}
class abs
{
    public static void main (String [] args)
    {
        rectangle rec = new rectangle ();
        rec.print_area ();
        triangle tri = new triangle ();
        tri.print_area ();
        circle cir = new circle ();
        cir.print_area ();
    }
}
```

OUTPUT:-

```
E:\>cd java  
E:\Java>javac abs.java  
E:\Java>java abs  
The area of rectangle is: 12  
The area of triangle is: 6  
The area of circle is: 28  
E:\Java>
```

LAB-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: 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.

WRITEUP:-

NAME: T.VAGHDEVI PRAVEEN
USN: IBM1ACCS17S
SIGNATURE: Taghdevi

DOJ
LAB-5

URBAN
EDGE

```
import java.util.*;  
class account {  
    String customer_name;  
    int account_number;  
    String account_type;  
}  
  
class curr_acct extends account {  
    Scanner x = new Scanner (System.in);  
    double temp = 0.0;  
    double amount = 0.0;  
    double fine = 0.0;  
    double minimum_amount = 1000.0;  
}  
  
void get_details () {  
    customer_name = x.nextLine();  
    account_number = x.nextInt();  
}  
  
void deposit () {  
    System.out.println ("Enter the deposit amount:");  
    temp = x.nextDouble();  
    amount += temp;  
}  
  
void showbalance () {  
    if (amount >= min_amount) {  
        System.out.println ("Balance is: " + amount);  
    }  
}
```

```
else {  
    fine = (Amount * 1.0 * 10) / 100;  
    amount -= fine;  
    System.out.println ("the fine imposed : " + fine);  
    System.out.println ("Balance is : " + amount);  
}  
}  
}  
void withdrawal () {  
    System.out.println ("Enter the withdrawal amount");  
    temp = x.nextDouble ();  
    amount -= temp;  
}  
class sav_account extends account {  
    Scanner x = new Scanner (System.in);  
    double temp = 0.0;  
    double amount = 0.0;  
    double interest = 0.0;  
    void get details () {  
        customer_name = x.nextLine ();  
        account_number = x.nextInt ();  
    }  
    void showbalance () {  
        System.out.println ("Balance is : " + amount);  
    }  
    void withdrawal () {  
    }
```

```
System.out.println("Enter the withdrawal amount:");
temp = x.nextDouble();
amount = temp;
}
}
void interest() {
interest = (amount * 1.0 * 3) / 100;
amount += interest;
System.out.println("interest added: " + interest);
System.out.println("Balance is: " + amount);
}
}
public class Main {
public static void main (String [] args) {
int Opt=0;
String type=null;
Scanner x = new Scanner (System.in);
System.out.println("Welcome to the bank Service");
System.out.println("Enter the type of account
(Curr_Acc/ Sav_Acc)");
type = x.nextLine();
if (type.equals ("curr_acc")) {
curr_acc a = new curr_acc ();
System.out.println ("Enter the customer name,
account number:");
a.get details();
while (true) {
System.out.println ("Press 1: Accept deposit and
update the balance");
System.out.println ("press 2: Display the balance");
}
}
}
```

```
System.out.println("press 3: withdrawal and update  
the balance");  
System.out.println("Enter option:");  
opt = x.nextLine();  
switch (opt) {  
    case 1: a.deposit();  
    a.showbalance();  
    break;  
    case 2: a.showbalance();  
    break;  
    case 3: a.withdrawal();  
    a.showbalance();  
    break;  
}  
}  
}  
  
if (type.equals ("sav-acct")){  
    sav-acct a = new sav-acct();  
    System.out.println("Enter the customer name,  
    account-number:");  
    a.getdetails();  
    while (true){  
        System.out.println("press 1: Accept details and  
        update the balance");  
        System.out.println("press 2: Display the amount");  
        System.out.println("press 3: Compute and deposit  
        interest");  
        System.out.println("press 4: withdrawal and  
        update the balance");  
        System.out.println("Enter Option:");  
        opt = x.nextLine();
```

Switch (opt) {

Case 1: a. deposit ()

a. Showbalance (3)

break;

case 2: a. Show balance C;

break;

Case 3: a. interest C;

a. Show balance();

break

Case 4: a withdrawal (C);

a. Show balance (C);

break;

}{

2 3

2

$\{$

1

3

卷之三

卷之三

1

— 1 —

100-100

Chlorophytum

1000

卷之三

卷之三

10 of 10

• • •

OUTPUT:-

