CSci 364
Spring, 2023
Programming Assignment #4

OpenMP reduction

**Phase I**
Write a C++ program that adds the integers from 0 to a positive integer specified on the command line and prints the sum to the console. If the command line value is less than 0, the sum should be 0.

Be sure to load Talon modules shared and gcc9 before compiling and running your program.

Use the provided Makefile to compile your program. Use the provided Slurm script to run your program as a batch job on Talon.

**Phase II**
Parallelize your program with the OpenMP reduction construct.

Use the provided Slurm script (see the line with cpus-per-task) so that the number of cores allocated by Slurm sets the OMP_NUM_THREADS environment variable and therefore sets the team size.

Test your program with different cpus-per-task values to confirm that the sum is still correct.

**Phase III**
Print to the console the difference between the current time before and after the parallel reduction using `double omp_get_wtime()` (see the OpenMP 4.5 Reference Guide). This difference represents the runtime performance of the parallel reduction. Be sure not to include processing command line arguments, allocating memory for variables, or printing the results.

Test your program to confirm that the elapsed time decreases as the number of threads increases.

Submit your source code, Makefile, and Slurm script files as a tar or zip file.