Part a

```
import networkx as nx
import pandas as pd
import numpy as np
import matplotlib.pylab as plt
import seaborn as sns
```

```
dataframe=np.array(pd.read_csv("/content/Part2_adj.csv",header=None))
print(dataframe)

network=nx.from_numpy_array(dataframe)
```

```
    [[0 1 1 ... 0 1 0]
     [1 0 1 ... 1 1 1]
     [1 1 0 ... 1 0 0]
     ...
     [0 1 1 ... 0 0 0]
     [1 1 0 ... 0 0 0]
     [0 1 0 ... 0 0 0]]
```

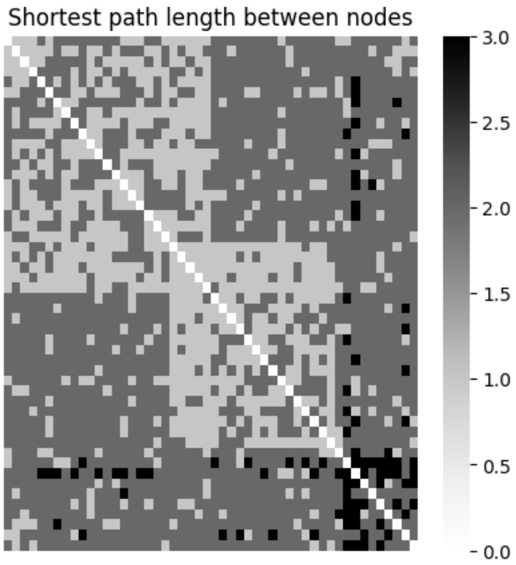
```
shortest_path=nx.all_pairs_shortest_path_length(network)
length = dict(shortest_path)
```

```
mat_mindistances= np.zeros((dataframe.shape[0], dataframe.shape[1]))

for i in range(dataframe.shape[0]):
  for j in range(dataframe.shape[1]):
    mat_mindistances[i,j]=length[i][j]
```

```
plt.figure(figsize=(5,5))
heat_map = sns.heatmap( mat_mindistances,cmap="Greys",xticklabels=False,yticklabels=False)

#heat_map = sns.heatmap( theta,cmap=sns.cubehelix_palette(as_cmap=True))
plt.title( "Shortest path length between nodes" )
plt.show()
```



```
print("Shortest path length between node 0 and node 1 is ", mat_mindistances[0,1])
print("Shortest path length between node 0 and node 25 is ",mat_mindistances[0,25])
print("Shortest path length between node 0 and node 48 is ", mat_mindistances[0,48])
print("Shortest path length between node 25 and node 48 is ",mat_mindistances[25,48])
```

```
    Shortest path length between node 0 and node 1 is  1.0
    Shortest path length between node 0 and node 25 is  2.0
    Shortest path length between node 0 and node 48 is  1.0
    Shortest path length between node 25 and node 48 is  2.0
```

Answer a

Shortest path length between node 0 and node 1 is 1.0

Shortest path length between node 0 and node 25 is 2.0

Shortest path length between node 0 and node 48 is 1.0

Shortest path length between node 25 and node 48 is 2.0

Part b

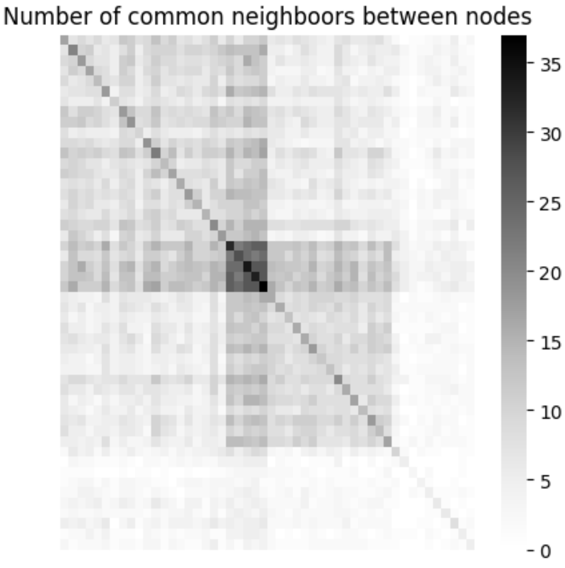
```
num_common_neighbors= np.zeros((dataframe.shape[0], dataframe.shape[1]))

for i in range(dataframe.shape[0]):
  for j in range(dataframe.shape[1]):
    c_neighbors=nx.common_neighbors(network,i,j)
    for nodes in c_neighbors:
      num_common_neighbors[i,j]=num_common_neighbors[i,j]+1



plt.figure(figsize=(5,5))
heat_map = sns.heatmap( num_common_neighbors,cmap="Greys",xticklabels=False,yticklabels=False)

#heat_map = sns.heatmap( theta,cmap=sns.cubehelix_palette(as_cmap=True))
plt.title( "Number of common neighboors between nodes" )
plt.show()
```



```
print("Number of common neighboors between node 0 and node 1 is ", num_common_neighbors[0,1])
print("Number of common neighboors between node 0 and node 25 is ",num_common_neighbors[0,25])
print("Number of common neighboors between node 0 and node 48 is ", num_common_neighbors[0,48])
print("Number of common neighboors between node 25 and node 48 is ",num_common_neighbors[25,48])
```

```
Number of common neighboors between node 0 and node 1 is  7.0
Number of common neighboors between node 0 and node 25 is  6.0
Number of common neighboors between node 0 and node 48 is  1.0
Number of common neighboors between node 25 and node 48 is  2.0
```

