

## Program 1

Java Program to print an Integer.

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

```
public class HelloWorld {
```

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

```
        Scanner variable reader = object new Scanner (System.in)
```

```
        System.out.print ("Enter a number: ")
```

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

```
        System.out.println ("You entered: " + number);
```

```
    }
```

```
}
```

Output:

Enter a number: 2

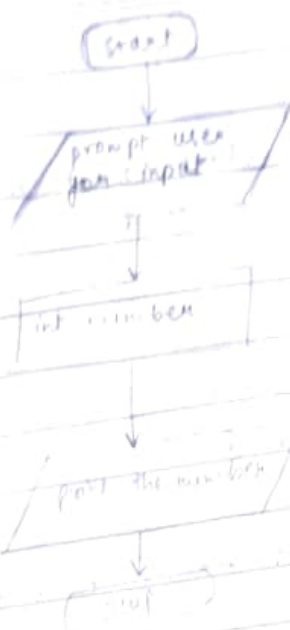
You entered: 2

## Flowchart

[ ] → Statement



• decision



## Algorithm

1. Start
2. Read number
3. Print the number
4. Stop

## Program 2

Java Program to check whether a Number is Even or Odd.

```
import java.util.Scanner;
public class javacexam {

    public static void main (String args[])
    {

        int num;
        System.out.print ("Enter an Integer number:")

        Scanner input = new Scanner (System.in);

        num = input.nextInt ();

        if (num % 2 == 0)

            System.out.println (num + " is an
                                even number.");

        else

            System.out.println (num + " is an odd
                                number.");

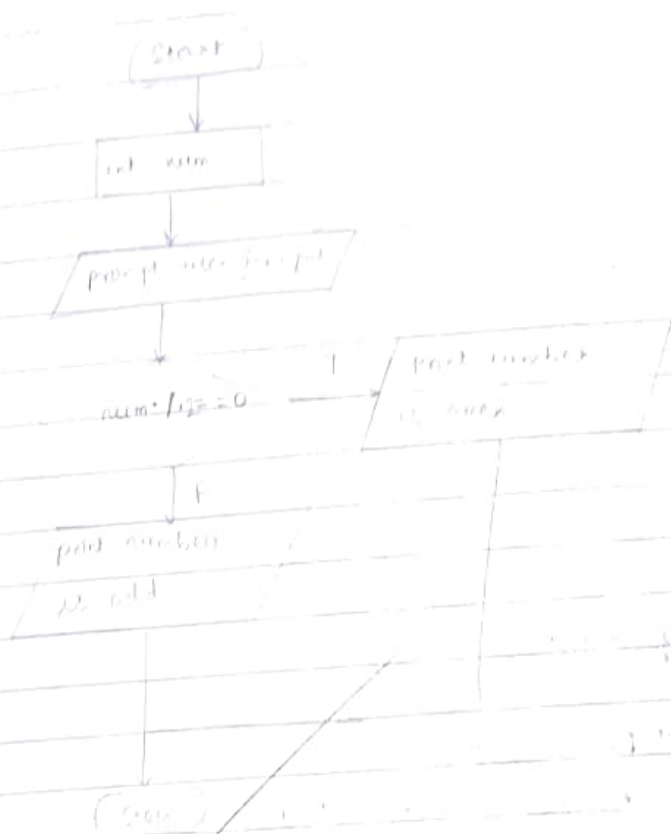
    }

}
```

Output:

Enter an integer number: 7  
7 is an odd number

Flowchart:

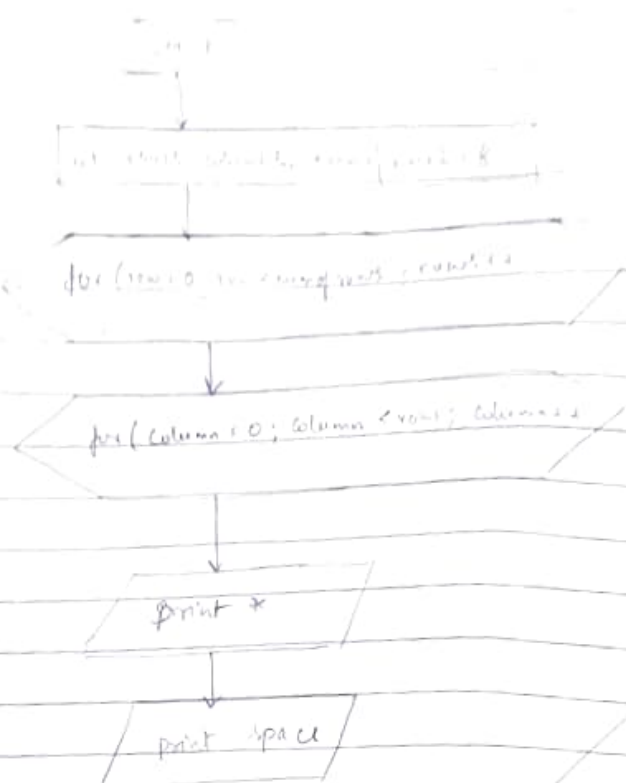


Algorithm:

1. Read start
2. Read num
3. Check for the condition, check for remainder = 0
4. If condition is true print that the number is even  
- else print that the number is odd
5. Stop

## Prog 3

### Flowchart



### Algorithm:

- ~~Step~~ 1. Start
2. Read row, column and num of rows = 8
3. Set row = 0, repeat steps 4 & 5 until row < num of rows
4. Set column = 0, repeat step 5 until column < row.
5. Print \*
6. Stop

Program 3

Java Program to Print Right Triangle  
Star Pattern.

