Kevin Wissink

2/22/2020

CSCE – 313 – 501

UIN: 826005542

PA2 – Client to Server Interactions

Introduction:

This project is about teaching us to write to a sever and read what comes back from it. In this case we are given 15 patients that all have three columns of data and time. We have different cases based on what is command line inputted. If they input a patient number, we return a copied version of the file to the user in the received folder. If they give us a new folder, we then return a copied version of the inputted file to the user. The user can also ask for specific data points and new channels to direct this information over.

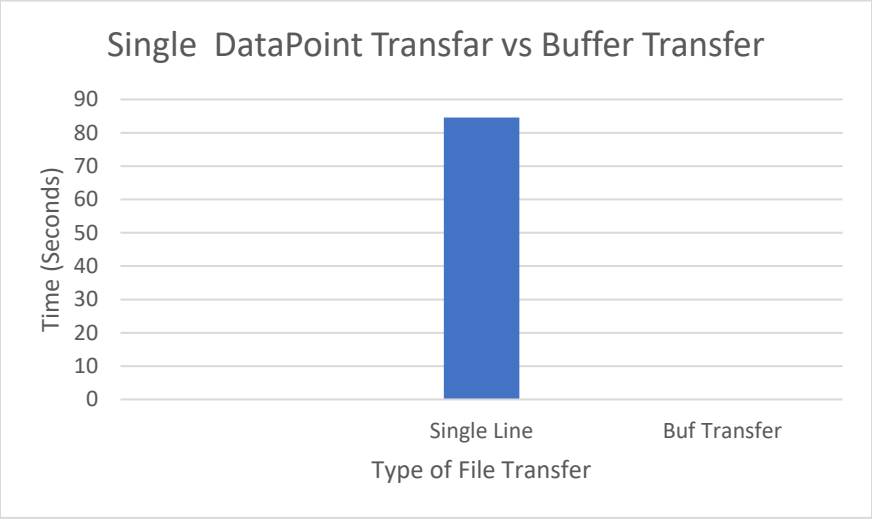
Data:

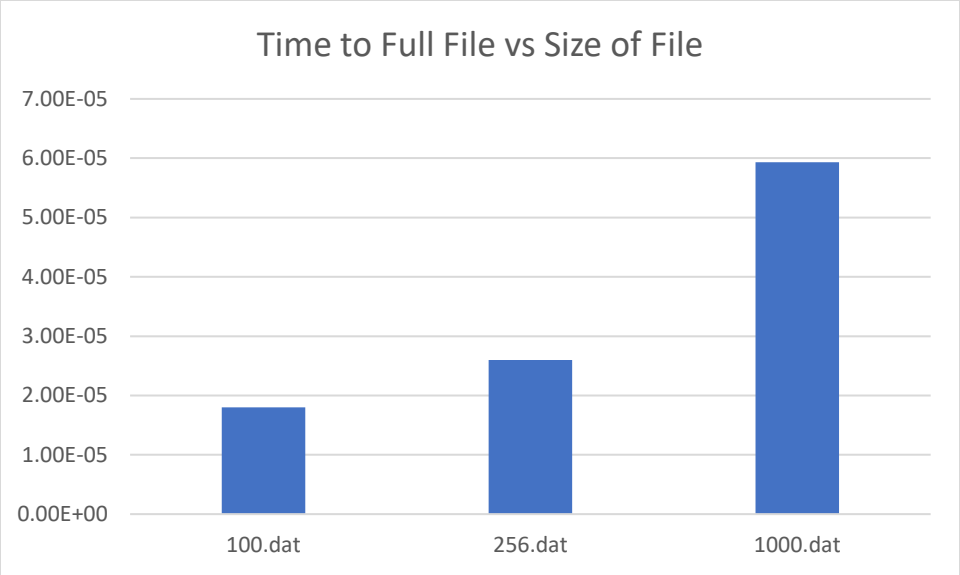
Single Line	84.5328
Buf Transfer	0.02233

100.dat	1.80E-05
256.dat	2.60E-05
1000.dat	5.93E-05

All Data is given in seconds

Graphs:





Analysis:

Overall, I believe my data shows that as the MAX_MESSAGE increases the time to move data decreases. The part of this code that bottlenecks it the most is the buffer size the lager the size the more data that can be sent at once. Although I could not get the buffer size to be modular my data shows that using the buffers transfer which takes a chunk of data and sends it through the pipe is more efficient that trying to get each single data point. The other graph I included shows the change in time for a truncated file that is filled with random data.

Conclusion:

Overall, this project took me a long time to mentally figure out what was going on and how to go about doing it. I went on many nights dealing with errors and segmentation faults. In the end, my biggest issue was with going through the buffer and seeing what is being given back from the server and what is inside it… I forgot that each read can be as another type hence the __int64_t for the length of the file or a *(double*) if your reading a data point. Thanks for reading have a great day/night!