

**ITESO**

**MÉTODOS NUMÉRICOS**

**LÓPEZ LAZARENO DIEGO ALBERTO IF722100**

**PRÁCTICA 05**

El siguiente programa extrae una matriz S de una matriz A.

clear

a=input('ingresa la matriz A');

n=length(a);

S=zeros(n);

for fila=1:n

    for columna=(fila+1):n

        S(fila,columna)=a(fila,columna);

    end

end

S

1. Modifica el programa de manera que extraiga la matriz D, S e I.

% Se ingresa una matriz cuadrada

a=input("Ingrese la matriz");

% Longitud de la matriz

len=length(a);

% Se rellenan 3 matrices (de la longitud de a) con 0's

superior=zeros(len);

diagonal=zeros(len);

inferior=zeros(len);

% Ciclo para obtener la parte superior a la diagonal de la matriz

for i=1:(len-1)

for j=1:len

if j>i

superior(i,j)=a(i,j);

else

continue

end

end

end

superior

% Ciclo para obtener la diagonal de la matriz

for i=1:len

diagonal(i,i)=a(i,i);

end

diagonal

% Ciclo para obtener la parte inferior a la diagonal de la matriz

for i=2:len

for j=1:len

if j<i

inferior(i,j)=a(i,j);

else

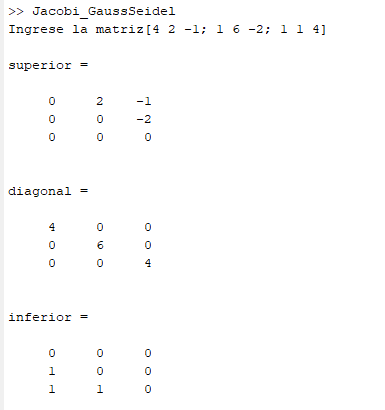
continue

end

end

end

inferior



Los algoritmos escritos en forma matricial para los métodos de Jacobi y de Gauss-Seidel se muestran en la siguiente tabla.





1. Desarrolla un programa que al ingresa la matriz A, el vector R y el vector X de valores iniciales, determine la solución de la ecuación mediante los métodos de Jacobi y de Gauss-Seidel.

% Método de Jacobi y Gauss-Seidel

% Se ingresa una matriz cuadrada

a=input("Ingrese la matriz");

% Longitud de la matriz

len=length(a);

% Se rellenan 3 matrices (de la longitud de a) con 0's

superior=zeros(len);

diagonal=zeros(len);

inferior=zeros(len);

% Inputs necesarios para realizar el Método de Jacobi y Gauss-Seidel

vector\_r=input("Ingrese el vector columna de resultados");

vector\_valores\_iniciales\_j=input("Ingrese el vector columna con valores iniciales");

vector\_valores\_iniciales\_g=vector\_valores\_iniciales\_j;

% Ciclo para obtener la parte superior a la diagonal de la matriz

for i=1:(len-1)

for j=1:len

if j>i

superior(i,j)=a(i,j);

else

continue

end

end

end

superior

% Ciclo para obtener la diagonal de la matriz

for i=1:len

diagonal(i,i)=a(i,i);

end

diagonal

% Ciclo para obtener la parte inferior a la diagonal de la matriz

for i=2:len

for j=1:len

if j<i

inferior(i,j)=a(i,j);

else

continue

end

end

end

inferior

% Método de Jacobi

for i=1:50

jacobi=inv(diagonal)\*(vector\_r-(inferior+superior)\*vector\_valores\_iniciales);

vector\_valores\_iniciales\_j=jacobi;

end

jacobi

% Método de Gauss-Seidel

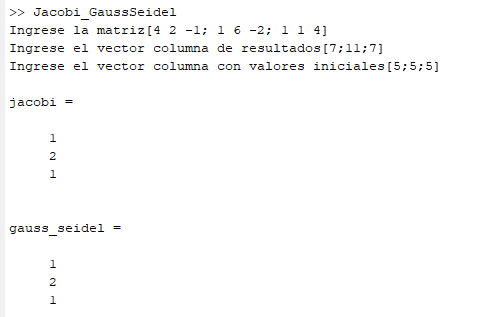
for i=1:50

gauss\_seidel=inv(diagonal+inferior)\*(vector\_r-superior\*vector\_valores\_iniciales);

vector\_valores\_iniciales\_g=gauss\_seidel;

end

gauss\_seidel



**Solución de la ecuación (x=1, y=2, z=1)**

**Conclusión**

En esta práctica de laboratorio se abordó al Método de Jacobi y Gauss-Seidel. Ambos métodos numéricos están adaptados a las operaciones matriciales, lo que nos permite obtener la solución a un sistema de ecuaciones de una forma eficiente. A su vez, Gauss-Seidel converge al vector solución de forma más rápida, pues incorpora, a diferencia de Jacobi, nuevas aproximaciones. Por otra parte, los códigos proporcionados por el profesor fueron difíciles de descifrar para mí, pues cada persona tiene una forma distinta de resolver los problemas. Para resolver lo anterior, opté por elaborar un código propio, desde mi particular forma para resolver los problemas.