PA4: Matrix Multiplication (Sequential)

Evan Grill

CS 415

April 13, 2017

Problem

This project asked us to implement matrix multiplication sequentially, timing a variety of different multiplicand sizes.

Procedure

Each matrix is represented by a 2D C-style array. To multiply them, two loops are used to iterate through each entry in the result. For each entry, a third loop is run, and the multiplications of the corresponding rows/columns are added and stored.

Data

Data Size	Execution Time
100	0.001211
200	0.012451
300	0.035734
400	0.104611
500	0.247311
600	0.576311
700	1.248369
800	2.698798
900	6.554976
1000	12.082251
1100	18.003646
1200	25.100451
1300	33.485400
1400	42.696824
1500	59.252977
1600	71.377102
1700	79.990671
1800	105.146240
1900	124.090307
2000	146.300030
2100	170.826572
2200	199.033843
2300	233.465028
2400	262.798810
2500	303.444208
2600	326.266405
2700	383.783720

2800	430.313305
2900	476.383772
3000	529.325919

Table 1: Execution time (in seconds) of sequential matrix multiplication. (Matrix is of size x size total entries)

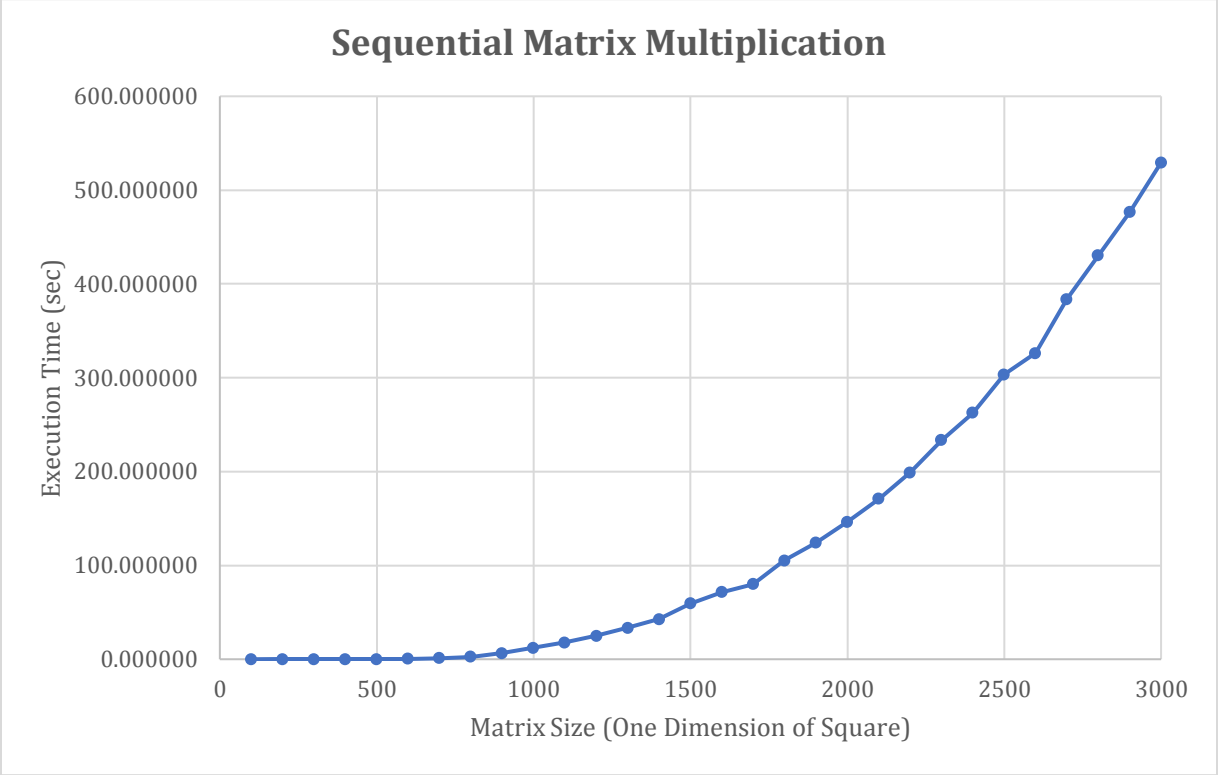

Figure 1: Chart showing execution times from Table 1 showing polynomial growth.

Results

The project presently only has the sequential method implemented, so there is not any data to compare it to, but it displays an expected polynomial growth rate. The algorithm is $O(n^3)$, so the growth rate reflects that. As always, we're tasked with determining the point at which the algorithm reaches five minute (300 seconds) run time. In this case, the code multiplied two 2500x2500 matrices in 303 seconds.

The next portion of the project and report will add parallel implementation with comparisons to sequential, speed up factors, and efficiencies, for multiple amounts of parallel processors.