CENTRALITY IMPLEMENTATION

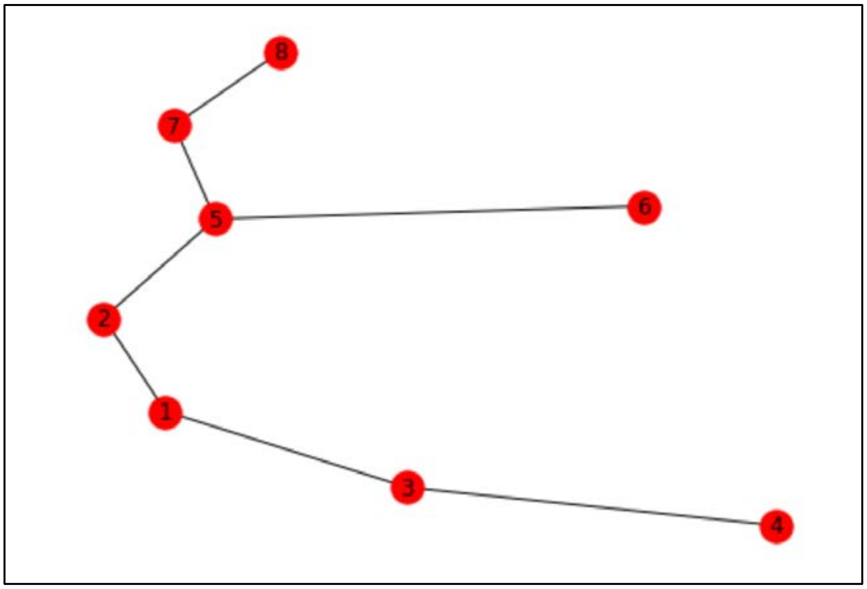
Kushal Hebbar - C13031425 | Tamanna Baig – C13768206

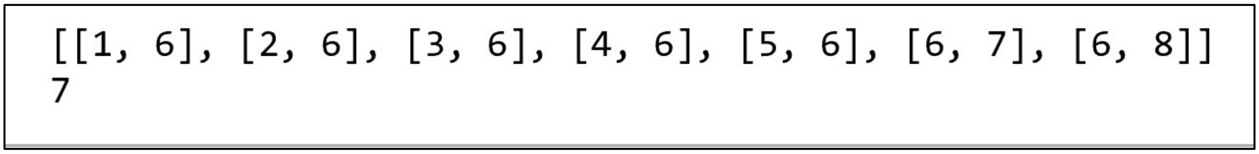
khebbar@clemson.edu | tbaig@clemson.edu

- We have decided to implement Traversal sets centrality index using python and Networkx graph library.

Traversal sets are represented as $|T_{ij}|$where i-j is the edge that is along the shortest path between vertices s,t.

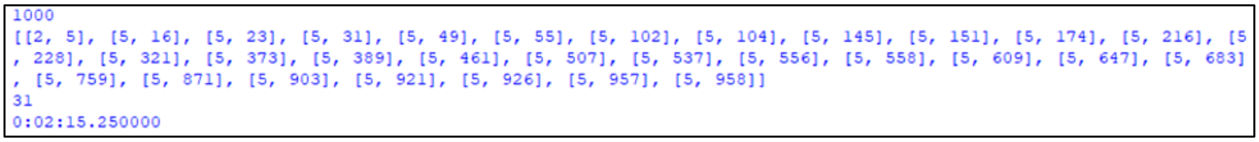
We first made use of 8 nodes and built our algorithm around the graph shown below to better analyze the results. Below is the graph and the traversal set as a result of our implementation.





