```
import java.io.*;
import java.util.*;
class nari
{
	public static void main(String args[])
	{
		try
		{
		int n,j,i;
		System.out.println("\tWOW!!!!! I GOT THE OUTPUT");
		InputStreamReader nnn=new InputStreamReader(System.in);
		BufferedReader br=new BufferedReader(nnn);
		do
		{
			System.out.println("1.MEAN\n2.MEDIAN\n3.MODE\n4.BISECTION
METHOD\n5.REGULAR FALSI METHOD\n6.Gauss-jacobi method\n7.Gauss-seidal
method\n8.EXIT");
			System.out.print("ENTER YOUR CHOICE:-");
			n=Integer.parseInt(br.readLine());
			switch(n)
			{
				case 1:
					System.out.println("\n1.RAW DATA\n2.DISCRETE
DATA\n3.CONTINUOS DATA");
					System.out.print("\nEnter your choice again:");
					int a=Integer.parseInt(br.readLine());
					switch(a)
					{
						case 1:
							double[] a1=new double[50];
							double b1=0;
							System.out.println("HOW MANY NUMBERS
ARE THERE TO CALCULATE MEAN");
							int n1=Integer.parseInt(br.readLine());
							System.out.println("ENTER THE NUMBERS
CONTINUOSLY BY PRESSING ENTER");
							for(i=0;i<n1;i++)
							{

a1[i]=Double.parseDouble(br.readLine());
								b1=b1+a1[i];
							}
							double c=b1/n1;
							System.out.println("MEAN="+c);
```

```
                break;
            case 2:
                double a2[]=new double[50];
                double b2[]=new double[50];
                double c2=0,d2,f2=0;
                System.out.println("HOW MANY NUMBERS
ARE THERE TO CALCULATE MEAN");
                double
n2=Double.parseDouble(br.readLine());
                System.out.println("Enter the numbers(x):");
                for(i=0;i<n2;i++)
                {

a2[i]=Double.parseDouble(br.readLine());
                }
                System.out.println("Enter the
frequencies(f):");
                for(i=0;i<n2;i++)
                {

b2[i]=Double.parseDouble(br.readLine());
                }
                for(i=0;i<n2;i++)
                {
                    c2= c2+(a2[i]*b2[i]);
                }
                for(i=0;i<n2;i++)
                {
                    f2=f2+b2[i];
                }
                d2=c2/f2;
                System.out.println("MEAN="+d2);
                break;
            case 3:
                double a3[]=new double[50];
                double b3[]=new double[50];
                double c3[]=new double[50];
                double d3[]=new double[50];
                double f3[]=new double[50];
                double g3[]=new double[50];
                double h3=0,i3=0;
                System.out.print("\nHOW MANY NUMBERS
ARE THERE TO CALCULATE MEAN\n");
                int n3=Integer.parseInt(br.readLine());
```

```
                                                    for(i=1,j=1;i<=n3;i++,j++)
                                                    {
                                                            System.out.print(i+".)Enter the lower
limit\n");

a3[i]=Double.parseDouble(br.readLine());
                                                            System.out.print(j+".)Enter the upper
limit\n");

b3[j]=Double.parseDouble(br.readLine());
                                                    }
                                                    System.out.println("Enter the frequency
continuosly by pressing enter one after another");
                                                    for(i=1;i<=n3;i++)
                                                    {

c3[i]=Double.parseDouble(br.readLine());

                                                    }
                                                    //To find mid value is :
                                                    for(i=1;i<=n3;i++)
                                                    {
                                                            d3[i]=(a3[i]+b3[i])/2;
                                                    }
                                                    int e3=(1+n3)/2;
                                                    double A=d3[e3];
                                                    double cl=d3[2]-d3[1];
                                                    //Calculation of d:
                                                    for(i=1;i<=n3;i++)
                                                    {
                                                            f3[i]=((d3[i]-A)/cl);
                                                            g3[i]=f3[i]*c3[i];
                                                    }
                                                    //Summation of fd and f:
                                                    for(i=1;i<=n3;i++)
                                                    {
                                                            h3=h3+c3[i];
                                                            i3=i3+g3[i];
                                                    }
                                                    //Calculation of mean:

                                                    double AM=A+(i3/h3)*cl;

System.out.println("\n_____");
```

```
                                                System.out.println("|    X    |  f  |M.V of
X|d=(X-A)/C|fd |");

System.out.println("*****************************************");
                                                for(i=1;i<=n3;i++)
                                                {

System.out.println("|"+a3[i]+"-"+b3[i]+" | "+c3[i]+" |   "+d3[i]+"  |  "+f3[i]+"  |"+g3[i]+"|");
                                                }

System.out.println("*****************************************");
                                                System.out.println("          | "+h3+" |
|"+i3+"|");
                                                System.out.println("          *******
******");
                                                System.out.println("MEAN="+AM);
                                                break;
                                        }
                                        break;
                                case 2:
                                        System.out.println("\n1.RAW DATA\n2.DISCRETE
DATA\n3.CONTINUOS DATA");
                                        System.out.print("\nEnter your choice again:");
                                        int b=Integer.parseInt(br.readLine());
                                        switch(b)
                                        {
                                                case 1:
                                                        int a4[]=new int[50];
                                                        int temp;
