Assignment 2 Writeup

1) Time 8 processes single threaded: 0.052 seconds, whereas for multithreading it takes 0.042 seconds. Which is a slight increase in performance, but at the same time I am not sure if I implemented multithreading correctly. The observed speed up is a 0.01 time increase

2) A likely bottleneck is my in my dispatcher, which is my main function and is due to how I handle the requests. I believe that there should be a more efficient way to send out multiple requests at the same time. I could increase concurrency in this area by finding a better structure to send out requests more efficiently. Also, there could be bottlenecks where we lock the offset variable every time that we want to modify it. To improve this, I could've created another function that solely takes care of logging to reduce the amount of locks in our program, which in turn would help to speed up our process.

3) In real life we would not log the entire contents of files because it is unnecessary in that we already know all the information needed without checking the Log file, for example, from the client we would know whether the request succeeded or whether or not there was an error, and for the contents of the files would be present in either the server side or client side if our program was working completely correctly.

How I tested my program: by running curl tests with an '&' character in between curl commands to run them all at the same time.

For ex. Two requests: Curl -v localhost:8080 --request-target someFile & localhost:8080 --request-target someFile & Curl -v localhost:8080 --request-target someFile & localhost:8080 --request-target someFile Four requests: Curl -v localhost:8080 --request-target someFile & localhost:8080 --request-target someFile & Curl -v localhost:8080 --request-target someFile & localhost:8080 --request-target someFile & Curl -v localhost:8080 --request-target someFile & localhost:8080 --request-target someFile & Curl -v localhost:8080 --request-target someFile & localhost:8080 --request-target someFile & Ect.