Homework 3

1. Modify your wave equation solver from Homework 1, ignoring all OpenMP directives (removing them if they are too confusing) to run in parallel using MPI. Your program should use P processors (left to be a runtime variable). Follow a data-parallel paradigm and split the gridpoints in equally sized contiguous blocks among the processors.

   - For simplicity you can assume $M = kP$ gridpoints (you may modify your program to always enforce this by redefining M if that is easier).
   - Use "ghost" or "shadow" cells for each process to signify values needed from other processes.
   - Use MPI_Bcast() to broadcast runtime parameters read from standard input or provided as command line options to the rest of the processes.
   - Can the program be written with MPI_Send()/MPI_Recv() alone?
   - How different is the message passing code depending on the choice of BC?

2. Eager Beaver[1]: Modify your program for the case of the 2D problem of Homework 1 to run in parallel using MPI.

   - What can be different about this case? Is deadlock a possibility for very fine discretizations (large numbers of gridpoints)?
   - Can the choice of decomposition direction lead to minimizing communications to the collection of solutions and errors?

12.950 Parallel Programming for Multicore Machines Using OpenMP and MPI
IAP 2010