                                                        System.out.println("\nHOW MANY
NUMBERS ARE THERE TO CALCULATE MEDIAN");
                                                        int n4=Integer.parseInt(br.readLine());
                                                        System.out.println("Enter the numbers to
find median");
                                                        for(i=1;i<=n4;i++)
                                                        {
                                                                a4[i]=Integer.parseInt(br.readLine());
                                                        }
                                                        //Sorting:
                                                        for(i=1;i<=n4;i++)

                                                        {
                                                                for(j=i+1;j<=n4;j++)
                                                                {
```

```
                                        if(a4[i]>a4[j])
                                        {
                                                temp=a4[i];
                                                a4[i]=a4[j];
                                                a4[j]=temp;
                                        }
                                }
                        }
                        System.out.println("_______");
                        System.out.println("| A.S |");
                        System.out.println("*******");
                        for(i=1;i<=n4;i++)
                        {
                                System.out.println("| "+a4[i]+" |");
                        }
                        if(n4%2==0)
                        {
                                i=(1+n4)/2;
                                double b4=a4[i];
                                double c4=a4[i+1];
                                double d4=(b4+c4)/2;
                                System.out.println("Your answer for
given question is(Median is):"+d4);
                        }
                        else
                        {
                                i=(1+n4)/2;
                                double e4=a4[i];
                                System.out.println("Your answer for
given question is(Median is):"+e4);
                        }
                        break;
                case 2:
                        double a5[]=new double[50];
                        double b5[]=new double[50];
                        double c5[]=new double[50];
                        System.out.println("\nHOW MANY
NUMBERS ARE THERE TO CALCULATE MEDIAN");
                        int n5=Integer.parseInt(br.readLine());
                        System.out.println("Enter the numbers");
                        for(i=1;i<=n5;i++)
                        {

a5[i]=Double.parseDouble(br.readLine());
```

```
                                        }
                                        System.out.println("Enter frequency
continuosly by pressing enter");
                                        for(i=1;i<=n5;i++)
                                        {

b5[i]=Double.parseDouble(br.readLine());

                                        }
                                        for(i=1;i<=n5;i++)
                                        {
                                                c5[i+1]=c5[i]+b5[i];
                                        }
                                        double top=n5;
                                        double N=c5[n5+1];
                                        double HF=N/2;

System.out.println("_____");
                                        System.out.println("|  X  |  F  | C.F|");
                                        System.out.println("******************");
                                        for(i=1;i<=n5;i++)
                                        {
                                                System.out.println("| "+a5[i]+"|
"+b5[i]+"| "+c5[i+1]+"|");
                                        }
                                        System.out.println("******************");
                                        System.out.println("    N=| "+N+"|");
                                        System.out.println("      ********");
                                        for(i=2;i<=top;i++)
                                        {
                                                if(HF<=c5[i])
                                                {
                                                        System.out.println("THE
MEDIAN CLASS FOR THE GIVEN QUESTION IS:"+c5[i]);
                                                        System.out.println("THE
MEDIAN FOR THE GIVEN QUESTION IS:"+a5[i-1]);
                                                        top=i;
                                                }
                                        }
                                        break;
                                case 3:
                                        double a6[]=new double[50];
                                        double a61[]=new double[50];
```

```
double b6[]=new double[50];
double c6[]=new double[50];
double d6;
double e6;
double cl2;
int top1;
double j6;
double g6=0;
double h6=0,i6=0;
System.out.println("\nHOW MANY
NUMBERS ARE THERE TO CALCULATE MEDIAN");
int n6=Integer.parseInt(br.readLine());
for(i=1,j=1;i<=n6;i++,j++)
{
        System.out.print(i+".)Enter the lower
limit\n");

a6[i]=Double.parseDouble(br.readLine());
        System.out.print(j+".)Enter the upper
limit\n");

a61[j]=Double.parseDouble(br.readLine());
}
System.out.println("Enter the frequency
continuosly by pressing enter one after another");
for(i=1;i<=n6;i++)
{

b6[i]=Double.parseDouble(br.readLine());


}
c6[0]=0;
for(i=1;i<=n6;i++)
{
        c6[i]=c6[i-1]+b6[i];
}
d6=c6[n6];
e6=d6/2;
top1=n6;
for(i=1;i<=top1;i++)
{
        if(e6<=c6[i])
        {
                g6=c6[i-1];
```

```
                                                h6=b6[i];
                                                top1=i;
                                        }
                                }
                                if(a61[1]==a6[2])
                                {
                                        i6=a6[top1];
                                        cl2=a61[1]-a6[1];
                                }
                                else
                                {
                                        i6=a6[top1]-0.5;
                                        cl2=a61[1]-a6[1]+1;
                                }