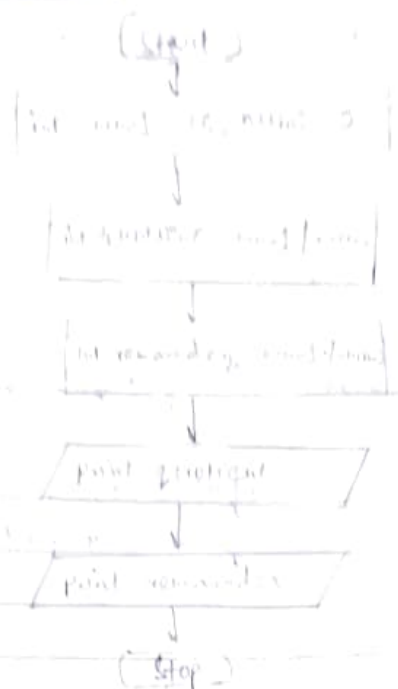
```
public class javacexam {  
    public static void main (String args[])  
    {  
        int row, column, numof rows = 8;  
        for (row = 0; row < numof rows; row++)  
        {  
            for (column = 0; column < row; column++)  
            {  
                System.out.print("*");  
            }  
            System.out.println();  
        }  
    }  
}
```

Output:

```
*  
**  
***  
****  
*****  
*****  
*****  
*****  
*****  
*****
```

## Prog 4

### Flowchart



### Algorithm:

1. Start
2. read num1 = 15 and num2 = 2
3. Read quotient as  $\text{num1} / \text{num2}$
4. Read remainder as  $\text{num1} \% \text{num2}$
5. Print Quotient
6. Print remainder
7. Stop

#### Program 4 :

Java Program to Find Quotient and Remainder.

```
public class javaexam {
```

```
    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);
```

```
    }
```

```
}
```

Output :

Quotient is : 7

Remainder is : 1



Prog 5

Flowchart



Algorithm:

1. Start
2. Read num1, num2
3. ~~Read~~ Read product to ~~be~~ be num1 \* num2.
4. Print the product
5. Stop

### Program 5:

Java Program to multiply two numbers

```
import java.util.Scanner;
public class javaexam {
    public static void main (String args[])
    {
        Scanner scan = new Scanner (System.in);
        System.out.print ("Enter first number:");
        int num1 = scan.nextInt();
        System.out.print ("Enter second number:");
        int num2 = scan.nextInt();
        scan.close();
        int product = num1 * num2;
        System.out.println ("Output: " + product);
    }
}
```

Output:

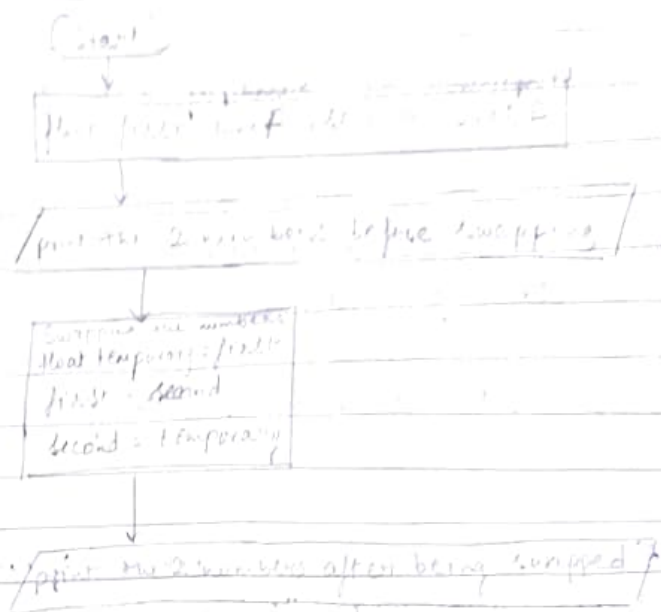
Enter first number: 3

Enter second number: 2

output: 6.

## Program 6:

### Flowchart



### Algorithm:

1. Start

2. Read First and Second

3. Print the numbers before swapping

4. Swap the numbers by taking an extra variable

5. read temp = first

first = second

second = temp

5. Print the numbers after swapping

6. Stop.

### Program 6.

Swap two numbers using temporary variable.

```
import java.util.Scanner;  
public class javacexam {
```

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

```
    {  
        float first = 1.20f, second = 2.45f
```

```
        System.out.println ("-- Before swap --");
```

```
        System.out.println ("First number = " + first);
```

```
        System.out.println ("Second number = " + second);
```

```
        float temporary = first;
```

```
        first = second;
```

```
        second = temporary;
```

```
        System.out.println ("-- After swap --");
```

```
        System.out.println ("First number = " + first);
```

```
        System.out.println ("Second number = " + second);
```

```
    }
```

```
}
```

Output:

