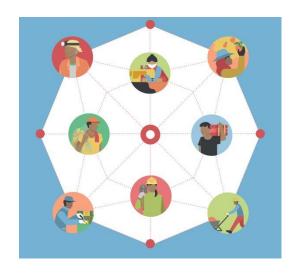


Social Network Analysis

PROJECT REPORT



Submitted By:

Divyansh Mishra (20ucs065) Manan Badaya (20ucs109) Chahak Jain (20dcs002)

Submitted to:

Dr. Sakthi Balan



Contents

1. Introduction	3.
2. BA algorithm	4.
3. Implementation	5
4. Conclusion	10.
5 References	11



INTRODUCTION

Objective: To Generate a Scale- free network by implementing BA algorithm and analyzing the network.

The Barabási–Albert (BA) model is a popular algorithm for generating scale-free networks, which are characterized by a small number of highly connected nodes (hubs) and a large number of poorly connected nodes. In social network analysis, the BA model can be used to simulate the growth and evolution of real-world networks, such as citation networks, collaboration networks, and online social networks.



BA Algorithm

The BA model starts with a small initial network of m nodes, and then adds new nodes to the network one at a time. Each new node is connected to m existing nodes, chosen with probability proportional to their degree (i.e., the number of connections they already have). This preferential attachment mechanism leads to the formation of hubs, as nodes with high degree are more likely to receive new connections.

To implement the BA model in social network analysis, the following steps can be followed:

- 1. Start with a small initial network of m nodes, either randomly connected or connected in a specific pattern.
- 2. Add new nodes to the network one at a time, and connect each new node to m existing nodes according to the preferential attachment mechanism.
- 3. Repeat step 2 until the desired number of nodes is reached.
- 4. Analyze the resulting network using various network analysis tools, such as centrality measures, clustering algorithms, and community detection methods.

4 | Page



CODE IMPLEMENTATION AND EXPLANATION

This .zip file also contains the code in a python file named **cg_project.py**. The code is as:

1. <u>Implementing the BA Algorithm to generate the scale-free</u> network S over 100,000 nodes.

```
node1, node2 = np.random.cnoice(S.nodes(), Size=2, replace=Faise)

# Add an edge between the two nodes
S.add_edge(node1, node2)
# initial parameters for BA model
m = 4 # Number of edges added at each time step
N = 10000 # Number of nodes in the final network

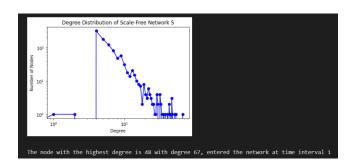
while len(S) < N:
nodes = list(S.nodes())
degrees = np.array([S.degree(node) for node in nodes])
probs = degrees / sum(degrees)#probability to be selected is depending on the degree ---> preferential attachment
chosen_nodes = np.random.choice(nodes, size=m, replace=False, p=probs)

# Adding new node with edges to the network
new_node = len(S)
S.add_node(new_node)
for node in chosen_nodes:
    S.add_edge(new_node)
# Calculate the degree distribution of the network
degree_sequence = sorted([d for n, d in S.degree()], reverse=True)
degree_counts = np.bincount(degree_sequence)
```

Here we implemented the BA algorithm which is used for generating random scale-free networks. We have initialized the parameters as -

m = 4 (these are the number of edges which are added at each time step). N = 10,000 (the number of nodes in the final network).

2. Plotting the degree distribution of the above scale-free network S.The node with the highest degree and the time interval this node came into the network when you generated the network S.

