

All times below are the recorded real time. These are ran on my PC at home. I used a program to randomly generate 10M chars in 3 different files, where the probability of the same char repeated was 10%, 50%, and 90%. I used the time command to record how long it took from start to finish. This was the 'real' time.

* Process VERSION *

Process: 10M Random letters up to 10% chance for letter to be repeated

1 Process:0.506s

2 Processes:0.300s

3 Processes:0.223s

4 Processes:0.181s

5 Processes:0.218s

6 Processes:0.216s

7 Processes:0.186s

8 Processes:0.174s

Process: 10M Random letters up to 50% chance for letter to be repeated

1 Process:0.596s

2 Processes:0.348s

3 Processes:0.257s

4 Processes:0.211s

5 Processes:0.184s

6 Processes:0.169s

7 Processes:0.144s

8 Processes:0.164s

Process: 10M Random letters up to 90% chance for letter to be repeated

1 Process:0.508s

2 Processes:0.302s

3 Processes:0.228s

4 Processes:0.185s

5 Processes:0.174s

6 Processes:0.213s

7 Processes:0.204s

8 Processes:0.187s

* THREAD VERSION *

Thread: 10M Random letters up to 10% chance for letter to be repeated

1 Thread:0.539s

2 Threads:0.344s

3 threads:0.259s

4 threads:0.214s

5 threads:0.224s

6 threads:0.210s

7 threads:0.209s

8 threads:0.214s

Thread: 10M Random letters up to 50% chance for letter to be repeated

1 Thread:0.687s

2 Threads:0.395s

3 threads:0.285s

4 threads:0.279s

5 threads:0.316s

6 threads:0.265s

7 threads:0.252s

8 threads:0.224s

Thread: 10M Random letters up to 90% chance for letter to be repeated

1 Thread:0.596s

2 Threads:0.350s

3 threads:0.262s

4 threads:0.343s

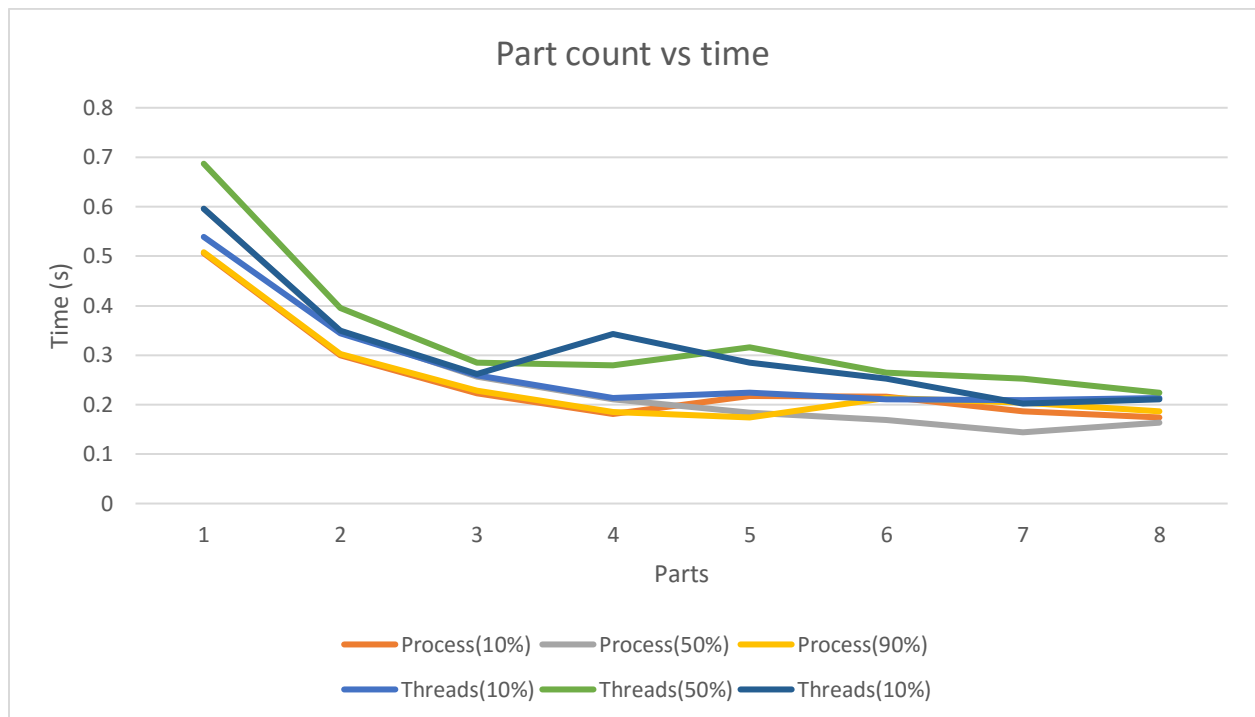
5 threads:0.285s

6 threads:0.252s

7 threads:0.202s

8 threads:0.210s

Compiling all the data together in an excel graph gives some interesting results. On average the Processes finished faster than the threaded version. Also after about 4 threads the time it took to finish was roughly constant for each run, so multi part did help to an extent. This may be due to not a bottleneck of the processor but the hard drive. This may be the fastest that my computer can read from the disk. As we learned in class one of the highest wait times is seeking from the hard drive, which is on the order of ms.



The process version is faster, even though I believed the threads would be quicker. I am not completely sure why this is the case, but it may have to do with loading the binary file into memory. It could also just be variables I cannot control such as other programs using cycles that take away time from my program. I thought by running it on my computer I could eliminate them. The time may be too little to generate reliable data without a large factor of error. I also feel running the tests many times over would only cause the average for every run to be the same.