ASSIGNMENT 2
HPC1 Fall 2014

Due Date: ***Thursday, October 2***
(please submit your report electronically to the instructor via email, as one PDF file named *hw2-yourUBitname.pdf*)

Problem 1: Write your own vector dot product benchmark code. For simplicity, just consider **double** (double precision).

a) Write a small benchmark code for the L1 BLAS routine for vector dot products. Note that for small vector lengths, depending on the granularity of your timer, you may need to average over a number of dot product evaluations.

b) Link with a reference copy of the BLAS (*e.g.*, -lblas on the CCR cluster, or you can download a copy of the source from netlib if you prefer), and plot your results in terms of MFlop/s as a function of vector length. Do you see any dependence on compiler flags?

c) Now use an optimized BLAS library (*e.g.*, Intel's MKL), and repeat the above, plotting both results together. You should see significantly different behavior than part **b**. Interpret your results. Note that Intel has a web tool to facilitate linking applications against the MKL,
http://software.intel.com/sites/products/mkl/MKL_Link_Line_Advisor.html

Problem 2: Use a simple MPI code to perform the "Ping-Pong" benchmark using blocking sends and receives. Time your results for buffer sizes ranging from, say, 8 Bytes to 8 MBytes, and plot the resulting message times and bandwidth (also identify the approximate latency) for:

a. CCR cluster nodes (inter and intra-node) using Infiniband.

b. CCR cluster nodes (inter and intra-node) using gigabit ethernet.

Note that the conventional way of implementing ping-pong uses two MPI processes, each of which does a send/recv pair, and then the single transit time is half the measured elapsed time. Also note that there are lots of ping-pong benchmark codes available in various places - if you prefer to use one of them rather than develop your own, just make sure that you provide an appropriate citation.