Nora Amer
DS210 Project

introduction
For the final project, I wanted to analyze a graph dataset about facebook following/friendship. I used the "Social circles: Facebook" dataset from SNAP (Stanford) that features an undirected graph of nodes (people on facebook), with edges representing friendship on facebook. The data featured 4039 nodes, and 88,234 edges (friendships). I personally am on social media very often, and wanted to analyze how interconnected connections on social networks, specifically facebook, can really be. Have you ever wondered how far on average you are from another peer on facebook? How many people are you away from a celebrity? Using this sample of data, I wanted to see how far on average people are away from each other, and also who is most connected within the sample of people. I decided to investigate this using BFS (breadth first search) and computing the average distance between nodes/degree centrality.

readfile.rs
I essentially created a module called readfile.rs that reads the .txt file and essentially formats it into a vector that has a node and its 'outputs'/other connected nodes This is formatted into a long vector in order for it to be accessible for my functions in bfs.rs.

bfs.rs
Bfs contains my functions for BFS as well as my degree centrality analysis. I created functions like "reverse_edges", "add_direction", "sort_graph_lists" and "create_directed" to aid in the process of creating an adjacency list with the function "create_adjacent" which uses the undirected graph. I have the function compute_distances_bfs- which essentially computes the distances of the nodes using breadth first search. Also, to compute average distances between nodes (people) I implement the function compute_average_distance - which essentially takes the average of the distances created with the BFS function. Lastly, I implement the function compute_degree_centrality which essentially computes how many connections (friends) a node (person has).

main.rs
In my main, I implement my readfile module to read the facebook .txt file. I also assign variables to functions I defined in my BFS in order to run it and receive values. I do this for my average distance and node centrality, ultimately to be able to organize the top 5 most connected users within the dataset.

how to run
1. Download "DS210 project"

2. Run the folder ‘finalproject’ in terminal
3. Use ‘cargo run’ to get main.rs results (cargo run –release also works too)
4. Use ‘cargo test’ to see test results

analysis

Essentially, with my code I was able to find out that on average (based on this data) you are 3.96 people away from another person while being friends on facebook. So, essentially, you are estimated to be on average 4 friends away from another person, based on this sample of facebook data. Also, when looking at the top 5 most connected individuals within the network- the most connected individual had 1045 friends. The second had 792 friends, the third had 755 friends, the fourth had 547 friends, and the fifth had 347 friends. It is interesting to see how many connections can be made out of a dataset with 4039 people- for example, an individual can have almost a fourth of the connections.

citations

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

https://doc.rust-lang.org/book/