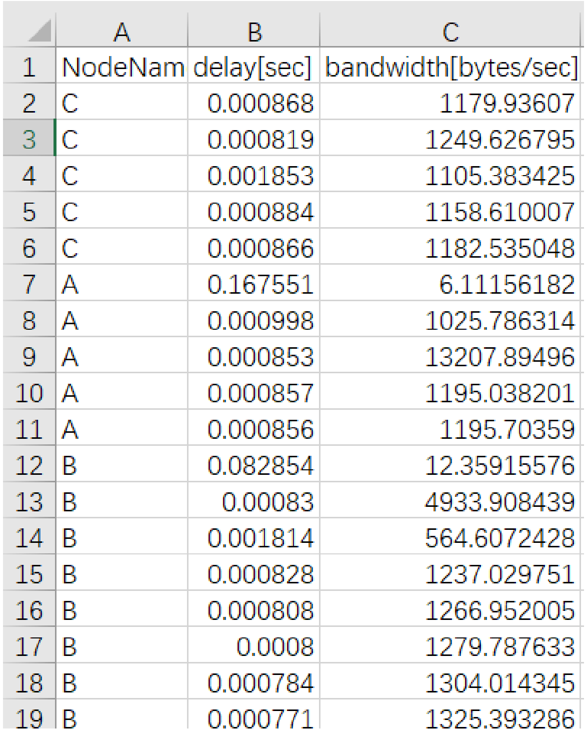
Mp0 Report

Yuhang Chen 3170111144
Wenbo Ye 3170112243

1. Outline of the codebase and introduction on the design
   As mentioned in the mp0.pdf, the main task of mp0 is to make a log file logging the data transmission between the client and server. My code is based on python language and work with TCP in python socket. The node.py file work as the client file to send messages produced by the local generator to the server (where we simply use the local server). The logger.py work to make logs in the local machine by receiving data from the server and doing the data processing and do data processing for drawing the graphs needed.

2. Libraries or package used in the codebase
   Sys, socket, time, csv, threading

3. Description on how to measure the delay and bandwidth
   The delay can be measured by the difference between the actual time we receive the message on the local server and the actual time that the message was sent from the node. Specifically, the actual time we receive the message can be recorded as we call time.time() in the logger.py file. The actual time we sent the message is recorded and sent by the node.py file and recorded as the log time in the log record. The bandwidth can be measured by the transmitted bytes per seconds from the client to the server, since we use the local machine as both the client and the server, we can simply use the delay as the difference of timestamps between the two node(which is actually one node). which means the transmitted bytes divided by the delay.



	A	B	C
1	NodeNam	delay[sec]	bandwidth[bytes/sec]
2	C	0.000868	1179.93607
3	C	0.000819	1249.626795
4	C	0.001853	1105.383425
5	C	0.000884	1158.610007
6	C	0.000866	1182.535048
7	A	0.167551	6.11156182
8	A	0.000998	1025.786314
9	A	0.000853	13207.89496
10	A	0.000857	1195.038201
11	A	0.000856	1195.70359
12	B	0.082854	12.35915576
13	B	0.00083	4933.908439
14	B	0.001814	564.6072428
15	B	0.000828	1237.029751
16	B	0.000808	1266.952005
17	B	0.0008	1279.787633
18	B	0.000784	1304.014345
19	B	0.000771	1325.393286

Figure 1. The screenshot on a csv file of some trial in logging

4. Testing

   Since we need to accurately print out the log file, we can do the testing on the cmd window and check if there exits bugs or something unreasonable by typing

   ```
   timeout 5 && python3 -u logger.py 8080 | python3 -u generator.py 10 20| python3 -u node.py A 127.0.0.1 8080 &
   timeout 5 && python3 -u logger.py 8080 | python3 -u generator.py 10 20| python3 -u node.py B 127.0.0.1 8080 &
   timeout 5 && python3 -u logger.py 8080 | python3 -u generator.py 10 20| python3 -u node.py C 127.0.0.1 8080 &
   ```

   By printing some variables we use in the codes also help a lot to figure out the bugs in the programming procedure (such as the deltat\trans_bytes)