PA#2 Benchmark

Dimension	200x200	400x400	600x600	800x800
No. of Threads				
1 (s)	11.4209	118.3245	416.5405	879.0510
4 (s)	6.3673	60.5356	120.3556	260.7920
8 (s)	5.9223	48.0774	102.9775	232.7871
16 (s)	6.4774	47.4742	98.7722	229.1102

- As the dimension (data size) increases, the elapsed time surely increases by a great factor of ~10 times for 200x200 to 400x400. But decreased to ~2 times when increasing by 200 in M and N after 400x400 for each thread number.
- On increasing the threads by 2 fold for each dimension, it does not seemingly reduce the processing times drastically. This is true for a single core process like this. A multi-core process may involve lesser processing time. This may be because of an increased synchronization time within the process.
- Synchronization times will surely increase due to the increased overhead on voluntary and involuntary switching between worker threads and the master thread when increasing the number of worker threads. They have a directly proportional relationship.

- In the sequential program, elapsed time if obviously very large as only one (thread) is processing it.
- Synchronization time here is less because there is no overhead required for changing between threads and only working on the two 2D arrays *u* and *w.*