Table of Contents

Experiment 1: Dijkstra Algorithm

```
% Name: Ventrapragada Sai Shravani
% PRN:17070123120
% Batch: G-5 (E&TC)
% This scrip demostrates the iterations of Dijkstra's algorithm
% The graph is assumed bi-directional
% NaN: Node is not in the reachable list yet.
% Inf: There is not known link yet to the node.
% p: is the “previous vector”
% N_prime: is the list of known shortest-path nodes (reachable list)
% D: is the current known shortest path
```

Code

```
close all
clc
clear
s       = [1 1 1 2 2 3  4 5 6 5];
t       = [2 4 6 6 3 5  6 6 7 7] ;
c       = [5 7 1 8 2 2  9 2 2 1];
Start_node =2; % starting mode
```

Dijsktra's Algorithm

```
[N_prime_cell,D_cell,p_cell,TotalSteps]
 =Dijkstra_table(t,s,c,Start_node);
```

Print the results

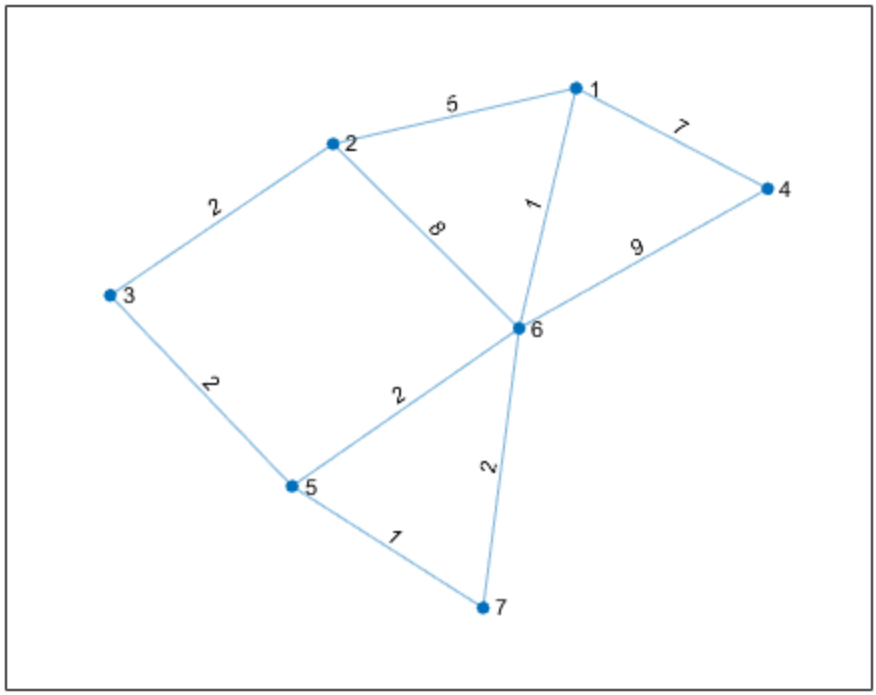
```
Dijkstra_print (N_prime_cell,D_cell,p_cell,TotalSteps);
```

```
Step 1:
N_prime = [2 ]
D = [5 0 2 Inf Inf 8 Inf ]
p = [2 2 2 NaN NaN 2 NaN ]

Step 2:
```

```
N_prime = [2 3 ]
D = [5 0 2 Inf 4 8 Inf ]
p = [2 2 2 NaN 3 2 NaN ]

Step 3:
N_prime = [2 3 5 ]
D = [5 0 2 Inf 4 6 5 ]
p = [2 2 2 NaN 3 5 5 ]

Step 4:
N_prime = [2 3 5 1 ]
D = [5 0 2 12 4 6 5 ]
p = [2 2 2 1 3 5 5 ]

Step 5:
N_prime = [2 3 5 1 7 ]
D = [5 0 2 12 4 6 5 ]
p = [2 2 2 1 3 5 5 ]

Step 6:
N_prime = [2 3 5 1 7 6 ]
D = [5 0 2 12 4 6 5 ]
p = [2 2 2 1 3 5 5 ]

Step 7:
N_prime = [2 3 5 1 7 6 4 ]
D = [5 0 2 12 4 6 5 ]
p = [2 2 2 1 3 5 5 ]
```

Using matlab built-in finction

```
G = graph(s,t,c);
figure(1)
plot(G,'EdgeLabel',G.Edges.Weight)
D_matlab = distances(G,Start_node,'Method','positive') % This is built
 in matlab function to clalculate the shotest path


D_matlab =

     5     0     2    12     4     6     5
```



Conclusion:

We find shortest distance in the nodes and conclude a shortest path for the mobile robot to move.

Published with MATLAB® R2020a