Dalton Harris and Ashton Binkley

MP5 Report

An increase of worked threads from 15 to 164 makes the program run faster. Worker threads fluctuating between 190 and 375 results in some higher and some lower times. Regardless, 164 worker threads (same intervals used as last time) was the most efficient, and resulted in the fastest completion. The highest time comes when there are 15 worker threads, which is the default. The program worked most efficiently when the buffer size is bigger, as it runs most efficiently. We found the buffer size of 500 (on average) to be the fastest at each of the worker thread intervals below, but the absolute fastest time was 1.6 seconds, and came with the following specifications:

\* Number of requests: 10,000

\* Worker threads: 164  
\* Buffer size: 50

Our final results are as follows.

(Buffer size of 500)  
\* 15 worker threads: 7.3 seconds

\* 100 worker threads: 3.6 seconds

\* 150 worker threads: 2.2 seconds

\* 164 worker threads: 1.7 seconds

\* 190 worker threads:  2.9 seconds

\* 200 worker threads: 2.5 seconds

\* 300 worker threads:  2.3 seconds

\* 350 worker threads: 2.4 seconds

\* 375 worker threads:  2.7 seconds

\* 390 worker threads: results in an overflow error

\* 400 worker threads: results in an overflow error

MP4 Time

(Buffer size of 500)

 15 worker threads: 7.25 sec

100 worker threads: 2.71 sec

150 worker threads: 1.4 sec

164 worker threads: 1.5 sec

190 worker threads: 2.4 sec

Compared to MP4, MP5 allows for the worker thread number to be higher as well as the buffer size to be smaller. Our best time for MP5 used a buffer size of 50 and 164 worker threads returning a time of 1.6 seconds. Our time of 164 worker threads and a buffer size of 500 resulted in the same time of 1.7 for both MP5 and MP4.

MP5 (Buffer 500)

Blue line: number of worker threads

Red line: seconds it took

MP4 (Buffer 500)

Blue line: number of worker threads

Red line: seconds it took

As the graphs show, MP5 was able to hold more worker threads and the timing was all around better.