ACMS60212/40212 Advanced Scientific Computing, Fall 2021

Instructor: Zhiliang Xu (Email: zxu2@nd.edu; Office: 152B Hurley Hall; Phone: 631-3423 (For this semester, the best way to reach me is through email.))

Class time and place (Aug. 23 – Dec. 16**):** TR 12:30PM – 1:45PM, 105 Pasquerilla Center

Fa2021 is still an unusual semester. Your well-being is your top priority. Please use https://covid.nd.edu/ **for University news and guidance. I am happy to accommodate your needs if they can be justified. If you need COVID related accommodations, please know that this and all other accommodations are handled through the Sara Bea Center. For more information, on the accommodations process, see** https://supportandcare.nd.edu/**.**

Masking Policies: I strongly encourage everyone to wear a mask when attending the class.

Attendance: Students are expected to attend all classes unless there is an allowed excuse. If you need to take some day off, please let me know. Exceptions can be made for pre-agreed arrangement or sickness. Please note that excessive unexcused absences may result in lowering the course grade or even failing the course.

Online office hours: R 3:00pm – 4:30pm, or by appointment. Office hours are flexible since they are held online. Zoom address of the office hours will be the same as the Zoom session of class (Zoom link is on Sakai and at the end of the syllabus). For fast connecting with me, please drop me an email to setup a Zoom time if you need to have a live discussion. Also, if you want to chat after the class, let's do it in an open space outside the building.

Course materials: notes and sample codes.

a) I will post course notes for materials we will be learning before class begins if possible. I will use iPad to write additional notes for additional explanation, and post those hand-written notes on Sakai after class. Lectures will be recorded and uploaded to Panopto/Meeting Recordings on Sakai.
b) Sample codes are stored on CRC super computer. If you do not have a CRC account, please apply for one at https://crc.nd.edu/services/use-our-services/. Keep in mind that a training session needs to be completed before you can use the CRC computer if you have not done so.

Prerequisite: ACMS60690/ACMS40390 or equivalent. Significant experience in C/C++ programming and applications to science or engineering.

This course covers fundamental programming techniques necessary to use high performance computing to support research in science and engineering. There is a special emphasis on computer implementation, and their application to specific problems in science, engineering.

Topics to be covered:

1. Review of C/C++ programming: pointer, dynamic memory allocation, class, template, etc.
2. Parallel Computing, MPI basics
3. Parallel algorithms for implementing direct and iterative methods for solving system of linear equations
4. Sub-domain partitioning method for solving time-dependent partial differential equations on large domains.
5. Computing on GPUs
6. OpenMP basics (if time permits)
7. Monte Carlo and stochastic simulations (simple examples)

Grades: Course grades will be based on homework and projects. All assignments are electronic and on Sakai. Students also submit finished work on Sakai. Students are permitted and encouraged to work together when doing homework and projects, but must write their own papers and codes. Copying work is not allowed. Undergraduate and graduate students will have similar and/or different assignments.

Homework, projects (5-6 tentative*) and in class assignment:	60%
Final project(including project presentation*):	35%
Attendance (*):	5%

Final project options (subject to discussion. Final decision will be made in 7th or 8th week (around midterm week)):

1. A team project (samples will be given). A team consists of 3-4 members. Every member is expected to have a reasonable amount of work. The team project allows you to more deeply explore some facets of computing which interests you.
2. Individual project. You independently work on a smaller project of your interest or something suggested.
3. For both team and individual project, you need to first submit a project proposal to get an approval. The submitted materials of the project include a short white paper (1-2 pages) summarizing project goals, ideas and computer algorithms for solving the proposed problem, codes, a separate code documentation consisting of instructions for program functions, variables. In case there is a project presentation, presentation slides are submitted after the presentation. Since there are no scheduled "department exams", time of presentation will be discussed.

Homework, projects (5-6 tentative*) and in class assignment:

- Best way to learn programming is through hands-on practice. I will assign small in class programming tasks from time to time to make sure we are on the same page. Please bring your laptop to the class. In class assignments will not be graded, but can be submitted for discussion.
- Homework and projects will be assigned regularly (every two weeks or so). Students have 1-2 weeks to finish them.
- **Late work:** Late work is normally not accepted, except for pre-arranged agreement or university excused absences.

Honor Code: As a member of the Notre Dame community, I will not tolerate academic dishonesty. All examinations, homework and computer projects are conducted under the Honor Code. You are encouraged to work together on the homework assignments and projects, but copying in any form or submitting work done by others as your own is a violation of the Honor Code.

Electronics: Please respect your fellow students and prevent your electronics from disrupting class. If you use Zoom, please mute microphones when not speaking. Use the raise hand function in Zoom to speak.