System.out.println("\nL="+i6+"\nN/2="+e6+"\nC.F="+g6+"\nCL="+cl2+"\nF="+h6);

System.out.println("_____");
                                System.out.println("|    X    |  F  | C.F |");
                                System.out.println("************************");
                                for(i=1;i<=n6;i++)
                                {

System.out.println("|"+a6[i]+"-"+a61[i]+"| "+b6[i]+" | "+c6[i]+" |");
                                }
                                System.out.println("************************");
                                System.out.println("          | "+d6+" |");
                                System.out.println("          ********");
                                j6=i6+(((e6-g6)*cl2)/h6);
                                System.out.println(" "+j6);
                                break;
                        }
                        break;
                case 3:
                        System.out.println("\n1.RAW DATA\n2.DISCRETE
DATA\n3.CONTINUOS DATA");
                        System.out.print("\nEnter your choice again:");
                        int c=Integer.parseInt(br.readLine());
                        switch(c)
                        {
                                case 1:
                                        int a7[]=new int[50];
                                        int b7[]=new int[50];
                                        int f7=0;
```

```
                                        System.out.println("\nHOW MANY
NUMBERS ARE THERE TO CALCULATE MODE");
                                        int n7=Integer.parseInt(br.readLine());
                                        System.out.println("\nEnter the no.s
continuosly to find mode");
                                        for(i=1;i<=n7;i++)
                                        {
                                                a7[i]=Integer.parseInt(br.readLine());
                                        }
                                        for(i=1;i<=n7;i++)
                                        {
                                                b7[i]=1;
                                                for(j=i+1;j<=n7;j++)
                                                {
                                                        if(a7[i]==a7[j])
                                                        {
                                                                b7[i]=b7[i]+1;
                                                        }
                                                }

                                        }
                                        int max=b7[1];
                                        for(i=2;i<=n7;i++)
                                        {
                                                if(b7[i]>max)
                                                {
                                                        max=b7[i];
                                                        f7=i;
                                                }
                                        }
                                        System.out.println("\tBy inspection method i
come to found\n that("+a7[f7]+")repeats "+b7[f7]+" times in above
data.\n\tHence,MODE:"+a7[f7]);
                                        break;
                                case 2:
                                        int a8[]=new int[50];
                                        int b8[]=new int[50];
                                        int f8=0;

                                        System.out.println("\nHOW MANY
NUMBERS ARE THERE TO CALCULATE MODE");
                                        int n8=Integer.parseInt(br.readLine());
                                        System.out.println("Enter the numbers to
find mode(X)");
```

```
for(i=1;i<=n8;i++)
{
	a8[i]=Integer.parseInt(br.readLine());

}
System.out.println("Enter frequency
continuosly by pressing enter");
for(i=1;i<=n8;i++)
{
	b8[i]=Integer.parseInt(br.readLine());

}
int max1=b8[1];
for(i=2;i<=n8;i++)
{
	if(b8[i]>max1)
	{
		max1=b8[i];
		f8=i;
	}
}
System.out.println("\tBy inspection method i
come to found\n that "+b8[f8]+" is the largest frequency.\n\tMODAL CLASS:
"+b8[f8]+"\n\tMODE: "+a8[f8]);
		break;
	case 3:
		double a9[]=new double[50];
		double a91[]=new double[50];
		double b9[]=new double[50];
		int c9=0;
		double d9;
		double e9;
		double cl3;
		double f9;
		double g9;
		double h9;
		System.out.println("\nHOW MANY
NUMBERS ARE THERE TO CALCULATE MODE");
		int n9=Integer.parseInt(br.readLine());
		for(i=1,j=1;i<=n9;i++,j++)
		{
			System.out.print(i+".)Enter the lower
limit\n");
```

```
a9[i]=Double.parseDouble(br.readLine());
                                        System.out.print(j+".)Enter the upper
limit\n");
a91[j]=Double.parseDouble(br.readLine());
                                    }
                                    System.out.println("Enter the frequency
continuosly by pressing enter one after another");
                                    for(i=1;i<=n9;i++)
                                    {
b9[i]=Double.parseDouble(br.readLine());
                                    }
                                    double max2=b9[1];
                                    for(i=1;i<=n9;i++)
                                    {
                                        if(b9[i]>=max2)
                                        {
                                            max2=b9[i];
                                            c9=i;
                                        }
                                    }
                                    d9=b9[c9];
                                    e9=b9[c9+1];
                                    f9=b9[c9-1];
                                    if(a9[2]==a91[1])
                                    {
                                        g9=a9[c9];
                                        cl3=a91[1]-a9[1];
                                    }
                                    else
                                    {
                                        g9=a9[c9]-0.5;
                                        cl3=a91[1]-a9[1]+1;
                                    }
                                    System.out.println("_________________");
                                    System.out.println("|    X    | F |");
                                    System.out.println("***************");
                                    for(i=1;i<=n9;i++)
                                    {
System.out.println("|"+a9[i]+"-"+a91[i]+"|"+b9[i]+"|");
                                    }
```

