

Java Program

Q1) Java Program to print an Integer

(Entered by me)

```

import java.util.*;
public class HelloWorld {
    public static void main(String args[]) {
        Scanner input = new Scanner(System.in);
        System.out.println("Enter an integer");
        int number = input.nextInt();
        System.out.println("ADITHYA PILLAI(IBM22CS013)");
        System.out.println("The integer is " + number);
    }
}
    
```

Output :-

Enter an integer

Adithya Pillai (IBM22CS013)

The integer is

Algorithm

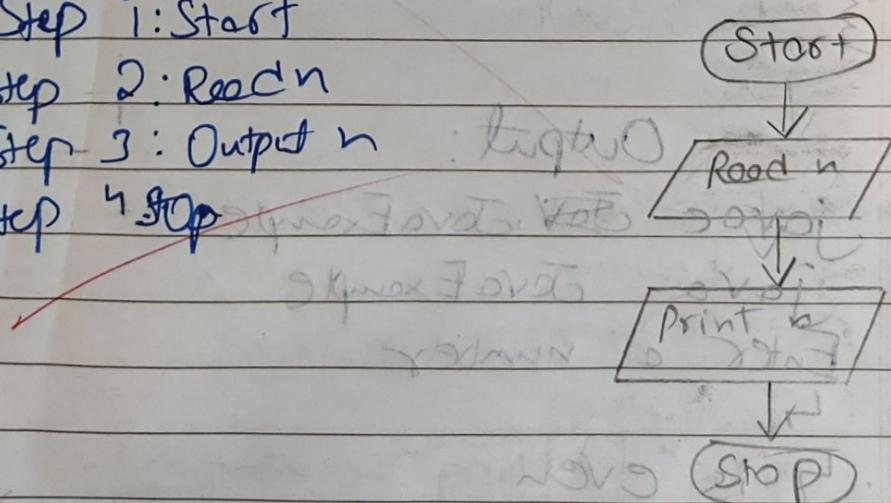
Step 1: Start

Step 2: Read n

Step 3: Output n

Step 4: Stop

Flowchart:



```
C:\Users\bmsce\Desktop\Adithya-013>javac HelloWorld.java  
C:\Users\bmsce\Desktop\Adithya-013>java HelloWorld  
Enter an integer  
5  
ADITHYA PILLAI(1BM22CS013  
The integer is5
```

2) Java Program to check whether a number is even or odd.

{
}

```
import java.util.*;  
public class JavaExample  
{  
    public static void main(String args[])  
    {  
        int num;  
        System.out.print("Enter an Integer");  
        Scanner input=new Scanner(System.in);  
        num=input.nextInt();  
        if (num % 2 == 0)  
            System.out.println(num+" is even");  
        else  
            System.out.println(num+" is odd");  
        System.out.println("ADITHYA PILLAI(IBM22CS013)");  
    }  
}
```

~~Output:~~

javac ~~Java~~.JavaExample

java JavaExample

Enter a number

{

{ is even

ADITHYA PILLAI(IBM22CS013)

Algorithm :

Task

Flowchart :

Step 1 : Start

Step 2 : Read num

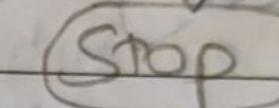
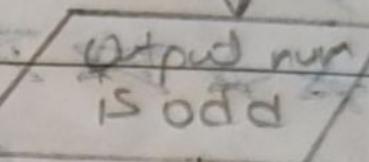
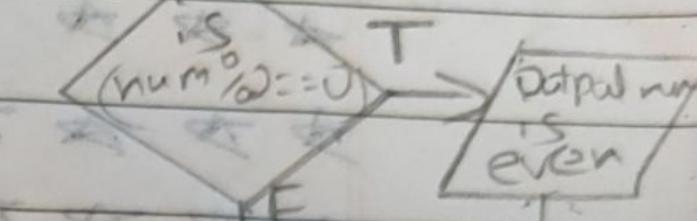
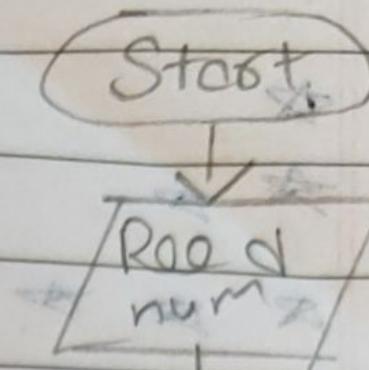
Step 3 : if ($\text{num} \bmod 2 = 0$)

Output num is even

else

Output num is odd

Step 4 : Stop



```
Enter a number
```

```
4
```

```
4 is even
```

```
ADITHYA PILLAI(1BM22CS013)
```

```
C:\Users\bmsce\Desktop\Adithya-013>java JavaExample
```

```
Enter a number
```

```
5
```

```
5 is odd
```

```
ADITHYA PILLAI(1BM22CS013)
```

3) Java Program to print Right Triangle Star Pattern

```
import java.util.*;
public class JavaExample2 {
    public static void main(String args[]) {
        int row, column, numberOfRows=8;
        for (row=0; row<=numberOfRows; row++) {
            for (column=0; column<=row; column++) {
                System.out.print("* ");
            }
            System.out.println();
        }
        System.out.println("Aditya Pillai");
    }
}
```

Output

```

* 
* *
* * *
* * * *
* * * * *
* * * * * *
* * * * * * *
* * * * * * * *

```

Date: 19/12
 Main board: C got 2
 (056 down run) pi: 8 got 2
 Nov 2nd lecture
 eals
 bba 2nd year lecture
 9/12/12 got 2.

Ardiye Pillai

Algorithm:

Step 1: Start

Step 2: for let number of Rows = 8
 number of Rows not (8)

Step 3: for row=0 to 8 do
 not 8 + 2

for column=0 to ^{row} do

Output: *

end for

End for

Step 4: Stop

Flowchart

number of Rows = 8

if row = 8 then stop

for column = 0 to row step by 1

Output "

Column

row

Stop

```
C:\Users\bmsce\Desktop\Adithya-013>javac JavaExample2.java
```

```
C:\Users\bmsce\Desktop\Adithya-013>java JavaExample2
```

Adithya Pillai(1BM22CS013)

4)

Java Program to Find Quotient and Remainder

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

```
public class JavaExample3 {
```

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

```
        int num1=15, num2=2;
```

```
        int quotient = num1/num2;
```

```
        int remainder = num1 % num2;
```

```
        System.out.println("Quotient is " + quotient);
```

```
        System.out.println("Remainder is " + remainder);
```

```
        System.out.println("Aditya Pillai");
```

```
    }
```

```
}
```

Output:

