

# Proyecto nº4:

## Inicialización

```
clc           % Limpia la pantalla de salida
clf           % Limpia ventana de figura
clear         % Inicializa variables
close         % Cierra cualquier ventana de una ejecución anterior
```

## Apartado a)

Generar un digrafo aleatorio, con un mínimo de 5 nodos y un máximo de 30.

```
n = 8 % randi([5 30]);
```

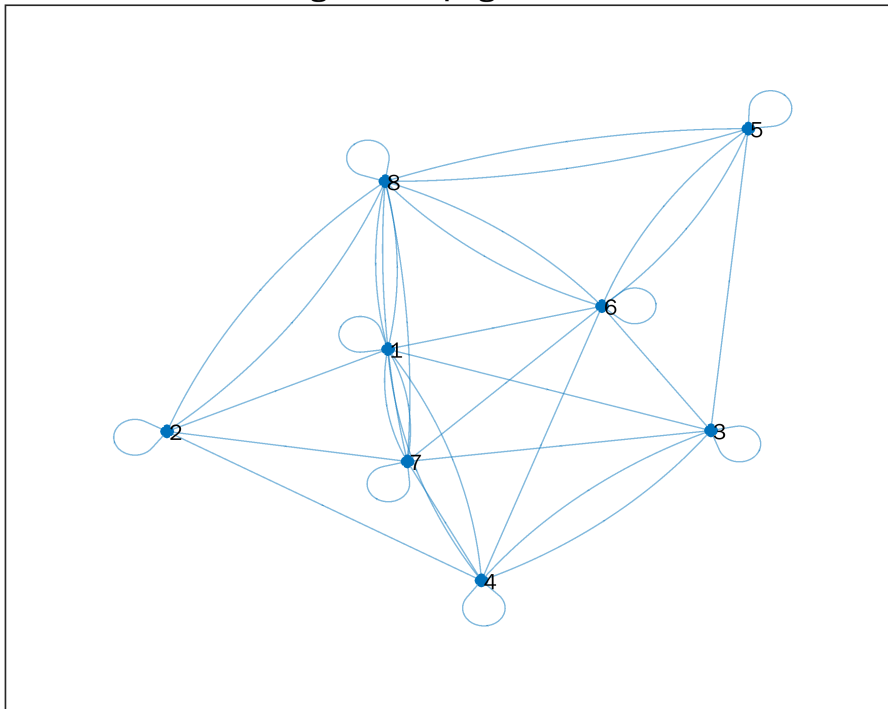
```
n = 8
```

```
A = [];
for i=1:n
    for j=1:n
        if i == j
            A(i,j)=1;
        else
            A(i,j)=randi([0 1]);
        end
    end
end
end
A
```

```
A = 8x8
    1     0     1     1     0     1     1     1
    1     1     0     1     0     0     1     1
    0     0     1     1     1     1     1     0
    1     0     1     1     0     1     1     0
    0     0     0     0     1     1     0     1
    0     0     0     0     1     1     1     1
    1     0     0     0     0     0     1     1
    1     1     0     0     1     1     1     1
```

```
M = digraph(A);
plot(M)
title('Digrafo de páginas web')
```

Digrafo de páginas web



## Apartado b)

Etiquetar con un nombre distinto a cada nodo.

```
letras = 'ABCDEFGH IJKLMNOPQRSTUVWXYZuvwyz';
r = randperm(30,n);

for i=1:n
    %Nombre aleatorio asociado a una letra:
    y = string(letras(r(i)));
    sw(i) = strcat('www.x',y,'x.com');

    %Etiquetamos los nodos con su número de página
    x = string({i});
    sn(i) = strcat('web', x);
end
sw
```

```
sw = 1x8 string
"www.xQx.com" "www.xJx.com" "www.xYx.com" "www.xLx.com" "www.xOx.com" "www.xGx.com" "w * * *
```

```
sn
```

```
sn = 1x8 string
"web1"      "web2"      "web3"      "web4"      "web5"      "web6"      "w * * *
```

### Apartado c)

Matriz de adyacencia del digrafo:

A

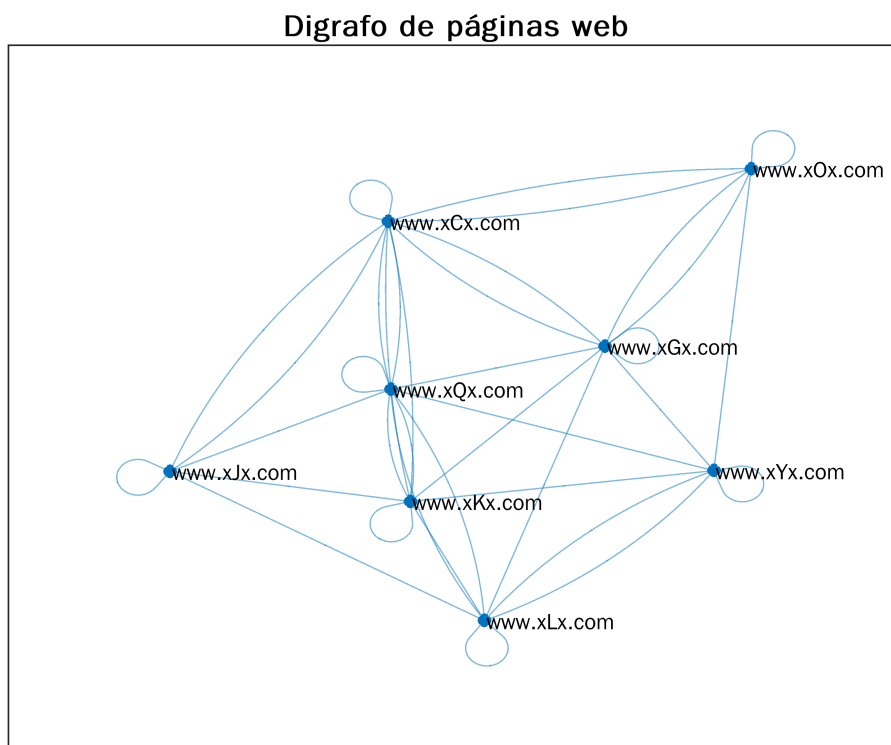
A = 8x8

1	0	1	1	0	1	1	1
1	1	0	1	0	0	1	1
0	0	1	1	1	1	1	0
1	0	1	1	0	1	1	0
0	0	0	0	1	1	0	1
0	0	0	0	1	1	1	1
1	0	0	0	0	0	1	1
1	1	0	0	1	1	1	1

### Apartado d)

Representación del digrafo creado.

```
M = digraph(A,sw);  
plot(M)  
title('Digrafo de páginas web')
```



### Apartado e)

Transformación de la matriz de adyacencia (A), en una matriz de transición (B).

```
% Suma de los elementos de cada fila / página
```

```

C = [];
suma = 0;
for i = 1:n
    for j = 1:n
        suma = suma + A(i, j);
    end
    C(i,:) = suma;
    suma = 0;
end
C

```

```

C = 8x1
     6
     5
     5
     5
     3
     4
     3
     6

```

```

% Suma de los elementos de cada columna (menciones)
D = [];
suma = 0;
for i = 1:n
    for j = 1:n
        suma = suma + A(j, i);
    end
    D(i,:) = suma;
    suma = 0;
end
D

```

```

D = 8x1
     5
     2
     3
     4
     4
     6
     7
     6

```

```

%Matriz de transición
B = [];
for i = 1:n
    for j = 1:n
        if A(i,j) == 0
            B(i,j) = 0.15/n;
        else
            B(i,j) = 0.85/C(i,:)+ 0.15/n;
        end
    end
end
end

```

```
end
```

Para comprobar que la matriz de transición es correcta, nos aseguramos de que las probabilidades de cada fila sumen 1 respectivamente. El vector  $v$  deberá estar formado únicamente por 1.

```
for i = 1:n
    suma = 0;
    for j = 1:n
        suma = B(i, j) + suma;
    end
    v(i) = suma;
end
v
```

```
v = 1x8
    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000
```

Si la suma de todos los elementos del vector  $v$  es igual a  $n$ , la matriz es correcta.

```
suma = 0;
for m = 1:n
    suma = suma + v(m);
end
sumal = uint8(suma)
```

```
sumal = uint8
```

```
8
```

Además, debería tener 1 como autovalor principal. (?)

```
e = eig(B);
e1 = uint8(e);
if sumal == n && e1(1) == 1
    fprintf('Como el autovalor principal de la matriz es 1 y cada fila suma 1, la matriz es correcta.\n');
end
```