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

System.out.println("L="+g9+"\nf0="+f9+"\nf1="+d9+"\nf2="+e9+"\ncl="+cl3);
                                        h9=g9+(((d9-f9)/(2*(d9)-f9-e9))*cl3);
                                        System.out.println("MODE="+h9);
                                        break;
                                }
                                break;
                        case 4:
                                int aj;
                                int aa[]=new int[200];
                                float ab[]=new float[200];
                                float ah[]=new float[200];
                                float ac=0,ad=1;
                                float ae=0,ae1=0,af;
                                float ag[]=new float[200];
                                int ai=200;
                                do
                                {
                                        System.out.println("BISECTION METHOD");
                                        System.out.println("*****************");
                                        System.out.println("WHICH IS THE HIGHEST
POWER IN YOUR EQN:");
                                        int nn=Integer.parseInt(br.readLine());
                                        System.out.println("\nJUST ENTER THE
COEFFICIENT NOW");
                                        for(i=nn;i>=1;i--)
                                        {
                                                System.out.print("Enter the coefficient of the
term whose degree is "+i+" :");
                                                aa[i]=Integer.parseInt(br.readLine());
                                                System.out.print("\n");
                                        }
                                        System.out.println("Now enter the constant term");
                                        aa[0]=Integer.parseInt(br.readLine());
                                        System.out.println("\nI think this is your eqn");
                                        System.out.print("("+aa[nn]+"X)^"+nn);
                                        for(i=nn-1;i>=2;i--)
                                        {
                                                System.out.print("+("+aa[i]+"X)^"+i);
                                        }
                                        System.out.print("+("+aa[1]+"X)");
                                        System.out.println("+("+aa[0]+")=0");
```

```
                                        System.out.println("AM I RIGHT\n(if right means
click 1 otherwise 2)");
                                        aj=Integer.parseInt(br.readLine());
                                        switch(aj)
                                        {
                                                case 1:
                                                        {
                                                                System.out.println("\nOn
substituting the values for X:\nI got:");
                                                                j=0;
                                                                ac=0;
                                                                for(i=nn;i>=1;i--)
                                                                {
                                                                        ad=1;
                                                                        for(int k=1;k<=i;k++)
                                                                        {
                                                                                ad*=j;
                                                                        }
                                                                        ac=ac+((aa[i])*(ad));
                                                                }
                                                                ab[j]=ac+(aa[0]);

System.out.println("f("+j+")="+ab[j]);
                                                                int l=100;
                                                                if(ab[0]<0)
                                                                {
                                                                        for(j=1;j<=l;j++)
                                                                        {
                                                                                ac=0;


for(i=nn;i>=1;i--)
                                                                                {
                                                                                        ad=1;
                                                                                        for(int
k=1;k<=i;k++)
                                                                                        {

ad*=j;
                                                                                        }

ac=ac+((aa[i])*(ad));
                                                                                }

ab[j]=ac+(aa[0]);
```

```
System.out.println("f("+j+")="+ab[j]);
                                                        if(ab[j]>0)
                                                        {
                                                                l=j;
                                                                ae1=l;

ae=ae1-1;
                                                        }
                                                }
                                        }
                                        if(ab[0]>0)
                                        {
                                                for(j=1;j<=l;j++)
                                                {
                                                        ac=0;

for(i=nn;i>=1;i--)
                                                        {
                                                                ad=1;
                                                                for(int
k=1;k<=i;k++)
                                                                {

ad*=j;
                                                                }

ac=ac+((aa[i])*(ad));
                                                        }

ab[j]=ac+(aa[0]);