Quotient is 7

Remainder is 1

Aditya Pillai

Algorithm :-

Step 1 : Start

Step 2 : $n_1 = 15 \quad n_2 = 2$

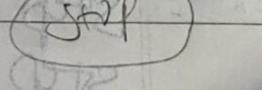
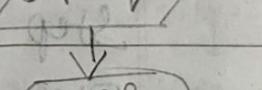
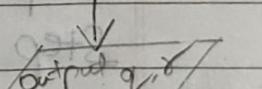
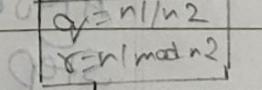
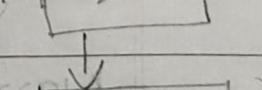
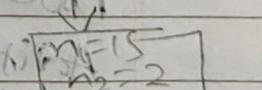
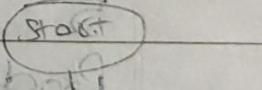
Step 3 : $q = n_1 / n_2$

Step 4 : $r = n_1 \bmod n_2$

Step 5 : Output q

Step 6 : Output r

Step 7 : Stop



```
C:\Users\bmsce\Desktop\Adithya-013>java JavaExample3
Quotient is7
Remainder is1
Adithya Pillai(1BM22CS013)
```

5 Java Program to Multiply Two Numbers

```
import java.util.*;  
public class Demo  
{  
    public static void main(String args[]) {  
        Scanner input = new Scanner(System.in);  
        System.out.println("Enter first number");  
        int num1 = input.nextInt();  
        System.out.println("Enter second number");  
        int num2 = input.nextInt();  
        int product = num1 * num2;  
        System.out.println("Product is " + product);  
        System.out.println("Aditya Pillai");  
    }  
}
```

Output :
Enter first number
4 7

Enter second number

5

Product is 20

Aditya Pillai

Algorithm :-

Step 1 : Start

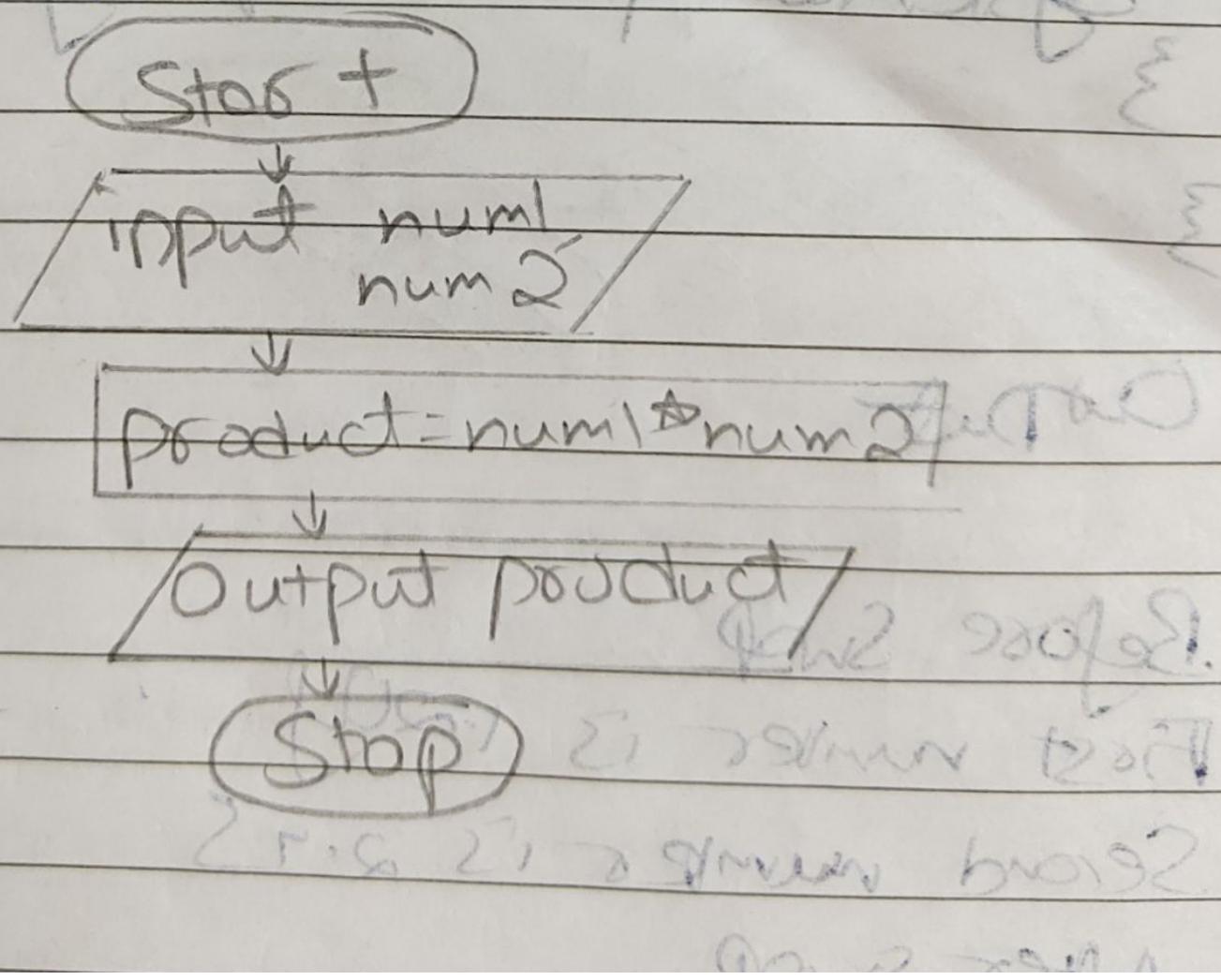
Step 2 : input num1, num2

Step 3 : product = num1 * num2

Step 4 : Output &

Step 5 : Stop

Flowchart



```
C:\Users\bmsce\Desktop\Adithya-013>javac Demo.java
```

```
C:\Users\bmsce\Desktop\Adithya-013>java Demo
```

```
Enter first number
```

```
4
```

```
Enter second number
```

```
5
```

```
Product is20
```

```
Adithya Pillai(1BN22CS013)
```

6. Swap two numbers using temporary variable

```
import java.util.*;
public class SwapNumbers {
    public static void main(String args[]) {
        float first = 1.2f, second = 2.4f;
        System.out.println("Before Swap");
        System.out.println("First number is " + first);
        System.out.println("Second number is " + second);
        float temporary = first;
        first = second;
        second = temporary;
        System.out.println("After Swap");
        System.out.println("First number is " + first);
        System.out.println("Second number is " + second);
```

System.out.println("Aditya Pillai");
3
3

Output

Before Swap

First number is 1.20

Second number is 2.45

After Swap

First number is 2.45

Second number is 1.20

Aditya Pillai

Algorithm:-

Step 1: Start

Step 2: ~~first=~~ first=1.20, second=2.45

second=2.45

Step 3: output first, second

Step 4: temporary=first

Step 5: first=second

Step 6: second=temporary

Step 7: output first, second

Step 8: Stop

(Nimish) 89 min Flowchart = tqni annos?

