

class Diagram :

Distance

~x1 : int
~x2 : int
~y1 : int
~y2 : int
~dis : double

Output :

Enter the coordinates of first point:

10
20

Enter the coordinates of second point:

30
40

Distance : 28.284271247461902

Prog. No.
Date.....

Page No.

Aim :

write a java program to find the distance between 2 points.

SOURCE CODE :

```
import java.io.*;
import java.util.*;
class Distance {
    public static void main(String args[])
    {
        Scanner in = new Scanner(System.in);
        int x1, x2, y1, y2;
        double dis;
        System.out.println("Enter the coordinates of first point:");
        x1 = in.nextInt();
        y1 = in.nextInt();
        System.out.println("Enter the coordinates of second point:");
        x2 = in.nextInt();
        y2 = in.nextInt();
        dis = Math.sqrt(((x2 - x1) * (x2 - x1)) + ((y2 - y1) * (y2 - y1)));
        System.out.println("Distance :" + dis);
    }
}
```

Prog. No. 2

Page No.

class Diagram

numbers

SOURCE CODE :

AIM:
write a program to find sum, difference, product, quotient and remainder of 2 numbers' passed as command line argument.

```

import java.util.Scanner;

class Numbers {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the value of 'a' : ");
        int a = scanner.nextInt();
        System.out.println("Enter the value of 'b' : ");
        int b = scanner.nextInt();
        System.out.println("Sum : " + (a+b));
        System.out.println("Difference : " + (a-b));
        System.out.println("Product : " + (a*b));
        System.out.println("Quotient : " + (a/b));
        if(b != 0) {
            System.out.println("Quotient : " + (a/b));
        }
    }
}

```

Output :

Enter the value of 'a' : 10

Enter the value of 'b' : 20

Sum : 30

Difference : -10

Product : 200

Quotient : 0

Remainder : 10

```
System.out.println("Remainder: "+  
(a%b));  
else {  
System.out.println(" Cannot divide  
by zero ");  
}  
Scanner.close();  
}
```

class Diagram:

```
fib
~n1 : int = 0
~n2 : int = 1
~n3 : int
~limit : int
~i : int
```

Output:

Enter the limit :

5

```
import java.io.*;
import java.util.*;
class fib {
    public static void main (String args[])
    {
        Scanner in = new Scanner (System.in);
        int n1=0, n2=1, n3, limit;
        System.out.println ("Enter the
        limit : ");
    }
}
```

Aim :
write java program to display
Fibonacci Series up to a limit.

SOURCE CODE:

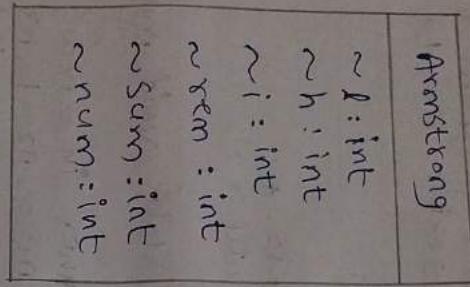
```
import java.io.*;
import java.util.*;
class fib {
    public static void main (String args[])
    {
        Scanner in = new Scanner (System.in);
        int n1=0, n2=1, n3, limit;
        System.out.println ("Enter the
        limit : ");
        limit = in.nextInt();
        System.out.print (n1 + "\t" + n2);
        for (i=2; i<limit; i++) {
            n3 = n1+n2;
            n1 = n2;
            n2 = n3;
            System.out.print (" " + n3);
        }
    }
}
```

Prog. No. 3

Date.....

Page No.

class Diagram:



• Aim:
java program to display Armstrong numbers within a range.

SOURCE CODE:

```
import java.io.*;
import java.util.*;
class Armstrong {
    public static void main (String args[]){
        Scanner in = new Scanner (System.in);
        int l, h, i, rem, sum;
        System.out.println("Enter the
lower limit & upper limit : ");
        l = in.nextInt();
        h = in.nextInt();
        System.out.println("Armstrong numbers");
        for(i=l; i<=h; i++){
            num = i;
            sum = 0;
            while (num != 0) {
                rem = num % 10;
                sum = sum + (rem * rem);
                num = num / 10;
            }
            if (sum == i)
                System.out.println(i);
        }
    }
}
```

Prog. No.....

Date.....

Page No.....