```
Enter the type of account (curr_acct/sav_acct)
sav_acct
Enter the customer_name,account_number:
VAGHDEVI PRAVEEN T
1234567890
press 1 : Accept deposit and update the balance
press 2 : Display the balance
press 3 : Compute and deposit interest
press 4 : Withdrawal and update the balance
Enter option :
1
Enter the deposit amount : 200000
Balance is : 200000.0
press 1 : Accept deposit and update the balance
press 2 : Display the balance
press 3 : Compute and deposit interest
press 4 : Withdrawal and update the balance
Enter option :
3
interest added : 6000.0
Balance is : 206000.0
Balance is : 206000.0
press 1 : Accept deposit and update the balance
press 2 : Display the balance
press 3 : Compute and deposit interest
press 4 : Withdrawal and update the balance
Enter option :
4
Enter the withdrawal amount : 100000
Balance is : 106000.0
press 1 : Accept deposit and update the balance
press 2 : Display the balance
press 3 : Compute and deposit interest
press 4 : Withdrawal and update the balance
Enter option :
```

LAB 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.

WRITEUP:-

USN: 1BM19CS175
SIGNATURE: Vaghdevi PRAVEEN
ODJ
LAB - 6
URBAN
EDGE

```
Package CIE;
import java.util.*;
public class personal
{
    public String name;
    public int Sem;
    public String USN;

    public void read()
    {
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter the Semester");
        Sem = sc.nextInt();
        System.out.println ("Enter the USN");
        USN = sc.next();
    }

    public void display()
    {
        System.out.println ("Student details:");
        System.out.println ("Name: " + name + "\n"
                           "USN: " + USN + "\n"
                           "Sem: " + Sem);
    }
}

Package SEE;
import java.util.*;
import CIE.*;
public class external extends personal
```

Scanned with CamScanner

```
{  
    public double SEE[];  
    public void get C)  
{  
  
    SEE = new double [5];  
    Scanner SC=new  
    Scanner (System.in);  
    for (int i=0 ;i<5 ;i++)  
{  
        System.out.println ("SEE mark for course" +(i+1)+"  
        SEE[i]=SC.nextDouble();  
    }  
}  
  
}  
  
package CIE;  
import java.util.*;  
public class internals extends personal  
{  
    public double CIE[];  
    public void accept C)  
{  
  
    CIE = new double [5];  
    Scanner Sc = new Scanner (System.in);  
    for (int i=0 ;i<5 ;i++)  
{  
        System.out.println ("CIE mark for course"  
        +(i+1)+"");  
        CIE[i]= Sc.nextDouble();  
    }  
}
```

{

```
import CIE.*;
import SEE.*;
import java.util.*;
```

Class Main

{

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

{

```
Scanner sx = new Scanner (System.in);
System.out.println ("Enter the number Of Students");
int n = sx.nextInt();
```

```
CIE.internals in [] = new CIE.internals [n];
```

```
SEE.internals en [] = new SEE.Externals [n];
```

```
int i, j;
```

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

{

```
System.out.println ("Student") + (i+1));
in [i] = new CIE.internals ();
```

```
en [i] = new SEE.Externals ();
in [i].read ();
```

```
System.out.print ln ("CIE MARKS:");
in[i].accept ();
```

```
System.out.print ln ("SEE MARKS:");
en[i].get ();
```

```
System.out.print ln ();
in[i].display ();
```

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

