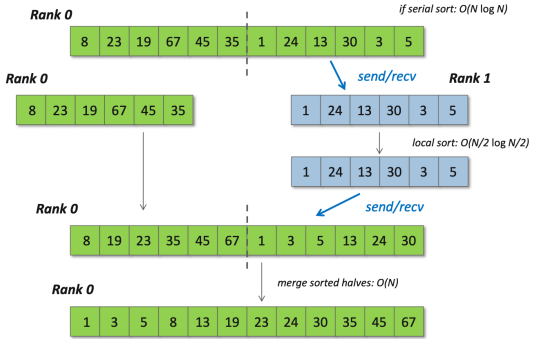
CS546 "Parallel and Distributed Processing"
Programming Assignment 1

Submission:
Due by 11:59pm of 09/30/2021
Late penalty: 10% penalty for each day late
Please upload your assignment on Blackboard with the following name:
CS546_SectionNumber_LastName_FirstName_PA1.
Please do NOT email your assignment to the instructor and/or TA!

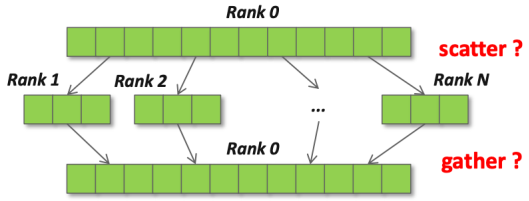
NOTE: Please use the example from the lecture resource https://anl.app.box.com/v/2021-IIT-MPI-Lecture as the reference. Modify it and the whole package should be compiled.

Exercise 1:

We described the 2-processes sorting using MPI send/receive in class. An example of parallel sort using MPI send/receive is:

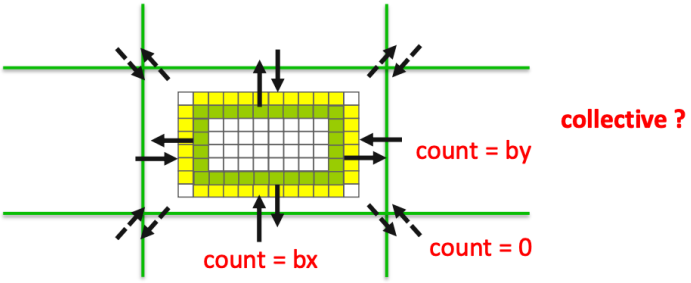


However, in the 2-processes version of sorting, only the first two processes are active, all other processes are waiting. Please implement the sorting algorithm for an arbitrary number of processes and try scattering and gathering chunks by using collective calls. Hint: please start from blocking_p2p/sort_2_procs.c



Exercise 2:

In the stencil example code described in class, it used non-blocking send/receive calls for each direction. Please implement this algorithm using a single collective call (Alltoallv). HINT: please Start from nonblocking_p2p/stencil.c



Submission Information

Each program must work correctly and be well documented. You should hand in:

1. **Report file (in PDF format)**: This is your change to explain your solution, possible optimization, any insights gained, problems encountered, etc. The report should include the performance results for your solution and your analysis of results.
2. **Readme:** This file should include the instructions to build and run your program.
3. **Source Code:** You must hand in all your source code.
4. **Output file with timings of your performance testing**. It should be consistent with your report.

Grading:

Out of 100 possible points we split the points as follows:

- Report: 30 points
  - 10 points for readability, use of English, organization
  - 20 points for performance evaluation and analysis of results
- Readme: 10 points for given instructions to compile and run your code
- Source code: 60 points
  - 10 points for readability and cleanness of code
  - 5 points for in-code comments
  - 30 points for correctness (if it is not even compiling you lose all points)
  - 15 points for performance and possible optimizations you have done

Note: We encourage collaboration between you and your classmates. Discuss various approaches and techniques to better understand the questions. However, we do NOT allow copying solutions or code. This is considered as cheating and falls under IIT code of honor. Penalties will be enforced. Please make sure you write your own solutions.

GOOD LUCK!