Chp 6 - Exercise 2 - Individual

•Get the basic message code to run and compile.
  •Be sure to include the print statements
•Run the code with number of processors set to 2, 4, and 8 to see how it responds.

Grading Rubric
_____ (1 Point) Copy of the code
_____ (2 Points) In top comments add in what was printed out when running with 2, 4, and 8 processors

```cpp
// A simple example of sending/receiving messages with MPI
// By: Nick from CoffeeBeforeArch

#include <iostream>

#include "mpi.h"

int main(int argc, char *argv[]) {
  // Initialize MPI
  MPI_Init(&argc, &argv);

  // Get the task ID
  int task_id;
  MPI_Comm_rank(MPI_COMM_WORLD, &task_id);

  // Rank 0 coordinates work
  if (task_id == 0) {
    // Get the total number of tasks
    int num_tasks;
    MPI_Comm_size(MPI_COMM_WORLD, &num_tasks);

    // Send an integer to each thread
    for (int i = 1; i < num_tasks; i++) {
      MPI_Send(&i, 1, MPI_INT, i, 0, MPI_COMM_WORLD);
    }

    // Receive and integer back from each thread
    for (int i = 1; i < num_tasks; i++) {
      int recv;
      MPI_Recv(&recv, 1, MPI_INT, i, 0, MPI_COMM_WORLD,
MPI_STATUS_IGNORE);
```

```cpp
      // Print the received value
      std::cout << "Received " << recv << " from rank " << i <<
'\n';
    }
  } else {
    // Receive an integer from rank 0
    int recv;
    MPI_Recv(&recv, 1, MPI_INT, 0, 0, MPI_COMM_WORLD,
MPI_STATUS_IGNORE);

    // Square the value
    recv *= recv;

    // Send back the squared value
    MPI_Send(&recv, 1, MPI_INT, 0, 0, MPI_COMM_WORLD);
  }

  // Finish our MPI work
  MPI_Finalize();
  return 0;
}
```

Output on Sloop:

```
sa7233@sloop:~/fall2024/HPC$ mpirun -n 8 ./1_mpi_send_recv
Received 1 from rank 1
Received 4 from rank 2
Received 9 from rank 3
Received 16 from rank 4
Received 25 from rank 5
Received 36 from rank 6
Received 49 from rank 7
sa7233@sloop:~/fall2024/HPC$ mpirun -n 6 ./1_mpi_send_recv
Received 1 from rank 1
Received 4 from rank 2
Received 9 from rank 3
Received 16 from rank 4
Received 25 from rank 5
sa7233@sloop:~/fall2024/HPC$ mpirun -n 4 ./1_mpi_send_recv
Received 1 from rank 1
Received 4 from rank 2
Received 9 from rank 3
sa7233@sloop:~/fall2024/HPC$ mpirun -n 2 ./1_mpi_send_recv
Received 1 from rank 1
```

sa7233@sloop:~/fall2024/HPC$