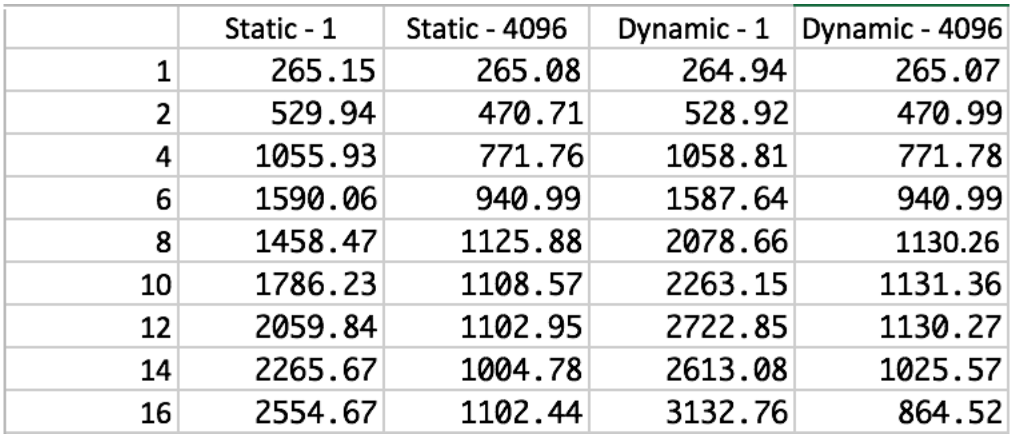
CS 475 Assignment #2

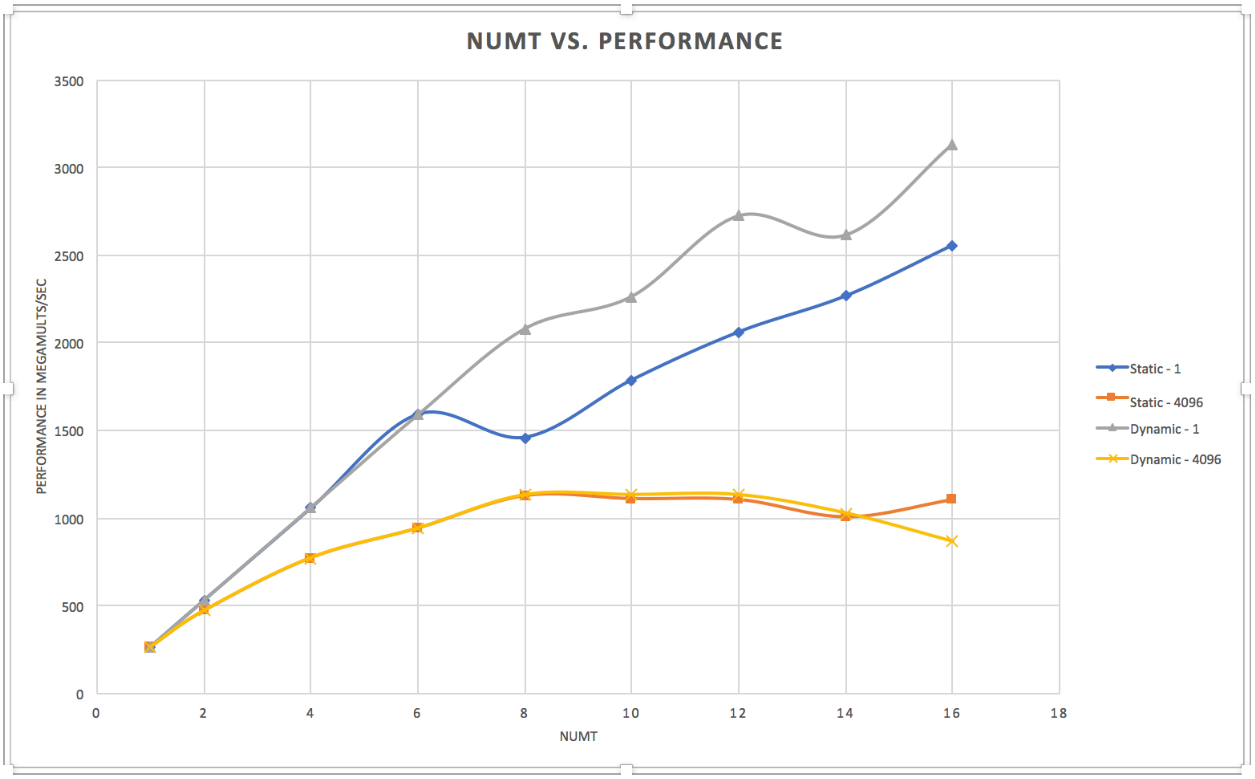
1) I ran this program on the FLIP server

2)



	Static - 1	Static - 4096	Dynamic - 1	Dynamic - 4096
1	265.15	265.08	264.94	265.07
2	529.94	470.71	528.92	470.99
4	1055.93	771.76	1058.81	771.78
6	1590.06	940.99	1587.64	940.99
8	1458.47	1125.88	2078.66	1130.26
10	1786.23	1108.57	2263.15	1131.36
12	2059.84	1102.95	2722.85	1130.27
14	2265.67	1004.78	2613.08	1025.57
16	2554.67	1102.44	3132.76	864.52

3)



4) From the above plot, I see the static and dynamic scheduling with 4096 chunk size have a lower performance than the static and dynamic scheduling with 1

chunk. Another pattern that I observe is the static-1 and dynamic-1 have the same increase in performance until 6 threads. After that, static-1 takes a dip in performance and recovers at 10 threads. Meanwhile, dynamic-1 continues to increase with a dip at 14 threads. So, dynamic-1 scheduling outperforms static-1 after 6 threads.

5) The chunksize matters because, when it is as large as 4096, each thread has 4096 iterations. So, it takes time to complete it and get a new assignment. However, when the chunksize is only 1, the performance increases as adding more threads will mean more threads finishing more iterations quickly and get a new assignment.
6) Static and dynamic scheduling matters because, dynamic scheduling takes advantage of having a master and pool of threads. In dynamic scheduling, the master allots some work at run time to the threads. Once a thread finishes its work, the master allots some more. However, in static, all the work is allotted at runtime and if a thread finishes early, then it just idly sits there. Therefore, dynamic scheduling has a higher performance.