**Parallel Programming**

**Project 0**

1. Tell what machine you ran this on

I used a flip server to run the program.

2. What performance results did you get?

Number of tries = 2000

|  |  |  |  |
| --- | --- | --- | --- |
| Array size | Single thread peak performance | 4 thread peak performance | Ratio |
| 100000 | 333.38 | 1215.67 | 3.64649949 |
| 200000 | 331.32 | 1231.16 | 3.715924182 |
| 300000 | 334.03 | 1191.23 | 3.566236566 |
| 400000 | 330.07 | 1251.63 | 3.792013815 |
| 500000 | 314.11 | 1252.33 | 3.986915412 |
| 600000 | 329.52 | 1332.08 | 4.04248604 |
| 700000 | 317.84 | 1277.62 | 4.019695444 |
| 800000 | 322.03 | 1246.64 | 3.871192125 |
| 900000 | 323.28 | 1317.69 | 4.076002227 |
| 1000000 | 326.32 | 1291.58 | 3.958016671 |

Average Single thread peak performance = 316.50 MegaMults/Sec

Average 4 thread peak performance = 1203.43 MegaMults/Sec

3. What was your 4-thread-to-one-thread speedup?

The average speed up = 3.80

4. If the 4-thread-to-one-thread speedup is less than 4.0, why do you think it is this way?

This might be because of the waiting caused by the shared memory, communications, and input/output resources. There also might be differences in the peak performances of each core. For instance, 12th Gen CPUs integrate two types of cores into a single die:

* Performance cores (P-cores)
* Efficiency cores (E-cores).

5. What was your Parallel Fraction, Fp?

Speedup(4) = 1

----------------------

(1 - Fp) + Fp / 4

3.80 = 1

---------------

1 – Fp\*(3/4)

0.263 = 1 – Fp \* (3/4)

Fp = 56

57