(START)

first = 120

second = 245

if first > second then, tqni = 0

Output first, second, tqni = 0

(C) if first > second, tqni = 1

temporary = first

first = second

second = temporary

fo.0 = 6

fo.0 = 182

fo.0 = 50

Output first, second

(STOP)

```
C:\Users\bmsce\Desktop\Adithya-013>javac SwapNumbers.java
```

```
C:\Users\bmsce\Desktop\Adithya-013>java SwapNumbers
```

Before Swap

First number is 1.2

Second number is 2.45

After Swap

First number is 2.45

Second number is 1.2

Adithya Pillai(1BM22CS013)

22/12/23

Quadratic eqns

Ex) Program 7 : WAP that prints all real
soln to the s. quadratic eqn
 $ax^2 + bx + c = 0$. 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.*;  
import java.util.Scanner;  
public class Quadratic  
{  
    public static void main(String args[]){  
        float a, b, c, d, s, r1, r2;  
        System.out.println("Enter a, b, c:");
```

Scanner input = new Scanner (System.in);

~~a = input.nextDouble();
b = input.nextDouble();
c = input.nextDouble();~~

a = input.nextDouble();

b = input.nextDouble();

c = input.nextDouble();

d = a * a - b * b;

~~d = 0.0f;~~

~~delta = 0.0f;~~

~~delta = 0.0f;~~

if (a == 0 || b == 0 || c == 0)

System.out.println("Invalid Input");

~~else~~ {
 if (d > 0) {
 double r1 = (-b + Math.sqrt(d)) / (2 * a);
 double r2 = (-b - Math.sqrt(d)) / (2 * a);
 System.out.println("Roots are real and
 distinct");
 System.out.println("r1 = " + r1 + " r2 = " + r2);
 }
 else if (d < 0) {
 System.out.println("Roots are imaginary");
 }
}

~~else if (d == 0) {
 double r = -b / (2 * a);
 System.out.println("Roots are real and
 equal");
 System.out.println("r = " + r);
}~~

~~r1 = (-b + Math.sqrt(d)) / (2 * a);
r2 = (-b - Math.sqrt(d)) / (2 * a);~~

~~System.out.println("Roots are real and
distinct");
System.out.println("r1 = " + r1 + " r2 = " + r2);~~

~~System.out.println("Roots are real and
equal");
System.out.println("r = " + r);~~

~~else if (d < 0) {
 System.out.println("Roots are imaginary");
}~~

~~System.out.println("Roots are complex");
}~~

~~CloseScanner();~~

$\{ \quad \} \quad \{ \quad \}$
 $x_1 = -b / (2a);$
 $x_2 = x_1;$
 System.out.println("Roots are equal");
 $x_1 = "x_1 + x_2";$
 $x_2 = "x_1 + x_2";$
 $\}$
 $\}$
 $\} \quad \rightarrow \text{Algorithm:}$

Step 1: Start

Step 2: Read a, b, c

Step 3: if ($a=0 \text{ || } b=0 \text{ || } c=0$)

Print Invalid input

go to step 8

end of if

Step 4: $d = b^2 - 4ac$

Step 5: if ($d > 0$)

Print roots are real and distinct

$$x_1 = -b + \sqrt{d} / (2a)$$

$$x_2 = -b - \sqrt{d} / (2a)$$

Print x_1, x_2

go to step 8

end of if

Step 6: if ($d < 0$)

Print roots are imaginary

go to step 8

end of if

Step 7: If ($d = 0$)

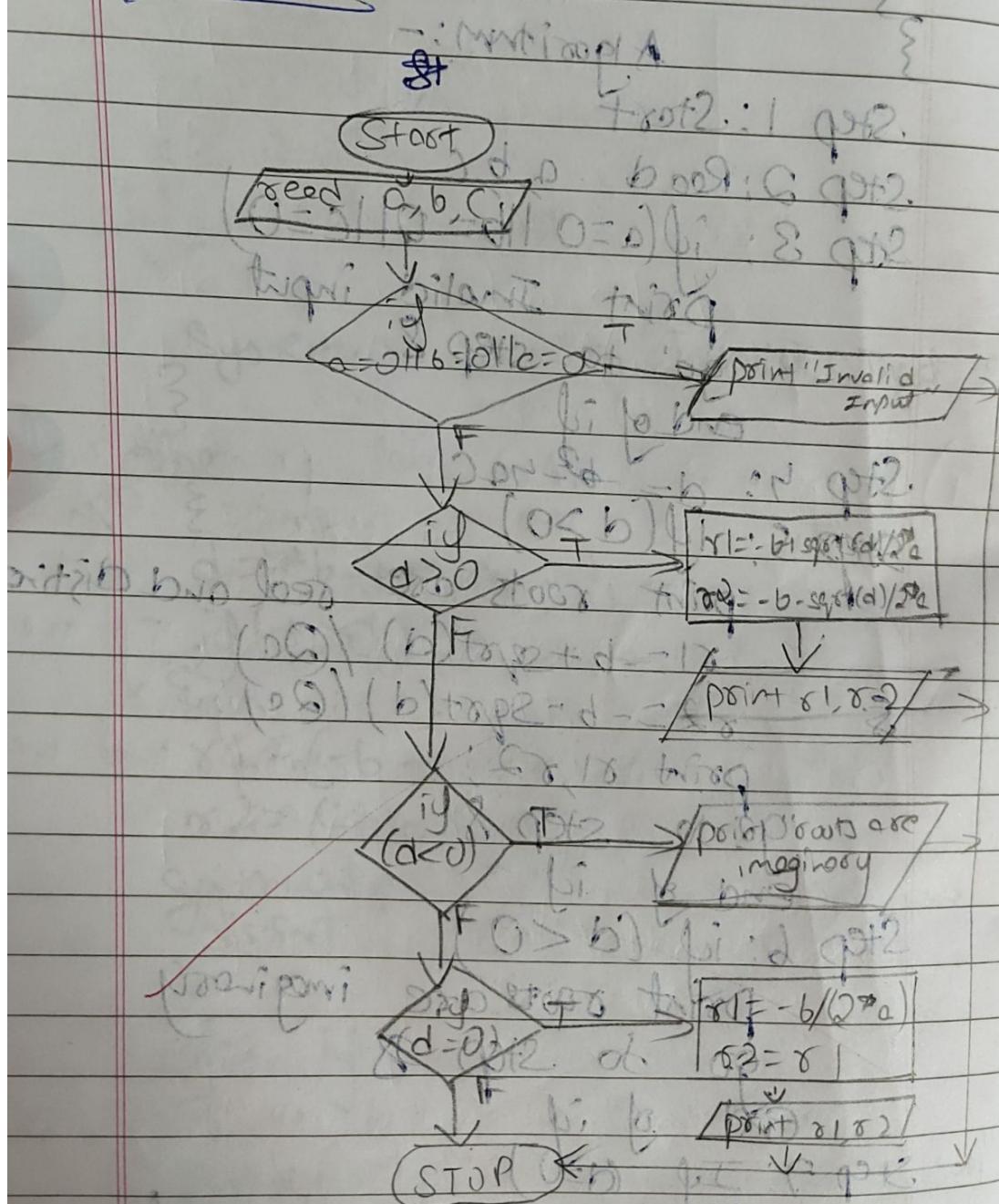
Print roots are equal

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

$$\alpha_2 = \gamma_1$$

Point α_1, α_2 (α_1, α_2) $| d = 18$
 goto step 8 $| 18 = 508$
 end of if
 skip 8 : stop.

Flowchart :-



Output:

Enter values of a, b and c

1

2

3

Roots are imaginary

Adimya Pillai (1BM22CS013)

Enter (a, b and c)

12.5

25

12.5 = [Tarsus] [Tibia] tri

Roots are equal. $\sqrt{1} = 1$. $\sqrt{2} = 1.414$

Enter values a, b and c

29

172

46

Roots are real and distinct $\sqrt{2} = -5.650305$

Enter a, b and c

O:("molt") althing.two.molt2

S6("molt") althing.two.molt2

O("molt") althing.two.molt2

Invalid input

