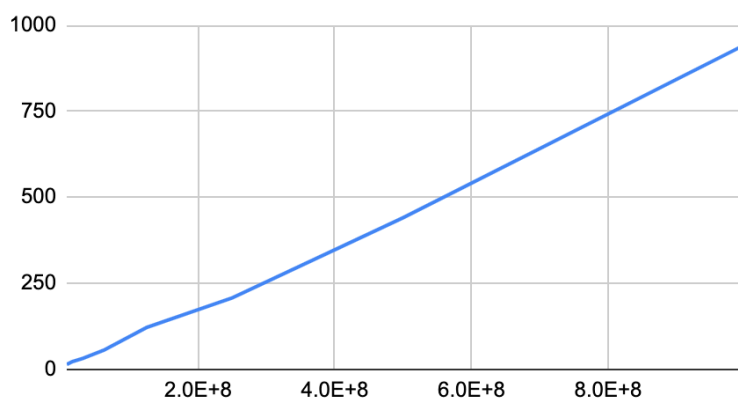


Benjamin Katz
CountString Writeup

In general, parallel programs do not work particularly well for relatively small inputs. The cost of creating and setting up the infrastructure of threads eats up at the cost savings produced by running parallel. With large inputs, the benefits of running a parallel program far outweigh the cost of setting up the infrastructure. With an input of one billion, I was able to run the program three times faster than a sequential program on my 4 core laptop. While this is not linear speed up it is still substantially faster. While theoretically, linear speedup is the fastest it could achieve, practically, with the cost of thread creation it is not possible. In terms of thresholds, I found that for large inputs, large thresholds increase performance, but for smaller inputs, large thresholds are not helpful.

Input size and time of a sequential algorithm



Speed up relative to sequential vs Input size