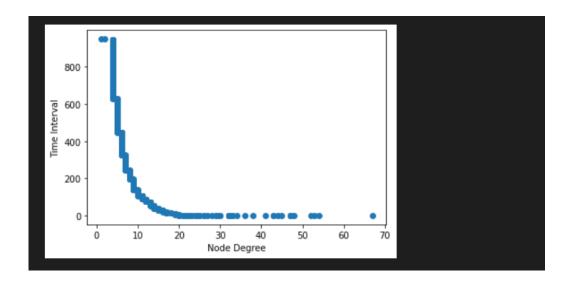




3. A plot where the x-axis is the node degree, and the y-axis is the time interval in which that node entered the network.

```
# Ploting the graph for node degree vs time interval
degrees = nx.degree(S)
degree_sequence = sorted([d for n, d in degrees], reverse=True)
time_intervals_sequence = [time_intervals[n] for n, d in degrees]

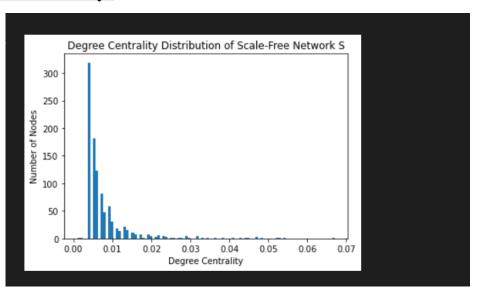
plt.plot(degree_sequence, time_intervals_sequence, 'o')
plt.xlabel('Node Degree')
plt.ylabel('Time Interval')
plt.show()
```



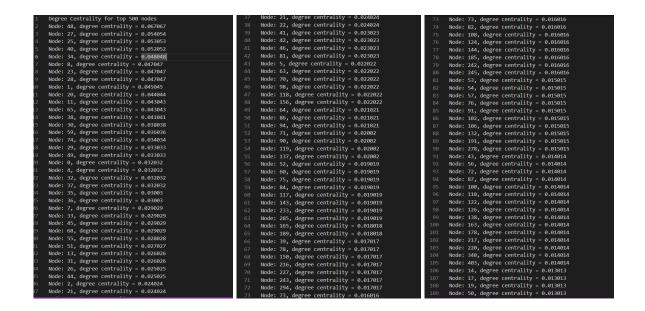


4. The centrality measures of all the nodes in S and through proper visualization.

• Degree Centrality-



Degree Centrality of 500 nodes in the network-





```
| 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100
```

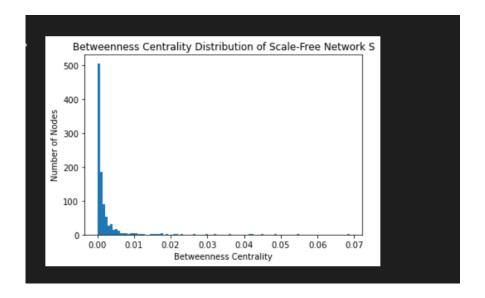


```
| Mode: 372, degree centrality = 0.0000007 | 309 | Mode: 502, degree centrality = 0.0000000 | 309 | Mode: 376, degree centrality = 0.0000000 | 309 | Mode: 616, degree centrality = 0.0000000 | 309 | Mode: 617, degree centrality = 0.0000000 | 309 | Mode: 617, degree centrality = 0.0000000 | 309 | Mode: 617, degree centrality = 0.0000000 | 309 | Mode: 617, degree centrality = 0.0000000 | 309 | Mode: 379, degree centrality = 0.0000000 | 309 | Mode: 379, degree centrality = 0.0000000 | 309 | Mode: 379, degree centrality = 0.0000000 | 309 | Mode: 309, degree centrality = 0.0000000 | 309 | Mode: 309, degree centrality = 0.0000000 | 309 | Mode: 309, degree centrality = 0.0000000 | 309 | Mode: 407, degree centrality = 0.0000000 | 309 | Mode: 407, degree centrality = 0.0000000 | 309 | Mode: 407, degree centrality = 0.0000000 | 309 | Mode: 415, degree centrality = 0.0000000 | 309 | Mode: 415, degree centrality = 0.0000000 | 309 | Mode: 415, degree centrality = 0.0000000 | 410 | Mode: 309, degree centrality = 0.0000000 | 410 | Mode: 309, degree centrality = 0.000000 | 410 | Mode: 309, degree centrality = 0.000000 | 410 | Mode: 309, degree centrality = 0.000000 | 410 | Mode: 309, degree centrality = 0.000000 | 410 | Mode: 309, degree centrality = 0.000000 | 410 | Mode: 415, degree centrality = 0.0000000 | 410 | Mode: 415, degree centrality = 0.0000000 | 410 | Mode: 415, degree centrality = 0.0000000 | 410 | Mode: 415, degree centrality = 0.0000000 | 410 | Mode: 415, degree centrality = 0.0000000 | 410 | Mode: 415, degree centrality = 0.0000000 | 410 | Mode: 415, degree centrality = 0.000000 | 410 | Mode: 316, degree centrality = 0.0000000 | 410 | Mode: 317, degree centrality = 0.0000000 | 410 | Mode: 317, degree centrality = 0.0000000 | 410 | Mode: 317, degree centrality = 0.0000000 | 410 | Mode: 317, degree centrality = 0.0000000 | 410 | Mode: 317, degree centrality = 0.0000000 | 410 | Mode: 317, degree centrality = 0.0000000 | 410 | Mode: 317, degree centrality = 0.0000000 | 410 | Mode: 317, degree centrality = 0.0000000
```

```
671, degree centrality
Node: 678, degree centrality = 0.006006
Node: 682, degree centrality = 0.006006
Node: 683, degree centrality = 0.006006
Node: 696, degree centrality = 0.006006
Node: 703, degree centrality = 0.006006
Node: 709, degree centrality = 0.006006
Node: 716, degree centrality = 0.006006
Node: 725, degree centrality = 0.006006
Node: 726, degree centrality = 0.006006
Node: 734, degree centrality = 0.006006
Node: 739, degree centrality = 0.006006
Node: 755, degree centrality = 0.006006
Node: 765, degree centrality = 0.006006
Node: 780, degree centrality = 0.006006
Node: 789, degree centrality = 0.006006
Node: 840, degree centrality = 0.006006
Node: 845, degree centrality = 0.006006
Node: 849, degree centrality = 0.006006
Node: 859, degree centrality = 0.006006
Node: 865, degree centrality = 0.006006
Node: 121, degree centrality = 0.005005
Node: 133, degree centrality = 0.005005
```



• Betweenness centrality-



Betweenness Centrality for top 500 nodes -

```
| Node: 48, betweeness centrality on top 500 modes | Node: 63, betweeness centrality = 0.005771 | Node: 24, betweeness centrality = 0.005771 | Node: 25, betweeness centrality = 0.005781 | Node: 27, betweeness centrality = 0.005864 | Node: 27, betweeness centrality = 0.005864 | Node: 27, betweeness centrality = 0.005864 | Node: 28, betweeness centrality = 0.005868 | Node: 28, betweeness centrality = 0.005874 | Node: 28, betweeness centrality = 0.005974 | Node: 29, bet
```