```
C:\Users\Admin\Desktop\CS-013>java Quadratic
```

```
Enter values of a, b, c:
```

```
1
```

```
2
```

```
3
```

```
Roots are imaginary
```

```
Adithya Pillai(1BM22CS013)
```

```
C:\Users\Admin\Desktop\CS-013>java Quadratic
```

```
Enter values of a, b, c:
```

```
12.5
```

```
25
```

```
12.5
```

```
Roots are real and equal
```

```
R1= -1.0 R2= -1.0
```

```
Adithya Pillai(1BM22CS013)
```

```
C:\Users\Admin\Desktop\CS-013>java Quadratic
```

```
Enter values of a, b, c:
```

```
0
```

```
56
```

```
0
```

```
Invalid Input
```

```
Adithya Pillai(1BM22CS013)
```

Q WAP to create a class Student
with members USN, name, array
credits and an array marks.

Include methods to accept and
display details and a method
to calculate CGPA of a student.

import java.util.*;

```
public class Student {  
    String usn, name;  
    int credits[], marks[];  
    public Student(String usn, name, int credits[],  
        int marks[]);
```

{ }
this.usn=usn;

this.name=name;

this.marks=marks;

this.credits=credits;

}

public void display()

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

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

System.out.println("Credits & marks: ");

for(int i=0; i<credits.length; i++)

{
 System.out.println("Subject "+(i+1)+"

"Marks : "+marks[i]+" in credits:

$t \leftarrow t + \text{credits}[i]$;

{
 t9PF colgp(marks[i]) * credits[i];

$$S_{\text{QPA}} = \left(\frac{\tau_{\text{QPL}}}{\tau_{\text{C}}} \right);$$

```
System.out.println("SGPA: " + student.getSGPA());
```

public double calOp(int m) remove;

$i \neq m \geq 90$

){ initTxn.beginTransaction = error ??

return 10.0; // taking two messages

$\text{C}_1\text{H}_2\text{S}_2\text{N}_2\text{F}_4\text{O}_4 = \text{NCI}$

else if ($m \geq 80$) : "Juggling two rings"

() $\text{Tr}[\text{TxSN, sig}] = \alpha$ fair

return: 9.0; 9.0 = 216927 fm

Lafni was a woman.

else if ($m >= 70$)

$\lim_{n \rightarrow \infty} \delta_n = 0$ if $\{x_n\}$ is

```

else if(m >= 60)
{
    return 7.0;
}
else if(m >= 50)
{
    return 6.0;
}
else
{
    return 0.0;
}

public static void main(String args[])
{
    Scanner input = new Scanner(System.in);
    System.out.println("Enter name of student");
    Student student = new Student();
    student.name = input.nextLine();
    System.out.println("Enter USN");
    student.usn = input.next();
    System.out.println("Enter number of subjects");
    int n = input.nextInt();
    float credits = new float[n];
    marks = new int[n];
    System.out.println("Enter credits & marks");
    for(int i=0; i < credits.length; i++)
    {
        System.out.println("For Subject " + (i+1));
    }
}

```

credits[i]

v = input.nextInt();

marks[i] = input.nextInt();

}

Student student = new Student(roll_no,
name,
credits, marks);

student.display();

student.sgp();

}

Algorithm

Step 1: Start

Step 2: input roll_no, name, n

Step 3: for int i=0 to credit.length
 credits[i] = input.nextInt();
 marks[i] = input.nextInt();
end of for

Step 4: for i=0 to credits.length; i++
 tc = tc + credits[i];

 tgpt = colgp(marks[i]) * credits[i];
end of for

Step 5: - sgp = (tgpt / tc)

point sgp

Step 6: output roll_no, name

Step 7: output marks, credits, point sgp

Step 8: Stop

colgp Function :-

Step 1: start

Step 2:

if ($m \geq 10$)
point return 10.0

else if ($m \geq 80$)
return 9.0

else if ($m \geq 70$)
return 8.0

else if ($m \geq 60$)
return 7.0

else if ($m \geq 50$)
return 6.0

else
return 0.0

Step 3: Stop

($\text{float} t = \text{input}$ value: 8 qf2,
($\text{float} f = \text{input}$ value: 5 qf2)

($\text{float} m = \text{input}$ value: 4 qf2)

if b == 1

float t (**Start** of $t = ?$ qf2: N)

$t = t + 10$ qf2: S

if $b == 0$ qf2: S

input w, n, m qf2: S

$n = ?$ qf2: S

$t = t + 10$ qf2: S

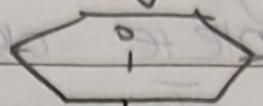
$y = (t - 1) * p$ qf2: S

$y = y + 10$ qf2: S

~~int i = 0; while (i < credits.length; i++)~~

$$tgcP = tgc + credits[i];$$

$$tgP = colgp(marks[i]) * credits[i]$$



$$SgPo = tgP/tgc$$

Output: marks, credits, Sg.Pc

STOP

29/12/23

```
C:\Users\Admin\Desktop\CS-013>java Student
Enter name of student
Adithya Pillai
Enter USN
1BM22CS013
Enter Credits and marks
For subject1
3
95
For subject2
4
90
For subject3
3
85
For subject4
1
98
For subject5
4
85
SGPA of student is:9.533333333333333
Name of student:Adithya Pillai
USN:1BM22CS013
For subject1
Credits:3
Marks:95
For subject2
Credits:4
Marks:90
For subject3
Credits:3
Marks:85
For subject4
Credits:1
Marks:98
For subject5
Credits:4
Marks:85
```

Q

Create a class Book which contains your members: name, author, price, num_page. 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.*;  
class Book {  
    String name, author;  
    int price, num_pages;
```

```
    Book (String name, String author, int price, int num_pages)  
    {
```

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

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

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

```
        this.num_pages = num_pages;
```

```
}
```

```
    void getVal ()
```

```
{}
```

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

~~```
System.out.println ("Enter name, author, price,
 num of pages");
```~~

```
name = input.nextLine();
```

```
author = input.nextLine();
```

```
price = input.nextInt();
```

```
numPage = input.nextInt();
}
void toString()
{
 System.out.println("Name of book " + name);
 System.out.println("Author " + Author);
 System.out.println("Price " + price);
 System.out.println("Number of page " + numPage);
}
public static void main()
{
 Scanner input = new Scanner(System.in)
 String name
 Book book[] = new Book[n];
 System.out.println("Enter no of books");
 int Scanner input = new Scanner(System.in);
 int n = input.nextInt();
 Book book[] = new Book[n];
 for (int i = 0; i < n; i++)
 {
 book[i] = new Book();
 book[i].getVal();
 }
 System.out.println("The Book Library");
 for (i = 0; i < n; i++)
 {
 book[i].toString();
 }
}
```

Algorithm: ~~Algorithm: begin = input number of books~~

Output :-

Enter the number of books

Enter name, author, price, number of pages

Author:

Name:

Price:

Number of Pages:

50

The Book LIBRARY starts.

Name of book: Adhi

Name of Author: Meena

Price of Book: 45

Number of Pages: 50

Algorithm:

Step 1: START

Step 2: Input n = [ ] void book

Step 3: Create a Book Array,

Step 4: for i=0 to n-1 do

    input name, author, price, number of pages

    For each book object created.

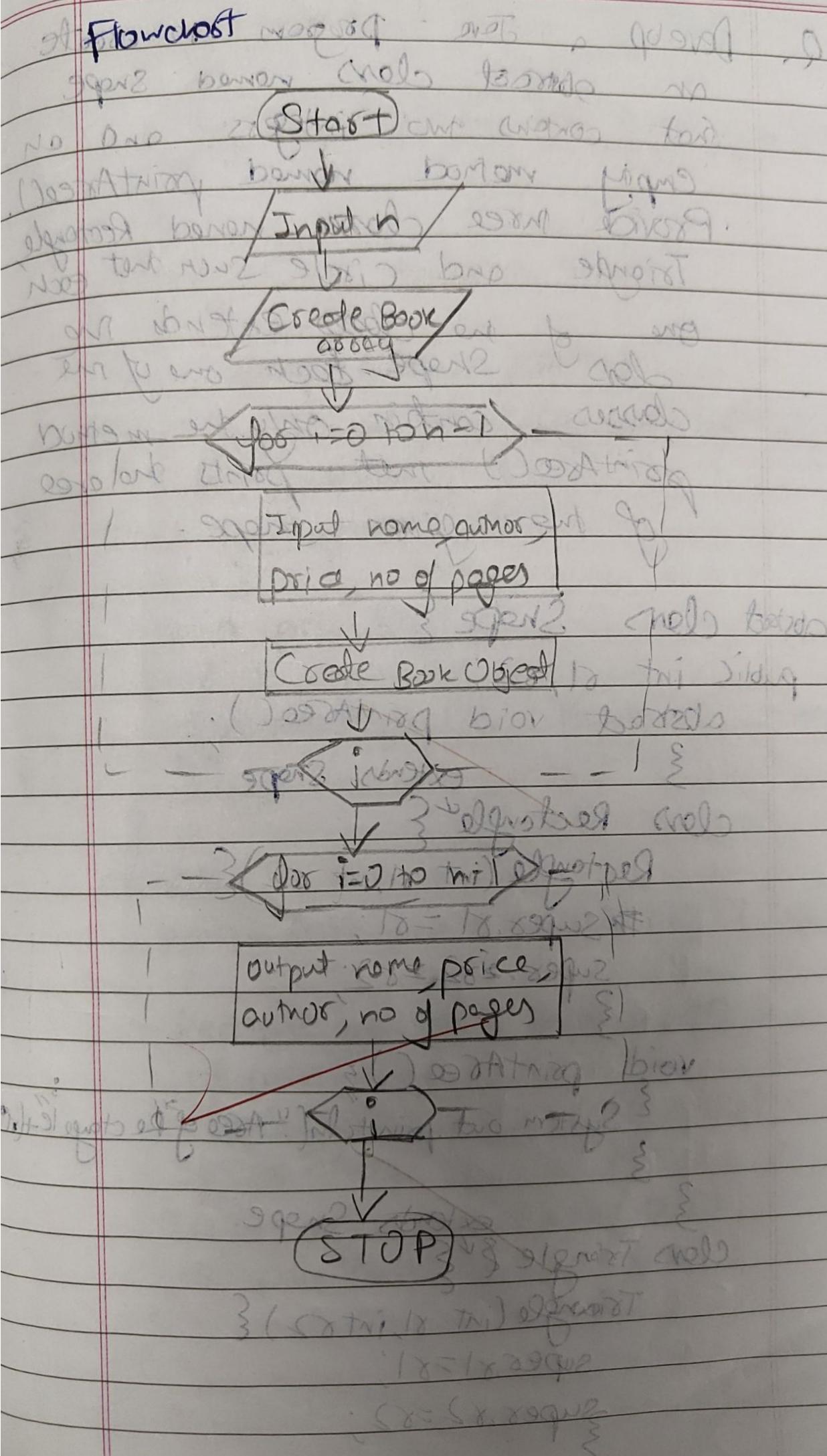
end for

Step 5: for i=0 to n-1 do

    output name, author, price, no. of

    pages, end for

Step 6: Stop



```
C:\Users\Admin\Desktop\CS-013>javac Book.java
```

```
C:\Users\Admin\Desktop\CS-013>java Book
```

```
Enter the number of books
```

```
1
```

```
Enter name
```

```
Adhi
```

```
Enter Author
```

```
Hello
```

```
Enter Price
```

```
45
```

```
Enter Number of pages
```

```
50
```

```
THE BOOK LIBRARY
```

```
Name of book: Adhi
```

```
Name of author: Hello
```

```
Price of book: 45
```

```
number of pages of book: 50
```

```
1BM22CS013 ADITHYA PILLAI
```

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 contains only the method printArea() that prints the area of the given shape.

abstract class Shape {

    public int x1, x2;

    abstract void printArea();

}

class Rectangle {

    Rectangle(int x1, int x2) {

        super.x1 = x1;

        super.x2 = x2;

}

    void printArea() {

        System.out.println("Area of Rectangle is " + (x2 - x1) \* (x2 - x1));

}

extends Shape

class Triangle {

    Triangle(int x1, int x2) {

        super.x1 = x1;

        super.x2 = x2;

```
void printArea() {
 System.out.println("Area of Triangle "(0.5 * 10 * 10))
}
```

```
class Circle extends Shape
```

```
{ circle }
```

```
Shape(int x) { }
```

~~System.out.println("Area of Circle "+(3.14 \* 10 \* 10))~~~~super.x = x;~~~~return super.x = x;~~~~return super.x = x;~~~~void printArea() { }~~~~System.out.println("Area of Circle "+(3.14 \* 61 \* 61))~~~~super.x = 61;~~~~: 904.2 : P QR~~

```
class Toy {
```

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

```
 Rectangle ob = new Rectangle(10, 2);
```

~~ob.printArea();~~~~Triangle ob1 = new Triangle(10, 2);~~~~ob1.printArea();~~~~Circle ob2 = new Circle(10);~~~~ob2.printArea();~~~~6840.2 : 1962~~~~, now 0.0 : C QR~~

Output:-

Area of Rectangle : 20

Area of Triangle : 25.0

Area of Circle : 153.0

### Algorithm

Step 1 : Start

Step 2 : Create class Rectangle, Triangle, Circle  
extending class Shape.

Step 3 : Call Rectangle(10, 2) constructor

Step 4 : Call pointArea() for Rectangle

Step 5 : Call Triangle(10, 2) constructor

Step 6 : Call pointArea() for Triangle

Step 7 : Call Circle(10) constructor

Step 8 : Call pointArea() for Circle

Step 9 : Stop.

pointArea(10) for Rectangle

Step 1 : ~~Step~~ Point

~~Step 2 : return~~

pointArea() for Triangle

Step 1 : ~~Print~~

~~Step 2 : return~~

pointArea() for Circle

Step 1 : Print

~~Step 2 : return~~

# Flowchart

bottom of main window is yellow 10  
best out exception for now only

start

draw rectangle

rectangle create Rectangle

draw triangle create Triangle

draw circle create Circle

call rectangle(10,2)

end by construction

triangle, rectangle on lid until

call point A(0,0)

draw rectangle rectangle

circle with downward as fi br

call triangle(10,2) call rectangle

triangle and down to door. because

triangle | call point A(0,0) for triangle

2nd part between to right hand up

triangle triangle | call circle(10,2) construction

new screen or triangle again bring

triangle right call point A(0,0) for circle

is absent no necessary end point

final call end

Stop

```
C:\Users\Admin\Desktop\CS-013>javac Try.java
```

```
C:\Users\Admin\Desktop\CS-013>java Try
```

```
Area of Rectangle:20
```

```
Area of Triangle:10.0
```

```
Area of Circle:314.0
```

```
Adithya Pillai(1BM22CS013)
```

# LAB 4

Q1 Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings, other current. 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 min balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account no and type of account. From this derive class for current account and savings account 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 min balance, impose

penalty of necessary and update  
the balance.

```
import java.util.*;
class Account {
 String customer_name;
 int acc;
 double bal;
 Account(String name, int no, double bal)
```

```
 customer_name = name;
```

```
 acc = no;
```

```
 this.bal = bal;
```

```
 void deposit(double d)
 {
```

```
 bal = bal + d;
```

```
 System.out.println("Deposit made In
```

```
 Account deposit:" + bal);
```

```
}
```

```
void display()
```

```
{
```

```
 System.out.println("Balance:" + bal);
```

```
}
```

```
class Currentacc extends Account
```

```
{
```

```
 double minbal=1500.0, boolean check;
```

```
 Currentacc(String name, int no, double bal, boolean
 chq,)
```

```
Super(name, no, bal);
check = chq;
}

void verify()
{
 if (check)
 System.out.println("Checkbook Provided");
 else
 System.out.println("Checkbook will be Provided
 in mail");
}

if (bal >= 1500)
 System.out.println("No Tax Levied since
 balance greater than
 1500");
else
 System.out.println("Tax of 10% levied
 bal = bal - (0.1 * bal); on balance");
 System.out.println("Current balance: " + bal);

}

void withdraw(double amt)
{
 this.verify();
 bal = bal - amt;
 System.out.println("Withdrawn successfully
 Current balance: " + bal);
}
```

Date \_\_\_\_\_  
Page \_\_\_\_\_

```

class Savingsacc extends Account {
 double rate; int time;
 Savingsacc (String name, int no, double bal,
 double rate, int time)
 {
 super(name, no, bal);
 this.rate = rate; this.time = time;
 }
 void Cint()
 {
 bal = bal + (bal * Math.pow((1 + rate), time) - bal);
 }
 void withdraw (double amt)
 {
 bal = bal - amt;
 System.out.println("Withdrawn successfully");
 System.out.println("Current balance : " + bal);
 }
}

class Bank {
 public static void main (String [] args)
 {
 System.out.println("Enter the number of customers");
 int n =
 Scanner input = new Scanner (System.in);
 int n = input.nextInt();
 for (i=0; i<n; i++)
 {
 Currentacc ca = new Currentacc [n];
 Savingsacc sa = new Savingsacc [n];
 }
 }
}

```

```
for(i=0; i<n; i++)
{
 // code
 System.out.println("For customer " + (i+1));
 System.out.print("Enter name ");
 String n=input.next();
 System.out.print("For savings: ");
 System.out.print("Enter account no
balance, rate, time");
```

```
int acc = input.nextInt();
double bal = input.nextDouble();
double rate = input.nextDouble();
int time = input.nextInt();
sa[i] = new Savingsacc(n, acc, bal, rate, time);
System.out.println("For current:");
System.out.print("Enter accountno, balance");
acc = input.nextInt();
bal = input.nextDouble();
```

Boolean chq = true;

ca[i] = new Currentacc(n, acc, bal, chq);

sa[i].deposit();

sa[i].withdraw();

sa[i].do[i].withdraw();

}

(n, acc, bal, chq)

sa[i].do[i].deposit();

sa[i].do[i].withdraw();

sa[i].do[i].deposit();

Output:

Enter the number of customers

For customers: 18/12/2017

Enter name: John Doe, Jr.

Adri peacock will be

Food savings :-

Enter account no, balance rate time

~~145 New Bedford April 1898~~

12000. to add branching;

10 pages no cost right

Review 17th Feb 1998

For currents:  $v_{\text{left}} = v_{\text{right}}$

17 *b* *schistosomiasis* *by* *elbowage*

12000 Enter withdrawal: 1000

withdrawn successfully from savings

Current between 110 LSO

~~We withdraw successfully from closed~~

~~Current~~ balance : ₹ 11900

~~1910.124~~

```
C:\Users\Admin\Desktop\CS-013\Java>javac Bank.java
```

```
C:\Users\Admin\Desktop\CS-013\Java>java Bank
```

```
Enter the number of customers
```

```
1
```

```
FOR CUSTOMER 1
```

```
Enter name
```

```
Adhi
```

```
For savings:
```

```
Enter account number, balance, rate of interest, time
```

```
145
```

```
12000
```

```
10
```

```
2
```

```
Deposit success. Balance is: 24000.0
```

```
Compound interest
```

```
New Balance: 29040.00000000004
```

```
For current:
```

```
Enter account number, balance, presence of check (Enter 1 if check present)
```

```
146
```

```
12000
```

```
1
```

```
Deposit success. Balance is: 24000.0
```

```
For savings account:
```

```
Enter withdrawal amount
```

```
12000
```

```
Remaining Balance is: 17040.00000000004
```

```
For current account:
```

```
Enter withdrawal amount
```

```
1200
```

```
Withdrawal success. Remaining Balance: 22800.0
```

```
Adithya Pillai(1BM22CS013)
```

LABS  
programs

Q. Create a package CIF which has two classes Student and Internals. The class Person has members like ~~name, sem,~~  
 The class Internals has ~~an array~~  
~~not stored~~ the internal marks ~~scored~~  
 in five courses of the current  
 semester of the student. Create  
~~another~~ ~~do~~ package SE which has  
 the class External which  
 is a derived class of Student.  
 This class has an array that  
 stores the SE marks scored  
 in five courses of one current  
 semester of the student.  
 Import the ~~int~~ package in  
 a file and declare time  
 and marks of n students in  
 all five courses.

Ans  
 package CIF;  
~~import java.util.\*;~~  
 public class Student

```

 public int sem,

 public String usn,

 public String name,

 public void accept()
 }
```

```
System.out.println("Enter WSN Name Sensor");
wsn = scan.nextLine();
name = scan.nextLine();
```

Scm := scannextInt();

3. 20. 76. 763 ~~available~~

~~poorly~~ ~~poorly~~ ~~poorly~~ ~~poorly~~ ~~poorly~~

public class functions  
{

public int[] new int[5];

package SEE; v. 1 mi. S.E.

import CJF.Student; // Line 172

Public class External extends Student

*[Handwritten signature]*

public int sum[] = new int[5];

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

impost. SEE Fig. 3. In this it is evident

import CIF;

public class FindMarks

public class {

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

~~public store voice over S-3~~

$\{ \text{at } 1, \text{ for } s = \text{sat}, \text{ in } [s] \}$

~~int ym[ ] = new int;~~

Scanner sc=new Scanner(  
System.in);  
System.out.print("Enter n:");

System.out.println("In file 1");

Initiation = scale next Sat ( ),

SEE: External St[] = new St.

CIE.Internal(s) & now, CIE.Internal(w);

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

{

s[i].now.SEE.External();

s[i]=now.CIE.Internal(c);

System.out.println("Enter details "+(i+1));

s[i].accept();

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

{

System.out.println("Enter int and sum  
of subject "+(j+1));

s[i].imf[j]=sc.nextInt();

s[i].sm[j]=sc.nextInt();

ymf[j]=s[i].imf[j]+s[i].sm[j];

}

System.out.println("Final marks of "+s[i]);

for (int k=0; k<5; k++)

{

System.out.println("Course "+(k+1)+" = "+ymf[k]);

{

20 30 40 50 60 70 80 90 100

Algorithm:

~~the CIE~~

Step 1: Create

Step 2: Create package CIE

Step 3: Create class student within

Step 4: Create class Internal within CIE

Step 5: Create package SEE

Step 6: Create class External within SEE  
which extends CIF Student.

Step 7: Create method accept() within  
class External which reads USN,  
name and sem.

Step 8: for i=0 to n-1 do

    call accept() for student[i];  
    go to 6

    input marks[i] for subject 'j' of student;  
    input smarks[i][j] for subject 'j' of student;  
    imarks = imarks[i] + smarks[i][j].

end for;

Output "Find marks of student"

for k=0 to k <= K do

    Output ("course+(k+1) = "+imarks[k])

End of for

Step 9: STOP

Output

Enter the number of students

1

Enter Details of student 1

Enter one student's USN, Name or Semester

13

Adi

1

Enter Internal Marks and Semester End Marks

15

60

subject 1

Under Interval Marks and SF Moon of Surface

١٦

60

## Enterin<sup>g</sup> Internal Markets and SE' markets of subject)

1A

(10)

Entered Interval Marks and SE. Marks of subjects

18

2

introduction 5

19

10

Find Max w of Ad; ~~Max~~

Course 1-25 bit log

Course Q-576 at Qin up.

Please see ~~Section~~ (1-2000) together.

Course 4-78

Course 5 = A

```
C:\Users\Admin\Desktop\CS-013\Java>javac FinalMarks.java
```

```
C:\Users\Admin\Desktop\CS-013\Java>java FinalMarks
```

```
Enter the number of students
```

```
1
```

```
Enter Details of student 1
```

```
Enter the student's USN,Name and Semester
```

```
13
```

```
Adi
```

```
1
```

```
Enter Internal Marks and Semester End Marks of subject1
```

```
15
```

```
60
```

```
Enter Internal Marks and Semester End Marks of subject2
```

```
16
```

```
60
```

```
Enter Internal Marks and Semester End Marks of subject3
```

```
17
```

```
60
```

```
Enter Internal Marks and Semester End Marks of subject4
```

```
18
```

```
60
```

```
Enter Internal Marks and Semester End Marks of subject5
```

```
19
```

```
60
```

```
Final marks of Adi
```

```
Course1=75
```

```
Course2=76
```

```
Course3=77
```

```
Course4=78
```

```
Course5=79
```

LAB 6

Q2. Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class "Son" which extends the base class.

In Father class, implement a constructor which takes the age and throws an exception WrongAge() when the input age < 0. In Son class, implement a constraint constructor that takes both father and son's age and throws an exception if son's age is > father's age.

