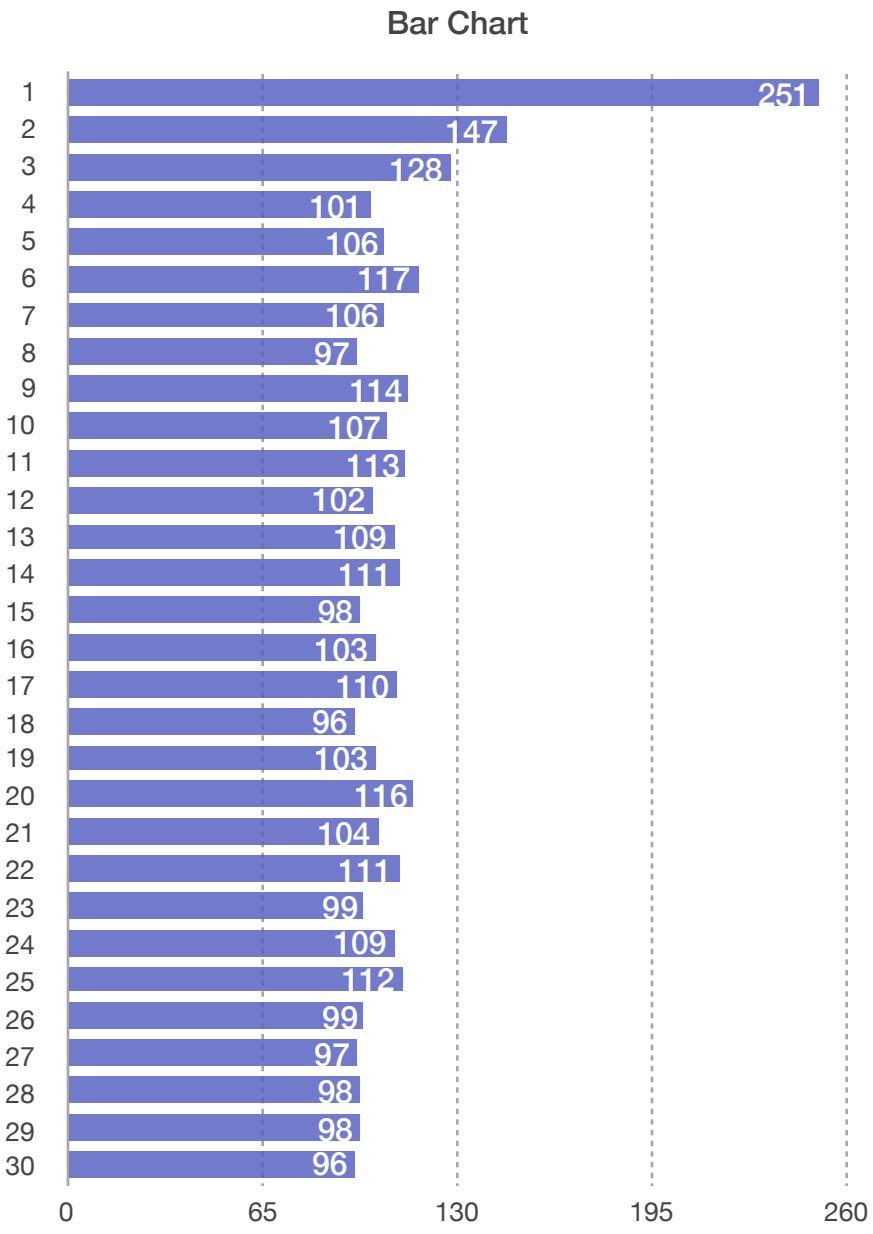
Analysis.pdf

Shane Bolding
Alex Fife
Systems and Networks 1 Fall 2019
Project 2

Description

Many tests were ran and averaged in order to get the histogram and other data. The thread count was started at 1 and incremented to 30. I set N = 100,000, due to RAM limitations on my personal computer.

Tables and Histograms



CPU Time (in ms) (measured using REALTIME)

THREADS	TIME
1	251
2	147
3	128
4	101
5	106
6	117
7	106
8	97
9	114
10	107
11	113
12	102
13	109
14	111
15	98
16	103
17	110
18	96
19	103
20	116
21	104
22	111
23	99
24	109
25	112
26	99
27	97
28	98
29	98

CPU Time (in ms) (measured using REALTIME)

THREADS	TIME
30	96

Analysis

For 1 thread, it is evident that since there is only one job for the calculation, that it would indeed take the most CPU time. For the following values however, it seems that there is a bottleneck in speed. Either a hardware bottleneck, or a software bottleneck.

Something else to consider, is that rather than a bottleneck, the threads reach max efficiency for N = 100,000 at 4 threads, therefore perhaps no more
are needed to accomplish the Collatz sequences.

Conclusion

In conclusion, the number of threads seem to hit a maximum value for either efficiency, or the memory of the system the program runs on determines a bottleneck. Due to my system not acting sluggish whatsoever, I conclude that for my parameters of N = 100,000 that 4 threads is simply the most efficient number of threads to have, any more is a waste of resources.