Output:

Enter the lower limit & upper limit:

100

500

Armstrong numbers:

153

370

371

407

```
if(Sum==i)
```

```
{  
    System.out.print(n(i));  
}
```

```
}
```

```
}
```

class Diagram:

```
class Triangle  
{  
    ~a: int  
    ~b: int  
    ~c: int  
    ~s: double  
    ~area: double}
```

Alm:

Given the sides of a triangle, write a program to check whether the triangle is equilateral, isosceles / scalene and find its area.

SOURCE CODE:

```
import java.*;  
import java.util.*;  
class Triangle {  
    public static void main (String args[]){  
        int a,b,c;  
        double s,area;  
        Scanner in = new Scanner (System.in);  
        System.out.println ("Enter the  
        sides of a triangle : ");  
        a = in.nextInt();  
        b = in.nextInt();  
        c = in.nextInt();  
        if (a==b & b==c)  
            System.out.println ("Equilateral triangle");  
        else if (a==b || b==c || a==c)  
            System.out.println ("Isosceles triangle");  
        else  
            System.out.println ("Scalene triangle");  
    }  
}
```

Output:

Enter the sides of a triangle :

20

10

150 scalene triangle

Area : 96.8245836551

```
else
    System.out.println ("scalene
    triangle");
    area = math.sqrt (s * (s-a) * (s-b) *
                      (s-c));
    System.out.println ("Area :" + area);
}
```

Class Diagram:

Numbers
~n: int
~i: int
~j: int
~temp: int
~a[]: int

AIM:

- Read an array of 10 or more numbers and write a program to find
a) Smallest element in the array
b) Largest element in the array
c) Second largest element in the array.

SOURCE CODE:

```
import java.io.*;
import java.util.*;
class Numbers{
    public static void main (String args[])
    {
        Scanner in = new Scanner (System.in);
        int n, i, j, temp;
        int a[] = new int[10];
        System.out.println ("Enter the
        number of elements : ");
        n = in.nextInt();
        System.out.println ("Enter the
        elements: ");
        for(i=0; i<n; i++)
        {
            a[i] = in.nextInt();
        }
    }
}
```

Output:

Enter the number of elements:

5

Enter the elements:

2

4

1

6

3

smallest element : 1

largest element : 6

Second largest element : 4

```
for(j=0; j<n-1; j++) {  
    if (a[j] > a[j+1]) {  
        temp = a[j];  
        a[j] = a[j+1];  
        a[j+1] = temp;  
    }  
}  
System.out.println ("smallest  
element :" + a[0] +  
"\n largest element :" + a[n-1] +  
"\n second largest element :" +  
a[n-2]);  
}
```

Class Diagram:

Base
~num : int
~rem : int
~base : int
~str : String
~dig[] : char

- AIM :
 - write a pgm to perform base conversion
 - a) Integer to binary
 - b) Integer to Octal
 - c) Integer to Hexadecimal.

SOURCE CODE :

```

import java.io.*;
import java.util.*;
class Base {
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);
        int num, rem, base;
        String str = "";
        char dig[] = {'0', '1', '2', '3', '4', '5', '6', '7',
                      '8', '9', 'A', 'B', 'C', 'D', 'E', 'F'};
        System.out.println("Enter the number:");
        num = in.nextInt();
        System.out.print("Enter the base To convert : ");
        base = in.nextInt();
        while (num > 0)
        {
            rem = num % base;
            str = str + dig[rem];
            num = num / base;
        }
        System.out.println(str);
    }
}
  
```

Output :

Enter the number:

10

Enter the base to convert:

2

1010

```
stry = dig[rem] + str;  
num = num / base;  
}  
System.out.println(str);  
}
```

Class Diagram:

Merge

```

~m : int
~n : int
~i : int
~j : int
~k : int
~a1[] : int
~a2[] : int
~a3[] : int
    
```

AIM:
write a program to merge 2 arrays.

