CS 214: Systems Programming, Fall 2016
Assignment 2: Procs vs Threads (round 0)

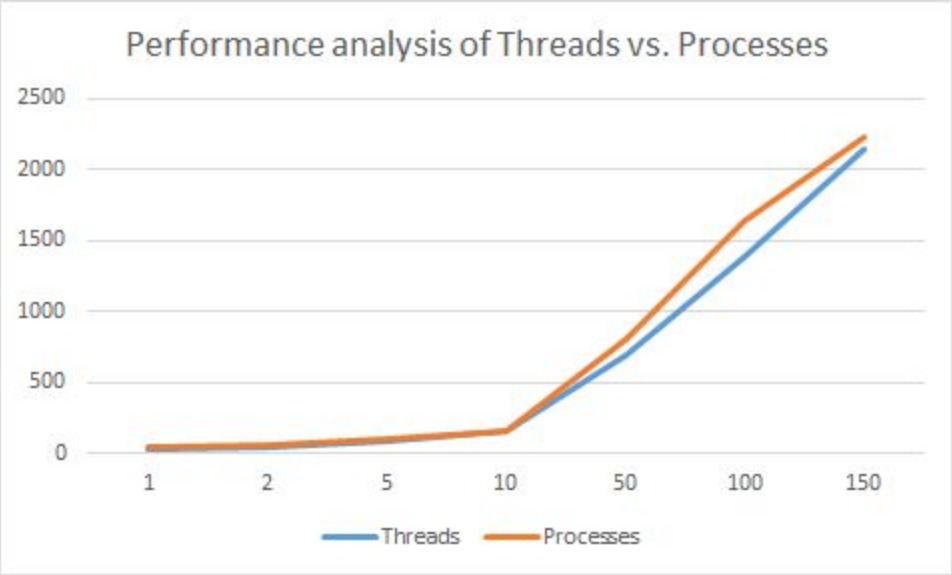
timetests.pdf

Jeffrey Huang, jh1127
Richard Li, rl606

We since utilized the same splitting algorithm and reading algorithm for both programs and that the thread and the process both utilize the compression method, we tested the timing by using the time command when executing the code.

time ./compressT_LOLS test1.txt 5 or time ./compressR_LOLS test1.txt 5

By running each program multiple times and taking their average then plotting them on a graph comparing the two run times, we would be able to see that the process program for compiling has a relatively longer runtime as the number of compression files generated increases and the number of events happening repeats more than one time.



From our analysis, the runtime for threads and processes are relatively the same when the number of partitions in the compression is low. However, as the number of partitions increase, the runtimes between the two separate and show that the threads would have a shorter runtime than processes. The reason behind this is that threads only deviate by one method while a process deviates by another program. As the number of compressions increase, more memory is needed to be allocated for a process while the thread just utilizes anything that is already provided.