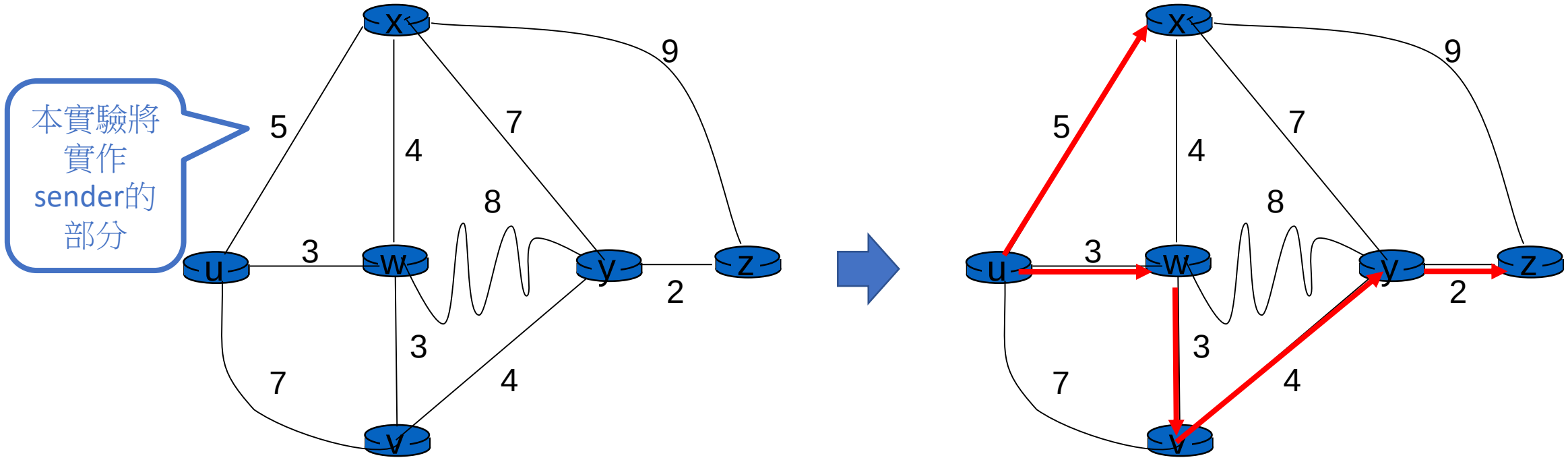
Lab 3: Dijsktra’s algorithm

HackMD: https://hackmd.io/@KentShen/Hy2F6Pgv5

實驗場景

- 輸入網路拓樸資料及起點
- 輸出shortest-path tree
- Python 3.8.10

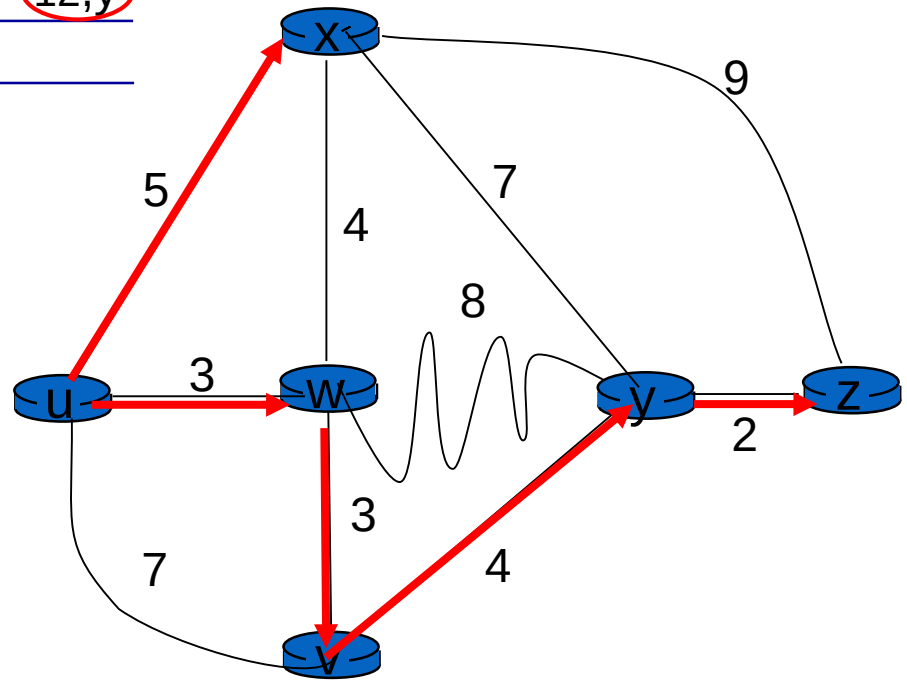


Dijkstra's algorithm: example

Step	N'	D(v) p(v)	D(w) p(w)	D(x) p(x)	D(y) p(y)	D(z) p(z)
0	u	7,u	3,u	5,u	∞	∞
1	uw	6,w		5,u	11,w	∞
2	uwx	6,w			11,w	14,x
3	uwxv				10,v	14,x
4	uwxvy					12,y
5	uwxvyz					

notes:

- construct shortest path tree by tracing predecessor nodes
- ties can exist (can be broken arbitrarily)



拓樸檔案

- input.txt

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

There are 6 nodes in the topology, which are node 0 to node 5

A link from node 3 to node 5 with link cost 2

topo.py

- Define a class named Topo
- Member variables:
  - numNodes: the number of nodes in the topology
  - links[<A>][<B>]: if larger than 0, there is a link from node <A> to node <B>
  - EX: links[2][3] = 5: the cost of link from node 2 to node 3 is 5
  - links[<A>][<B>] is equal to links[<B>][<A>]

sp.py

```
from topo import Topo
import numpy as np

start = 0

myTopo = Topo('input.txt')

N = np.zeros((myTopo.numNodes, 1))
D = np.zeros((myTopo.numNodes, 1))
p = np.zeros((myTopo.numNodes, 1))

for i in range(myTopo.numNodes):
    D[i] = -1
    p[i] = -1
    N[i] = -1

N[0] = start
D[start] = 0
p[start] = start

# TODO: your codes here

for i in range(1, myTopo.numNodes):
    print(int(p[i]), ' --> ', i, ' cost = ', int(D[i]))
```

成果繳交

- 繳交檔案：sp.py
- 繳交時間：July 2, 2022