-- Before swap --

First number = 1.2

Second number = 2.45

-- After swap --

First number = 2.45

Second number = 1.2

## Lab. Program - 1

- \* ① Develop a Java Program that print all real solutions to the quadratic equation  $ax^2 + bx + c = 0$ .  
Read in  $a, b, c$  and use the quadratic formula. If the discriminate  $b^2 - 4ac$  is negative, display a message stating that there are no real solutions.

```
import java.util.Scanner;
import static java.lang.Math.sqrt;
import static java.lang.Math.abs;

public class Squad {
    public static void main (String[] args) {

        Scanner in = new Scanner (System.in);

        System.out.println ("Enter coefficient:");

        int a = in.nextInt();
        int b = in.nextInt();
        int c = in.nextInt();

        if (a == 0) {

            System.out.println ("Invalid input");
        }

        else {
            int d = b*b - 4*a*c;
            if (d > 0) {
                System.out.println ("Roots are real")
            }
        }
    }
}
```

```

float r1 = (float) (-b + sqrt(d)) / (2*a);
float r2 = (float) (-b - sqrt(d)) / (2*a);
System.out.println(r1);
System.out.println(r2);
}

```

```

else if (d < 0) {

```

```

    System.out.println("Roots are imaginary,
    There are no real solutions");

```

```

float r1 = (float) -b / (2*a);
float r2 = (float) sqrt(abs(d)) / (2*a);
System.out.println(r1 + " + i" + r2);
System.out.println(r1 + " - i" + r2);
}

```

```

else {

```

```

    System.out.println("Roots are equal");
    float r = (float) -b / (2*a);
    System.out.println(r);
}
}
}

```

Output:-

4

5

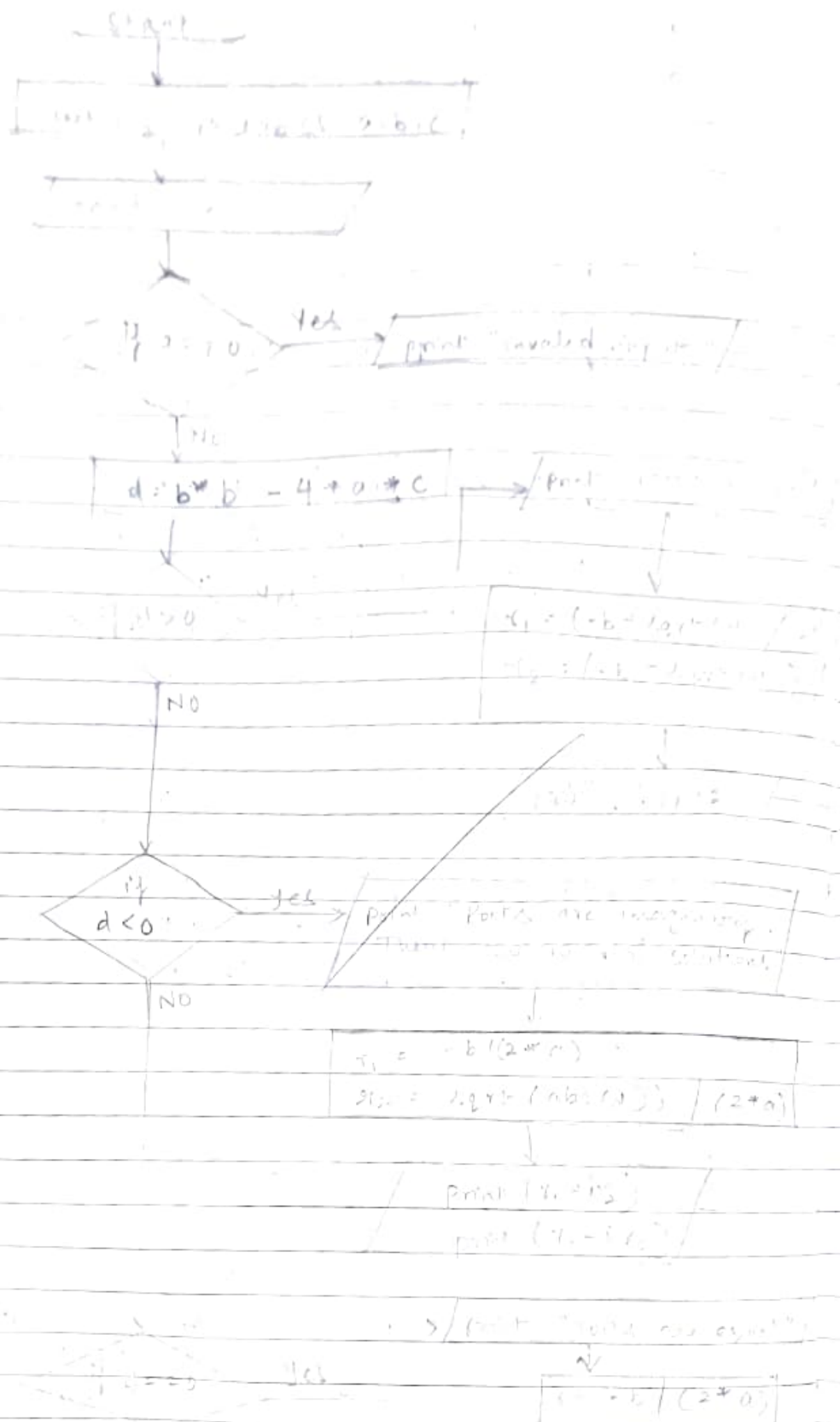
6

Roots are imaginary. There are no real solutions

-0.625 + i 1.0532687

-0.625 - i 1.0532687

# Flowchart:





classmate  
Date  
Page

## Algorithm

Step 1: Start

Step 2: Initialize variable  $a, b, c, d$  &  
read  $a, b, c$

Step 3: if  $(a=0)$  print "Invalid input" goto  
step 8

Step 4:  $d = b*b - 4*a*c$

Step 5: if  $d \geq 0$

print "roots are real"

