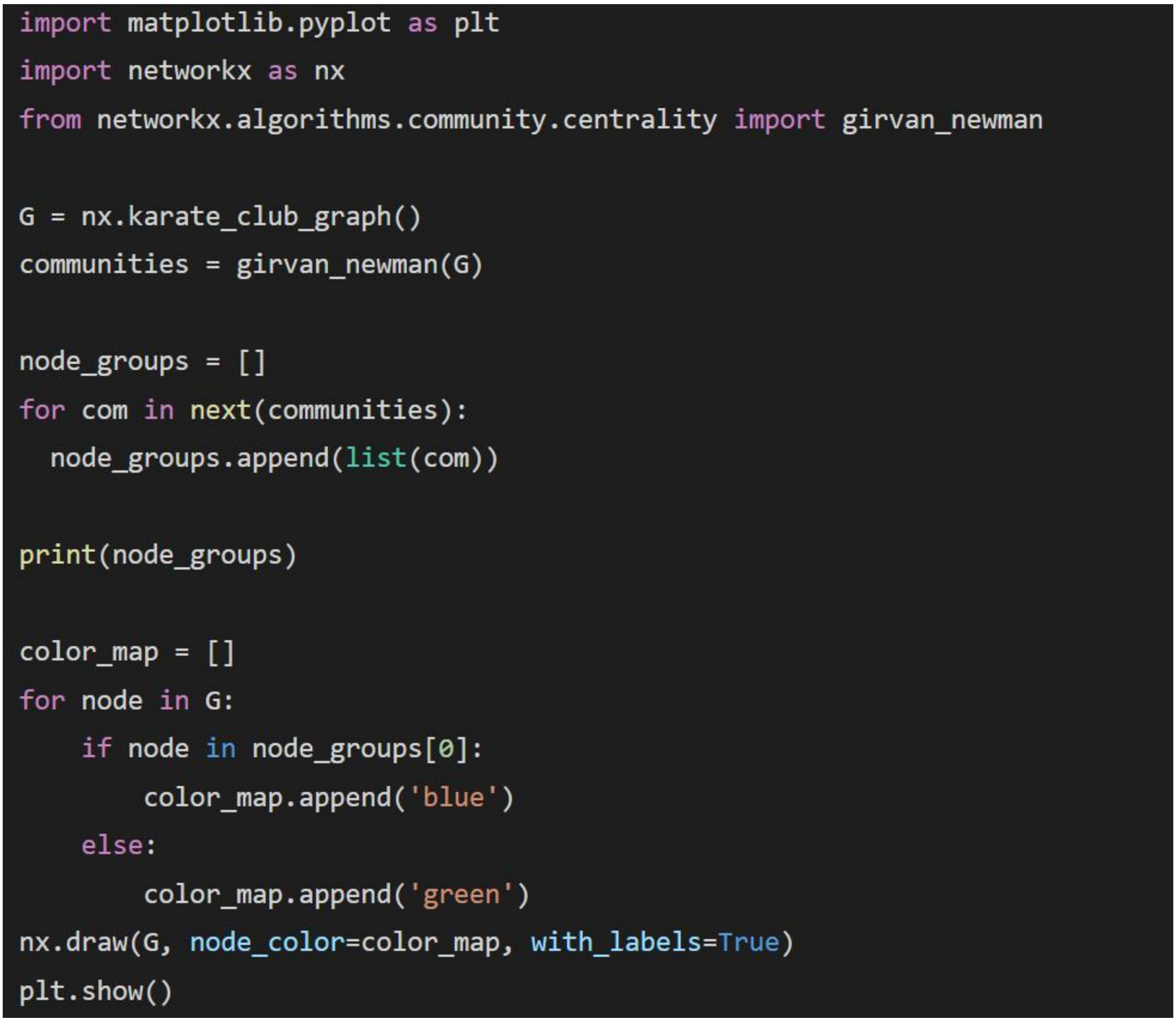
SNA: Laboratory Session 3

Manipulation using NetworkX and other python libraries for Community detections A)

Use the karateClubDataset used in Laboratory session 1 (see file: karate_club_coords.pkl). Submit your code and graphical illustrations, variables as a one single zip file in Submission return box-

1) Study and execute the program below for community detection and display the plot showing the communities detected from KarateClubDataset and save the result as variables. Suggest alternative visualization that enhances readability of the communities (various color and thickness or annotation).



```python
import matplotlib.pyplot as plt
import networkx as nx
from networkx.algorithms.community.centrality import girvan_newman

G = nx.karate_club_graph()
communities = girvan_newman(G)

node_groups = []
for com in next(communities):
  node_groups.append(list(com))

print(node_groups)

color_map = []
for node in G:
    if node in node_groups[0]:
        color_map.append('blue')
    else:
        color_map.append('green')
nx.draw(G, node_color=color_map, with_labels=True)
plt.show()
```

2) We want to study the result of community detection using Ratio Cut Method. For this purpose, use the in-built function in NetworkX from community.kernighan_lin_bisection() and display the result of the community detection, and save the result in a separate variables.

3) Repeat 2) when using the Louvain community detection algorithm, which has inbuilt function in NetworkX

4) Repeat 2) when using label-propagation community detection algorithm.

5) We want to evaluate the quality of the communities detected in 1)-4). For this purpose, study the available community quality performance algorithms available in NetworkX and write a script that outputs the quality performance (s) for each of the detection algorithm (Girvan Neuman, Ratio Cut, label propagation and Louvain method).

6) We want to compare the performance of the algorithms in 1)-4) in terms of algorithmic complexity. For this purpose, write a script that reports the execution time of each algortihm (Girvan Neuman, Ratio Cut, label propagation and Louvan method).

7) Calculate the average path length in the above graph.

8) Write a script that generates additional 20 nodes, randomly located within the smallest rectangular region containing all Karate nodes, to the karate network using a geometric random graph (using nx.random_geometric_graph function in NetworkX) such as an edge is established if its distance is at least equal to the average path length. Display the graph using different for colors for karate subgraph and random subgraph.

9) We want to repeat 1)-6) when using the newly expanded graph, and exhibit the communities detected by each algorithm.