

## CS475 Project 0 Written Commentary

The machine this project was run on was the Oregon State flip server. The command `uname -a` returns:

```
"Linux kec1130-11.engr.oregonstate.edu 3.10.0-693.11.1.el7.x86_64 #1 SMP Mon Dec 4  
23:52:40 UTC 2017 x86_64 x86_64 x86_64 GNU/Linux"
```

The program was run with array sizes of 100,000, and with 16 tries.  
Performance results for 1 thread:

Peak Performance: 2309.82 MegaMults/Sec  
Avg. Performance: 2137.32 MegaMults/Sec

Performance results for 4 threads:

Peak Performance: 7919.62 MegaMults/Sec  
Avg. Performance: 7224.80 MegaMults/Sec

4-thread-to-one-thread speedup:  
 $S = 0.29$

It seems that the speedup is a fraction, or smaller than what is was with 1 thread. This may be happening because the operations that are being performed on the array isn't optimized by multiple threads. Since I don't believe the work isn't being divided in any way, some of the operations will be run more than once, and that's why the speedup result is a fraction.

The Parallel Fraction:  
 $F_p = -3.238236$