ECE5640 High Performance Computing Spring 2016

Homework 3: Due Wednesday, February 17 at midnight

Homework is due Wednesday, February 17 at midnight on blackboard.

Submit one file, a zip file named `lastname_firstname_hw#.zip` containing code, `readme.txt`, bash file and answers to questions named according to the question numbers. For example the code in question 1 should be named `Q1`.

Programs should be run on the Discovery cluster.

1. (15 pts) Run `hello world from <your_username>`, based on the code from lecture 5 slide 51, on the discovery cluster using MPI. Run on 20 and 30 processes. Report your run times. Turn in your code, shell script and the output.

2. (15 pts) Repeat question 1 using an MPI broadcast command in place of the for loop to broadcast the message to all the processes from the master node. Run on 20 and 30 processes. Report your run times. Turn in your code, shell script and the output.

3. (30 pts) Implement the ring communications from Lecture 6, slides 55 through 57. Run on 20 and 30 processes. Report your run times. Turn in the shell script and the output.

4. (40 pts) Write MPI code to compute the Euclidean distance between two vectors. The vectors should be split among the available processes and each process should compute the distance of partial pairs. Then, each process should have its own partial sum and then add to a shared variable called `totdist`. Use an MPI reduction to compute `totdist`. Note that this is the MPI version of your code from Homework 2. Your code should work on any number of processes. Compute the runtime of your code. Run with different number of processes and different length vectors. Turn in your code and bash script. Give advice to someone running this code regarding how they should parallelize it. How many processes is optimal? How large does your vector need to be to see speedup? Would you advise that a user use OpenMP or MPI to parallelize their code? Explain your answer.