1.) I divided the tasks with each processor getting a section of a column. For example, if there were 2 processors, each one would get half a column. When processor 1 finishes its first row, it passes the data to the next processor and begins to work on the next row. This creates a pipeline. The subtasks get distributed to each processor that isn't the master processor. The master processor divides the tasks and gathers the end result from the last processor. The data would be passed along. The message being passed by each task is the computation it did. It would pass the part of the row that it did.
2.) Some challenges that I met were getting the make file to work. I ended up using mpicc. Another problem I had was setting the number of processors. After looking around on the web, I figured out how to set the number of threads for MPI. Figuring out what was wrong with my syntax in MPI was another struggle.
3.) Sequential time = 18.44102 seconds
    16 cores = 11.604 seconds
    8 cores = 50.484 seconds
    I did not test 4, 2 and 1 core because the computation time took too long.
4.) I got speedup for a 16 core but if the core number went lower, the time got increased. These times may be skewed since other people were working on the lab at the same time.