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

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

print  $(x_1, x_2)$  goto step 8

Step 6: if  $d < 0$

print ("Roots are imaginary. There  
are no real solution")

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

$$x_2 = \sqrt{\text{abs}(d)} / (2*a)$$

print  $(x_1 + i x_2)$

print  $(x_1 - i x_2)$  goto step 8

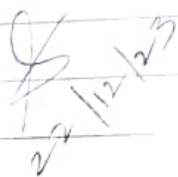
Step 7: if  $d = 0$

print "Roots are equal"

$$x_1 = x_2 = -b / (2*a)$$

print  $x_1$

Step 8: Stop

  
27/12/23



## Program - 2

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

```
class Subject
```

```
{
```

```
    int subjectMarks;
```

```
    int credits;
```

```
    int grade;
```

```
}
```

```
class Student
```

```
{
```

```
    Subject subject[];
```

```
    String name;
```

```
    String USN;
```

```
    double SGPA;
```

```
    Scanner s;
```

```
    Student()
```

```
{
```

```
    int i;
```

```
    Subject = new Subject[9];
```

```
    for (i=0; i<9; i++)
```

```
{
```

```
        subject[i] = new Subject();
```

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

```
    }
```

```
void getStudentDetails()
```

```
{
```

```
    System.out.print ("Enter your name");
```

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

```
    System.out.println ("Enter your USN:");
```

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

```
}
```

```

void getmarks() {
    for (int i = 0; i < 8; i++)
    {
        System.out.print("Enter marks for subject" + (i+1)
            + " : ");
        subject[i].subjectmarks = s.nextInt();
        System.out.print("enter your credits for subject",
            (i+1) + " : ");
        subject[i].credits = s.nextInt();
        subject[i].grade = (subject[i].subjectmarks/10)+1;

        if (subject[i].grade == 11)
            subject[i].grade = 10;
        if (subject[i].grade <= 4)
            subject[i].grade = 0;
    }
}

```

```

void computeSGPA()
{
    int effectiveScore = 0;
    int totalCredits = 0;
    for (int i = 0; i < 9; i++)
    {
        effectiveScore += (subject[i].grade * subject[i].credits);
        totalCredits += subject[i].credits;
    }
    SGPA = (double) effectiveScore / (double) totalCredits;
}
}
class Main
{
    public static void main (String args[])
    {

```

```
Student s1 = new Student();  
s1.getStudentDetails();  
s1.getMarks();  
s1.computeSGPA();  
System.out.println("Name: " + s1.name);  
System.out.println("USN: " + s1.USN);  
System.out.println("SGPA: " + s1.SGPA);  
}  
}
```

O/P: ?

Output:-

Enter your USN: IBM22CS055

Enter name: Arya

Enter marks:

95 96 98 94 99 92 95 100

SGPA: ~~9.10~~ 9.65

### Program 3 :-

```
import java.util . Scanner ;  
class book  
{
```

```
String name;  
String author;  
float price;  
int num-pages;
```



```
void set-details()
```

```
{  
    Scanner sc = new Scanner(System.in);  
    System.out.println("Enter bookname, author,  
    price, num-pages");
```

```
    name = sc.next();  
    author = sc.next();  
    price = sc.nextFloat();  
    num-pages = sc.nextInt();
```

```
}  
void get-details()  
{
```

```
    String details = toString();  
    System.out.println(details);
```

```
}
```

```
public String toString()  
{
```

```
    return "The book" + name + " was written  
    by " + author + ". It consists of "  
    + num-pages + " pages and costs around "  
    + price;
```

```
}
```

classmate  
Date: / /  
Page: /

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

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

```
System.out.println ("enter no of books you  
want to generate");
```

```
int n = scan.nextInt();
```

```
    // Array of type Book  
    book b[] = new book [n];
```

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

```
        b[i] = new book();
```

```
        b[i].set-details();
```

```
    }
```

```
System.out.println ("book details");
```

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

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

```
    b[i].get-details();
```

```
}
```

```
}
```

Output :- enter no of books you want  
to generate 2

enter bookname, author, price, num. pages  
through as sudha 200 300

enter bookname, author, price, num. pages

valley of adven Enid 300 2000

book details

the book therefore was written by Sudha  
it consists of 300 pages and costs around 200.0

the book valleyadvent was written by and it consists  
of 200 pages and costs around 300.0

#### Program 4 :-

```
import java.util.Scanner;  
abstract class Shape {  
    int x, y;  
    abstract void area();  
    public static void main(String args[])  
    {  
        * Shape obj1 = new Circle();  
        obj1.area();  
        Shape obj2 = new Rectangle();  
        obj2.area();  
        Shape obj3 = new Triangle();  
        obj3.area();  
    }  
}
```

```
class Circle extends Shape {  
    Circle() {  
        Scanner sc = new Scanner(System.in);  
        System.out.println("enter the radius of the  
        circle");  
        x = sc.nextInt();  
        y = x;  
    }  
    void area()  
    {
```

```
        System.out.println("area of circle is"  
        + 3.14 * x * y);
```

```
    }
```



```
class Rectangle extends Shape {
```

```
    Rectangle()
```

```
{
```

