For this homework I followed different websites to help me modify the code for pi1-3. I was having trouble using the mpi library with my code as it would not allow me to get results but it would allow me to compile. For the pi1 file I used MPI_Sendrecv for the two-sided communication. In pi2 I followed the example pseudocode from the website https://web.cels.anl.gov/~thakur/papers/rma-impl.pdf. The website provided an example of fence synchronization which I followed by creating a Mpi Window, and using a Mpi Barrier, and in between barriers using Mpi get and put. For pi3 also followed the website pseudocode for the lock-unlock synchronization. I created a Mpi window and then implemented Mpi lock and unlock. For pi4.cpp I used an openMp method. I chose this method because it helped me speed up performance in previous assignments. I used #pragma omp parallel in the code and the performance increased from 12ms compared to the pi.cpp default code's 14ms. The following error below is what happens when I run srun for jobs pi1-3.

*** An error occurred in MPI_Init

*** on a NULL communicator

*** MPI_ERRORS_ARE_FATAL (processes in this communicator will now abort,

*** and potentially your MPI job)

[r041.ib.bridges2.psc.edu:45384] Local abort before MPI_INIT completed completed successfully, but am not able to aggregate error messages, and not able to guarantee that all other processes were killed!

Commands used to compile pi1-4:

mpic++ -o pi1 pi1.cpp

mpic++ -o pi2 pi2.cpp

mpic++ -o pi3 pi3.cpp

g++ -fopenmp pi4.cpp -o pi4