SOURCE CODE :

```

import java.io.*;
import java.util.*;
public static void main (String args[]){
    Scanner in = new Scanner (System.in);
    int m, n, i, j, k = 0;
    int a1[] = new int [10];
    int a2[] = new int [10];
    int a3[] = new int [20];
    System.out.println ("Enter the size
        of array 1:");
    m = in.nextInt();
    System.out.println ("Enter the
        elements = ");
    for (i = 0; i < m; i++) {
        a1[i] = in.nextInt();
    }
    System.out.println ("Enter the
        size of array 2:");
    n = in.nextInt();
    
```

```

System.out.println("enter the elements");
for(i=0; i<n; i++){
    a2[i] = in.nextInt();
}
i=0;
j=0;
k=0;
while(i < m && j < n){
    if(a1[i] < a2[j]){
        a3[k] = a1[i];
        i++;
    }
    else{
        a3[k] = a2[j];
        j++;
    }
    k++;
}
if(i >= m){
    while(j < n){
        a3[k] = a2[j];
        j++;
        k++;
    }
}
}

```

Output :

enter the size of array 1:

3

Enter the elements:

10

30

50

Enter the size of array 2:

2

Enter the elements:

20

40

After merging :

10

20

30

40

50

Class Diagram:

HCFLCM

```
~a : int  
~b : int  
~x : int  
~y : int  
~t : int  
~hcf : int  
~lcm : int
```

Prog. No. 9

Date.....

Page No.

• Aim : write a program to find HCF LCM of 2 numbers.

• SOURCE CODE :

```
import java.io.*;  
import java.util.*;  
class HCFLCM {  
    public static void main(String args[]){  
        int a,b,x,y,t,hcf,lcm;  
        Scanner in = new Scanner(System.in);  
        System.out.println("Enter 2 numbers:");  
        x = in.nextInt();  
        y = in.nextInt();  
        a = x;  
        b = y;  
        while(b != 0) {  
            t = b;  
            b = a % b;  
            a = t;  
        }  
        hcf = a;  
        lcm = (x * y) / hcf;
```

Prog. No.....

Date.....

Page No.....

Output:
enter 2 numbers :

16
20

HCF = 4

LCM = 80

```
System.out.println("HCF = "+hcf);
System.out.println("LCM = "+lcm);
```

class Diagram:

```
cm convert
~cm : double
~inch : double
~mtr : double
~km : double
```

Output:

Enter the centimeter value :

20
20.0 cm is equal to 7.874 inches
20.0 cm is equal to 0.2 mtr
20.0 cm is equal to 2.0E-4 km.

AIM:

write a program to convert centimeter to inch ,meter and kilometer.

SOURCE CODE:

```
import java.io.*;
import java.util.*;
class cmconvert {
    public static void main (String args[])
    {
        Scanner in = new Scanner (System.in);
        double cm, inch, mtr, km;
        System.out.println ("Enter the
        centimeter value : ");
        cm = in.nextDouble ();
        inch = 0.3937 * cm;
        mtr = 0.01 * cm;
        km = 0.00001 * cm;
        System.out.println ("(" + cm + "cm is
        equal to " + inch + "inch");
        System.out.println ("(" + cm + "cm is
        equal to " + mtr + "mtr");
        System.out.println ("(" + cm + "cm is
        equal to " + km + "km");
```

class Diagram:

Matrix

```
n : int  
m : int  
i : int  
j : int  
trace : int = 0  
naccc : int  
nbcc : int
```

AIM:

write a program to find the trace
and transpose of a matrix.

SOURCE CODE:

```
import java.io.*;  
import java.util.*;  
class Matrix  
public static void main (String args[]){  
    Scanner in = new Scanner (System.in);  
    int n,m,i,j,trace=0;  
    int a[][] = new int [10][10];  
    int b[][] = new int [10][10];  
    System.out.println ("Enter the order of  
    the matrix :");  
    m = in.nextInt();  
    n = in.nextInt();  
    System.out.println ("Enter the elements :");  
    for(i=0;i<m;i++){  
        for(j=0;j<n;j++) {  
            a[i][j] = in.nextInt();  
        }  
    }
```

Output:

Enter the order of the matrix :

3
3

Enter the elements :

1 2 3
4 5 6
7 8 9

trace : 15

Transpose :

1 4 7
2 5 8
3 6 9

```
for(i=0; i<m; i++) {  
    for(j=0; j<n; j++) {  
        b[j][i] = a[i][j];  
        if(i==j) {  
            trace = trace + a[i][j];  
        }  
    }  
}  
System.out.println("Trace : " + trace);  
System.out.println("Transpose : ");  
for(i=0; i<m; i++) {  
    for(j=0; j<n; j++) {  
        System.out.print(b[j][i]);  
    }  
    System.out.println();  
}
```

class Diagram :

```

class Ten {
    int num;
    int rem;
    int sum = 0;
    int rev = 0;

    ~sumrev()
}

```

AIM:

write a program to find the sum of the digits and reverse of a given number using class and object.

