```
#include<bits/stdc++.h>
#include<conio.h>
#include<time.h>
using namespace std;

#define SIZE 10
float a[SIZE][SIZE], x[SIZE], ratio;
int i,j,k,n, cont=0,cont2=0;

double sum,x1=1;

void gaussJordan()
{
   for(i=1;i<=n;i++)
   {
      if(a[i][i] == 0.0)
      {
         printf("Mathematical Error!");
         break;
      }
      for(j=1;j<=n;j++)
      {
         if(i!=j)
         {
            ratio = a[j][i]/a[i][i];
            for(k=1;k<=n+1;k++)
            {
               a[j][k] = a[j][k] - ratio*a[i][k];
```

```
            }
        }
    }
    cont++;
}
/* Obtaining Solution */
for(i=1;i<=n;i++)
{
    x[i] = a[i][n+1]/a[i][i];
}

printf("Starting of Execution Gauss Jordan Method:");
printf("\nThe solution of linear equation is:\n");
for(i=1;i<=n;i++)
{
    printf("x[%d] = %0.3f\n",i, x[i]);
}
printf("End of Execution.....\n");

}

void gaussSeidal()
{
    i=0,j=0;
    double x[n];
    //double x[n],sum,x1=1;

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

```
    x[i]=0;
while(1)
{
    for(i=0;i<n;i++)
    {
        sum=0;
        for(j=0;j<n;j++)
        {
            if(i!=j)
                sum+=a[i][j]*x[j];
        }
        x[i]=(a[i][n]-sum)/a[i][i];

        cont2++;
    }
    if(abs(x1-x[0])/x1<0.005)
    {
        break;
    }
    x1=x[0];
}
printf("Starting of Execution Gauss Seidal Method:");
printf("\nThe solution of linear equation is:\n");
for(i=1;i<=n;i++)
{
    printf("x[%d] = %0.3f\n",i, x[i]);
}
printf("End of Execution.....\n");
```

```
}

int main()
{
    printf("Enter the size of the equations: ");
    scanf("%d", &n);

    printf("Enter the elements of coefficients:\n");
    for(i=1;i<=n;i++)
    {
        for(j=1;j<=n+1;j++)
        {
            printf("a[%d][%d] = ",i,j);
            scanf("%f", &a[i][j]);
        }
    }

    clock_t t;
    t = clock();
    gaussJordan();
    t = clock() - t;
    double time_taken = ((double)t)/CLOCKS_PER_SEC;
    printf("\nRunning Time for Gauss Jordan Method = %lf seconds\n", time_taken);
    printf("Number of iterations: %d", cont);

    clock_t t2;
    t2 = clock();
    gaussSeidal();
```

```
    t2 = clock() - t2;
    double time_taken2 = ((double)t2)/CLOCKS_PER_SEC;
    printf("\nRunning Time for Gauss Seidal Method = %lf seconds\n", time_taken2);
    printf("Number of iterations: %d", cont2);


    getch();
    return(0);
}
```

Comparison

Here, we use 2 methods. They are: Gauss Jordan Method & Gauss Seidel Method.

Gauss Seidel needs fewer iterations. But Gauss Jordan needs more iterations than Gauss Seidel. For this reason Gauss Seidel method is better.

Number of Iterations in Gauss Jordan method: 3

Running Time for Gauss Jordan Method: 0.001 seconds