

# COMP3230 Programming Assignment 2 – Report

- By Zhao Wenzhe (3035844883)

Here are my benchmark results:

Thread Numbers	Speed (tok/s)	User Time	System Time	User Time/System Time
0 (Sequential)	23.3170	11.4306	0.1699	67.2784
1	21.7090	10.512383	1.363277	7.7111
2	30.8049	11.697586	2.341562	4.9956
4	41.3043	12.069727	3.536805	3.4126
6	38.9572	12.888067	4.445567	2.8991
8	36.2398	12.794686	5.838247	2.1915
10	32.1256	13.667464	5.929852	2.3049
12	30.6240	13.029757	7.653552	1.7024
16	26.0376	13.630102	9.643349	1.4134

For the performance, as the number of threads increases, the speed increases at the beginning (1 -> 2-> 4). When the threads numbers exceed 4, the speed decreases gradually but is still higher than the speed of the single thread process. The best performance is when thread number = 4.

The reason for the improvement in the performance when the thread is from 1 to 4 is that multi-thread divides the work in parallel so that multiple workloads can be done in the same time. Since the thread is not so many, the time for creating, handling, and cancelling threads is not important at this stage. However, when the threads are too many, although the process can divide work in parallel, the time for dealing with multi-thread also increases. We can notice that for the User Time/System Time, it decreases gradually as threads increase. So, this means the system has to spend more and more time on dealing threads, and this amount of time wasted is gradually more than the time saved in parallel working.

As a result, the performance increase first and decrease latter as the thread number increases.