

CS342

Operating Systems

Spring 2018

Project 2: Multi-threaded Server, Shared Memory and Synchronization

Yalchin Aliyev - ID: 21500571

Kanan Asadov - ID: 21503382

General Information

We measured the average execution time (in seconds) for values of $N = \{1, 3, 5, 7, 9\}$ and $L = \{20, 30, 40, 50, 60\}$, where N is the number of clients and L is the number of lines of the input file in thousands, in order to see how concurrently running clients and the size of the file affect the performance. We also checked the effect of the frequency of the keyword in the file on the execution time for values $F = \{0.05, 0.01\}$. Execution time was measured using `/usr/bin/time` program.

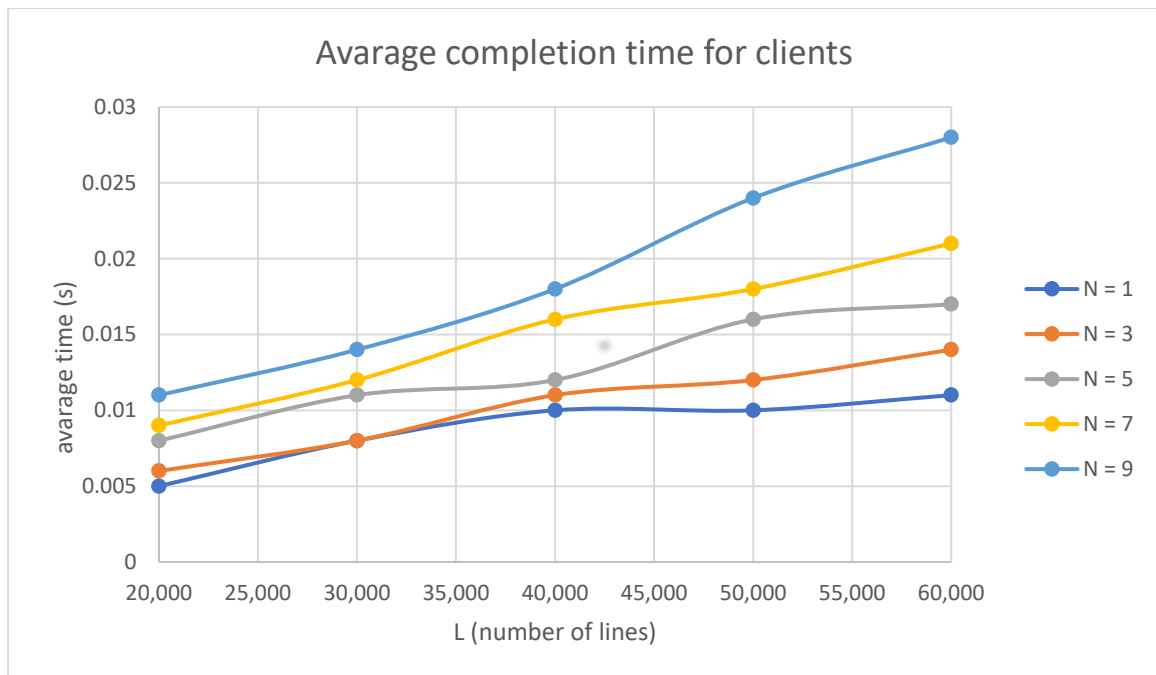
Environment and Setup

The experiments were run on Ubuntu 16.04 dual booted with 4 cores.

Results and Discussions

Line numbers in a file vs time for different number of clients running concurrently

<div>M</div> <div>L</div>	1	3	5	7	9
20,000	0.005	0.006	0.008	0.009	0.011
30,000	0.008	0.008	0.011	0.012	0.014
40,000	0.010	0.011	0.012	0.016	0.018
50,000	0.010	0.012	0.016	0.018	0.024
60,000	0.011	0.014	0.017	0.021	0.028



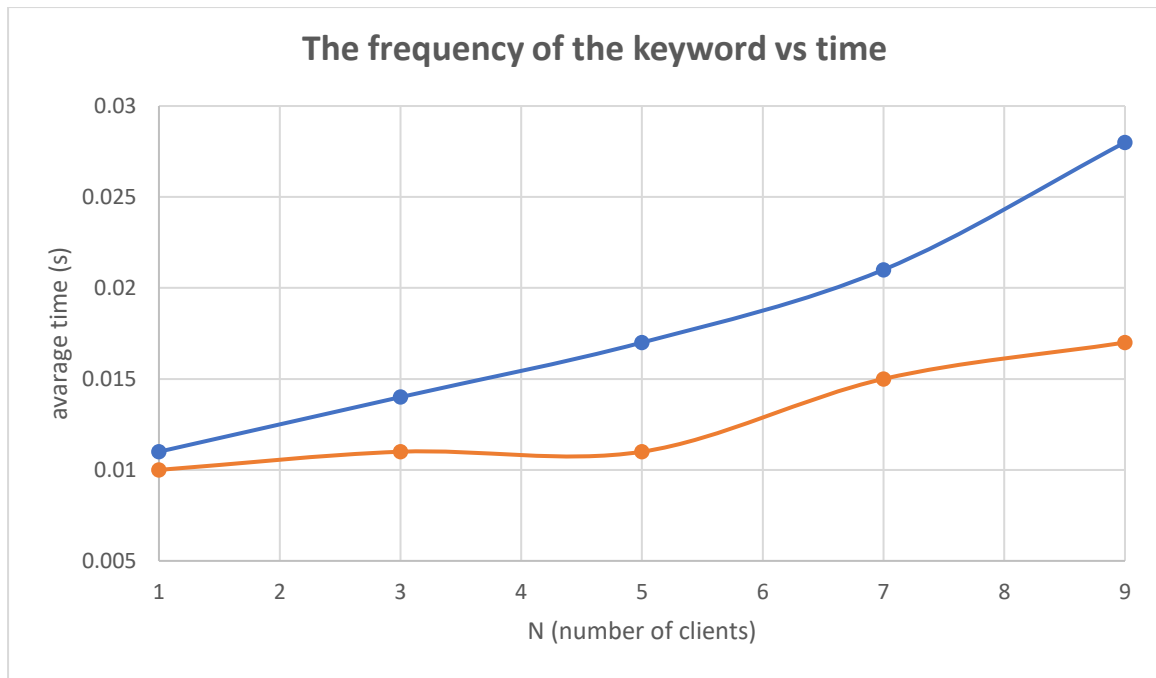
In this example, the number of keywords in the input file equals to $0.05 \cdot N$.

Increasing the number of clients increases the execution time because it increases the workload of the CPU. Furthermore, there is a sharp increase in execution time between 7 and 9 concurrent clients because the environment this program was ran on has only 4 cores that can handle 8 threads concurrently. When a 9th thread comes in at least one thread will be in the waiting state at a time which increases the average execution time.

The number of lines increases the average execution time since every thread needs to search for the keyword linearly from the input file.

The frequency of the keyword vs time for clients running concurrently

F \ M					
	1	3	5	7	9
0.05	0.011	0.014	0.017	0.021	0.028
0.01	0.010	0.011	0.011	0.015	0.017



In this example, the number of lines of the input file is 60000.

When the frequency of the keyword is increased, the thread handling the request must send more data to the client. Since the result queues has a limited size, it causes the thread to wait for an empty space in the buffer. Therefore, this extra waiting increases the average client completion time.