For this MP, we created a program that created a bunch of requests and sent them to the server. Later we would get the responses from the server and categorize them. First the program gets the input from the command line and stores them in the correct variables. Next it sets the correct buffer sizes. Next the worker threads are initiated and the requests are generated. Then three request threads are made, one for each person. Next the requests are deposited in the correct buffers. Then the items from the deposited items are removed by workers.

In test 2, increasing the number of workers decreased the time. The time was highest when the workers were fewest. However there was a point when adding workers slowed the time. If there were too many workers such as in test 1, the time increased however this only held true if the optimum number of workers was not reached. For example, when there was only 10 requests adding more workers greatly increased the time. However when there were 10,000 requests adding more workers decreased the time.

One problem that occurred was that the pipe would break if the workers were more than 110 workers. I tried adding more sleep time and that increased the amount of workers before it broke.