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

```
    System.out.println("enter the length and  
    breadth of the rectangle");
```

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

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

```
}
```

```
    void area()
```

```
{
```

```
        System.out.println("area of rectangle is"  
        x * y);
```

```
}
```

```
}
```

```
class Triangle extends Shape {
```

```
    Triangle()
```

```
{
```

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

```
    System.out.println("enter the base and  
    height of the triangle");
```

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

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

```
    void area()
```

```
{
```

System.out.println ("area of triangle is " + 0.5 \* x \* y);

}

12/11/24  
output

enter the radius of the circle

2

area of circle is 12.56

enter the length and breadth of the rectangle

2 3

area of rectangle is 6

enter the base and height of the triangle

2 4

area of triangle is 4.0.



## Program - 5

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

```
class Account {
```

```
    String CustomerName;
```

```
    long accno;
```

```
    String accountType;
```

```
    double balance;
```

```
    public Account (String CustomerName,  
                    long accno, String accountType) {
```

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

```
        this.accno = accno;
```

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

```
        this.balance = 0.0;
```

```
    }
```

```
    public void displayBalance () {
```

```
        System.out.println ("Account-Number: " +  
                             accno);
```

```
        System.out.println ("CustomerName: " +  
                             CustomerName);
```

```
        System.out.println ("AccountType: " +  
                             accountType);
```

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

```
}
```

```
class CurAcct extends Account {
```

```
    double minBalance;
```

```
    double serviceCharge;
```

```
    public CurAcct(String customerName,  
                    long accno) {
```

```
        super(customerName, accno,  
               "Current");
```

```
        this.serviceCharge = 50.0;
```

```
    }  
    serviceCharge
```

```
}
```

```
    public void withdraw(double amount)
```

```
{
```

```
        if (balance - amount >= minBalance)
```

```
{
```

```
            balance -= amount;
```

```
            System.out.println("Withdrawal successful.
```

```
            Current Balance: $ + balance);
```

```
        }
```

```
        else {
```

```
            System.out.println("Insufficient funds.
```

```
            withdrawal not allowed.");
```

```
        }
```

```
    }
```

```
public void imposeServiceCharge ()  
{
```

```
    if (balance < minBalance) {
```

```
        balance -= serviceCharge;
```

```
        System.out.println ("Service charge  
        imposed. Current Balance :  
        Rs. " + balance);
```

```
    }
```

```
}
```

```
}
```

```
class SavAcct extends Account {
```

```
    double interestRate;
```

```
    public SavAcct (String customerName, accno,  
        "Savings");
```

```
        this.interestRate = 0.05;
```

```
}
```

```
    public void depositInterest () {
```

```
        double interest = balance * interestRate;
```

```
        balance += interest;
```

```
        System.out.println ("Interest deposited.  
        Current Balance : $ " + balance);
```

```
}
```

```

public void CompoundInterest (double initialAmount,
int term) {

double compoundInterest = initialAmount * Math.pow
((1 + interestRate),
term) - initialAmount;

```

```

balance + = compoundInterest;

```

```

System.out.println ("Compound Interest deposited,
Current Balance: Rs." + balance);
}

```

```

}

public class Bank {
    public static void main (String [] args)
    {
        Scanner scanner = new Scanner (System.in);

        System.out.println ("choose account type:");

        System.out.println ("1. current");
        System.out.println ("2. Savings");
        System.out.print ("Enter choice (1 or 2) :");

        int choice = scanner.nextInt();

        System.out.print ("Enter customerName =");

        String customerName = scanner.next();

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

        long accno =
        scanner.nextLong();
    }
}

```

```
if (choice == 1) {
```

```
    CurAcct curAccount = new
```

```
    CurAcct (customerName, accno);
```

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

```
    double initialBalance =
```

```
        scanner.nextDouble();
```

```
    curAccount.balance = initialBalance;
```

```
    System.out.print ("Enter withdrawal  
amount : $ ");
```

```
    double withdrawalAmount =
```

```
        scanner.nextDouble();
```

```
    curAccount.withdraw (withdrawalAmount);
```

```
    curAccount.imposeServiceCharge();
```

```
    curAccount.displayBalance();
```

```
}
```

```
else if (choice == 2) {
```

```
    SavAcct savAccount = new
```

```
    SavAcct (customerName, accno);
```

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

double initialBalance =  
scanner.nextDouble();

savAccount.balance = initialBalance;

System.out.print("Enter withdrawal amount: \$");

double withdrawalAmount =  
scanner.nextDouble();

savAccount.balance -= withdrawalAmount;

System.out.println("Withdrawal successful.  
Current Balance: \$" + savAccount.balance);

System.out.print("Enter interestRate = ");

double interestRate = interestRate;

savAccount.displayBalance();

System.out.print("Enter term (in years) for  
Compound interest calculation:");

int term = scanner.nextInt();

savAccount.compoundInterest(initialBalance,  
term);

savAccount.displayBalance();