```
system.out.println("Total marks for course" + (j+1) + ":"  
+ cin[i].cic[j] + cin[i].sec[j]/2));  
}  
}  
}
```

OUTPUT:-

```
gnome-terminal[1]:~$ python3 student.py
Enter the number of students
2
Student 1
Enter the name
vaghdevi
Enter the semester
3
Enter the USN
1BM19CS175
CIE MARKS:
CIE mark for course 1 :
34
CIE mark for course 2 :
40
CIE mark for course 3 :
39
CIE mark for course 4 :
38
CIE mark for course 5 :
36
SEE MARKS:
SEE mark for course 1 :
99
SEE mark for course 2 :
100
SEE mark for course 3 :
98
SEE mark for course 4 :
97
SEE mark for course 5 :
96

Student details:
Name: vaghdevi
USN: 1BM19CS175
Sem: 3
Total Marks for course 1: 83.5
Total Marks for course 2: 90.0
Total Marks for course 3: 88.0
Total Marks for course 4: 86.5
Total Marks for course 5: 84.0
Student 2
Enter the name
YASHWANTH
Enter the semester
3
Enter the USN
1BM19CS187
CIE MARKS:
CIE mark for course 1 :
```

Command Prompt

```
1BM19CS187
CIE MARKS:
CIE mark for course 1 :
39
CIE mark for course 2 :
40
CIE mark for course 3 :
40
CIE mark for course 4 :
40
CIE mark for course 5 :
40
SEE MARKS:
SEE mark for course 1 :
100
SEE mark for course 2 :
99
SEE mark for course 3 :
98
SEE mark for course 4 :
100
SEE mark for course 5 :
98

Student details:
Name: YASHWANTH
USN: 1BM19CS187
Sem: 3
Total Marks for course 1: 89.0
Total Marks for course 2: 89.5
Total Marks for course 3: 89.0
Total Marks for course 4: 90.0
Total Marks for course 5: 89.0
```

E:\Java\packages>

LAB-7

Write a program to demonstrate generics with multiple object parameters.

WRITEUP:-

Name : T. VAGHDEVI PRAVEEN
U&N : IBM19CS175
SIGNATURE: Vaghdevi DOJ URBAN EDGE

LAB-7

```
// A Simple generic class with the two parameters : T and V
class TwoGen <T,V> {
    T ob1;
    V ob2;
    // Pass the constructor a reference to an Object of
    // type T and an Object of type V.
    TwoGen(T o1, V o2) {
        ob1 = o1;
        ob2 = o2;
    }
}

// Show types of T and V
void showTypes() {
    System.out.println("Type of T is " + ob1.getClass().getName());
    System.out.println("Type of V is " + ob2.getClass().getName());
}

T getOb1() {
    return ob1;
}

V getOb2() {
    return ob2;
}

// Demonstrate TwoGen.
class SimpGen {
    public static void main(String args[]) {
        TwoGen<Integer, String> tgObj =
            new TwoGen<Integer, String>(88, "Generics");
    }
}
```

// Show the types.
tgObj.showTypes();

and v

// Obtain the show values.

```
int v = tgObj.getObject();
System.out.println("value:" + v);
String str = tgObj.getString();
System.out.println("value:" + str);
}
```

());

OUTPUT:-

The screenshot shows a Windows desktop environment. At the top, there are several open browser tabs: "Vodafone Mobile Connect", "OOJ-LAB/LAB 8 at main · YASHV", "vaghdevipraveen/OOJ-LAB", and "Online Java Compiler - online ed". The active tab is "OnlineGDB beta" from onlinegdb.com/online.java_compiler. The page displays a Java code editor with the following content:

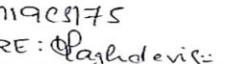
```
Type of T is java.lang.Integer  
Type of V is java.lang.String  
value: 88  
value: Generics  
...Program finished with exit code 0  
Press ENTER to exit console.
```

The browser's address bar shows the URL https://onlinegdb.com/online.java_compiler. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray indicating the date and time as 12/21/2020 at 7:38 PM.

LAB-8

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 cases both father and son's age and throws an exception if son's age is >=father's age.

WRITEUP:-

NAME: T.VAGHDEVI PRAVEEN
USN: 1BM19CS175
SIGNATURE:  OOS
T.Vaghdevi Praveen LAB-8
URBAN EDGE