System.out.println("f("+j+")="+ab[j]);
                                                        if(ab[j]<0)
                                                        {
                                                                l=j;
                                                                ae=l;

ae1=ae-1;
                                                        }
                                                }
                                        }
                                        af=(ae+ae1)/2;
```

```
                                                        System.out.println("Root lies
between "+ae+" & "+ae1);
                                                        for(i=1;i<=ai;i++)
                                                        {
                                                                System.out.println(i+"
APPROXIMATION:");
System.out.println("******************");
System.out.println("\ta="+ae+" , b="+ae1);
System.out.println("\tX"+i+"=(a+b)/2");
                                                                System.out.println("\t
=("+ae+"+"+ae1+")/2");
System.out.println("\tX"+i+"="+af);
System.out.print("\tf(X"+i+")=");
System.out.print(aa[nn]+"(X"+i+")^"+nn);
                                                                for(j=nn-1;j>=2;j--)
                                                                {
System.out.print("+("+aa[j]+"(X"+i+")^"+j+")");
                                                                }
System.out.print("+("+aa[1]+"X"+i+")");
System.out.println("+("+aa[0]+")");
System.out.print("\tf("+af+")=");
System.out.print(aa[nn]+"("+af+")^"+nn);
                                                                for(j=nn-1;j>=2;j--)
                                                                {
System.out.print("+("+aa[j]+"("+af+")^"+j+")");
                                                                }
System.out.print("+("+aa[1]+"("+af+"))");
System.out.println("+("+aa[0]+")");
                                                                ac=0;
                                                                for(j=nn;j>=1;j--)
```

```
                                                                {
                                                                        ad=1;
                                                                        for(int
k=1;k<=j;k++)
                                                                        {
                                                                                ad*=af;
                                                                        }

ac=ac+((aa[j])*(ad));
                                                                }
                                                                ag[i]=ac+(aa[0]);

System.out.println("\tf("+af+")="+ag[i]);
                                                                if(ag[i]<0)
                                                                {
                                                                        ae=af;
                                                                }
                                                                if(ag[i]>0)
                                                                {
                                                                        ae1=af;
                                                                }
                                                                af=(ae+ae1)/2;

System.out.println("Root lies between "+ae+" & "+ae1);
                                                                ah[i]=af;
                                                                if(ah[i]==ah[i-1])
                                                                {
                                                                        ai=i;
                                                                }
                                                        }
                                                        System.out.println("Therefore
by "+i+"approximation process i come to find the\npositive root of the
equation.\n\t\tROOT="+af);
                                                                break;
                                                        }
                                                case 2:
                                                        break;
                                                }
                                        }while(aj>=2);
                                        break;
                                case 5:
                                        int bj;
                                        int ba[]=new int[200];
                                        float bb[]=new float[200];
```

```
float bh[]=new float[200];
float bc=0,bd=1,bbn=0,bbp=0;
float be=0,be1=0,bf;
float bg[]=new float[200];
int bi=200;
do
{
	System.out.println("\tREGULAR FALSI METHOD");
	System.out.println("\t*********************");
	System.out.println("WHICH IS THE HIGHEST
POWER IN YOUR EQN:");
	int nn=Integer.parseInt(br.readLine());
	System.out.println("\nJUST ENTER THE
COEFFICIENT NOW");
	for(i=nn;i>=1;i--)
	{
		System.out.print("Enter the coefficient of the
term whose degree is "+i+" :");
		ba[i]=Integer.parseInt(br.readLine());
		System.out.print("\n");
	}
	System.out.println("Now enter the constant term");
	ba[0]=Integer.parseInt(br.readLine());
	System.out.println("\nI think this is your eqn");
	System.out.print("("+ba[nn]+"X)^"+nn);
	for(i=nn-1;i>=2;i--)
	{
		System.out.print("+("+ba[i]+"X)^"+i);
	}
	System.out.print("+("+ba[1]+"X)");
	System.out.println("+("+ba[0]+")=0");
	System.out.println("AM I RIGHT\n(if right means
click 1 otherwise 2)");
	bj=Integer.parseInt(br.readLine());
	switch(bj)
	{
		case 1:
			{
				System.out.println("\nOn
substituting the values for X:\nI got:");
				j=0;
				bc=0;
				for(i=nn;i>=1;i--)
				{
```

```
                bd=1;
                for(int k=1;k<=i;k++)
                {
                        bd*=j;
                }
                bc=bc+((ba[i])*(bd));
        }
        bb[j]=bc+(ba[0]);

System.out.println("f("+j+")="+bb[j]);

        int l=100;
        if(bb[0]<0)
        {
                for(j=1;j<=l;j++)
                {
                        bc=0;

for(i=nn;i>=1;i--)

                        {
                                bd=1;
                                for(int

k=1;k<=i;k++)

