Social Network Analysis

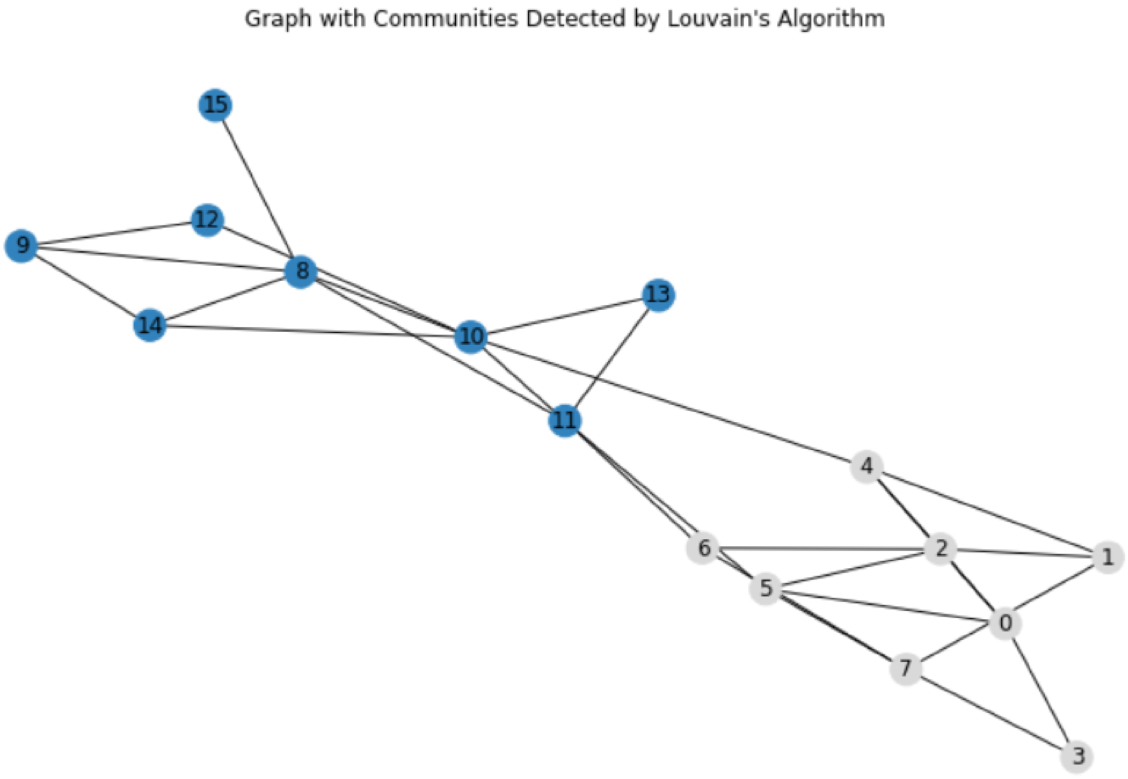
Assignment 5: Louvain's algorithm

U20CS005
BANSI MARAKANA

Detect the communities in the graph using Louvain's algorithm for the following graphs (Use louvain_communities method of NetworkX.):

1) Generate a graph shown in the figure below using NetworkX and detect the communities.

```
import networkx as nx
import matplotlib.pyplot as plt
from community import community_louvain
G = nx.Graph()
G.add_nodes_from([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15])
G.add_edges_from([(0, 2), (0, 4), (0, 5), (0, 3), (1, 2), (1, 4), (1, 7), (2, 4), (2, 5), (2, 6), (3, 7), (4,
10), (5, 7), (5, 11), (6, 7), (6, 11), (8, 9), (8, 10), (8, 11), (8, 14), (8, 15), (9, 12), (9, 14), (10, 11),
(10, 12), (10, 13), (10, 14), (11, 13)])

# Detect communities using Louvain's algorithm
partition = community_louvain.best_partition(G)

# Visualize the graph with communities
pos = nx.spring_layout(G)
plt.figure(figsize=(10, 6))
nx.draw(G, pos, node_color=list(partition.values()), cmap='tab20c', node_size=300,
with_labels=True)
plt.title("Graph with Communities Detected by Louvain's Algorithm")
plt.show()

# Print the communities
communities = {}
for node, community_id in partition.items():
    if community_id not in communities:
        communities[community_id] = []
    communities[community_id].append(node)
print("Communities detected by Louvain's algorithm:")
for community_id, nodes in communities.items():
    print(f"Community {community_id}: {nodes}")
```