Como el autovalor principal de la matriz es 1 y cada fila suma 1, la matriz de transición es correcta

Se cumplen los puntos anteriores, entonces la matriz de transición que hemos calculado es:

```
B
```

```
B = 8x8
    0.1604    0.0187    0.1604    0.1604    0.0187    0.1604    0.1604    0.1604
    0.1887    0.1887    0.0187    0.1887    0.0187    0.0187    0.1887    0.1887
    0.0187    0.0187    0.1887    0.1887    0.1887    0.1887    0.1887    0.0187
    0.1887    0.0187    0.1887    0.1887    0.0187    0.1887    0.1887    0.0187
    0.0187    0.0187    0.0187    0.0187    0.3021    0.3021    0.0187    0.3021
    0.0187    0.0187    0.0187    0.0187    0.2312    0.2312    0.2312    0.2312
    0.3021    0.0187    0.0187    0.0187    0.0187    0.0187    0.3021    0.3021
    0.1604    0.1604    0.0187    0.0187    0.1604    0.1604    0.1604    0.1604
```

**Apartado f)**

Planteamiento de la ecuación de relevancia por páginas.

```
% Como la matriz B tiene autovalor 1, x*(B - 1*I) = vector cero.
x = sym('x', [1 n]);
Y = x*B == x; % Calculamos x, autovector asociado al autovalor 1
```

### Apartado g)

Resolución de la ecuación.

```
assume(x(1,1) == 0.3);
sol = solve(Y,x)

sol = struct with fields:
  x1: 3/10
  x2: 1981775970550/16479764452993
  x3: 8575392998313/65919057811972
  x4: 9923000658287/65919057811972
  x5: 4458403101612/16479764452993
  x6: 27902603094207/82398822264965
  x7: 6494244169830/16479764452993
  x8: 69712766515479/164797644529930
```

### Apartado h)

Presentación correcta de los resultados de la resolución de la ecuación. (?)

```
E = struct2array(sol); % Convertimos la estructura en un array de soluciones
```

En caso de haber empate: (?)

```
for i=1:n
if D(i) == D(i+1)
disp('empate')
end
end
```

### Apartado i)

Ordenación de los resultados.

```
[E index] = sort(E, 'descend');
sw = sw(index)
```

```
sw = 1x8 string
"www.xCx.com" "www.xKx.com" "www.xGx.com" "www.xQx.com" "www.xOx.com" "www.xLx.com" "w * * *
```

```
D = D(index)
```

```
D = 8x1
6
```

7  
6  
5  
4  
4  
3  
2

Presentación final mediante una tabla.

```
VarTypes = {'string', 'double', 'double', 'double'};  
VarNames = {'Dirección URL', 'Menciones', 'Nº de página', 'Posición'};  
sz = [n 4];  
T1 = table('size',sz,'VariableTypes',VarTypes, 'VariableNames', VarNames, 'RowNames', s
```

```
for j=1:n  
    i=1;  
    T1(j,i) = {sw(j)};  
    i=2;  
    T1(j,i) = {D(j)};  
end  
  
% En cada página hay k-1 nº de webs  
k=4
```

```
k = 4
```

```
m=1
```

```
m = 1
```

```
pos=1
```

```
pos = 1
```

```
for j=1:n  
    i = 3  
    if j/k == 1  
        m = m+1;  
        T1(j,i) = {m};  
        i=4;  
        pos=1;  
        T1(j,i) = {pos};  
    else  
        T1(j,i) = {m};  
        i=4;  
        T1(j,i) = {pos};  
        pos=pos+1;  
    end
```

end

i = 3  
i = 3  
i = 3  
i = 3  
i = 3  
i = 3  
i = 3  
i = 3

T1

T1 = 8×4 table

	Dirección URL	Menciones	Nº de página	Posición
1 web1	"www.xCx.com"	6	1	1
2 web2	"www.xKx.com"	7	1	2
3 web3	"www.xGx.com"	6	1	3
4 web4	"www.xQx.com"	5	2	1
5 web5	"www.xOx.com"	4	2	1
6 web6	"www.xLx.com"	4	2	2
7 web7	"www.xYx.com"	3	2	3
8 web8	"www.xJx.com"	2	2	4