

# Lab1 report

## Tables

Table 1 contains the real part of the time command.

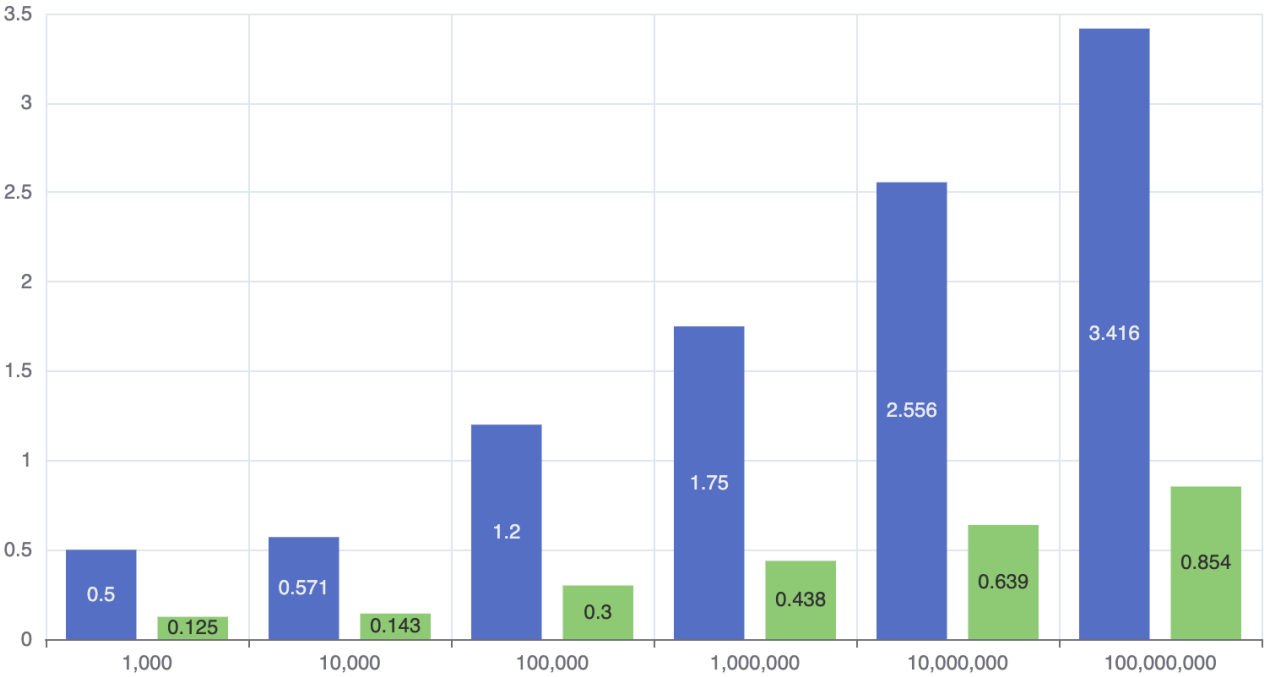
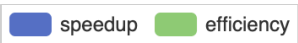
N	1,000	10,000	100,000	1,000,000	10,000,000	100,000,000
Seq	0.004s	0.004s	0.006s	0.014s	0.092s	0.837s
1-thread	0.005s	0.005s	0.005s	0.013s	0.086s	0.785s
4-threads	0.008s	0.007s	0.005s	0.008s	0.036s	0.245s

Table 2 contains the cpu+sys part of the time command.

N	1,000	10,000	100,000	1,000,000	10,000,000	100,000,000
Seq	0.003s	0.003s	0.004s	0.013s	0.090s	0.834s
1-thread	0.004s	0.003s	0.004s	0.012s	0.085s	0.785s
4-threads	0.005s	0.005s	0.005s	0.015s	0.088s	0.882s

## Figures

### OpenMP graphs



## Questions

1. Which machine have you used to do the experiment (crunchy1, crunchy3, ...)? Doing them on your laptop will not be accepted.

Machine: crunchy1

2. Given the Table 1 and graph 1 what is the relationship between  $N$  and the speedup?

As  $N$  is increasing, the speedup increases.

3. How can you justify that relationship you showed in the above question?

The IO time is the same, which occupies most of the sequential time. In our experiment, we can roughly conclude the fraction of the parallelizable part increases. According to Amdahl's Law, the speedup increases.

4. Given Table 1 and Table 2: Was the real time mostly bigger, smaller, equal to  $\text{cpu} + \text{sys}$  time?

Mostly bigger, but some smaller and equal.

5. What is your justification for the trend in question 4 above?

Basically, real time is the time consumption of the program. When it comes to multithreading, say 4-thread, the  $\text{cpu} + \text{sys}$  time consists of the time of 4 threads. Hence, it is larger than one thread or sequential program.

6. What is the trend you see in graph 2?

The efficiency is growing.

7. How do you explain the trend you found in question 6 above?

It has overhead when creating threads, terminating threads, and communicating. When the parallelizable fraction of the task increases, more parallelizable portions of the task can be assigned to each thread. Hence, the efficiency of each thread increases.