**CSC410**

*Assignment G3*

Diana Lewis

Alex Houser

Marty Moradian

Yanlin Li

Stephen Chilcote

**Time Testing**

Sequential

|  |  |  |  |
| --- | --- | --- | --- |
|  | 300 | 1000 | 2000 |
| Real | 0m0.274s | 0m9.636s | 1m29.187s |
| User | 0m0.272s | 0m9.609s | 1m29.126s |
| Sys | 0m0.000s | 0m0.020s | 0m0.024s |

Pthread

|  |  |  |  |
| --- | --- | --- | --- |
|  | 300 | 1000 | 2000 |
| Real |  |  |  |
| User |  |  |  |
| Sys |  |  |  |

Omp

|  |  |  |  |
| --- | --- | --- | --- |
|  | 300 | 1000 | 2000 |
| Real | 0m0.117s | 0m1.941s | 0m9.780s |
| User | 0m1.052s | 0m23.461s | 3m7.972s |
| Sys | 0m0.008s | 0m0.396s | 0m0.060s |

**Conclusions/comparisons**

Ease of writing: It was very easy to write the OpenMP code. Everything could be written as if for a sequential program, and adding one pragma (and compiling with -fopenmp) is enough to add parallelism.

Performance: Using OpenMP led to a jump in the system time required versus the sequential code, probably due to scheduling. The real time was significantly less in the OpenMP version, however. For human purposes, real time is usually the most important.

Timing & speed-up tests: