Name: Rachel Young
DS210 Final Project

Explanation of Dataset:

The dataset describes connections between friends on a social network, specifically Facebook. It is an undirected graph with nodes representing individuals and the edges representing friendship connections between each person. As provided by the dataset statistics on the site, there are 4,039 nodes and 88,234 edges.

Inspiration:

I used the ideas provided in the project examples Professor Kontothanassis posted earlier in the semester to decide what my graph analysis should look like. I was curious to understand how social networks can be analyzed and interpreted as they are complex networks with thousands of nodes. I decided to calculate the average distance between node pairs in the graph and calculate centrality measures to determine the nodes with the most connections.

About the Project:

This project will analyze the graph to determine the average distance between all the pairs of nodes to evaluate on average how many friends of friends you would need to go through in order to reach every other person on the graph. I also wanted to determine centrality by finding the top 10 nodes with the greatest degrees, the number of vertices a node is connected to, as a representation of the people with the greatest number of friends in the social network and would most likely have the greatest amount of influence, popularity, and connections within in the social network.

Explanation of Code Design:

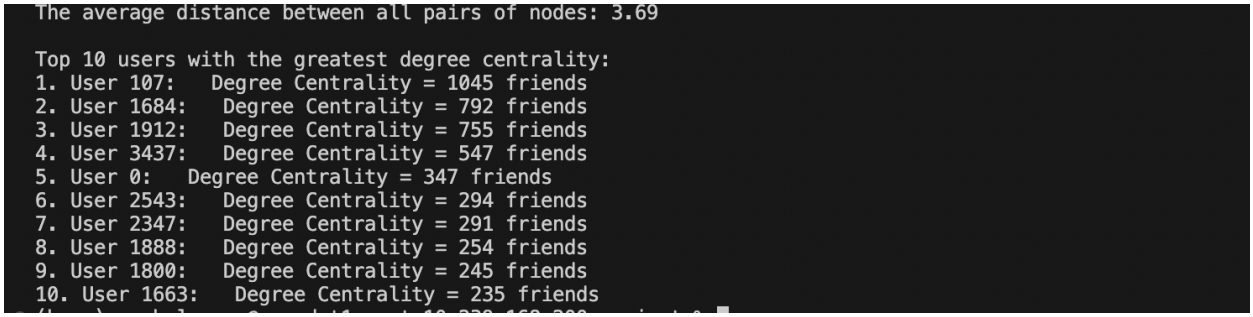
The code has a main.rs that includes the two tests of functions that are used to calculate average distance between nodes and degree centrality. It also has code that will print the calculated results. There are 2 modules: reader.rs and analysis.rs. The reader.rs has code that reads the .txt file and returns a vector of all the pairs of nodes. The analysis.rs has code that uses a struct that creates the adjacency lists that will be analyzed by the following functions. The compute_distances_bfs function that will search through the graph using breadth first search in order to record the distances between nodes and determine the average of all those distances. The compute_degree_centrality function calculates the number of edges each node has. These edges represent the number of friend connections each person, or node, has.

1) Average distances: The average distance shows how interconnected the graph is. If the results are low then the social network has greater connectivity meaning that on average you are fewer connections away from everyone else in the network. I used breadth first search to find the distances between a node and all other nodes in a graph. Then, I calculated the average of all the distances collected.
1) Centrality Measures: Since it is a social network, individuals in the graphs with more friends will be considered the most popular people in the social network. After reading the *Medium* article linked on Piazza, I choose to select node degrees as my centrality

measure, which is equal to the number of node neighbors. My interpretation of the output will be that the calculation will determine the top 10 most influential people in the graph.

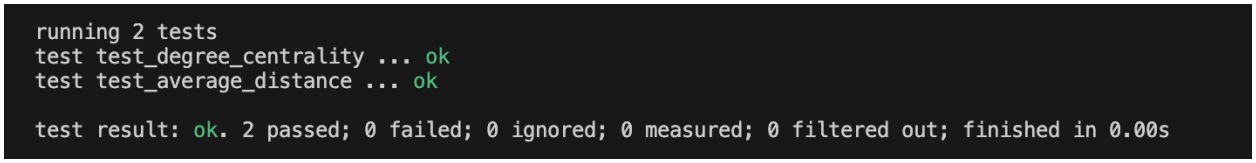
How to run the code

You would enter 'cargo run' to get the average distance between all nodes. It will take a moment to run, but the code output should look like the following:



To run the tests on the code, you would enter 'cargo test'. The first test is used to check that the calculated average distance between the nodes is greater than 1, which is what you would expect of a graph that has any edges.

The second test is used to check that a sample graph provided outputs the expected degree centralities for each node using the functions coded. The output of the tests should look like the following:



Results & Interpretation

My interpretation of the average distance between nodes being 3.69 is that the graph is highly interconnected with many individuals having friend connections with other nodes rather than being more distantly connected by friends of friends. On average, nodes are 3.69 friend connections apart from any other node in the network.

My interpretation of my degree centrality results is that user 107 has the greatest number of friend connections in the social network with 1045 friends, and can be considered a popular and well known individual. Subsequently, the 10 users returned would have the greatest number of connections within this social network, which says something about their popularity within this social circle and in society.

Also, looking at the pattern of the degrees of the top 10 users, they do somewhat follow the power-law distribution with degree centrality exponentially decreasing as you go down the top 10 user list, which is another point of analysis that could be further explored and evaluated in the future.

Sources/Citations for Dataset & Code

The dataset was retrieved from the Stanford Network Analysis Project: https://snap.stanford.edu/data/ego-Facebook.html. The aggregate data set which has the friends of everyone else is here: https://snap.stanford.edu/data/facebook_combined.txt.gz.

The dataset citation:

J. McAuley and J. Leskovec. Learning to Discover Social Circles in Ego Networks. NIPS, 2012.

References

Aomar, A. A. (2021, December 15). "Notes on graph theory — Centrality measures" Towards Data Science, *Medium*. https://towardsdatascience.com/notes-on-graph-theory-centrality-measurements-e37d2e49550a