CS 320/CSE 302/ECE 392

Introduction to Parallel Programming for Scientists and Engineers
MP5 - Due Friday, April 16

The objective of this MP is to write two MPI versions (three if you registered for one unit) of the odd-even transposition sort described in Section 8.18).

First, you are requested to implement the algorithm presented in class assuming a single value per processor. That is, there is a processor for each element of the vector to be sorted. Notice that the compare-exchange phase can be implemented using MPI_SENDRECV (see below how to find a description of this command) and letting each processor decide whether or not to replace its value with the one received from its partner. *Note: Use only a few (4-6) processes to test your program. Otherwise we may bring the system down.*

For the second part, implement the same algorithm, but now assume each processor holds M items (M>1). The compare-exchange phase can be implemented by having processor I and I+1 exchange the whole M-element array (I=1,3,.... for the odd phase, and I=2,4,.. for the even phase). Both processors sort the combined arrays and then select the upper half (processor I+1) or the lower half (processor I) of the array before moving to the next phase of the algorithm.

The value (for the first problem) or the collection of values (for the second problem) to be sorted should be generated by the master processor using a random number generator. The master should send the value to each of the other processors and keep one value or collection of values for itself. The values should be in the range 1 to 10,000. The value M for the second problem should be read by processor 0 and then broadcast to all processors. Each processor should send to the master its rank (myid) and the sorted M item list to be printed at the end of the algorithm.

Those registered for one unit are also requested to extend the algorithm so that, instead of exchanging all M items at once, several exchanges of N items (N<M) are carried out. This can be done using the algorithm described in the Chapter 18 of *Solving Problems on Concurrent Processors* by G. Fox et. al. to be distributed in class Thursday April 8, 1999.

To run MPI programs, you should compile using

`mpif77` *<fortran source file (`*.f`)>*

or

`mpicc` *<c source file (`*.c`)>*

(`mpif77` and `mpicc` are in `/usr/commercial/ptools/mpich/bin`) and then run the program using

`mpirun -np` *<number of processors> <executable file (`a.out`)>*.

To get information about MPI commands, you can go to `http://www.mcs.anl.gov/mpi` and there click **MPI Standard 1.1** on the left bar. Then go to the end of the frame on the right and click **Index**.