                                {

bd*=j;

                                }

bc=bc+((ba[i])*(bd));

                        }

bb[j]=bc+(ba[0]);

System.out.println("f("+j+")="+bb[j]);

                        if(bb[j]>0)
                        {
                                l=j;
                                be1=l;

be=be1-1;

                        }
                }
        }
        if(bb[0]>0)
        {
```

```
for(i=nn;i>=1;i--)

k=1;k<=i;k++)

bd*=j;

bc=bc+((ba[i])*(bd));

bb[j]=bc+(ba[0]);

System.out.println("f("+j+")="+bb[j]);

be=be1-1;

bf=((be*bb[l])-(be1*bb[l-1]))/(bb[l]-bb[l-1]);

between "+be+" & "+be1);

APPROXIMATION:");

System.out.println("******************");

System.out.println("\ta="+be+" , b="+be1);
```

```
for(j=1;j<=l;j++)
{
        bc=0;

        {
                bd=1;
                for(int

                {

                }

        }

        if(bb[j]<0)
        {
                l=j;
                be1=l;

        }
}
}

System.out.println("Root lies

bbn=bb[l-1];
bbp=bb[l];
for(i=1;i<=bi;i++)
{
        System.out.println(i+"
```

```
System.out.println("\tX"+i+"=af(b)-bf(a)/f(b)-f(a)");
                                                        System.out.println("\t
="+be+"("+bbp+")-"+be1+"("+bbn+")/("+bbp+"-("+bbn+"))");
System.out.println("\tX"+i+"="+bf);
System.out.print("\tf(X"+i+")=");
System.out.print(ba[nn]+"(X"+i+")^"+nn);
                                                        for(j=nn-1;j>=2;j--)
                                                        {
System.out.print("+("+ba[j]+"(X"+i+")^"+j+")");
                                                        }
System.out.print("+("+ba[1]+"X"+i+")");
System.out.println("+("+ba[0]+")");
System.out.print("\tf("+bf+")=");
System.out.print(ba[nn]+"("+bf+")^"+nn);
                                                        for(j=nn-1;j>=2;j--)
                                                        {
System.out.print("+("+ba[j]+"("+bf+")^"+j+")");
                                                        }
System.out.print("+("+ba[1]+"("+bf+"))");
System.out.println("+("+ba[0]+")");
                                                        bc=0;
                                                        for(j=nn;j>=1;j--)
                                                        {
                                                                bd=1;
                                                                for(int
k=1;k<=j;k++)
                                                                {
                                                                        bd*=bf;
                                                                }
bc=bc+((ba[j])*(bd));
                                                        }
```

```
                                                        bg[i]=bc+(ba[0]);

System.out.println("\tf("+bf+")="+bg[i]);
                                                        if(bg[i]<0)
                                                        {
                                                                bbn=bg[i];
                                                                be=bf;
                                                        }
                                                        if(bg[i]>0)
                                                        {
                                                                bbp=bg[i];
                                                                be1=bf;
                                                        }

bf=((be*bbp)-(be1*bbn))/(bbp-bbn);

System.out.println("Root lies between "+be+" & "+be1);
                                                        bh[i]=bf;
                                                        if(bh[i]==bh[i-1])
                                                        {
                                                                bi=i;
                                                        }
                                                }
                                                int bk=i-1;
                                                System.out.println("Therefore
by "+bk+" approximation process i come to find the\npositive root of the
equation.\n\t\tROOT="+bf);
                                                        break;
                                                }
                                        case 2:
                                                break;
                                }
                        }while(bj>=2);
                        break;
                case 6:
                        int c0[][]=new int[300][300];
                        int c1[][]=new int[300][300];
                        int c2[]=new int[300];
                        float ca=0,cb=0,cc=0,cd=100,caa,cba,cca;
                        int ce=1;
                        System.out.println("\tGAUSS-JACOBI METHOD");
                        System.out.println("\t*******************");
                        System.out.println("The equation form is in:");
```

```
System.out.println("\ta11x1+a12x2+a13x3=b1\n\ta21x1+a22x2+a23x3=b2\n\ta31x1+a32x2+a33
x3=b3");
                                        System.out.println("From this equation form i come to find
that you \nhave to enter a11,a12,....a33 and b1,b2,b3.");
                                        do
                                        {
                                    for(i=1;i<=3;i++)
                                        {
                                        for(j=1;j<=3;j++)
                                         {
                                                System.out.print("Enter a"+i+j+":");
                                                c1[i][j]=Integer.parseInt(br.readLine());
                                                if(c1[i][j]<0)
                                                {
                                                c0[i][j]=c1[i][j]-(2*c1[i][j]);
                                                }
                                                else
                                                {
                                                        c0[i][j]=c1[i][j];
                                                }
                                         }
                                                System.out.print("Enter b"+i+":");
                                                c2[i]=Integer.parseInt(br.readLine());
                                      }
                                for(i=1;i<=3;i++)
                                     {

