Team: Alexa White and Justin Morgan

Timing results for five sample runs on PSC bridges with 5000x5000 matrices and using all 28 cores, results verified against sequential computation for correctness. For the extra credit chunks implementation, -n 9 and 5001x50001 are used to properly divide the input matrices into square chunks in a 3x3 grid.

| sequential | openMP | mpi<br>-n 4 | mpi<br>-n 8 | openMP +<br>mpi -n 4 | openMP +<br>mpi -n 8 | chunks<br>-n 4 | chunks<br>-n 9 |
|---|---|---|---|---|---|---|---|
| 1317.62 s | 36.73 s | 334.00 s | 174.55 s | 9.37 s | 4.66 s | 9.95 s | 4.67 s |
| 1396.76 s | 33.85 s | 333.49 s | 174.51 s | 9.27 s | 4.68 s | 10.01 s | 4.63 s |
| 1389.92 s | 36.33 s | 333.76 s | 174.15 s | 9.31 s | 4.70 s | 9.92 s | 5.10 s |
| 1368.31 s | 34.94 s | 335.85 s | 174.73 s | 9.32 s | 4.69 s | 9.88 s | 4.63 s |
| 1392.37 s | 35.67 s | 336.48 s | 174.72 s | 9.31 s | 4.68 s | 9.97 s | 4.61 s |

<u>to run:</u>

to compile sequential: gcc -g -Wall -o seq matrix_seq.c
to run sequential: ./seq

to compile mpi: mpicc -g -Wall -o mpi matrix_mpi.c
to run mpi: mpiexec -n 4 ./mpi

to compile openMP: gcc -g -Wall -std=c99 -fopenmp -o3 -o openMP matrix_MP.c
to run openMP: ./openMP

to compile opemMP + mpi: mpicc -g -Wall -std=c99 -fopenmp -o3 -o mpiMP matrix_mpiMP.c
to run openMP + mpi: mpiexec -n 4 ./mpiMP

to compile block implementation: mpicc -g -Wall -std=c99 -fopenmp -o3 -o blocks matrix_blocks.c -lm
to run block implementation: mpiexec -n 9 ./blocks