```

class WrongAge extends Exception
{
 public WrongAge()
 {
 super("Age is incorrect");
 }
}

class Father extends Exception
{
 public int age;
}

class Son extends Father
{
 public int age;
}

```

if(men (int age) throws wrongAge {  
    if(j-age > 0) {  
        throw new wrongAge();  
    }  
    this.sage = age;  
}

class Son extends Father {  
    int sage;  
    @Son(int j-age, int sage) throws Input  
    {  
        super(j-age);  
        if(j-age <= 0 & sage <= 0) {  
            throw new wrongAge();  
        }  
        this.sage = sage;  
    }  
}

Exception Prog.  
public class {  
    public static void main(String[] args)  
    {  
        try {  
            Father f = new Father(40);  
        }  
    }

```
System.out.println("Farmer's age : " + age);
Son s = new Son(40, 56);
System.out.println("Son's age : " + s.age);
} catch (WrongAgeException e) {
 System.out.println(e.toString());
}
catch (InputException e) {
 System.out.println(e.toString());
}
}
```

### Algorithm

Step 1: Create class WrongAge and inherit from Exception.

Step 2: Create Base class Farmer and child class Son which extends Farmer.

Step 3: try,  
 Create Farmer object f.  
 Print Farmer's age;

Step 4 create new Son object s.  
Print Son's age

Step 5 catch,

WrongAge Exception and print Age is incorrect  
catch,

~~InputException and print Son's age~~

~~cannot be lesser than Farmer's~~

Step 5: Stop

Output :-

javac ExceptionProg.java

java ExceptionProg

input: son's age cannot be more than father's  
~~Age~~

```
C:\Users\bmsce\Desktop\Adithya-013>javac ExceptionProg.java
C:\Users\bmsce\Desktop\Adithya-013>java ExceptionProg
input: Sons age cannot be more then Fathers
Adithya Pillai
1BM22CS013
```

Q8 Write a program which create 2 threads where one is executed every 10 seconds & another for 2 second.

```
class One extends Thread {
 public void run() {
```

```
 int i = 0;
 while (i < 5) {
```

```
 i++;
 try {
```

System.out.println("BMS College of  
Engineering");

```
 Thread.sleep(10000);
```

```
} catch (Exception e) {
```

System.out.println("Exception caught");

```
}
```

```
{ }
```

```

class Two extends Thread {
 public void run() {
 int i=0;
 while(i<5) {
 i++;
 try {
 System.out.println("CSE");
 Thread.sleep(2000);
 } catch (Exception e) {
 System.out.println(e);
 }
 }
 }
}

```

### Thread Programs

```

public class {
 public static void main(String[] args) {
 One t1 = new One();
 Two t2 = new Two();
 t1.start();
 t2.start();
 }
}

```

### Algorithm:-

Step 1: Create ClassOne and ClassTwo which extends Thread.

Step 2: In Class One override run

method :-

while( $i < 5$ ),  $i++$   
~~try, System.out.println("BMS college")~~  
~~Thread.sleep(10000);~~  
~~catch Exception e and print it~~  
~~end while.~~

Step 3: In class TWO override Run :-  
while i < 5

    i++;

    try, point ("CSE");  
    Thread.sleep(2000)

Catch Exception e and print it  
end of while

Step 4: Create Threads t1 and t2  
from Thread class One and Two

Step 5: t1.start();

Step 6: t2.start();

Step 7: Stop

Output:-

javac ThreadProg.java

java ThreadProg

BMS College of Engineering

CSE

CSF

CSE

CSE

CSE

BMS College of Engineering

BMS College of Engineering

BMS College of Engineering

BMS College of Engineering

24  
16.02.24

```
C:\Users\bmsce\Desktop\Adithya-013>javac ThreadProg.java
```

```
C:\Users\bmsce\Desktop\Adithya-013>java ThreadProg
```

BMS College of Engineering

CSE

CSE

CSE

CSE

CSE

BMS College of Engineering

BMS College of Engineering

BMS College of Engineering

BMS College of Engineering

Q9). Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were zero, the program would throw an ArithmeticException. Display the exception in a message dialog box.

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

class SwingDemo {
 SwingDemo() {
 JFrame jfr = new JFrame("Divider App");
 jfr.setSize(275, 150);
 jfr.setLayout(new FlowLayout());
 jfr.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 }
}
```

JLabel jl1 = new JLabel("Enter no divisor and dividend"),

JTextField a1 = new JTextField(8);  
JTextField b1 = new JTextField(8);

```
JButton button = new JButton("Calculate");
JLabel err = new JLabel();
JLabel alab = new JLabel();
JLabel blab = new JLabel();
JLabel onslab = new JLabel();

jfr.add(err);
jfr.add(jlab);
jfr.add(ajt);
jfr.add(bjt);
jfr.add(button);
jfr.add(alab);
jfr.add(blab);
jfr.add(onslab);
```

```
ActionListener I = new ActionListener(){
 public void actionPerformed(ActionEvent evt){
 if(err.getText() != null)
 System.out.println("Action event from " + evt);
 }
};

ajt.addActionListener(I);
bjt.addActionListener(I);
button.addActionListener(new ActionListener()
{
```

```
 public void actionPerformed(ActionEvent evt)
 {
 if(err.getText() == null)
 err.setText(" ");
 }
});
```

```
try {
```

```
 int a = Integer.parseInt(tf1.getText());
 int b = Integer.parseInt(tf2.getText());
 int ans = a / b;
 alab.setText("In A=" + a);
 blab.setText("In B=" + b);
 anslab.setText("In Ans=" + ans);
```

```
}
```

```
catch (NumberFormatException e)
```

```
{
```

```
 alab.setText("");
 blab.setText("");
 anslab.setText("");
 err.setText("Enter only Integers");
```

```
}
```

```
catch (ArithmaticException e) {
```

```
 alab.setText("");
```

```
 blab.setText("");
```

```
 anslab.setText("");
```

```
 err.setText("B should be NON zero");
```

```
}
```

~~```
}
```~~~~```
});
```~~

```
jfr.setVisible(true);
```

```
public static void main(String args[]) {
 SwingUtilities.invokeLater(new Runnable
 {
```

```
 public void run() {
 new SwingDemo(); } }); }
```

Output

Aditya Pillai (1BM22CS013)

Divider App

Enter the divisor and dividend:

50      10  
Calculate      A=50, B=10 Ans=5

Explanation

javax.swing is a package which has the swing sub package used for integrating GUI elements.

java.awt.★ contains classes for creating GUI components.

java.awt.event.★ contains Event Listener interfaces and classes for handling events.

JFrame object is created which represents main window of application

g.jfr.setSize(200, 100) → is to set size of window 200 being width and 100 being height

`jfr.setLayout(new FlowLayout());`  
→ used to set flow layout to the JFrame object

`jfr.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);`  
→ sets default close operation

JLabel is the class used to create label objects.

JTextField is used to create textfield

JButton is used to create object of button

`jfr.add()` → is used to add objects into JFrame.

ActionListener class is used to create ActionListener objects which respond to a specific action.

~~lab.setText(" ")~~ is to set texts into a label.

~~lab.getText()~~ is to obtain texts from a label.

`jfr.setVisible(true);` - is to set JFrame object to be visible.

```
C:\Users\bmsce\Desktop\Adithya-013>javac SwingDemo.java
```

```
C:\Users\bmsce\Desktop\Adithya-013>java SwingDemo
Adithya Pillai(1BM22CS013)
```

