



COMPLEX NETWORK

Quiz 8

WANG BIYUAN

18M38156

DEPARTMENT OF COMPUTER SCIENCE

LECTURER: TSUYOSHI MURATA

January 8, 2019

Code

```
import networkx as nx
import matplotlib.pyplot as plt

G = nx.karate_club_graph()
N = nx.nodes(G)

known = [0] * nx.number_of_nodes(G)
dist = [-1] * nx.number_of_nodes(G)

colors = ['red', 'blue', 'deepskyblue', 'yellow', 'springgreen',
          'lightskyblue', 'darkgreen', 'green', 'lightgreen', 'gray', 'lightgray']
color_map = ['black'] * nx.number_of_nodes(G)

start = 0 # start from 0
dist[start] = 0
color_map[start] = colors[dist[start]]
known[start] = 1
queue = []

d = 0
while sum(known) != nx.number_of_nodes(G) :
    for n in nx.nodes(G) :
        if nx.shortest_path_length(G, start, n) == d :
            queue.append(n)
            for nb in G.neighbors(n) :
                if known[nb] != 1 :
                    dist[nb] = d + 1
                    color_map[nb] = colors[dist[nb]]
                    known[nb] = 1
            d = d + 1

print('distance', dist)
print('nodes', queue)

plt.figure(figsize=(5, 5))
nx.draw_spring(G, node_size=400, node_color=color_map, with_labels=True,
               font_weight='bold')

>>>distance [0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 3, 3, 2, 1, 3, 1, 3,
            1, 3, 3, 2, 2, 3, 2, 2, 3, 2, 1, 2, 2]
nodes [0, 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 17, 19, 21, 31, 9, 16,
       24, 25, 27, 28, 30, 32, 33]
```

Results

(1) Make a program of breadth-first search (BFS).

See the **Code** section above.

(2) Show the final status of distance array and queue after BFS is done from vertex 0 of Karate club network.

See the **Code** section above.

(3) Explain why BFS is not good for networks with varying edge lengths.

Because BFS only visits every node once. If a node has a direct edge from starting node with longer length, but also has an indirect edge through another node connected to the starting node with a sum-up shorter edge length, as long as this node is visited by BFS via the direct path from starting node, it will be annotated as seen node and never being visited again. The shortest path recorded for this node is actually not the real shortest one but an edge of longer length.