SOURCE CODE:

```

import java.io.*;
import java.util.*;
public class Ten {
    int num, rem, sum = 0, rev = 0;
    public static void main(String args[]) {
        Ten obj = new Ten();
        obj.sumrev();
    }

    void sumrev() {
        Scanner in = new Scanner(System.in);
        System.out.print("Enter the number: ");
        num = in.nextInt();
        while (num != 0) {
            rem = num % 10;
            sum = sum + rem;
            rev = rev + (rev * 10);
            num = num / 10;
        }
    }
}

```

Output :

Enter the number :
123
sum : 5
reverse : 321

Prog. No.....
Date.....

Page No.....

```
System.out.println ("sum: " + sum);
System.out.println ("reverse: " + rev);
}
}
```

class Diagram :

Anagram	
~str1 : String	
~str2 : String	
~len1 : int	
~len2 : int	
~st1 : char[]	
~st2 : char[]	

Aim:

write a program to check given string Anagram or not. If the 2 strings are Anagram to each other, then 1 string can be rearranged to form the other string.

SOURCE CODE:

```
import java.io.*;
import java.util.*;
class Anagram {
    public static void main (String args[]){
        Scanner in = new Scanner(System.in);
        String str1, str2;
        int len1, len2;
        System.out.println ("Enter the first string:");
        str1 = in.next();
        System.out.println ("Enter the second string:");
        str2 = in.next();
        char[] st1 = str1.toCharArray();
        char[] st2 = str2.toCharArray();
        Arrays.sort(st1);
        Arrays.sort(st2);
```

Output :

Enter the first string :
silent
Enter the second string :

strings are Anagram

```
[len1 = str1.length();  
len2 = str2.length();  
if (len1 == len2) {  
    if (Array.equals(st1, st2))  
        System.out.println("Strings  
        are Anagram");  
    else  
        System.out.println("Strings  
        are not Anagram");  
} else {  
    System.out.println("Different  
    length ... Not Anagram");  
}
```

class Diagram:

Vowel
str : String

Output :

Enter the String :

welcome

wcm

SOURCE CODE :

```

import java.io.*;
import java.util.*;
class Vowel {
    public static void main(String args[]) {
        Scanner in = new Scanner(System.in);
        String str;
        System.out.println("Enter the
String : ");
        str = in.nextLine();
        str = str.replaceAll("[aeiouAEIOU]", "");
        System.out.println(str);
    }
}

```

Class Diagram:

```

Complex
~real : int
~image : int
add(Complex c1, Complex c2) : Complex.

```

- AIM:
Using class and objects, write a program to find the sum of 2 complex numbers.
- SOURCE CODE:

```

import java.io.*;
import java.util.*;
class Complex {
    int real, image;
    public static void main (String args[])
    {
        Scanner in = new Scanner (System.in);
        Complex c1 = new Complex();
        Complex c2 = new Complex();
        System.out.println ("Enter the first complex number:");
        c1.real = in.nextInt();
        c1.image = in.nextInt();
        System.out.println ("Enter the second complex number:");
        c2.real = in.nextInt();
        c2.image = in.nextInt();
        c1.add(c1, c2);
    }
}

```

Prog. No.....
Date.....

Page No.....

Output:
Enter the first complex number:

10
20

Enter the second complex number:

30
40

sum : 40+60

```
void add(Complex c1,Complex c2){  
    Complex c3=new Complex();  
    c3.real = c1.real + c2.real;  
    c3.image = c1.image + c2.image;  
    System.out.println("Sum:" +  
        (c3.real+"+"+c3.image));  
}
```

Class Diagram :

```
Numobj
~Count: int = 0
Numobj()
```

Output :

Number of objects created : 3

AIM :
write a program to count and display total number of objects created to a class.