}  
else

{  
System.out.println("Invalid choice");



}  
3  
Output

Arya Himanshu

18M22CS055

Choose account type :

1. Current

2. Savings

Enter choice (1 or 2) : 2

Enter customer name : aru

Enter account number : 2

Enter initial balance : \$245

Enter withdrawal amount : \$235

Withdrawal successful. Current Balance : \$10.0

Enter Interest rate : 3

Account Number : 2

Customer Name : aru

Account Type : Savings

Balance : \$10.0

Enter term (in years) for compound

interest calculation : 20

Compound Interest deposited. Current Balance

Rs. 2.693803488

Customer Name : aru

Account Type : Savings

Balance : \$ 2.69380

## Prog-6

```
public class Cart {
```

```
    private String itemName;  
    private int price;  
    private int quantity;
```

```
    public void setItemName (String  
    itemName)
```

```
    {  
        this.itemName = itemName;
```

```
    }
```

```
    public String getItemName() {
```

```
        return itemName;
```

```
    }
```

```
    public void setPrice (int price) {
```

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

```
    }
```

```
    public int getPrice() {
```

```
        return price;
```

```
    }
```



public static void main (String[] args)

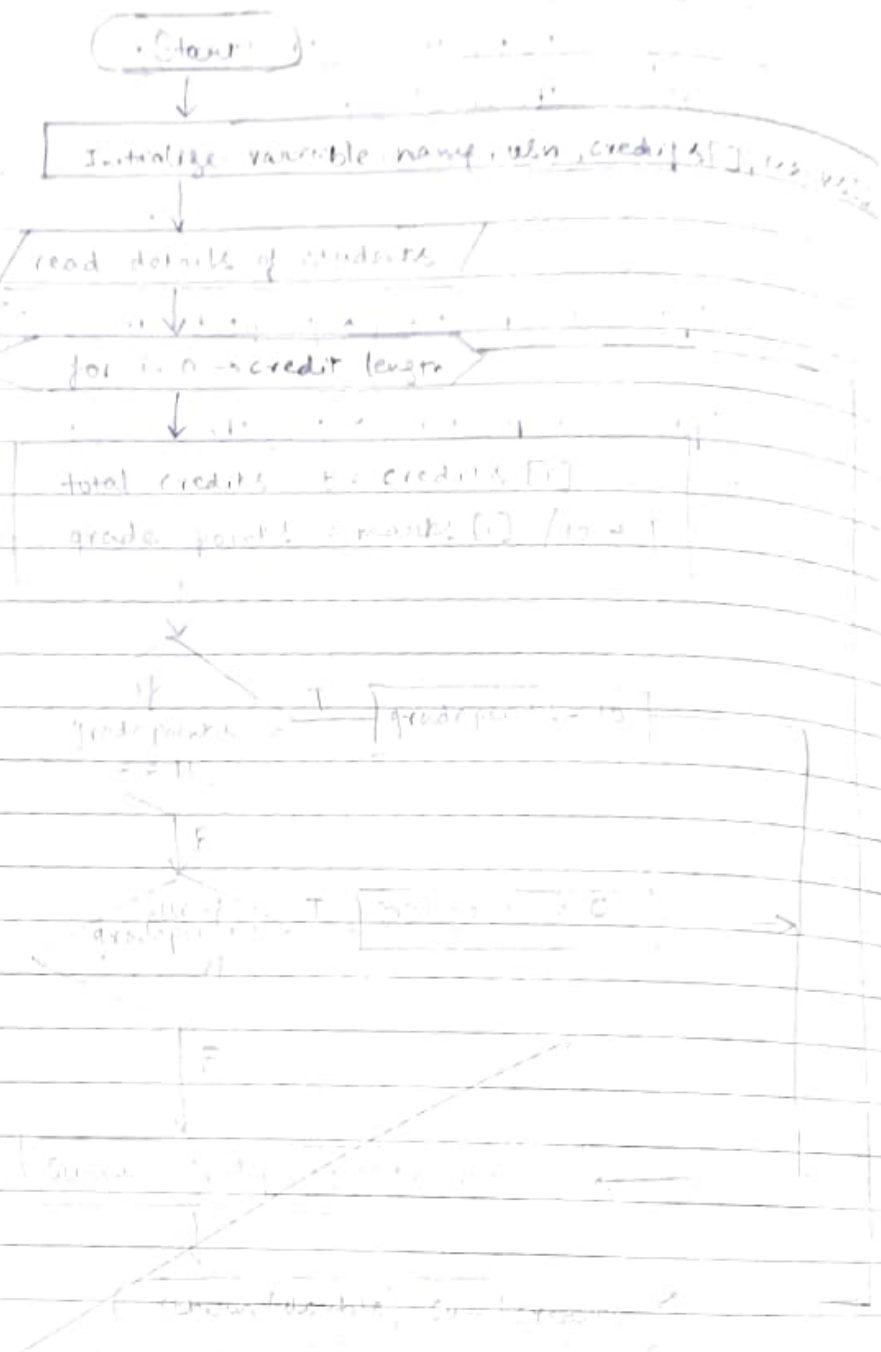
```
{  
    Cart obj = new Cart();  
    obj.setItemName("Butter");  
    obj.setPrice(50);  
    System.out.println("The details we have set  
                        are :");  
}
```

```
System.out.println(obj.getItemName());
```

```
System.out.println(obj.getPrice());
```

```
}
```

## Program - 2 : Flowchart and Algorithm:



## Algorithm

Step 1: Start

Step 2: Initialize variable arr[], marks, usn, name, sgpa.

Step 3: Calling class student in which calling func<sup>n</sup> first()

Step 4: input: "Enter usn" + usn

Step 5: Input: "Enter name" + name

Step 6: Print "Enter marks"

Step 7: for (i=0; i<8; i++) {

arr[i] = ~~next~~ in.nextInt();

}

Step 8: Read array credits as {4,4,3,3,3,1,1,1}

Step 9: for (i=0; i<arr.length; i++) {

if (arr[i] >= 100) {

arr[i] = arr[i] - 10;

}

else if (arr[i] < 40) {

arr[i] = 0;

}

Marks += credits[i] \* ((arr[i] / 10) + 1)

}

Step 10: sgpa = marks / 20

Step 11: Print "SGPA" + sgpa

Step 12: Stop.

## Program : 7

```
package CSE;
import java.util.Scanner;
public class Student Student
{
    public String usn, name;
    public int sem;
    public void inputStudentDetails()
    {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the student
                               usn:");

        usn = sc.nextLine();
        System.out.println("Enter the student
                               name:");
        name = sc.nextLine();

        System.out.println("Enter student
                               semester:");
        sem = sc.nextInt();
    }

    public void display()
    {
        System.out.println("Student USN: " + usn);
        System.out.println("Student Name: " + name);
        System.out.println("Student Sem: " + sem);
    }
}
```

```
package CIE;
import java.util.Scanner;
public class Internals extends Student
```

```
public int marks[] = new int[5];
public void inputCIEmarks()
```

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

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

```
    {
        System.out.println("Enter the marks  
for subject " + (i+1) + ":");
        marks[i] = sc.nextInt();
```

```
    }
```

```
}
```

```
package SEE;
```

```
import CIE.Internals;
```

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

```
public class Externals extends CIE.Internals
```

```
{
```

```
    public int marks[];
```

```
    public int finalMarks[];
```

```
    public Externals()
```

```
{
```

```
        marks = new int[5];
```

```
        finalMarks = new int[5];
```

```
}
```

```

public void inputSEEMarks()
{
    Scanner sc = new Scanner(System.in);

    for (int i = 0; i < 5; i++)
    {
        System.out.println("Enter subject  

        "+ (i+1) + " marks:");
        marks[i] = sc.nextInt();
    }
}

```

```

public void calculateFinalMarks()
{
    for (int i = 0; i < 5; i++)
    {
        finalMarks[i] = marks[i] / 2 + Super.marks[i];
    }
}

```

```

public void displayFinalMarks()
{
    inputStudentDetails();

    for (int i = 0; i < 5; i++)
    {
        System.out.println("Subject " +
        (i+1) + " final marks: " + finalMarks[i]);
    }
}

```

import SEE.Externals;  
class Package Main {

public static void main(String args[])  
{

int numofStudent : 2;  
Externals finalmarks[] = new Externals  
[numofStudent];

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

{  
finalmarks[i] : new Externals();

finalmarks[i].inputStudentDetails();  
System.out.println("Enter CIE marks:");

finalmarks[i].inputCIEmarks(),  
System.out.println("Enter SEE marks:");  
finalmarks[i].inputSEEMarks();

}

System.out.println("Displaying data:\n");

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

{  
finalmarks[i].calculateFinalMarks();  
finalmarks[i].displayFinalMarks();  
Output : -  
}

}

}



## Program - 3 Algorithm

- Step 1: Start
- Step 2: Initialize variables, names, author, price, num-pages
- Step 3: Enter no of books
- Step 4: Enter name, author, price, num-pages
- Step 5: for ( $i=0$ ;  $i < n$ ;  $i++$ )  
     $b[i] = \text{new books}()$   
     $b[i] = \text{set}[i];$
- Step 6: ~~display books~~  
    ~~for ( $i=0$ ;  $i < n$ ;  $i++$ )~~  
        ~~prints book details~~
- Step 7: Stop

### Program 7: Output

Enter the value of  $n$

1

Enter name: Arya

Enter usn: 1bm22CS055

Enter semester: 3.

Enter internal marks of Course 1: 23

" " 2 : 35

" " 3 : 37

" " 4 : 38

" " 5 : 32

Enter semester: 3

Enter external marks of Course 1: 30

" " 2 : 56

" " 3 : 89

4 : 90

5 : 95



## Program - 4 Algorithm.

- Step 1: Create abstract class named Shape
- Step 2: Include 2 members  $x$  &  $y$ .
- Step 3: Declare abstract method `area()`;
- Step 4: Create sub-class Rectangle that extends Shape
- Step 5: Override area method to calculate area of rectangle
- Step 6: Repeat Steps 4 and 5 for triangle and circle
- Step 7: In main method create object rectangle, triangle and circle
- Step 8: Stop.

Name: arya

USN: 1bm22 cs055

Sem: 3

Course 1: 38.0

Course 2: 63.0

Course 3: 82.0

Course 4: 83.0

Course 5: 80.0

### Program 8:

Write a program that demonstrates handling of exceptions in inheritance tree.

class AccountsThread

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 takes 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 {  
    public WrongAge(String message)  
{  
    }  
    super(message);  
}
```

```
class Father {  
    int fatherAge;  
    public Father(int fatherAge) throws  
WrongAge {
```

```
        if (fatherAge < 0) {  
            throw new WrongAge("Age cannot be  
            negative");  
        }
```

```
        this.fatherAge = fatherAge;
```

```
    }  
}
```

↳ Accessing the particular fatherAge Attribute.

```
class Son extends Father {
```

```
    int sonAge;  
    public Son (int fatherAge, int sonAge)  
        throws WrongAge {
```

```
        super(fatherAge);  
        if (sonAge >= fatherAge) {
```

```
            throw new WrongAge ("son's  
            age must be less than Father's age");  
        }
```

```
        this.sonAge = sonAge;  
    }
```

```
}
```

```
public class FatherSon {
```

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

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

```
        System.out.println ("Enter father's age  
        and son's age :");
```

```
        int fa = sc.nextInt();
```

```
        int so = sc.nextInt();
```

```
        try {
```

```
            Son s = new Son (fa, so);
```

```
            System.out.println ("Father's age"  
            + s.fatherAge);
```

```
System.out.println ("Son's age: " + ts.sonAge);
```

```
} // Name of the exception  
catch (WrongAge e) {
```

```
System.out.println ("Error: " + e.getMessage());
```

```
}
```

```
}
```

Output :-

Enter father's age and son's age:

56

23

Father's age: 56

Son's age: 23

C:\55 > java fatherSon

Enter father's age and son's age:

34

57

Error: Son's age must be less than Father's age

~~C:\55~~

Assign

IBM 22CS055

1000 - 10 seconds  
2000 - 20 seconds

### Program 3:

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

```
class A extends Thread
{
    int t1, time; int t1 = 0, time;
    A() {
        t1 = 10000; t1 = 10000;
        time = 21000;
    }
    public void run()
    {
        while (t1 <= time)
        {
            System.out.println("BMS College of Engineering");

            try {
                sleep(10000);
            } catch (Exception e) {
                System.out.println("error");
            }

            t1 += 10000;
        }
    }
}
```

```
class B extends Thread {
```

```
int t2, time;
```

```
int t2 = 0, time;
```

```
B() {
```

```
time = 21000;
```

```
t2 = 2000;
```

```
}
```

```
public void run() {
```

```
{
```

```
while (t2 <= time)
```

```
{
```

```
System.out.println("CSE");
```

```
try {
```

```
sleep(2000);
```

```
}
```

```
catch (Exception e)
```

```
{
```

```
System.out.println("error");
```

```
}
```

```
t2 += 2000;
```

```
}}
```

```
}
```

```
class tn
```

```
{
```

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

```
{
```

```
A a = new A();
```

```
B b = new B();
```

```
a.start();
```

```
b.start();
```

```
}
```

Output

BMS COLLEGE OF ENGINEERING

CSE

CSE

CSE

CSE

CSE

BMS COLLEGE OF ENGINEERING

CSE

CSE

CSE

CSE

CSE

Arya Himanshu

IBM23CS055

~~16/2/24~~



Write a program that creates a 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~~ not an integer, the program would throw a NumberFormatException. If Num2 were zero, the program would throw an ArithmeticException. Display the exception ~~to the~~ in a message dialog box.

Code:

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

```
class SwingDemo {
    SwingDemo() {
        JFrame jfrm = new JFrame("Dividers App");
        jfrm.setSize(275, 150);
        jfrm.setLayout(new FlowLayout());
        // to terminate on close
        jfrm.setDefaultCloseOperation(JFrame.EXIT-ON-
            CLOSE);

        // text label
        JLabel jlab = new JLabel("Enter the divider
            and dividend.");

        // add text field for both numbers
        JTextField aJtf = new JTextField(8);
        JTextField bJtf = new JTextField(8);

        // Calc button,
```

JButton button = new JButton("Calculate");

// labels

JLabel err = new JLabel();

JLabel alab = new JLabel();

JLabel blab = new JLabel();

JLabel anslab = new JLabel();

// add in order :

jfrm.add(err); // to display error box.

jfrm.add(jlab);

jfrm.add(a1tf);

jfrm.add(b1tf);

jfrm.add(button);

jfrm.add(alab);

jfrm.add(blab);

jfrm.add(anslab);

ActionListener l = new ActionListener() {

public void actionPerformed(ActionEvent

System.out.println("Action event from  
a text field");

}

};

```
ajtf  
ajtf  
addActionListener(i);  
addActionListener(i);
```

```
button.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent evt) {
```

```
        try {
```

```
            int a = Integer.parseInt(ajtf.getText());  
            int b = Integer.parseInt(bjtf.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 (ArithmeticException e) {
```

```
            alab.setText("");  
            blab.setText("");  
            ansLab.setText("");  
            err.setText("B should be non zero!");
```

```
        }
```

// display frame

frm.setVisible(true);

}

public static void main (String args[])

// create frame on event dispatching thread.

SwingUtilities.invokeLater (new Runnable

public void ~~run~~ run () {

new & SwingDemo();

}

});

}

}

Output:

Arya Himanshu

BM22CS055

Divider App -  X

Enter the divider and dividend

12

4

A=12 B=4 Ans=3