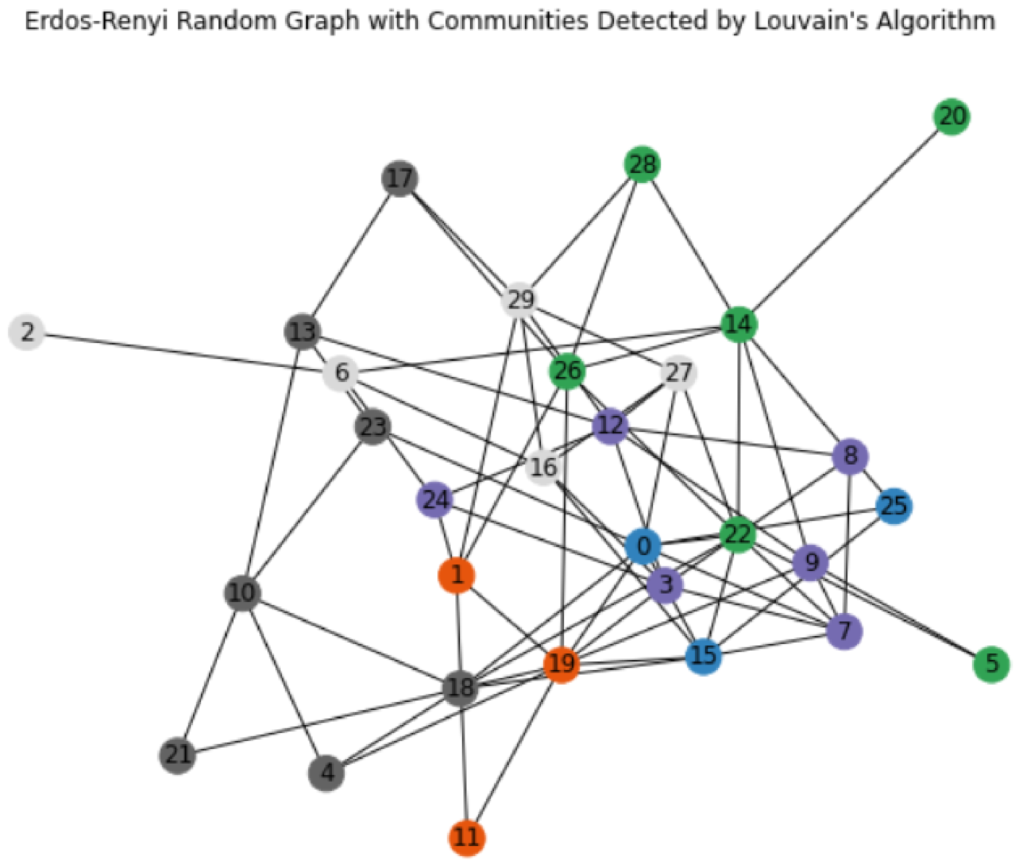


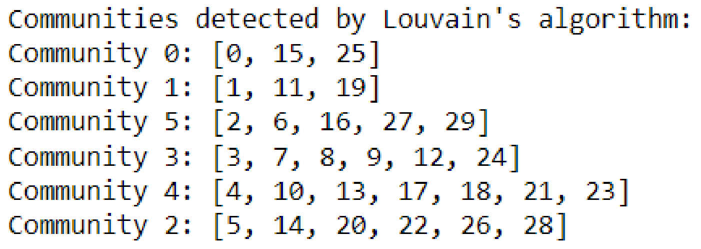
```
Communities detected by Louvain's algorithm:
Community 1: [0, 1, 2, 3, 4, 5, 6, 7]
Community 0: [8, 9, 10, 11, 12, 13, 14, 15]
```

2) Generate an Erdos Renyi random graph and detect communities.

```
import networkx as nx
import matplotlib.pyplot as plt
from community import community_louvain

# Generate Erdos-Renyi random graph
n = 30  # Number of nodes
p = 0.15  # Probability of edge creation
G = nx.erdos_renyi_graph(n, p)

# Detect communities using Louvain's algorithm
partition = community_louvain.best_partition(G)

# Visualize the graph with communities
pos = nx.spring_layout(G)
plt.figure(figsize=(8, 6))
nx.draw(G, pos, node_color=list(partition.values()), cmap='tab20c', node_size=300,
with_labels=True)
```

