Course: Social Computing(CSC555)
Instructor: Dr. Munindar P. Singh
Description: Fall 2016, ***P2: Social Analytics using Graph Databases***
Student Name: Himangshu Ranjan Borah
Student ID: 200105222
Unity ID: hborah

Graph properties:

The project tries to set up the social ego networks of different user using the SNAP Facebook Dataset and Neo4j to calculate some of the measures of the Graph and also Proving/Disproving the two Hypothesis selected in the voting process of the first part.

The program code is included along with this submission. the ReadMe files includes all the necessary information needed to run the code. Some of the most important points about the whole project is noted here. The edges of the Egonet are given twice for the bidirectional relationships, so they are added twice for signifying the two sided relationships. The Features and the Circle Ids of the Nodes are added as properties of the Nodes. The following stoichiometric properties are calculated from the graph and reported to the console.

We also add the EGO node to all it's ALTERs to complete the graph. So, all the measures calculated takes into consideration those connections too.

Sample Output from the program:

```
Going to Create The Graph for EGO NODE = 0
:::::The Graph No. of Nodes and Edges :::::
The no. of Nodes = 334
The no. of Edges/Relationships(Taking the relationships between
the altars twice for bi-direction) = 5704
```

Also, we calculate the Between ness Centralities and the Clustering coefficients for every node in the Ego Network, some of the outputs are shown here. All the outputs are not shown for space constraints.

:::::The Graph Betweenness Centralities :::::
Betweenness Centrality of Node with ID: 236 is = 65.335696310218
Betweenness Centrality of Node with ID: 186 is = 65.77287115783334
Betweenness Centrality of Node with ID: 122 is = 209.04031167337536
Betweenness Centrality of Node with ID: 285 is = 123.15833343956038
Betweenness Centrality of Node with ID: 24 is = 1.1144841269841272
Betweenness Centrality of Node with ID: 346 is = 45.12514430014429
Betweenness Centrality of Node with ID: 271 is = 224.10220037399432
Betweenness Centrality of Node with ID: 304 is = 158.1109910434331
Betweenness Centrality of Node with ID: 176 is = 2.6137454120190693
Betweenness Centrality of Node with ID: 9 is = 134.11359319779032
Betweenness Centrality of Node with ID: 130 is = 20.600011100011102
Betweenness Centrality of Node with ID: 329 is = 110.81000389063269
Betweenness Centrality of Node with ID: 204 is = 22.247961760461767 etc.

:::::The Graph Clustering Coefficients :::::
The Clustering Coefficients of the Nodes are Below : mapped By Node IDs
Node ID = 236 :: Clustering Coefficient = 0.4099099099099099
Node ID = 186 :: Clustering Coefficient = 0.4978858350951374
Node ID = 122 :: Clustering Coefficient = 0.38556067588325654
Node ID = 285 :: Clustering Coefficient = 0.43293246993524515
Node ID = 24 :: Clustering Coefficient = 0.9
Node ID = 346 :: Clustering Coefficient = 0.5299145299145299
Node ID = 271 :: Clustering Coefficient = 0.386986301369863
Node ID = 304 :: Clustering Coefficient = 0.3872053872053872
Node ID = 176 :: Clustering Coefficient = 0.7252747252747253
Node ID = 9 :: Clustering Coefficient = 0.39724310776942356
Node ID = 130 :: Clustering Coefficient = 0.35833333333333334
Node ID = 329 :: Clustering Coefficient = 0.22528735632183908
Node ID = 204 :: Clustering Coefficient = 0.5887445887445888
Node ID = 213 :: Clustering Coefficient = 0.30364372469635625
Node ID = 252 :: Clustering Coefficient = 0.4033653846153846
etc.

Hypothesis Testing for the selected hypothesis:

The Hypothesis that i will be testing in my code are below:

- **Hypothesis 1: <dding3>-<01>** People who graduated from or are currently enrolled in the same university are more likely to connect with each other and form a social circle.

- **Hypothesis 2: <ssingh31>-<01>** Shorter the path between two nodes, more likely they tend to be in the same circle.

Hypothesis 1:

Here we are trying to prove that the "People form same university" and the "people from same circle" are correlated variables. We first set up an experiment where we pick up random samples of 2 users from any network(No. of samples is a tweak able parameter) and then check two things for them. First are they from the same university? This can be done by checking the feature vectors from 24 to 52 index values in the respective feature vectors for the nodes. The function proveHypothesisOne() has the code for it. Second, we see are they from the same circle? These two pieces of information are then stored in two different series. Like this we do the experiment N times where N is the no of times we do sampling. Finally, we have two vectors of TRUE and FALSE values. The rows correspond to one sampling of 2 users from the population. First series, TRUE if they from same university, FALSE if not. The second vector is similar, just for circles. Then we find the matches in the two variables and also find Pearson correlations by mapping the boolean values to 0 and 1. From the code output,

```
Testing the First Hypothesis : <dding3>-<01>
The True - True Matches in the SeriesA and B = 30
The False - False Matches in the SeriesA and B = 8494
Total No of Samples = 10000
Correlation Between the Double Series = 0.1302736827696469
```

We run this experiment for all the Networks and for different sample sizes and saw that the correlation generally is very less. It didn't seem like there was a strong inference between being in the same university and in the same circle. Hence , based on the experiments we did, the **claim has to be rejected.**

Hypothesis 2:

The second hypothesis states that shorter the path between two nodes, more likely they tend to be in the same circle. This point can be alternatively stated as, "**the people in the same circle are more likely to have less average shorts path lengths as compared to the people from different circles.**" To prove this, we again do a similar approach as before. We take random samples of 2 users overtime and find out what is the shortest path length between them. Then we add the user tuple's best path length to one of the following pools. The first pool, if they are in the same circle, the second pool, if they are in different circles. Thus we again get 2 series. We expected that the mean of the Same Circle Series be less that the mean of the Different Circle Series for the hypothesis to be true. And this is exactly what we found in the experiments that we did. The following in the output from the program.

```
Testing the Second Hypothesis : <ssingh31>-<01>
No of people in the same circle from this Sample = 1478
No of people in the different circle from this Sample = 5393
Sample Size = 10000
The Average Shortest Path Length of People in Same Circle =
1.8504736129905277
The Average Shortest Path Length of People in Different Circle =
1.954199888744669
```

So based on the experiments, we will prefer to **accept the hypothesis.**

For the hypothesis testing part, whatever we have done is based on pure experimental observations of what we felt. Given more time, we can set up a more appropriate statistical testbed and perform standard hypothesis testing using Z-Test and ChiSquare tests for these variables to formally prove or disprove the points for the whole population.