System.out.println(c1[i][1]+"x+("+c1[i][2]+")y+("+c1[i][3]+")z="+c2[i]);
                                        }
                                        System.out.print("Is this your equation\n\tAm i right
??\nEnter 1 if right:\n      2 if wrong:");
                                        ce=Integer.parseInt(br.readLine());
                                        switch(ce)
                                        {
                                                case 1:

if(c0[1][1]>=c0[1][2]+c0[1][3]&&c0[2][2]>=c0[2][1]+c0[2][3]&&c0[3][3]>=c0[3][1]+c0[3][2])
                                                                {
                                                                        System.out.println("since the
elements are diagonally\n\tdominant.\nLet us solve by Gauss-jacobi method:");
                                                                        System.out.println();

System.out.println("X=1/"+c1[1][1]+"["+c2[1]+"-("+c1[1][2]+")Y-("+c1[1][3]+")Z]");
```

```
System.out.println("Y=1/"+c1[2][2]+"["+c2[2]+"-("+c1[2][1]+")X-("+c1[2][3]+")Z]");

System.out.println("Z=1/"+c1[3][3]+"["+c2[1]+"-("+c1[3][1]+")X-("+c1[3][2]+")Y]");
                                                        System.out.println("Let the initial
value be (0,0,0):");
                                                        for(i=1;i<=cd;i++)
                                                        {
                                                                System.out.println(i+".
ITERATION:");

System.out.println("**************");

System.out.println("\tX="+ca+",Y="+cb+",Z="+cc);

System.out.println("\tX=1/"+c1[1][1]+"["+c2[1]+"-("+c1[1][2]+")Y-("+c1[1][3]+")Z]");

System.out.println("\tX=1/"+c1[1][1]+"["+c2[1]+"-("+c1[1][2]+")"+cb+"-("+c1[1][3]+")"+cc+"]");
                                                                float
caa1=(c2[1]-((c1[1][2])*(cb))-((c1[1][3])*(cc)));
                                                                caa=caa1/c1[1][1];
                                                        System.out.println("\tX="+caa);
                                                                System.out.println();

System.out.println("\tY=1/"+c1[2][2]+"["+c2[2]+"-("+c1[2][1]+")X-("+c1[2][3]+")Z]");

System.out.println("\tY=1/"+c1[2][2]+"["+c2[2]+"-("+c1[2][1]+")"+ca+"-("+c1[2][3]+")"+cc+"]");
                                                                float
cba1=(c2[2]-((c1[2][1])*(ca))-((c1[2][3])*(cc)));
                                                                cba=cba1/c1[2][2];
                                                        System.out.println("\tY="+cba);
                                                                System.out.println();

System.out.println("\tZ=1/"+c1[3][3]+"["+c2[3]+"-("+c1[3][1]+")X-("+c1[3][2]+")Y]");

System.out.println("\tZ=1/"+c1[3][3]+"["+c2[3]+"-("+c1[3][1]+")"+ca+"-("+c1[3][2]+")"+cb+"]");
                                                                float
cca1=(c2[3]-((c1[3][1])*(ca))-((c1[3][2])*(cb)));
                                                                cca=cca1/c1[3][3];
                                                        System.out.println("\tZ="+cca);
                                        if(ca==caa&&cb==cba&&cc==cca)
                                                                {
                                                                        cd=i;
                                                                }
```

```
                                                ca=caa;
                                                cb=cba;
                                                cc=cca;
                                        }
                                        int cf=i-1;
                                        System.out.println("Finally i got the
correct answer by "+cf+" iteration process.");

System.out.println("\tX="+ca+",Y="+cb+",Z="+cc);
                                }
                                else
                                        System.out.println("The given
elements are not diagonally dominant!!\nSo please enter a valid equation.");
                        }
                        }while(ce>=2);
                break;
        case 7:
                        int d0[][]=new int[300][300];
                        int d1[][]=new int[300][300];
                        int d2[]=new int[300];
                        float
da=0,db=0,dc=0,dd=100,daa=da+1,dba=db+1,dca=dc+1;
                        int de=1;
                        System.out.println("\tGAUSS-SEIDAL METHOD");
                        System.out.println("\t********************");
                        System.out.println("The equation form is in:");

