CS 470 Cluster Submission and Performance Analysis Lab

Name(s): Teddy Pugh, Jacob Zorniak

This lab will introduce you to running programs on our cluster as well as the basics of performance analysis.

1. Copy the `nas-benchmarks` folder from `/shared/cs470` to your home directory and examine the resulting files. Make sure you **read the `README.md` file in its entirety**, as it contains helpful examples of how to run programs on the cluster.

2. Choose a benchmark to work on. *Benchmarks* are well-defined, standardized programs that are generally accepted to be representative of "real" workloads. You have been provided two types of benchmarks: OpenMP and MPI. OpenMP benchmarks run on a single node and scale up to 8 or 16 threads while MPI benchmarks run on multiple nodes with larger workloads and scale up to 64 processes (8 nodes). **CLEARLY MARK** the benchmark you've chosen below by highlighting it below. Note that EP is not a valid choice for this lab because it is naturally parallel and because it is the target of the provided code samples in the readme.

- **BT: Block Tri-diagonal numerical linear system solver (OpenMP and MPI)**
- CG: Conjugate Gradient approximation of smallest eigenvalue (MPI only)
- FT: Fast Fourier Transform-based differential equation solver (OpenMP only)
- LU: Lower-Upper Gauss-Seidel numerical linear system solver (OpenMP and MPI)
- MG: Multi-Grid mesh solution to the discrete Poisson problem (OpenMP only)
- UA: Unstructured Adaptive mesh-based heat transfer simulation (OpenMP only)

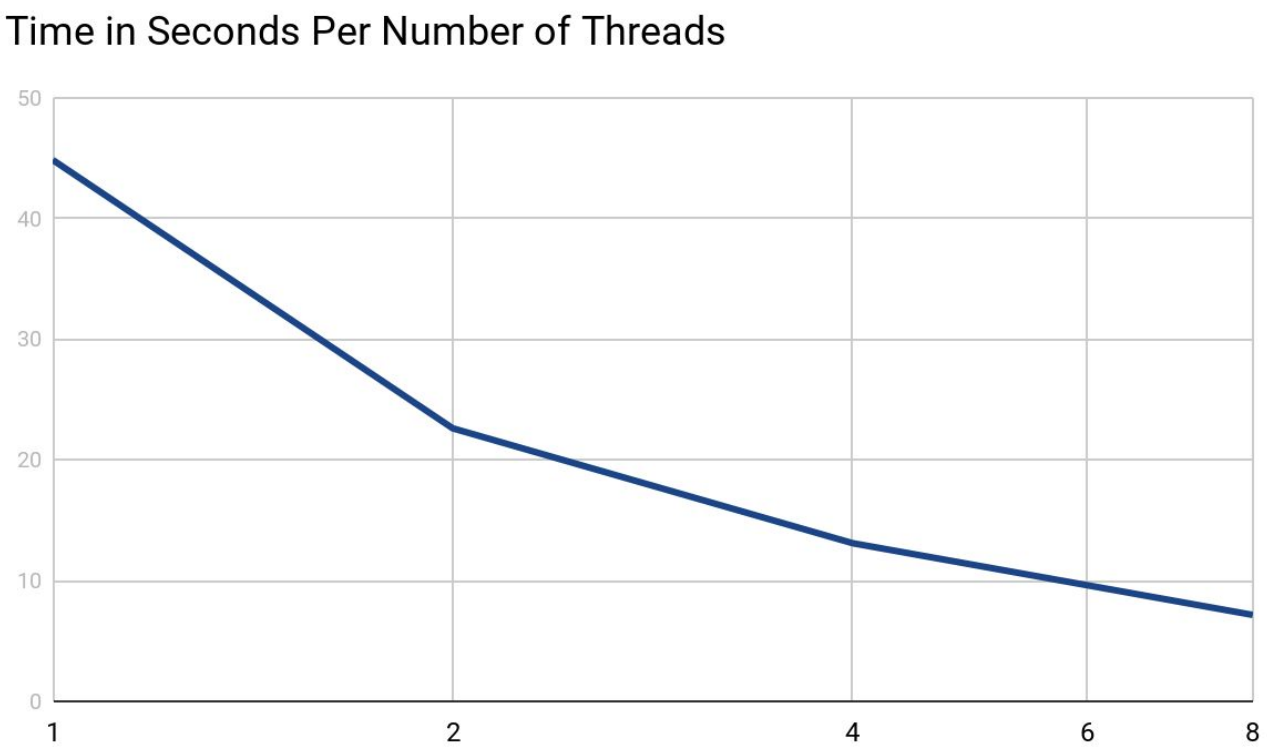
3. (Optional) Create a SLURM batch submission script to run your chosen benchmark. It is recommended that you start with one of the sample scripts on the cluster website or in the provided lab files. You may skip this step if you wish to use interactive job submission (e.g., the `srun` or `salloc` utilities) exclusively.

