Bookmark

LAB 2 - TASK 4 (EXTERNAL RESOURCE) (1.0 points possible)

Lab 2 - Task 4

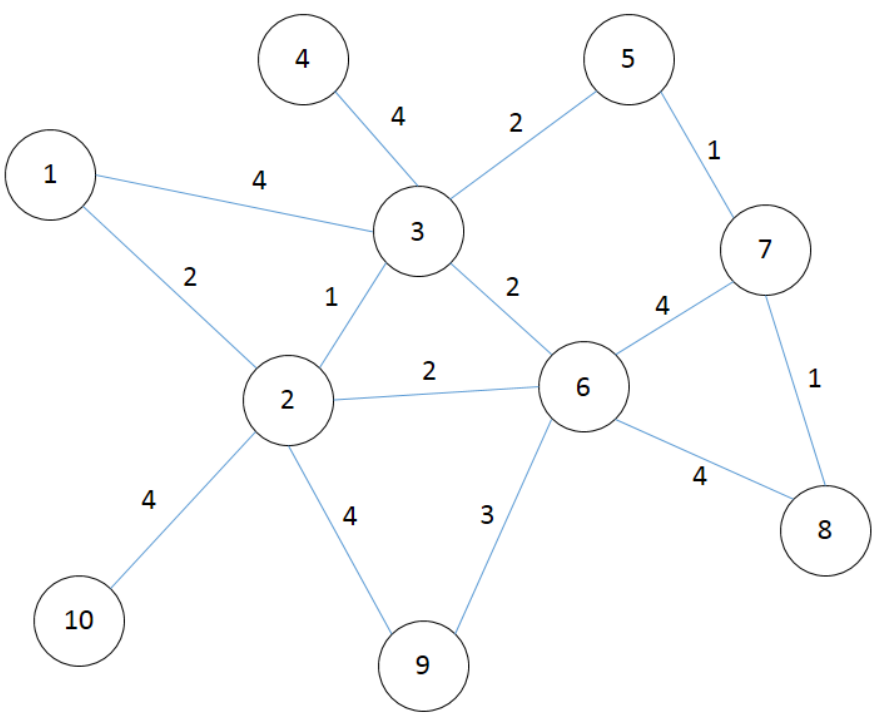
In this task, we will learn how Dijkstra's algorithm used in the link-state routing can be used to deter shortest path between any two nodes.

INSTRUCTIONS

Routing using the link-state algorithm is different than routing using the distance vector algorithm, l link-state algorithm first estimates the graph of the complete network at every node. The informatio complete network is obtained by message flooding, where each node broadcasts information abou connections to its neighbors, and these neighbors then forward that information on and so on. Thu information is propagated globally over the entire network, allowing each node to construct an estir complete network. Given the graph of the complete network, each node then computes the shortes every other node in the network using Dijkstra's algorithm. It then uses this information to construc routing/fowarding table. On the other hand, nodes in the distance vector algorithm never estimate the complete network. Although nodes know the cost of the route to get to each of the nodes throu their neighbors, they do not know anything about the path taken by the packets after they arrive at neighbors. The distance vector algorithm is a decentralized algorithm where each node broadcasts its own routing table iteratively. However, this information is not re-broadcast explicitly, but rather o through changes in its neighbors' routing tables.

The MATLAB code in the window below uses the Dijkstra algorithm to compute the shortest path b nodes. The implementation of the Dijkstra's algorithm is too complicated to be covered in this cours exercise, your task is to learn how to represent the network connection by a matrix and input the m function implementing Dijkstra's algorithm (**getPathDijkstra**).

In order to run Dijkstra's algorithm and get credit for this task, you must set the value of the variabl which is initialized to represent the networks used in Task 1 through 3, so that it represents the net in the figure below. We have already discussed the matrix representation of a graph in Task 2 and video. Once you have revised the current matrix, try to change the source and the destination nod check whether the retrieved path is the one with the smallest cost.



If you are looking for a challenge, try implementing your own version of Dijkstra's algorithm and che
the results of getPathDijkstra.

Your Solution

Reset MATLAB Documentation (https://www.mathworks.

```
12
13 % find the shortest path from source to destination
14 source = 8;
15 dest = 10;
16
17 [path, cost] = getPathDijkstra(graph, source, dest);
18 path
19 cost
20
```

Assessment Tests: Passed

✔ Is the graph correct?

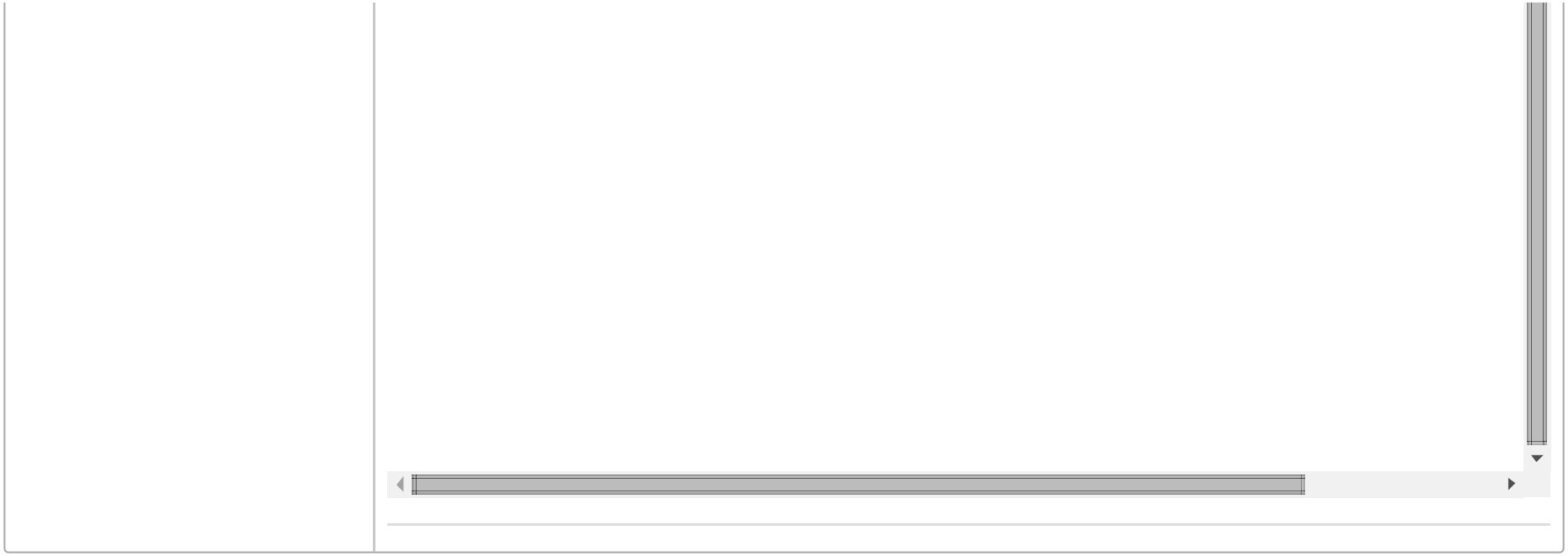
Output

```
path =

     8
     7
     5
     3
     2
    10


cost =

     9
```




Download on the
App Store


ANDROID APP ON
Google play