

Ejercicio 12.1

Jose Antonio Lorencio Abril

Introducimos la matriz de adyacencias:

```
A = [0, 1, 1, 1, 2, 1.5, 0.5, 1.5;  
     1, 0, 1.5, 1.5, 1, 1.5, 0.5, 1.5;  
     1, 0.5, 0, 1.5, 1, 1, 1.5, 1.5;  
     1, 0.5, 0.5, 0, 1, 1, 2, 2;  
     0, 1, 1, 1, 0, 1, 1, 1.5;  
     0.5, 0.5, 1, 1, 1, 0, 1.5, 1;  
     1.5, 1.5, 0.5, 0, 1, 0.5, 0, 1;  
     0.5, 0.5, 0.5, 0, 0.5, 1, 1, 0]
```

```
A = 8x8  
      0      1.0000      1.0000      1.0000      2.0000      1.5000      0.5000      1.5000  
      1.0000      0      1.5000      1.5000      1.0000      1.5000      0.5000      1.5000  
      1.0000      0.5000      0      1.5000      1.0000      1.0000      1.5000      1.5000  
      1.0000      0.5000      0.5000      0      1.0000      1.0000      2.0000      2.0000  
      0      1.0000      1.0000      1.0000      0      1.0000      1.0000      1.5000  
      0.5000      0.5000      1.0000      1.0000      1.0000      0      1.5000      1.0000  
      1.5000      1.5000      0.5000      0      1.0000      0.5000      0      1.0000  
      0.5000      0.5000      0.5000      0      0.5000      1.0000      1.0000      0
```

```
m = length(A)
```

```
m = 8
```

Promediamos los vectores columna (notemos que en esta ocasión no hay ninguna columna totalmente a 0, por lo que no :

```
M = zeros(m);  
for j = 1 : m  
    if sum(A(:, j)) > 0  
        M(:, j) = A(:, j)/sum(A(:, j));  
    else  
        M(:, j) = 1/m;  
    end  
end  
disp("La matriz M promediada es:")
```

La matriz M promediada es:

```
disp(M)
```

0	0.1818	0.1667	0.1667	0.2667	0.2000	0.0625	0.1500
0.1818	0	0.2500	0.2500	0.1333	0.2000	0.0625	0.1500
0.1818	0.0909	0	0.2500	0.1333	0.1333	0.1875	0.1500
0.1818	0.0909	0.0833	0	0.1333	0.1333	0.2500	0.2000
0	0.1818	0.1667	0.1667	0	0.1333	0.1250	0.1500
0.0909	0.0909	0.1667	0.1667	0.1333	0	0.1875	0.1000
0.2727	0.2727	0.0833	0	0.1333	0.0667	0	0.1000
0.0909	0.0909	0.0833	0	0.0667	0.1333	0.1250	0

Y obtenemos la matriz de Google poniendo a 0 el factor sigma de aleatoriedad, tal y como dice el enunciado:

```
C = ones(m)/m;
sigma = 1;
G = sigma*M + (1 - sigma)*C;
disp("La matriz G de Google es:")
```

La matriz G de Google es:

```
disp(G)
```

0	0.1818	0.1667	0.1667	0.2667	0.2000	0.0625	0.1500
0.1818	0	0.2500	0.2500	0.1333	0.2000	0.0625	0.1500
0.1818	0.0909	0	0.2500	0.1333	0.1333	0.1875	0.1500
0.1818	0.0909	0.0833	0	0.1333	0.1333	0.2500	0.2000
0	0.1818	0.1667	0.1667	0	0.1333	0.1250	0.1500
0.0909	0.0909	0.1667	0.1667	0.1333	0	0.1875	0.1000
0.2727	0.2727	0.0833	0	0.1333	0.0667	0	0.1000
0.0909	0.0909	0.0833	0	0.0667	0.1333	0.1250	0

Calculamos el vector de importancias a partir de un vector unitario cualquiera y el número de iteraciones necesarias para hacerlo:

```
w = rand(m, 1);
w = w/sum(w);
epsilon = 10^(-6);
iteraciones = 0;
while sum(abs(G*w - w)) > epsilon
    w = G*w;
    iteraciones = iteraciones + 1;
end
w = transpose(w);
disp("El vector w de importancias es:")
```

El vector w de importancias es:

```
disp(w)
```

0.1459	0.1518	0.1386	0.1294	0.1151	0.1180	0.1237	0.0775
--------	--------	--------	--------	--------	--------	--------	--------

```
fprintf("Iteraciones necesarias para obtenerlo: %u\n\n", iteraciones)
```

Iteraciones necesarias para obtenerlo: 12

Escribimos la ordenación obtenida:

```
%format shortG
[importancia, jugador] = sort(w, 'descend');
disp("La ordenación obtenida es:")
```

La ordenación obtenida es:

```
jugadores = ["Carlsen", "Kramnik", "Svindler", "Aronian", "Gelfand", "Grischuk", "Ivanchuk", "Fischer"]
```

```
jugadores = 1x8 string
"Carlsen"    "Kramnik"    "Svindler"    "Aronian"    "Gelfand"    "Grischuk"    ...
```

```
disp([jugadores(jugador); importancia])
```

"Kramnik"	"Carlsen"	"Svindler"	"Aronian"	"Ivanchuk"	"Grischuk"	"Gelfand"	"Radjabov"
"0.15182"	"0.14591"	"0.13858"	"0.12939"	"0.12371"	"0.11802"	"0.11509"	"0.077487"

Según esta forma de establecer los desempates el ganador hubiera sido Kramnik. Además, vemos como efectivamente Ivanchuk sale bastante beneficiado, escalando dos posiciones en la clasificación.