

Riley Payung

CDS 292

04/03/2020

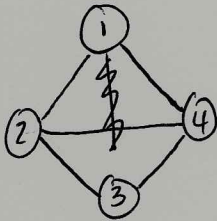
Homework 7

Imports

In [23]:

```
import networkx as nx
import numpy as np
```

Question 2



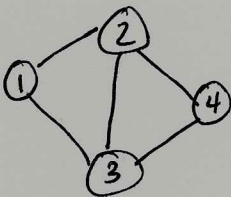
$1 \rightarrow 2 \rightarrow 4$	$3 \rightarrow 2 \rightarrow 1$	$1 \rightarrow 2 \rightarrow 3$
$1 \rightarrow 2 \rightarrow 3$	$3 \rightarrow 2 \rightarrow 4$	$1 \rightarrow 4 \rightarrow 2$
$1 \rightarrow 4 \rightarrow 3$	$3 \rightarrow 4 \rightarrow 1$	
$2 \rightarrow 4 \rightarrow 1$	$3 \rightarrow 4 \rightarrow 2$	
$2 \rightarrow 3 \rightarrow 4$	$4 \rightarrow 1 \rightarrow 2$	
$2 \rightarrow 1 \rightarrow 4$	$4 \rightarrow 2 \rightarrow 1$	
$2 \rightarrow 4 \rightarrow 3$	$4 \rightarrow 3 \rightarrow 2$	
$2 \rightarrow 2 \rightarrow 4$	$4 \rightarrow 2 \rightarrow 3$	

Question 4

It is invalid because there is not a link indicator containing a_{jh}

Question 5

Take for example:



$1 \rightarrow 2 \rightarrow 4$	$4 \rightarrow 3 \rightarrow 1$
$1 \rightarrow 2 \rightarrow 3$	$4 \rightarrow 2 \rightarrow 1$
$1 \rightarrow 3 \rightarrow 2$	$3 \rightarrow 2 \rightarrow 1$
$1 \rightarrow 3 \rightarrow 4$	$2 \rightarrow 3 \rightarrow 1$

$4 = 4$

Its equal because there are the same number as when it goes out.

Question 7

Yes, because $a_{ij}a_{jh} = 1$, therefore $a_{ih} = 1$. There is transitivity. They are connected by a path of length 2.

Question 10

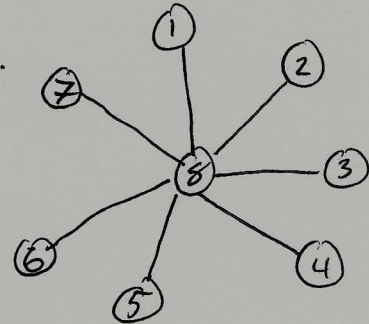
Lets say that we have an $n = 8$ spoke network.

$A =$

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0	1
3	0	0	0	0	0	0	0	1
4	0	0	0	0	0	0	0	1
5	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	0	1
7	0	0	0	0	0	0	0	1
8	1	1	1	1	1	1	1	0

$A^2 =$

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	2
2	0	0	0	0	0	0	0	2
3	0	0	0	0	0	0	0	2
4	0	0	0	0	0	0	0	2
5	0	0	0	0	0	0	0	2
6	0	0	0	0	0	0	0	2
7	0	0	0	0	0	0	0	2
8	2	2	2	2	2	2	2	0



$A^3 =$

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	7
2	0	0	0	0	0	0	0	7
3	0	0	0	0	0	0	0	7
4	0	0	0	0	0	0	0	7
5	0	0	0	0	0	0	0	7
6	0	0	0	0	0	0	0	7
7	0	0	0	0	0	0	0	7
8	7	7	7	7	7	7	7	0

$(7)(A) =$

	1	2	3	4	5	6	7	8
1	0	0	0	0	0	0	0	7
2	0	0	0	0	0	0	0	7
3	0	0	0	0	0	0	0	7
4	0	0	0	0	0	0	0	7
5	0	0	0	0	0	0	0	7
6	0	0	0	0	0	0	0	7
7	0	0	0	0	0	0	0	7
8	7	7	7	7	7	7	7	0