System.out.println("\ta11x1+a12x2+a13x3=b1\n\ta21x1+a22x2+a23x3=b2\n\ta31x1+a32x2+a33
x3=b3");
                        System.out.println("From this equation form i come to find
that you \nhave to enter a11,a12,....a33 and b1,b2,b3.");
                        do
                        {
                                for(i=1;i<=3;i++)
                                {
                                        for(j=1;j<=3;j++)
                                        {
                                                System.out.print("Enter a"+i+j+":");

d1[i][j]=Integer.parseInt(br.readLine());
                                                if(d1[i][j]<0)
                                                {
                                                        d0[i][j]=d1[i][j]-(2*d1[i][j]);
                                                }
```

```
                                        else
                                        {
                                                d0[i][j]=d1[i][j];
                                        }
                                }
                                System.out.print("Enter b"+i+":");
                                d2[i]=Integer.parseInt(br.readLine());
                        }
                        for(i=1;i<=3;i++)
                        {

System.out.println(d1[i][1]+"x+("+d1[i][2]+")y+("+d1[i][3]+")z="+d2[i]);
                        }
                        System.out.print("Is this your equation\n\tAm i right
??\nEnter 1 if right:\n      2 if wrong:");
                        de=Integer.parseInt(br.readLine());
                        switch(de)
                        {
                                case 1:

if(d0[1][1]>=d0[1][2]+d0[1][3]&&d0[2][2]>=d0[2][1]+d0[2][3]&&d0[3][3]>=d0[3][1]+d0[3][2])
                                        {
                                                System.out.println("since the
elements are diagonally\n\tdominant.\nLet us solve by Gauss-jacobi method:");
                                                System.out.println();

System.out.println("X=1/"+d1[1][1]+"["+d2[1]+"-("+d1[1][2]+")Y-("+d1[1][3]+")Z]");

System.out.println("Y=1/"+d1[2][2]+"["+d2[2]+"-("+d1[2][1]+")X-("+d1[2][3]+")Z]");

System.out.println("Z=1/"+d1[3][3]+"["+d2[1]+"-("+d1[3][1]+")X-("+d1[3][2]+")Y]");
                                                System.out.println("Initially
Y=0,Z=0");
                                                for(i=1;i<=dd;i++)
                                                {
                                                        System.out.println(i+".
ITERATION:");

System.out.println("**********");

System.out.println("\tX="+da+",Y="+db+",Z="+dc);

System.out.println("\tX=1/"+d1[1][1]+"["+d2[1]+"-("+d1[1][2]+")Y-("+d1[1][3]+")Z]");
```

```
System.out.println("\tX=1/"+d1[1][1]+"["+d2[1]+"-("+d1[1][2]+")"+db+"-("+d1[1][3]+")"+dc+"]");
                                                                        float
caa1=(d2[1]-((d1[1][2])*(db))-((d1[1][3])*(dc)));
                                                                        da=caa1/d1[1][1];

System.out.println("\tX="+da);
                                                                        System.out.println();

System.out.println("\tY=1/"+d1[2][2]+"["+d2[2]+"-("+d1[2][1]+")X-("+d1[2][3]+")Z]");

System.out.println("\tY=1/"+d1[2][2]+"["+d2[2]+"-("+d1[2][1]+")"+da+"-("+d1[2][3]+")"+dc+"]");
                                                                        float
cba1=(d2[2]-((d1[2][1])*(da))-((d1[2][3])*(dc)));
                                                                        db=cba1/d1[2][2];

System.out.println("\tY="+db);
                                                                        System.out.println();

System.out.println("\tZ=1/"+d1[3][3]+"["+d2[3]+"-("+d1[3][1]+")X-("+d1[3][2]+")Y]");

System.out.println("\tZ=1/"+d1[3][3]+"["+d2[3]+"-("+d1[3][1]+")"+da+"-("+d1[3][2]+")"+db+"]");
                                                                        float
cca1=(d2[3]-((d1[3][1])*(da))-((d1[3][2])*(db)));
                                                                        dc=cca1/d1[3][3];

System.out.println("\tZ="+dc);

if(da==daa&&db==dba&&dc==dca)
                                                                        {
                                                                                dd=i;
                                                                        }
                                                                        daa=da;
                                                                        dba=db;
                                                                        dca=dc;
                                                                }
                                                                int cf=i-1;
                                                                System.out.println("Finally i
got the correct answer by "+cf+" iteration process.");

System.out.println("\tX="+da+",Y="+db+",Z="+dc);
                                                        }
                                                        else
```

```
                                                                System.out.println("The given
elements are not diagonally dominant!!\nSo please enter a valid equation.");
                                                                    }
                                                }while(de>=2);
                                                break;
                                        case 8:
                                                break;
                                    }
                            }while(n<8);
                            }
                            catch(Exception e){}
                    }
}
```