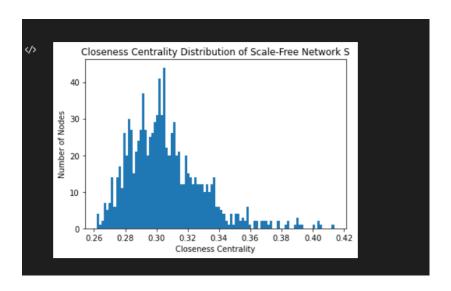
```
| Node: 83, betweeness centrality = 0.001218 | 873 | Node: 284, betweeness centrality = 0.001042 | 910 | Node: 43b, betweeness centrality = 0.00000 | 935 | Node: 475, betweeness centrality = 0.001042 | 911 | Node: 475, betweeness centrality = 0.000007 | 936 | Node: 475, betweeness centrality = 0.000007 | 936 | Node: 675, betweeness centrality = 0.000007 | 936 | Node: 675, betweeness centrality = 0.000007 | 937 | Node: 806 | Node: 806
```

```
Node: 699, betweeness centrality = 0.000713
 Node: 670, betweeness centrality = 0.000711
 Node: 314, betweeness centrality = 0.00071
 Node: 737, betweeness centrality = 0.00071
 Node: 682, betweeness centrality = 0.000708
 Node: 332, betweeness centrality = 0.000706
 Node: 666, betweeness centrality = 0.000704
 Node: 377, betweeness centrality = 0.000702
 Node: 526, betweeness centrality = 0.000698
Node: 164, betweeness centrality = 0.000696
Node: 174, betweeness centrality = 0.000694
 Node: 139, betweeness centrality = 0.000693
Node: 532, betweeness centrality = 0.000693
Node: 819, betweeness centrality = 0.000693
 Node: 311, betweeness centrality = 0.00069
 Node: 615, betweeness centrality = 0.000684
 Node: 262, betweeness centrality = 0.000682
 Node: 586, betweeness centrality = 0.000682
 Node: 559, betweeness centrality = 0.000681
 Node: 671, betweeness centrality = 0.000681
```

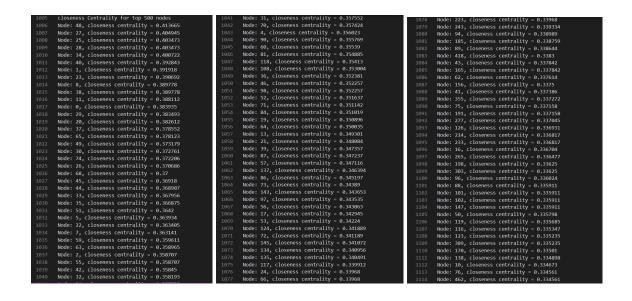
<u>Eigen-Vector Centrality -</u>



Closeness Centrality-



Closeness Centrality of top 500 nodes -





		_	
1226	Node: 221, closeness centrality = 0.320295	1263 Node: 969, closeness centrality = 0.316741	1300 Node: 515, closeness centrality = 0.313461
1227	Node: 260, closeness centrality = 0.32009	1264 Node: 169, closeness centrality = 0.31664	1301 Node: 790, closeness centrality = 0.31297
1228	Node: 12, closeness centrality = 0.319987	1265 Node: 466, closeness centrality = 0.31664	1302 Node: 148, closeness centrality = 0.312872
1229	Node: 301, closeness centrality = 0.319885	1266 Node: 745, closeness centrality = 0.31654	1303 Node: 283, closeness centrality = 0.312774
1230	Node: 320, closeness centrality = 0.319782	1267 Node: 254, closeness centrality = 0.316339	1304 Node: 509, closeness centrality = 0.312774
1231	Node: 403, closeness centrality = 0.319782	1268 Node: 268, closeness centrality = 0.316139	1305 Node: 269, closeness centrality = 0.312676
1232	Node: 748, closeness centrality = 0.31968	1269 Node: 373, closeness centrality = 0.316139	1306 Node: 363, closeness centrality = 0.312676
1233	Node: 133, closeness centrality = 0.319578	1270 Node: 589, closeness centrality = 0.316039	1307 Node: 482, closeness centrality = 0.312676
1234	Node: 483, closeness centrality = 0.319578	1271 Node: 217, closeness centrality = 0.315839	1308 Node: 115, closeness centrality = 0.312578
1235	Node: 533, closeness centrality = 0.319476	1272 Node: 528, closeness centrality = 0.315839	1309 Node: 868, closeness centrality = 0.312578
1236	Node: 92, closeness centrality = 0.319271	1273 Node: 146, closeness centrality = 0.31574	1310 Node: 140, closeness centrality = 0.31248
1237	Node: 195, closeness centrality = 0.319169	1274 Node: 496, closeness centrality = 0.31574	1311 Node: 192, closeness centrality = 0.312383
1238	Node: 819, closeness centrality = 0.319169	1275 Node: 305, closeness centrality = 0.31564	1312 Node: 309, closeness centrality = 0.312383
1239	Node: 100, closeness centrality = 0.318966	1276 Node: 797, closeness centrality = 0.31544	1313 Node: 341, closeness centrality = 0.312383
1240	Node: 141, closeness centrality = 0.318864	1277 Node: 521, closeness centrality = 0.315142	1314 Node: 930, closeness centrality = 0.312383
1241	Node: 378, closeness centrality = 0.318864	1278 Node: 536, closeness centrality = 0.315142	1315 Node: 937, closeness centrality = 0.312383
1242	Node: 294, closeness centrality = 0.318762	1279 Node: 433, closeness centrality = 0.314943	1316 Node: 364, closeness centrality = 0.312285
1243	Node: 415, closeness centrality = 0.318762	1280 Node: 251, closeness centrality = 0.314844	1317 Node: 516, closeness centrality = 0.31209
1244	Node: 330, closeness centrality = 0.31866	1281 Node: 312, closeness centrality = 0.314844	1318 Node: 596, closeness centrality = 0.311993
1245	Node: 263, closeness centrality = 0.318559	1282 Node: 336, closeness centrality = 0.314745	1319 Node: 976, closeness centrality = 0.311993
1246	Node: 559, closeness centrality = 0.318559	1283 Node: 408, closeness centrality = 0.314745	1320 Node: 464, closeness centrality = 0.311798
1247	Node: 665, closeness centrality = 0.318559	1284 Node: 176, closeness centrality = 0.314646	1321 Node: 308, closeness centrality = 0.3117
1248	Node: 199, closeness centrality = 0.318457	1285 Node: 517, closeness centrality = 0.314646	1322 Node: 407, closeness centrality = 0.3117
1249	Node: 206, closeness centrality = 0.318457	1286 Node: 186, closeness centrality = 0.314547	1323 Node: 809, closeness centrality = 0.3117
1250	Node: 79, closeness centrality = 0.318356	1287 Node: 394, closeness centrality = 0.314547	1324 Node: 420, closeness centrality = 0.311603
1251	Node: 400, closeness centrality = 0.318356	1288 Node: 162, closeness centrality = 0.314349	1325 Node: 573, closeness centrality = 0.311506
1252	Node: 412, closeness centrality = 0.318153	1289 Node: 455, closeness centrality = 0.314349	1326 Node: 626, closeness centrality = 0.311506
1253	Node: 523, closeness centrality = 0.318153	1290 Node: 716, closeness centrality = 0.314349	1327 Node: 353, closeness centrality = 0.311312
1254	Node: 501, closeness centrality = 0.318052	1291 Node: 907, closeness centrality = 0.314349	1328 Node: 468, closeness centrality = 0.311312
1255	Node: 655, closeness centrality = 0.317849	1292 Node: 285, closeness centrality = 0.31425	1329 Node: 687, closeness centrality = 0.311312
1256	Node: 210, closeness centrality = 0.317647	1293 Node: 152, closeness centrality = 0.314052	1330 Node: 919, closeness centrality = 0.311312
1257	Node: 58, closeness centrality = 0.317546	1294 Node: 280, closeness centrality = 0.314052	1331 Node: 131, closeness centrality = 0.311118
1258	Node: 188, closeness centrality = 0.317546	1295 Node: 793, closeness centrality = 0.314052	1332 Node: 337, closeness centrality = 0.311021
1259	Node: 181, closeness centrality = 0.317445	1296 Node: 229, closeness centrality = 0.313953	1333 Node: 190, closeness centrality = 0.310924
1260	Node: 354, closeness centrality = 0.317042	1297 Node: 298, closeness centrality = 0.313953 1298 Node: 448, closeness centrality = 0.313855	1334 Node: 548, closeness centrality = 0.310924
1261	Node: 413, closeness centrality = 0.316942		1335 Node: 121, closeness centrality = 0.310828
1262	Node: 616, closeness centrality = 0.316841	1299 Node: 202, closeness centrality = 0.313559	1336 Node: 212, closeness centrality = 0.310828
	·	•	