Question 11

In [10]:

```
n = 10
```

```
G = netx.Graph();
```

```
for i in range(n):
    if (not (i == n-1)):
        G.add_edge(i, i+1);
print(G.nodes())
```

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

In [52]:

```
A = netx.adjacency_matrix(G);  
print(A.toarray())
```

```
[[0 1 0 0 0 0 0 0 0 0]  
 [1 0 1 0 0 0 0 0 0 0]  
 [0 1 0 1 0 0 0 0 0 0]  
 [0 0 1 0 1 0 0 0 0 0]  
 [0 0 0 1 0 1 0 0 0 0]  
 [0 0 0 0 1 0 1 0 0 0]  
 [0 0 0 0 0 1 0 1 0 0]  
 [0 0 0 0 0 0 1 0 1 0]  
 [0 0 0 0 0 0 0 1 0 1]  
 [0 0 0 0 0 0 0 0 1 0]]
```

In [48]:

```
A2 = A*A  
print(A2.toarray())
```

```
[[1 0 1 0 0 0 0 0 0 0]  
 [0 2 0 1 0 0 0 0 0 0]  
 [1 0 2 0 1 0 0 0 0 0]  
 [0 1 0 2 0 1 0 0 0 0]  
 [0 0 1 0 2 0 1 0 0 0]  
 [0 0 0 1 0 2 0 1 0 0]  
 [0 0 0 0 1 0 2 0 1 0]  
 [0 0 0 0 0 1 0 2 0 1]  
 [0 0 0 0 0 0 1 0 2 0]  
 [0 0 0 0 0 0 0 1 0 1]]
```

In [47]:

```
A4 = A2*A2  
print(A4.toarray())
```

```
[[2 0 3 0 1 0 0 0 0 0]  
 [0 5 0 4 0 1 0 0 0 0]  
 [3 0 6 0 4 0 1 0 0 0]  
 [0 4 0 6 0 4 0 1 0 0]  
 [1 0 4 0 6 0 4 0 1 0]  
 [0 1 0 4 0 6 0 4 0 1]  
 [0 0 1 0 4 0 6 0 4 0]  
 [0 0 0 1 0 4 0 6 0 3]  
 [0 0 0 0 1 0 4 0 5 0]  
 [0 0 0 0 0 1 0 3 0 2]]
```

In [46]:

```
A6 = A2*A4  
print(A6.toarray())
```

```
[[ 5  0  9  0  5  0  1  0  0  0]  
 [ 0 14  0 14  0  6  0  1  0  0]  
 [ 9  0 19  0 15  0  6  0  1  0]  
 [ 0 14  0 20  0 15  0  6  0  1]  
 [ 5  0 15  0 20  0 15  0  6  0]  
 [ 0  6  0 15  0 20  0 15  0  5]  
 [ 1  0  6  0 15  0 20  0 14  0]  
 [ 0  1  0  6  0 15  0 19  0  9]  
 [ 0  0  1  0  6  0 14  0 14  0]  
 [ 0  0  0  1  0  5  0  9  0  5]]
```

In [53]:

```
A8 = A4*A4  
print(A8.toarray())
```

```
[[14  0 28  0 20  0  7  0  1  0]  
 [ 0 42  0 48  0 27  0  8  0  1]  
 [28  0 62  0 55  0 28  0  8  0]  
 [ 0 48  0 69  0 56  0 28  0  7]  
 [20  0 55  0 70  0 56  0 27  0]  
 [ 0 27  0 56  0 70  0 55  0 20]  
 [ 7  0 28  0 56  0 69  0 48  0]  
 [ 0  8  0 28  0 55  0 62  0 28]  
 [ 1  0  8  0 27  0 48  0 42  0]  
 [ 0  1  0  7  0 20  0 28  0 14]]
```

In [60]:

```
def sumPower(GG):  
    summed = 0;  
    A = GG.toarray();  
    for i in A:  
        for j in i:  
            summed += j;  
    return summed;  
  
print(sumPower(A))  
  
print(sumPower(A2))  
  
print(sumPower(A4))  
  
print(sumPower(A6))  
  
print(sumPower(A8))
```

18
34
122
444
1626

In []: