**CSS 534A Programming Assignment 2 Grading Sheet**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Max. Pts** | **Your Grade** |
| **Documentation** of your parallelization strategies including explanations and illustration in one page. | 20pts | 20pts |
| **Source code** that adheres good modularization, coding style, and an appropriate amount of commends.   * 25pts: well-organized and correct code receives * 23pts: messy yet working code or code with minor errors receives * 20pts: code with major bugs or incomplete code receives | 25pts | 23pts |
| **Execution output** that verifies the correctness of your implementation and demonstrates any improvement of your program’s execution performance.   * 25pts: Correct execution and better results than the two requirements: (1) four MPI ranks perform 2.2+ times better, and (2) four MPI ranks with multithreading perform 4.0+ times better than the sequential version. * 24pts: Correct execution and better results than requirement (1) four MPI ranks perform 2.2+ times better, but requirement (2) is just satisfied: four MPI ranks with multithreading perform 4.0+ times better than the sequential version. * 23pts: Correct execution and the two requirements just satisfied: (1) four MPI ranks perform 2.2 times better, and (2) four MPI ranks with multithreading perform 4.0 times better than the sequential version. * 22pts: Correct execution and requirement (1) was satisfied but requirement (2) was not satisfied. * 20pts: Correct execution and better performance improvement but none of the two requirements satisfied. * 15pts: Correct execution but little performance improvement, (i.e., max. 1.3 times better or less) * 10pts: Wrong execution | 25pts | 25pts |
| **Discussions** about the parallelization, the limitation, and possible performance improvement of your program in one page. | 25pts | 25pts |
| **Lab Session 2** Please turn in your lab 2 by the due date of program 2. Your source code and execution outputs are required. | 5pts | 5pts |
| **Total**  Note that program 2 takes 20% of your final grade. | 100pts | 98pts |

**Comments:**

Good performance improvements and discussions with potential uses of MPI\_ISend/Recv. Follow my instruction for deadlock avoidance in communication: even ranks should send first, while odd ranks should receive first.