#include<bits/stdc++.h>

#include <algorithm>

#include <time.h>

using namespace std;

int i,j,k,n,itr;

double arr[50][50],arr2[50][50],x[50];

void gaussJordan()

{

double pivot;

for(i=1; i<=n; i++)

{

for(j=1; j<=n; j++)

{

if(i!=j)

{

pivot=arr[j][i]/arr[i][i];

for(k=1; k<=n+1; k++)

{

arr[j][k]=arr[j][k]-pivot\*arr[i][k];

}

}

}

}

cout<<"\nThe solution of linear equations is:\n";

x[1]=arr[1][n+1]/arr[1][1];

cout<<"x: "<<x[1]<<endl;

x[2]=arr[2][n+1]/arr[2][2];

cout<<"y: "<<x[2]<<endl;

x[3]=arr[3][n+1]/arr[3][3];

cout<<"z: "<<x[3]<<endl;

}

void gaussSeidel()

{

double sum,x1=0;

while(1)

{itr++;

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

{

sum=0;

for(j=0; j<n; j++)

{

if(i!=j)

sum+=arr2[i][j]\*x[j];

}

x[i]=(arr2[i][n]-sum)/arr2[i][i];

}

if(fabs(x1-x[0])/x1<0.005)

{

break;

}

x1=x[0];

}

cout<<"The solution of linear equations is:"<<endl;

cout<<"x: "<<x[0]<<endl;

cout<<"y: "<<x[1]<<endl;

cout<<"z: "<<x[2]<<endl;

cout<<"Number of Iterations: "<<endl;

cout<<itr<<endl;

}

void flopCounter(int n)

{

double ans,ans2;

ans= (n\*n\*n)/2;

ans2 = pow(n,4);

printf("\nTotal Number of Flops for Gauss Jordan Method %lf \n",ans );

printf("\nTotal Number of Flops for Gauss Seidel Method %lf \n",ans2 );

}

/\* Observation:

Here from gauss jordan formula we get: 3.5,1,2.5

from gauss seidel formula we get higher precision of 3.49535,0.999023,2.50188

so get more accurate result in seidel method.

\*/

int main()

{

clock\_t start\_time, end\_time;

cout<<"Enter the size of the equations: ";

cin>>n;

cout<<"Enter the elements of Coefficients: "<<endl;

for(i=1; i<=n; i++)

{

for(j=1; j<=(n+1); j++)

{

cin >> arr[i][j];

}

}

cout<<endl;

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

{

for(j=0; j<=n; j++)

{

arr2[i][j] = arr[i+1][j+1];

}

}

cout<<"Starting of Execution Gauss Jordan Method:"<<endl;

start\_time = clock();

gaussJordan();

end\_time = clock();

double jordan\_time\_taken = (double)(end\_time - start\_time) /CLOCKS\_PER\_SEC;

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

x[i]=0;

cout<<"Starting of Execution Gauss Seidel Method:"<<endl;

start\_time = clock();

gaussSeidel();

end\_time = clock();

double sid\_time\_taken = (double)(end\_time - start\_time) /CLOCKS\_PER\_SEC;

printf("\nTime for gaussJordan: %lf sec\n\n",jordan\_time\_taken);

printf("\nTime for seidal: %lf mili-sec\n\n",sid\_time\_taken\*1000);

flopCounter(3);

flopCounter(6);

return(0);

}