**Programming Assignment 1**

# Design of Code

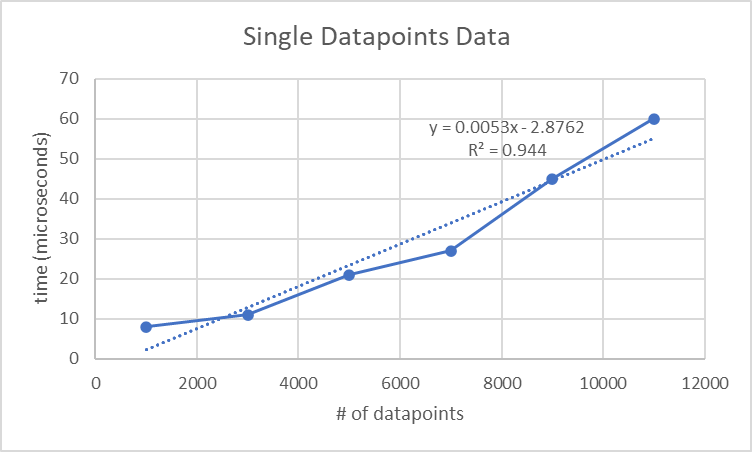
In this assignment, I wrote a client program that connects to a given server. I made sure that the client and server could effectively communicate data points, CSV files, and binary files. I did this by using getopt to identify flags and used if statements to determine the action that needs to be taken based on the flags presented in the terminal. I opted out of using functions so that I could take advantage of some of the global variables included in the starter code. Additionally, I incorporated some functions like cwrite(), cread(), and memcpy() in order to effectively communicate between the client and server.

Through this Programming Assignment I successfully learned how to:

* Run server as a child of the client
* Request data points
  + a single data point
  + 1000 data points
* Request files (with and without differing buffer capacity)
  + CSV file
  + binary file (of differing sizes)
* Request a new channel
* Close channels

# Single Datapoint Data

On average, requesting one data point takes about 4000 - 7300 microseconds to accomplish using the chrono::high\_resolution\_clock::now() function.

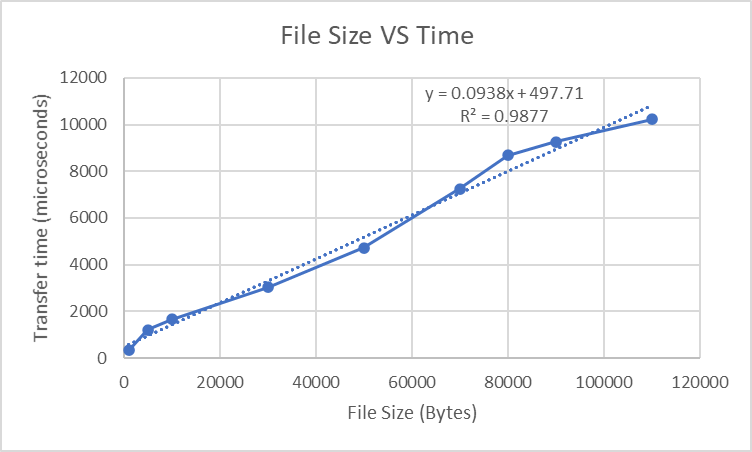


there is a strong (Shown by the strong R^2 value) positive linear relationship (shown by the line equation) between number of datapoints and time.

# 1000 Datapoints Data

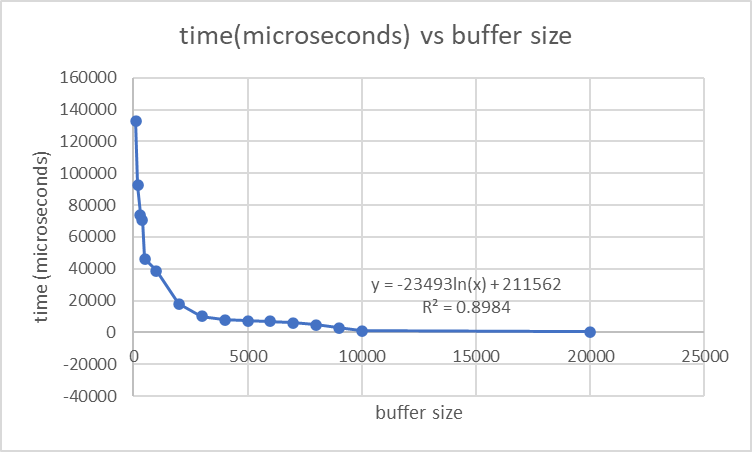
On average, the amount of time it takes for 1000 data points to be requested and put into a file is 2.5721\*10^7 microseconds using the chrono::high\_resolution\_clock::now() function.

# File Transfer Data



In this first instance of observing time and its relationship with file transfer, I looked at how file size relates to time. I increased the file size while keeping everything else consistent (including the default buffer size of 256 bytes). As shown above, there is a strong (Shown by the strong R^2 value) positive linear relationship (shown by the line equation) between file size and time.

| file size | time(microseconds) |
| --- | --- |
| 1000 | 323 |
| 5000 | 1214 |
| 10000 | 1654 |
| 30000 | 3021 |
| 50000 | 4713 |
| 70000 | 7245 |
| 80000 | 8678 |
| 90000 | 9256 |
| 110000 | 10231 |



In this second instance of observing time and its relationship with file transfer, I looked at how buffer size relates to time. I increased the buffer size while keeping everything else consistent.

As shown above, there is a strong (Shown by the strong R^2 value) negative logarithmic (shown by the line equation) relationship between buffer size and time. The graph shows that as buffer size increases it takes less time to transfer the file.

When I tested file transfer with bigger files I also got a similar logarithmic relationship and noticed a similar trend of decreasing transfer time with buffer size increases.

| buffer size | time(microseconds) |
| --- | --- |
| 100 | 132959 |
| 200 | 92646 |
| 300 | 74000 |
| 400 | 70735 |
| 500 | 46324 |
| 1000 | 38838 |
| 2000 | 18131 |
| 3000 | 10022 |
| 4000 | 8034 |
| 5000 | 7343 |
| 6000 | 6962 |
| 7000 | 5976 |
| 8000 | 4652 |
| 9000 | 3020 |
| 10000 | 1050 |
| 20000 | 502 |