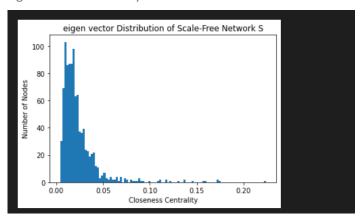


```
| 1377 | 100de: 70; | 100mens centrality = 0.38048 | 150mens c
```

```
Node: 419, closeness centrality = 0.303095
Node: 554, closeness centrality = 0.303095
Node: 636, closeness centrality = 0.303095
Node: 662, closeness centrality = 0.303095
Node: 767, closeness centrality = 0.303095
Node: 852, closeness centrality = 0.303003
Node: 209, closeness centrality = 0.302819
Node: 431, closeness centrality = 0.302819
Node: 700, closeness centrality = 0.302727
Node: 432, closeness centrality = 0.302636
Node: 476, closeness centrality = 0.302544
Node: 244, closeness centrality = 0.302452
Node: 314, closeness centrality = 0.302452
Node: 680, closeness centrality = 0.302452
Node: 858, closeness centrality = 0.302452
Node: 372, closeness centrality = 0.302361
Node: 425, closeness centrality = 0.302361
Node: 538, closeness centrality = 0.302361
Node: 722, closeness centrality = 0.302361
Node: 498, closeness centrality = 0.302269
```



Eigen Vector Centrality-



Eigen Vectore centraolity of 500 nodes -

```
| 1500 | Bode: 48, cigen vector centrality = 0.232831 | 1544 | Bode: 78, eigen vector centrality = 0.086233 | 1545 | Bode: 79, eigen vector centrality = 0.086233 | 1545 | Bode: 79, eigen vector centrality = 0.086233 | 1546 | Bode: 79, eigen vector centrality = 0.085233 | 1546 | Bode: 79, eigen vector centrality = 0.085233 | 1546 | Bode: 79, eigen vector centrality = 0.085233 | 1547 | Bode: 79, eigen vector centrality = 0.085233 | 1548 | Bode: 79, eigen vector centrality = 0.085233 | 1548 | Bode: 79, eigen vector centrality = 0.085233 | 1549 | Bode: 79, eigen vector centrality = 0.085233 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.085234 | 1549 | Bode: 79, eigen vector centrality = 0.0
```



5. The giant component in S as G and the ratio of the nodes in G to S -

```
Giant Component Size: 1000
Total Nodes in S: 1000
Ratio of Nodes in Giant Component to S: 1.0
```

```
#giant component in S
connected_components = list(nx.connected_components(S))
giant_component = max(connected_components, key=len)

#ratio
giant_component_size = len(giant_component)
total_nodes = len(S.nodes)
ratio = giant_component_size / total_nodes

print(f"Giant Component Size: {giant_component_size}")
print(f"Total Nodes in S: {total_nodes}")
print(f"Ratio of Nodes in Giant Component to S: {ratio}")

print()
```

9 | Page



- 6. Steps required to pass information to the maximum number of nodes in S.
 - We have combined the part a) and part b) of the question i.e for each value of p we calculated the average number of steps required by

performing the experiment 10 times-

```
for probabability 0.25 = 1472.1
for probabability 0.5 = 1924.9
for probabability 0.75 = 1335.1
for probabability 1 = 999.0
```

• Conclusion -

As every node will get one chance to pass information to its neighbors, with lower probability steps for information to reach the maximum number of nodes will be greater ,with increasing probability steps will decrease. But when probability is very low, steps again decreases as number of nodes susceptible to information will be far more compared to higher probabilities.



REFERENCES