CS214 Project 2

A Multi-threaded Sorter

Brian Huang & Matthew Stone

Only works 100% when running with VALGRIND :(

Analysis

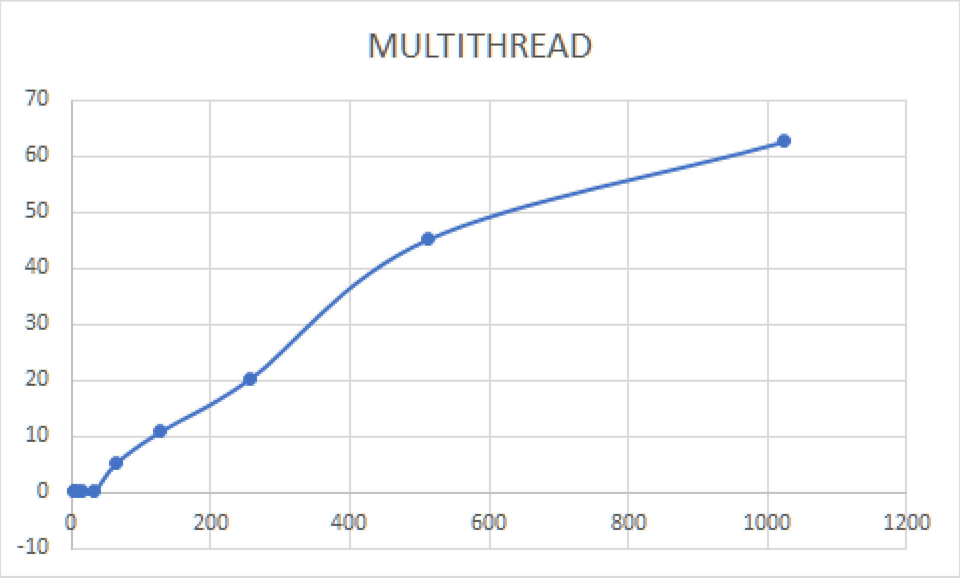
Multithread- The times are averaged 3 trials. Each file had 500 lines.

2	4	8	16
real 0m0.643s	real 0m0.719s	real 0m0.082s	real 0m0.285s
user 0m0.045s	user 0m0.077s	user 0m0.055s	user 0m0.114s
sys 0m0.009s	sys 0m0.027s	sys 0m0.016s	sys 0m0.032s

32	64	128	256
real 0m0.728s	real 0m5.461s	real 0m11.507s	real 0m21.204s
user 0m0.135s	user 0m5.293s	user 0m10.984s	user 0m20.340s
sys 0m0.056s	sys 0m0.385s	sys 0m0.933s	sys 0m2.063s

512	1024
real 0m48.133s	real 1m4.767s
user 0m45.236s	user 1m2.673s

sys 0m6.010s	sys 0m9.095s



USER
TIME GRAPHED

Multiprocess- Average of 3 trials, each file had 500 lines

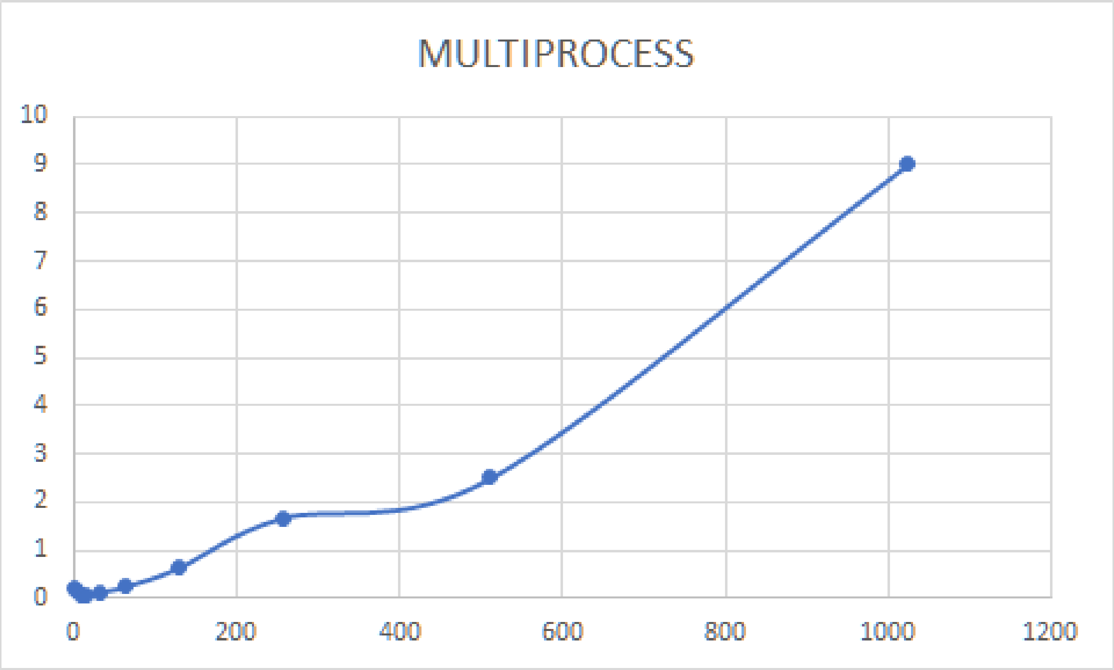
2	4	8	16
real 0m0.223s	real 0m0.144s	real 0m0.055s	real 0m0.058s
user 0m0.020s	user 0m0.012s	user 0m0.004s	user 0m0.005s
sys 0m0.005s	sys 0m0.006s	sys 0m0.004s	sys 0m0.005s

32	64	128	256
real 0m0.122s	real 0m0.243s	real 0m0.624s	real 0m1.668s
user 0m0.004s	user 0m0.003s	user 0m0.005s	user 0m0.005s

sys 0m0.006s	sys 0m0.012s	sys 0m0.020s	sys 0m0.038s

512	1024
real 0m2.496s	real 0m9.016s
user 0m0.006s	user 0m0.017s
sys 0m0.091s	sys 0m0.166s

Real time graphed



Is the comparison between run times a fair one? Why or why not?

No. Sometimes the ilab had high traffic, causing the processing speed of the current computer to slow down. Also, the programs aren't doing the exact same task, the multithreaded program is doing more.

What are some reasons for the discrepancies of the times or for lack of discrepancies?

Some reasons for the discrepancies of the times was due to the ilab being too populated, which caused the time to be slower. Our multi-process sorter seemed to be faster than our threaded sorter. This may be due to not having to merge all of the sorted files into one or maybe starting a p-thread requires more time than starting a new process.

If there are differences, is it possible to make the slower one faster? How? If there were no differences, is it possible to make one faster than the other? How?

Yes, it is possible to make the slower one faster. We could implement the multiprocess sorter to be more efficient with multiprocessing. We could also try implementing multithreading together with multiprocessing to take full advantage of parallelism.

Is mergesort the right option for a multithreaded sorting program? Why or why not?

Mergesort is the best sorting algorithm for a multithreaded sorting algorithm. This is because mergesort allows the program to divide and conquer in parallel. It allows the program to sort smaller chunks of data in parallel, and then merge them together efficiently.