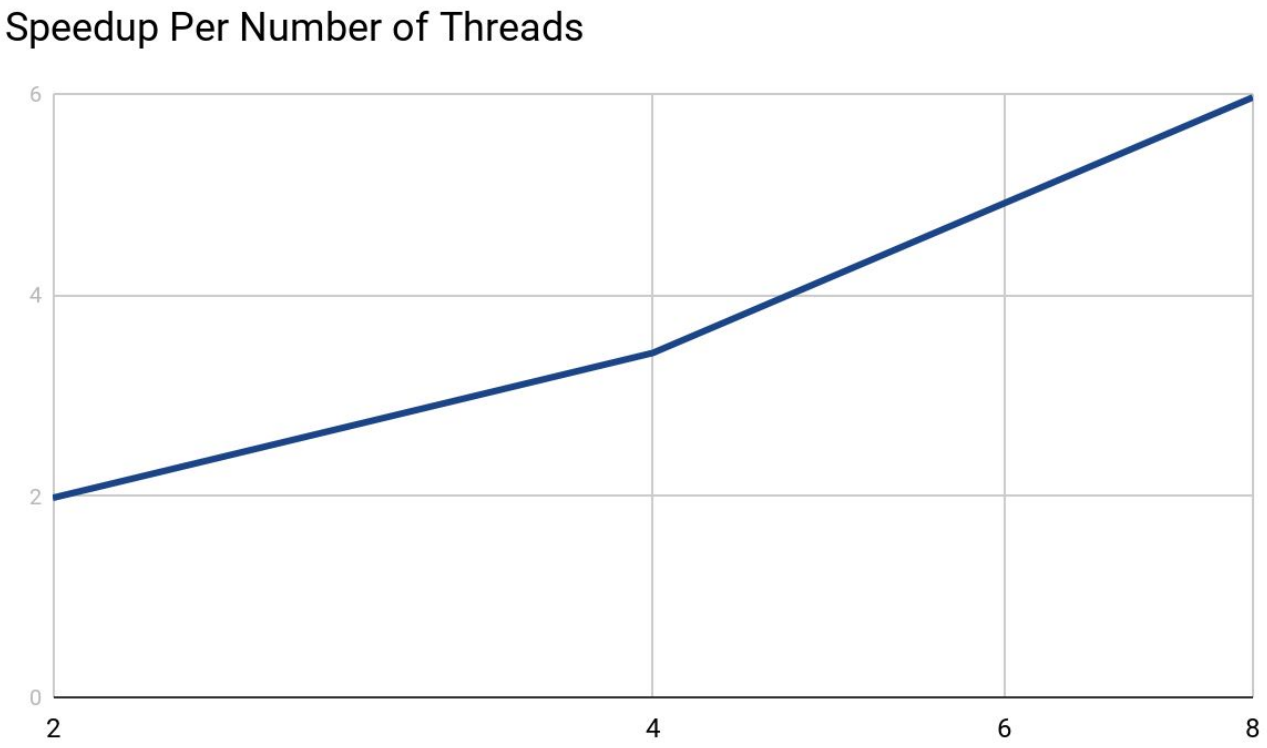
4. Using the SLURM job submission system, run a series of experiments on your chosen benchmark, varying the number of threads (for OpenMP benchmarks) or processes (for MPI benchmarks) for each experiment. Run at least five trials and report the minimum. Use the output of your experiments to fill out the following table (you may not need all of the rows):