SOURCE CODE :

```
import java.io.*;
import java.util.*;
class Numobj {
    static int count = 0;
    Numobj() {
        count++;
    }
    public static void main(String args) {
        Numobj obj1 = new Numobj();
        Numobj obj2 = new Numobj();
        Numobj obj3 = new Numobj();
        System.out.println("Number of objects created :" + count);
    }
}
```

class Diagram:

```

class Seventeen
{
    ~length : double
    ~width : double
    ~height : double
    ~radius : double
    ~area : double
    ~volume (double l)
    ~volume (double l, double w, double h)
    ~volume (double r, double h).
}

```

- AIM :
write a java program to find the volume
of cube, rectangular box, cylinder using
function overloading.
- SOURCE CODE :

```

import java.io.*;
import java.util.*;
class Seventeen {
    public static void main (String args[])
    {
        Seventeen obj = new Seventeen();
        Scanner in = new Scanner (System.in);
        double length, width, height, radius,
               area;
        System.out.println ("Enter the length
                           of cube:");
        length = in.nextDouble();
        obj.volume (length);
        System.out.println ("Enter the
                           radius and height of cylinder:");
        radius = in.nextDouble();
        height = in.nextDouble();
        obj.volume (length, height);
        System.out.println ("Enter the length
                           width, height of rectangular box:");
    }
}

```

Output:

Enter the length of cube :
10
volume : 1000.
Enter the radius and height of cylinder:

Volume : 1570.0
5
10

enter the length, width, height of rectangular box:

Volume : 100.0

```

length = in.nextInt();
width = in.nextInt();
height = in.nextInt();
obj volume (length, width, height);

}
void volume (double l) {
    System.out.println("volume : " + l * l * h);
}

void volume (double l, double w, double h) {
    System.out.println ("volume : " + l * w * h);
}

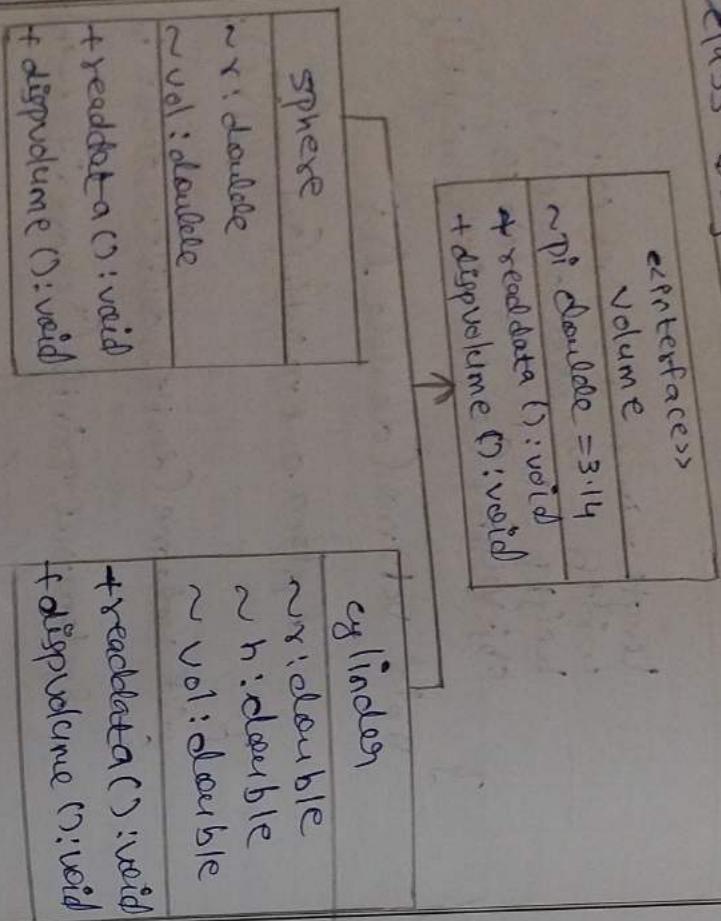
void volume (double x, double y) {
    System.out.println ("volume : " + 3.14 * x * y * h);
}

```

Prog. No.....

Page No.—

Class Diagram:



AIM: Create an interface Volume with member variable pi and methods readData() and dispVolume(). Create 2 classes Sphere and Cylinder to implement this interface. Write a program to find the volume.

SOURCE CODE:

```

import java.io.*;
import java.util.*;
interface Volume {
    static final double PI=3.14;
    public abstract void readData();
    public abstract void dispVolume();
}
class Sphere implements Volume {
    Scanner in = new Scanner(System.in);
    double r, vol;
    public void readData () {
        System.out.println("Enter the radius sphere:");
        r = in.nextDouble();
    }
}
  
```

```
public void dispvolume() {
    vol = (4/3) * pi * r * r * r;
    System.out.println("Volume of
Sphere : " + vol);
```

}

```
} class cylinder implements Volume {
    Scanner in = new Scanner (System.in);
    double r, vol, h;
    public void readdata() {
        System.out.println("Enter the
Radius of cylinder : ");
        r = in.nextDouble();
        System.out.println("Enter
the height of cylinder : ");
        h = in.nextDouble();
```

}

```
} public void dispvolume() {
    vol = pi * r * r * h;
    System.out.println("Volume of
cylinder : " + vol);
```

}

```
} class eighteen {
    static void main (String args) {
        public cylinder c = new cylinder();
```

Prog. No.....
Date.....

Page No.

Output:

Enter the radius of cylinder:

10

Enter the height of cylinder:

2

Volume of cylinder : 628.0

Enter the radius sphere :

5

Volume of sphere : 392.5

```
sphere sp = new sphere();  
cyl .readdata();  
cyl .disvolume();  
sp .readdata();  
sp .disvolume();
```

Class Diagram:

Odd	~i : int	~n : int = 10	+run : void
Even	~i : int	~n : int = 10	+run : void

Odd Even Thread

obj1 : odd
obj2 : even

- AIM : write a multi thread program for displaying odd numbers and even numbers up to a limit.

SOURCE CODE :

```

import java.io.*;
import java.util.*;
class Odd implements Runnable {
    int p, n = 10;
    public void run() {
        for (p = 1; p <= n; p += 2)
            System.out.println ("odd:" + p);
    }
}
class Even implements Runnable {
    int i, n = 10;
    public void run() {
        for (i = 0; i < n; i += 2)
            System.out.println ("even:" + i);
    }
}

```

Output:

```

Odd: 1
Odd: 3
Odd: 5
Odd: 7
Odd: 9
Even: 0
Even: 2
Even: 4
Even: 6
Even: 8
}

```

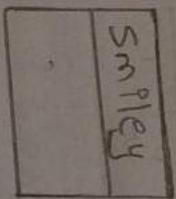
```

}
class OddEvenThread {
public static void main(String args) {
    Even obj2 = new Even();
    Thread obj2 = new Thread(obj2);
    obj2.start();
}

Odd obj1 = new Odd();
Thread obj1 = new Thread(obj1);
obj1.start();
}
}

```

class Diagram:



Smiley

- AIM:

create an applet for displaying smiling face

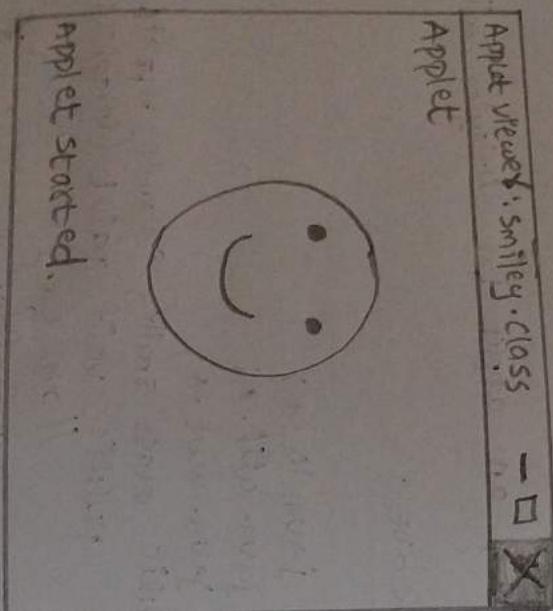
- SOURCE CODE :

```
import java.io.*;
import java.awt.*;
import java.awt.event.*;
public class Smiley extends Applet {
    public void paint(Graphics g) {
        // oval for face outline
        g.setColor(Color.YELLOW);
        g.fillOval(80, 10, 150, 150);
        // ovals for eyes
        // with black color filled
        g.setColor(Color.BLACK);
        g.fillOval(120, 60, 50, 180);
        g.fillOval(170, 120, 50, 180);
    }
}
```

// Arc for the smile

```
g.drawArc(125, 140, 60, 50, 180, 180);
```

Output :



Prog. No.....
Date.....

Page No.....