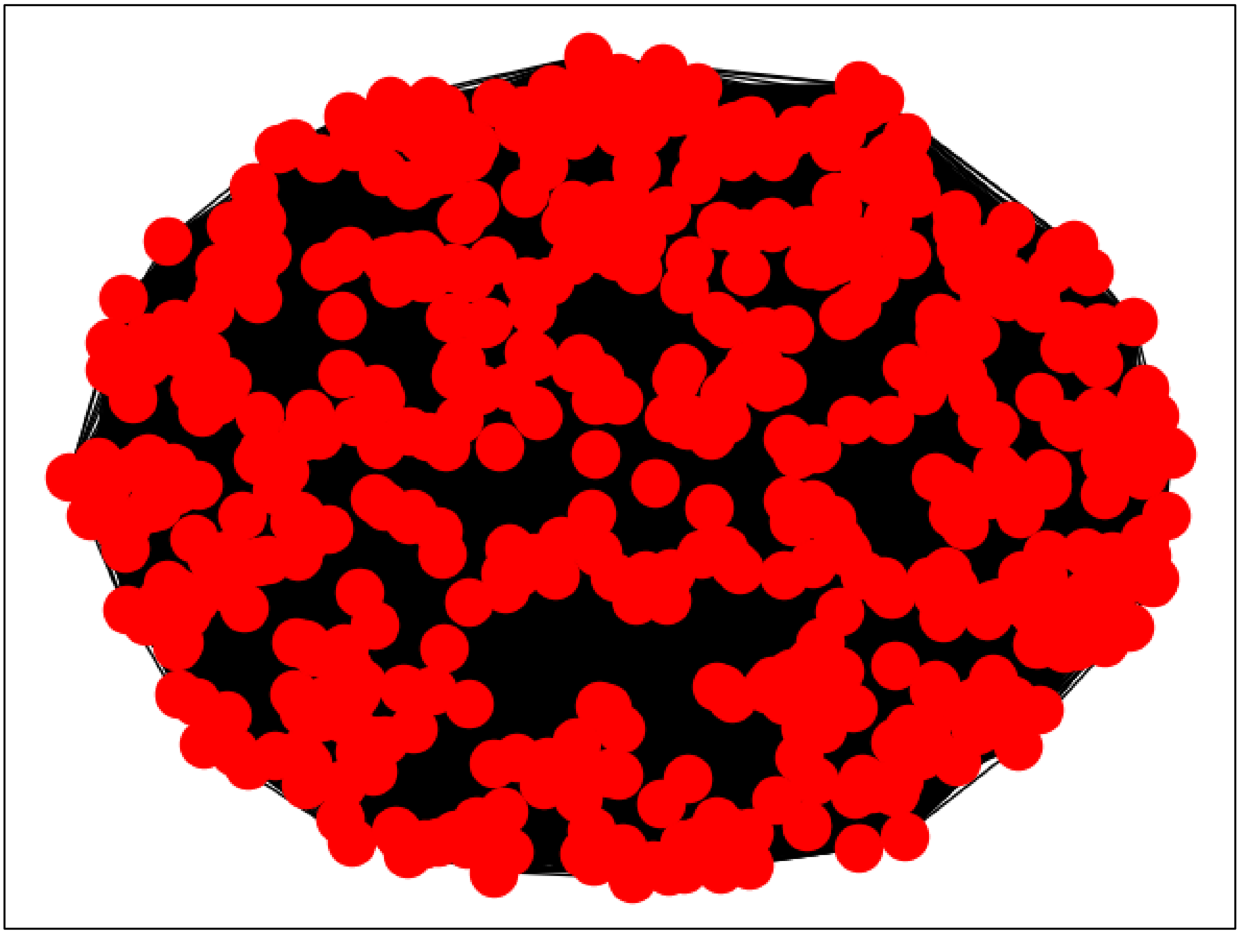
```
[[1, 6], [2, 6], [3, 6], [4, 6], [5, 6], [6, 7], [6, 8]]
7
```

We then made use of fast_gnp_random_graph to generate a random graph and implemented traversal set algorithm on that graph after converting the node labels to integers. The traversal set on 1000 nodes is as follows, for further analysis we implemented the time function to calculate the amount of time taken for program execution.



```
1000
[[2, 5], [5, 16], [5, 23], [5, 31], [5, 49], [5, 55], [5, 102], [5, 104], [5, 145], [5, 151], [5, 174], [5, 216], [5
, 228], [5, 321], [5, 373], [5, 389], [5, 461], [5, 507], [5, 537], [5, 556], [5, 558], [5, 609], [5, 647], [5, 683]
, [5, 759], [5, 871], [5, 903], [5, 921], [5, 926], [5, 957], [5, 958]]
31
0:02:15.250000
```

Further analyses was done by replacing fast_gnp_random_graph with random_regular_graph with number of nodes set to 10,000 and degree 3.



Please find attached the source code named as traversal.py.