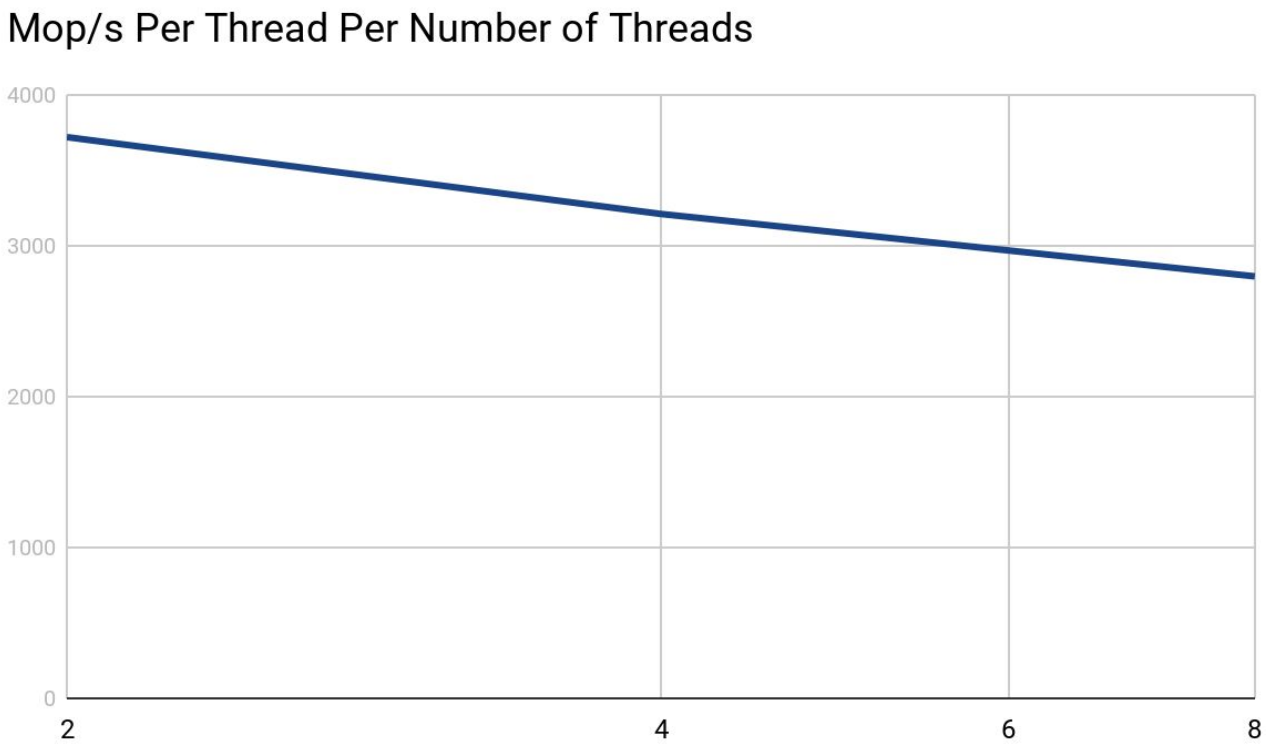
Number of threads/processes	Time in seconds	Speedup (T_s/T_p)	Mop/s total	Mop/s per thread/process
1	44.86	N/A	3751.58	3751.58
2	22.62	1.983	7439.74	3719.87
4	13.10	3.424	12842.74	3210.69
8	7.16	5.965	22376.67	2797.08

Note: For this lab, just use the 1-thread/process run time for T_s

5. Create three graphs of this data, each with thread/process count on the x-axis (logarithmic). For the y-axis (not logarithmic), one graph should have run time, one should have speedup, and one should have Mop/s per thread/process.







6. Write a 1-2 paragraph performance analysis of your results. How well does your chosen benchmark scale? Use your data and graphs to support your argument. Be as specific and quantitative as possible. Do your results suggest that the program demonstrates strong scaling, weak scaling, or both? Does it scale linearly? Estimate the percentage of the program that cannot be parallelized (i.e., *r* in our equation for Amdahl's Law).

Our benchmark seems to display strong scaling extremely well. This can be seen by the almost linear speedup that occurs as we increase the number of threads that are used. For example when the threads doubled from 1 to 2 the linear speedup was 1.983, very close to the value 2 which would indicate linear speedup. Because the programs run time and speedup were very clearly inversely proportional, and the speedup was almost linear, we can say that this is an example of strong scaling. We can not say anything about weak scaling for our tests as we never made any changes to the sample size. Since we can only seem to double our speedup, it makes sense for the percentage of the program that cannot be parallelized (or r in the equation for Amdahl's Law) to be about 50%.

7. Submit this document as a PDF to the appropriate Canvas assignment by the due date. If you worked in a group, please submit ONLY ONE copy per group and make sure the list of names at the top is accurate.