Answer b

Number of common neighboors between node 0 and node 1 is 7.0

Number of common neighboors between node 0 and node 25 is 6.0

Number of common neighboors between node 0 and node 48 is 1.0

Number of common neighboors between node 25 and node 48 is 2.0

Part c

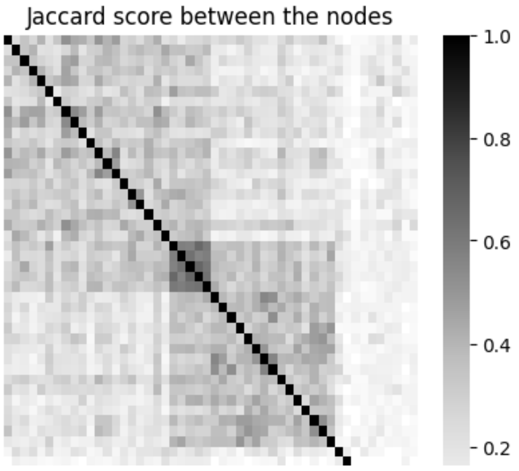
```
jaccard_score= np.zeros((dataframe.shape[0], dataframe.shape[1]))

for i in range(dataframe.shape[0]):
  for j in range(dataframe.shape[1]):
    c_neighbors=nx.common_neighbors(network,i,j)
    neighboors_i=network.neighbors(i)
    neighboors_j=network.neighbors(j)
    num_n_i=0
    num_n_j=0
    for nodes in neighboors_i:
      num_n_i=num_n_i+1
    for nodes in neighboors_j:
      num_n_j=num_n_j+1
    jaccard_score[i,j]=num_common_neighbors[i,j]/(num_n_i+num_n_j-num_common_neighbors[i,j])



plt.figure(figsize=(5,5))
heat_map = sns.heatmap( jaccard_score,cmap="Greys",xticklabels=False,yticklabels=False)

#heat_map = sns.heatmap( theta,cmap=sns.cubehelix_palette(as_cmap=True))
plt.title( "Jaccard score between the nodes" )
plt.show()
```



```
print("Jaccard score between node 0 and node 1 is ", np.round(jaccard_score[0,1],3))
print("Jaccard score between node 0 and node 25 is ",np.round(jaccard_score[0,25],3))
print("Jaccard score between node 0 and node 48 is ", np.round(jaccard_score[0,48],3))
print("Jaccard score between node 25 and node 48 is ",np.round(jaccard_score[25,48],3))
```

```
    Jaccard score between node 0 and node 1 is  0.226
    Jaccard score between node 0 and node 25 is  0.222
    Jaccard score between node 0 and node 48 is  0.048
    Jaccard score between node 25 and node 48 is  0.105
```

Answer c

Jaccard score between node 0 and node 1 is 0.226

Jaccard score between node 0 and node 25 is 0.222

Jaccard score between node 0 and node 48 is 0.048

Jaccard score between node 25 and node 48 is 0.105

Part d

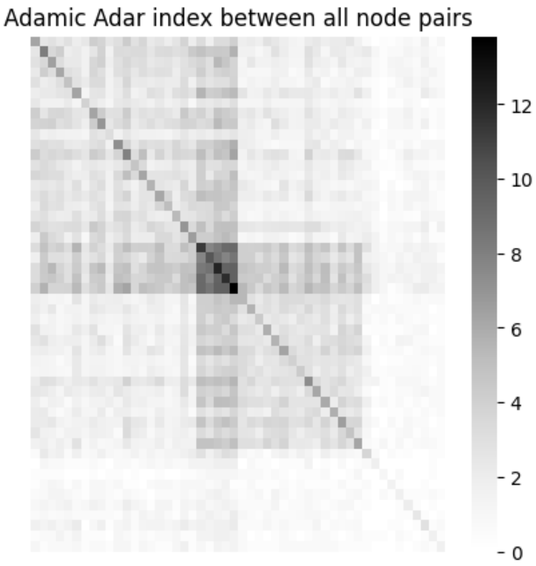
```
adamic_adar= np.zeros((dataframe.shape[0], dataframe.shape[1]))

def num_neighbors_node(node):
  neighboors_node=network.neighbors(node)
  num=0
  for neighboor in neighboors_node:
    num=num+1
  return num

for i in range(dataframe.shape[0]):
  for j in range(dataframe.shape[1]):
    c_neighbors=nx.common_neighbors(network,i,j)
    sum_fact=0
    for node in c_neighbors:
      sum_fact=sum_fact+1/np.log(num_neighbors_node(node))
    adamic_adar[i,j]=sum_fact
```

```
plt.figure(figsize=(5,5))
heat_map = sns.heatmap( adamic_adar,cmap="Greys",xticklabels=False,yticklabels=False)

#heat_map = sns.heatmap( theta,cmap=sns.cubehelix_palette(as_cmap=True))
plt.title( "Adamic Adar index between all node pairs" )
plt.show()
```



```
print("Adamic adar index between node 0 and node 1 is ", np.round(adamic_adar[0,1],3))
print("Adamic adar index between node 0 and node 25 is ",np.round(adamic_adar[0,25],3))
print("Adamic adar index between node 0 and node 48 is ", np.round(adamic_adar[0,48],3))
print("Adamic adar index between node 0 and node 48 is ", np.round(adamic_adar[0,48],3))
print("Adamic adar index between node 25 and node 48 is ",np.round(adamic_adar[25,48],3))
```

```
    Adamic adar index between node 0 and node 1 is  2.524
    Adamic adar index between node 0 and node 25 is  1.838
    Adamic adar index between node 0 and node 48 is  0.328
    Adamic adar index between node 25 and node 48 is  0.74
```

Answer d

Adamic adar index between node 0 and node 1 is 2.524

Adamic adar index between node 0 and node 25 is 1.838

Adamic adar index between node 0 and node 48 is 0.328

Adamic adar index between node 25 and node 48 is 0.74

✓ 0s completed at 2:31 PM