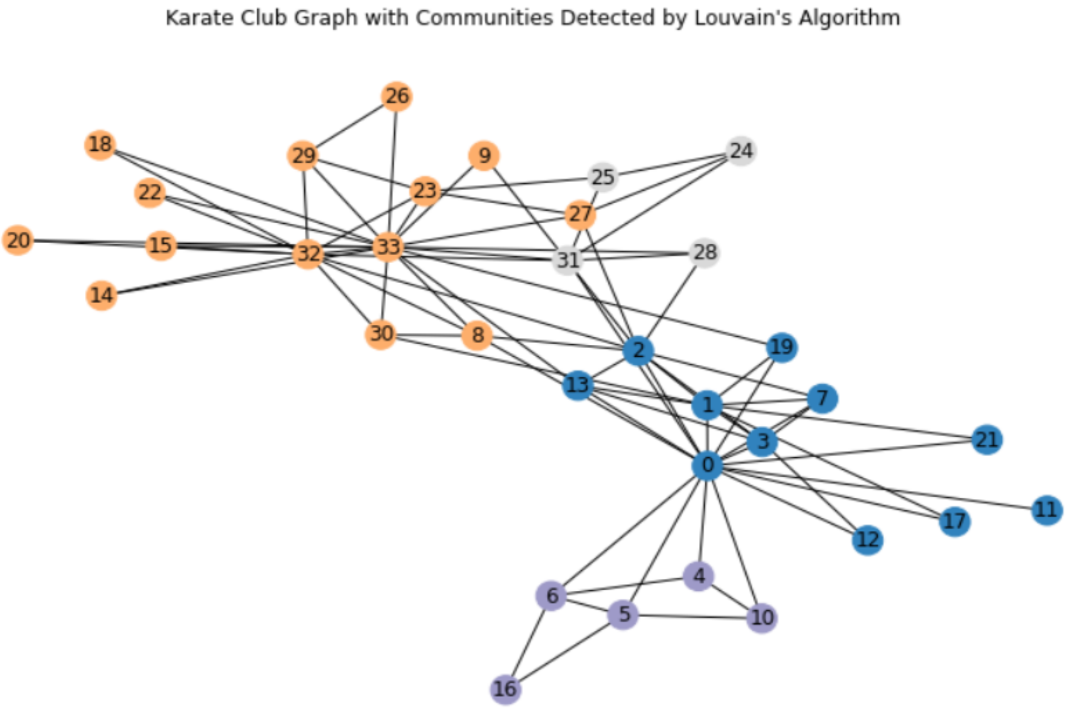
```
plt.title("Erdos-Renyi Random Graph with Communities Detected by Louvain's Algorithm")
plt.show()

# Print the communities
communities = {}
for node, community_id in partition.items():
    if community_id not in communities:
        communities[community_id] = []
    communities[community_id].append(node)
print("Communities detected by Louvain's algorithm:")
for community_id, nodes in communities.items():
    print(f"Community {community_id}: {nodes}")
```

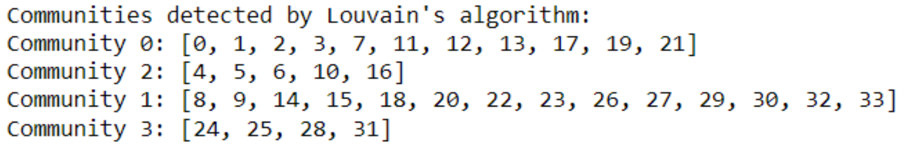

Erdos-Renyi Random Graph with Communities Detected by Louvain's Algorithm


```
Communities detected by Louvain's algorithm:
Community 0: [0, 15, 25]
Community 1: [1, 11, 19]
Community 5: [2, 6, 16, 27, 29]
Community 3: [3, 7, 8, 9, 12, 24]
Community 4: [4, 10, 13, 17, 18, 21, 23]
Community 2: [5, 14, 20, 22, 26, 28]
```

3) Use the inbuilt graph of Zachary's karate club and detect the communities.

```
import networkx as nx
import matplotlib.pyplot as plt
from community import community_louvain
G = nx.karate_club_graph()
partition = community_louvain.best_partition(G)
pos = nx.spring_layout(G)
plt.figure(figsize=(10, 6))
nx.draw(G, pos, node_color=list(partition.values()), cmap='tab20c', node_size=300,
with_labels=True)
plt.title("Karate Club Graph with Communities Detected by Louvain's Algorithm")
plt.show()
communities = {}
for node, community_id in partition.items():
    if community_id not in communities:
        communities[community_id] = []
    communities[community_id].append(node)
print("Communities detected by Louvain's algorithm:")
for community_id, nodes in communities.items():
    print(f"Community {community_id}: {nodes}")
```


Karate Club Graph with Communities Detected by Louvain's Algorithm
26
18
29
9
24
25
22
23
27
20
15
32
33
31
28
14
30
8
2
19
13
1
7
3
21
0
11
17
12
4
6
5
10
16



```
Communities detected by Louvain's algorithm:
Community 0: [0, 1, 2, 3, 7, 11, 12, 13, 17, 19, 21]
Community 2: [4, 5, 6, 10, 16]
Community 1: [8, 9, 14, 15, 18, 20, 22, 23, 26, 27, 29, 30, 32, 33]
Community 3: [24, 25, 28, 31]
```