



Lab 6: Collective Communication Practice

Due: 4:00 PM, Thursday, October 8, 2026

In this lab, you will learn about collective communication using MPI. Until now, we have used only point-to-point communication, *i.e.*, sends and receives.

You may work alone or with a partner on this lab.

Learning goals:

1. To learn the basics of collective communication in MPI.

Getting Set Up

In Canvas, you will find a link to the starter GitHub repository for this lab, and you should fork that repository into your own GitHub account and make sure it is shared with your instructor. Only one member of the group should fork the repository on GitHub, then others should request a link to be granted write access.

You may choose to answer the lab questions in the `README.md` file in the top-level directory of your repository, or upload a document with your responses to your repository, or add a link to a shared document containing your responses to the `README.md` file.

Reading

Read Pacheco Ch. 3, Section 4, to learn about MPI's collective communication capabilities.

Collective Communication

In addition to the example `mpi_trap3.c` from the text chapter, there are two examples to consider in the repository for this lab: `mpicoll.c` and `mpiscatgath.c`.

For the first task, worth 20 points total, instead of writing answers in your response document, you will open GitHub Issues and provide your answers there. This process will be demonstrated in class.

For each line or group of adjacent lines that uses MPI collective communication calls in `mpi_trap3.c`, `mpicoll.c`, and `mpiscatgath.c`:

- if it is the first example of this collective communication function, briefly describe its effect in general terms,

- describe what it is doing specifically in this case, and
- describe what data exists on each process before and after the call.

Question 1: Other than the fact that it simplifies your code and reduces the chance of programming errors, what other advantages are there to using these higher-level collective communication functions rather than programming your own using basic point-to-point communication functions? (5 points)

Question 2: Answer Pacheco Exercise 3.8 on p. 141. You may submit pictures of hand drawings or use a drawing program. (10 points)

Submission

Make sure your answers to lab questions are provided using one of the mechanisms mentioned in the “Getting Set Up” part of the lab, and that your responses to the first part are in GitHub Issues in your repository.

Grading

This assignment will be graded out of 35 points.

Feature	Value	Score
MPI call descriptions	20	
Question 1	5	
Question 2: Ex. 3.8 a	5	
Question 2: Ex. 3.8 b	5	
Total	35	