```
import java.util.*;  
class WrongAge extends Exception {  
    int f, s;  
    WrongAge (int fage, int sage) {  
        f = fage;  
        s = sage;  
    }  
    public String toString() {  
        return "Please enter the correct age as father's  
        age can't be less than or equal to the son's age";  
    }  
}  
  
class NegativeAge extends Exception {  
    int x;  
    NegativeAge (int fage) {  
        x = fage;  
    }  
    public String toString() {  
        return "Age can't be a negative value";  
    }  
}  
  
class Father {  
    int fage;  
    Scanner in = new Scanner (System.in);  
    Father () throws NegativeAge {  
        fage = in.nextInt();  
        if (fage <= 0) {  
            throw new NegativeAge();  
        }  
    }  
}
```

```
{  
    system.out.println("Enter the father's age");  
    fage = in.nextInt();  
    if(fage < 0) {  
        throw new NegativeAge(fage);  
    } } }
```

```
class Son extends Father  
{
```

```
    int sage;  
    Scanner in = new Scanner(System.in);  
    Son() throws NegativeAge, WrongAge {  
        super();
```

```
        system.out.println("Enter the son's age:");  
        Sage = in.nextInt();  
        if(sage < 0)
```

```
            throw new NegativeAge(Sage);  
        } }
```

```
        if(sage >= fage)  
            throw new WrongAge(fage, sage);  
    } }
```

```
class AgeDisplay {  
    public static void main(String args[]) {  
        try {  
            Son s = new Son();  
        } . . . . .
```

```
catch (NegativeAge n) {  
    System.out.println ("Exception : " + n);  
}  
  
catch (WrongAge w) {  
    System.out.println ("Exception : " + w);  
}  
}
```

OUTPUT:-

```
Enter the father's age:  
20  
Enter the son's age :  
50  
Exception:Please enter the correct ages as father's age can't be less than or equal to the son's age.  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

```
Enter the father's age:  
50  
Enter the son's age :  
20  
  
...Program finished with exit code 0  
Press ENTER to exit console.
```

LAB-9

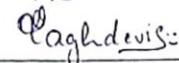
Write a program which creates two threads, one thread displaying "BMS College of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.

WRITEUP:-

NAME: T.VAGHDEVI PRAVEEN
USN: IBM19CS17S
SIGNATURE: Vaghdevi
ODJ
LAB - 9
URBAN EDGE

```
class thread1 implements Runnable
{
    Thread t;
    thread 1()
    {
        t = new Thread (this, "thread1");
        t.start ();
    }
    public void run()
    {
        for (;;)
        {
            try
            {
                System.out.println ("BMS College of Engineering");
                Thread.sleep (1000);
            }
            catch (InterruptedException i.e.)
            {
                System.out.println ("Interrupted");
            }
        }
    }
}

class thread2 implements Runnable
{
    Thread t2;
    thread 2()
    {
    }
```

NAME: TVAGHDEVI PRAVEEN
USN: 1BMA1acs175
SIGNATURE: 

OOP

URBAN
EDGE

NAN
UST
SIG

```
t2 = new Thread (this, "thread 2");
t2.start ();
}

public void run()
{
    for (i++)
    {
        try
        {
            System.out.println ("CSE");
            Thread.sleep (2000);
        }
        catch (InterruptedException i.e.)
        {
            System.out.println ("Interrupted");
        }
    }
}
```

```
class threadmain
{
    public static void main (String args[])
}
```

```
System.out.println ("Enter Control + C to stop");
thread1 t1 = new thread1 ();
thread2 t2 = new thread2 ();
```

```
}
```

```
}
```

OUTPUT:-