CSCE 435 Fall 2024

Assignment 1: Matrix Multiplication using MPI

Instructions to compile and execute the matrix multiplication code:

1. Upload the files (**mpi_mm.cpp, mpi.grace_job, build.sh, CMakeLists.txt**) to your scratch directory after logging into grace portal.

2. Open the current directory in the terminal using the "Open in terminal" option on the top.

3. Authenticate using your net id's password and duo-2 factor authentication.

4. Initialize the cmake build:

    - `$ . build.sh`

5. If further changes are made to the code, use make to re-build:

    - `$ make`

6. Run the batch file, giving matrix size and number of processors where m is the matrix size and p is the number of processors:

    - `$ sbatch mpi.grace_job <m> <p>`

7. After a job is complete, you'll be able to see the output in the output file corresponding to your jobid in the same directory as the source code. (You'll be able to find out whether a recent job has been completed or not by going to: grace dashboard > jobs > active jobs)

Assignment:

Summary

- Use MPI_Reduce to calculate the minimum, maximum and the average runtime taken by the "**receiving**" part, the "**calculation**" part, and the "**sending**" part of the **worker** processes. Plot **minimum time, maximum time, and average time** as functions of the
  number of processes where the number of processes will vary between {2, 4, 8, 16, 32, 64}. (You may plot all the three times: min, max, avg on a single graph with different colors)
  - There will be different plots for:
    - Matrices of size 128x128 **[10 points]**
    - Matrices of size 1024x1024 **[10 points]**
    - Matrices of size 8192x8192 **[10 points]**
- Similarly, in the master process, compute the runtime for the "**whole computation**", for the "**initialization**" part, and for the "**sending and receiving**" part and plot graphs for
  initialization time, **send/receive time**, and **total time** vs number of processes where the number of processes will vary between {2, 4, 8, 16, 32, 64}.
  - There will be different plots for:
    - Matrices of size 128x128 **[10 points]**
    - Matrices of size 1024x1024 **[10 points]**
    - Matrices of size 8192x8192 **[10 points]**
- Write down your observations on the variation of runtimes with the number of processes for various matrix sizes. **[20 points]**
- Correctness of the MPI_Reduce code changes **[20 points]**

Part 1 - MPI

1. Implement timers – Use MPI_Wtime to create timers to measure over the same regions that you marked with caliper.
2. MPI_Reduce - Use MPI_Reduce to calculate min, max, and sum (used to compute average) times for each of your worker timers. Be careful, if you use
   MPI_COMM_WORLD as your communicator, your values will be incorrect. This is because your master process will be implicitly included in your worker calculations. You need to either:
   a. Create a new MPI communicator that excludes the master process
   b. Initialize the variables you are reducing to in a way that avoids this problem.

Part 2 – Plotting and Observations

You can plot the graphs in excel.

Upload a .zip file on canvas containing:

- **A pdf with your answers.**
- **mpi_mm.cpp file with your code changes.**