**Report for Home Work 2 CME2201**

Student number: 2020510130

Student name: Rinat Zhulfayev

Main task of the Home Work is to find Centrality metrix(betweenness and closeness) for the given 2 graphs.

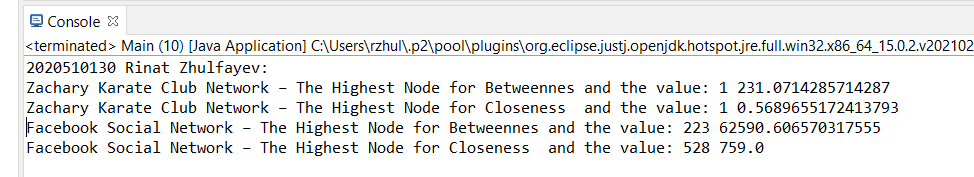
So, in way to find betweenness of the node(lets say node1), I need to find number of all shortest paths between nodes except node1 and search how many times node1 exist in path between this two nodes. Then delete this number to number of all paths. Do it for all pair nodes and sum values.

In way to find closeness(for node1) I need to find all shortest paths between node1 and all other nodes. Sum lengths of these paths(sum1). Then divide total number of vertices to sum1.

To accomplish this task, I used 2 algorithms:

1)Algorithm that uses DFS traversal for finding all paths, but gives only shortest ones. Algorithm is easy to write, but it takes very long time to execute it with large amount of data. Although for small graph it is perfect since ease to implement.

2)Algorithm that uses BFS instead of DFS. Initially, it only gave one path between two nodes and I had to modified it. And most probably one of the parts of this algorithm works inefficient and affects the execution time.



Then I decided to watch lesson, on which home work was described. And I understood that actually I don’t have to find all existed shortest paths between two nodes(one is enough). But it was too late to change the code.

I think it’s even better this way, since I made the task more difficult for myself. And of course, the algorithm calculates the actual values, which is more practical.