

**Spring 2019: Advanced Topics in Numerical Analysis:
High Performance Computing
Assignment 5 (due Apr. 29, 2019)**

Handing in your homework: Hand in your homework as for the previous homework assignments (git repo with Makefile), answering the questions by adding a text or a \LaTeX file to your repo.

1. **MPI ring communication.** Write a distributed memory program that sends an integer in a ring starting from process 0 to 1 to 2 (and so on). The last process sends the message back to process 0. Perform this loop N times, where N is set in the program or on the command line.
 - Start with sending the integer 0 and let every process add its rank to the integer it received before it sends it to the next processor. Use the result after N loops to check if all processors have properly added their contribution each time they received and sent the message.
 - Time your program for a larger N and estimate the latency on your system (i.e., the time used for each communication). If you have access to the CIMS network, try to test your communication ring on more than one machine such that communication must go through the network.¹ If you use MPI on a single processor with multiple cores, the available memory is logically distributed, but messages are not actually sent through a network.²
 - Hand in your solution using the filename `int_ring.c`.
 - Modify your code such that instead of a single integer being sent in a ring, you communicate a large array of about 2MByte in a ring. Time the communication and use these timings to estimate the bandwidth of your system (i.e., the amount of data that can be communicated per second).
2. **Provide details regarding your final project.** Provide a more detailed list/table of what tasks you have done and are planning on doing for your final project, including an estimate of the week you will be working on each task, and who will work on it. We are thinking of something like this:

¹See the computing CIMS information sheet posted in the first week of the class on how to use multiple hosts when using `mpirun`. Note that on each host, the same compiled file must be available, i.e., they need to have a shared directory such as the home directories in CIMS.

²It depends on the implementation of MPI how it handles sending/receiving of data that is stored in the same physical memory.

Project: A parallel compression algorithm to build a new internet		
Week	Work	Who
04/15-04/21	Read paper underlying idea and think about algorithm	Richard, Erlich
04/22-04/28	More literature search and sketch MPI parallel version, write pseudocode, think of extensions in CUDA	Richard
04/29-05/05	Implement basic MPI functions, update profs on status, ask for help in office hours, write tests to verify implementation	Erlich
05/06-05/12	Finish implementation, fix remaining bugs, experiment with compression capability, time permitting: write CUDA kernel	Richard, Erlich
05/13-05/19	Run scalability tests, work on presentation slides and report	Richard, Erlich