Administrivia

- Reminder – Homework 1 due today; turn in by e-mail.

Message Passing with MPI, Review/Recap

- Simplest form of message-passing is blocking point-to-point (MPI_Send, MPI_Recv).
- Collective communication functions encapsulate frequently-used operations involving all processes (or all within a communicator) — e.g., MPI_Bcast, MPI_Reduce.
Using MPI Tags

- One parameter to MPI_Send, MPI_Recv is a “tag”. Can be used as a way to distinguish among messages, selectively receive, etc.

- Example — simplified master/worker program (master-worker.c on sample programs page).
  Basic idea — one or more “worker” processes that do the desired computation, plus a “master” that assigns them work.

Overlapping Computation and Communication

- If there’s useful work a process can be doing while waiting for a message to arrive, can use other forms of send/receive:
  Asynchronous communication — MPI_Isend and MPI_Irecv (plus MPI_Wait).
  Persistent communication — MPI_Send_init and MPI_Recv_init to set up, then MPI_Start and MPI_Wait.
- Example — exchange.c, etc., on sample programs page.
Minute Essay

- What did you find most difficult about Homework 1? What did you find most interesting?