Course Website: Everything will be on Sakai and CRC computer.

Textbooks (no need to buy. Copies are available at the library):

1. A. Grama, A. Gupta, G. Karypis, V. Kumar, Introduction to Parallel Computing, ISBN-0-201-64865-2

2. J. Cheng, M. Grossman, T. Mckercher, Professional CUDA C Programming, ISBN-978-1-118-73932-7

References:

1. V. Eijkhout, E. Chow, R. van de Geijn, Introduction to High-Performance Scientific Computing by (Public draft)
2. Numerical Recipes in C: The Art of Scientific Computing. Cambridge University Press, second edition, 2002.
3. Iterative Methods for Linear and Nonlinear Equations by C.T. Kelley, SIAM 1995
4. J. Sanders, E. Kandrot, CUDA by Example: An Introduction to General-Purpose GPU Programming, ISBN-10: 0131387685

Useful online resources:

- http://www.tutorialspoint.com/cprogramming
- https://www.tutorialspoint.com/cplusplus/index.htm
- https://www.geeksforgeeks.org/c-plus-plus/
- http://www.cplusplus.com/doc/tutorial/

Quick Guide for Getting Started

1. **Requesting an Account**

   https://crc.nd.edu/index.php/services/usersupport

2. Remember to bring your laptop to the class to check out samples codes and work on in class assignments.

3. **Accessing the CRC machine**

   Supporting documentations for utilizing the CRC computer is at:
   `https://docs.crc.nd.edu/index.html`

   CRC provides the following front-end machines for compilation and job submission. Each machine is configured with identical software stacks.

   crcfe01.crc.nd.edu (4 16 core 2.4 GHz AMD Opteron processors with 256 GB RAM)
   crcfe02.crc.nd.edu (4 16 core 2.3 GHz AMD Opteron processors with 256 GB RAM)
   crcfeIB01.crc.nd.edu* (4 16 core 2.4 GHz AMD Opteron processors with 256 GB RAM)
   You can securely log into the front-end machines (enabling X forwarding for GUI displays) using a ssh client e.g.:

   1.On Linux/unix/MacOS:

   a. Open a terminal

   b. Then type:

   > ssh –Y yournetid@crcfe01.crc.nd.edu

   2. On Windows: use Putty (http://www.chiark.greenend.org.uk/~sgtatham/putty/) or other programs.

4. All of the CRC HPC infrastructure is running Red Hat Enterprise Linux (RHEL).

   - A good basic Linux guide is available at `https://docs.crc.nd.edu/new_user/linux_guide.html`.
   - Another Linux/unix starting guide (*unix_tutorial.pdf*) is under Resources on Sakai.

5. **The sample codes on CRC machine are available at: ~zxu2/Public**

6. Software and Development Environment

6.1 Interaction with machine through an Unix Shell

The C shell. This shell uses a command structure and syntax similar to the C programming language.

On Linux, the csh is a link to tcsh.

A system cshrc is executed, followed by the following files located in your home directory (in order of their execution). You may override settings in the system chsrc from within your .cshrc and .login.

.cshrc

.login (executed only at login)

.logout (executed only at logout)

6.2 Development Environment

The software environment on the front-end machines is managed with *modules*. You can easily modify your programming and/or application environment by simply loading and removing the required modules. The most useful module commands are:

- module avail (view the complete module list)
- module load xyz (load the module xyz)
- module unload xyz (unload the module xyz)
- module list (list currently loaded modules)

We typically use "intel/19.0" compiler and "mpich/3.3/intel/19.0" for MPI. To setup both of them, use the following commands:
> module load intel/19.0
> module load mpich/3.3/intel/19.0

6.3 Debugging tools

gdb (for serial code)

6.4 Text Editor

vi, vim, emacs or anything of your preference

The class Zoom address

Topic: FA21-ACMS-60212-01
Time: This is a recurring meeting Meet anytime

Join Zoom Meeting

https://notredame.zoom.us/j/91753597702?pwd=NGxQTUIyYnlFRFRidWN6RlJTRnJvQT09

Meeting ID: 917 5359 7702

Passcode: 8097186410

One tap mobile

+13126266799,,91753597702#,,,,*8097186410# US (Chicago)
+13017158592,,91753597702#,,,,*8097186410# US (Washington DC)

Dial by your location

+1 312 626 6799 US (Chicago)
+1 301 715 8592 US (Washington DC)
+1 646 558 8656 US (New York)
+1 669 900 6833 US (San Jose)
+1 253 215 8782 US (Tacoma)
+1 346 248 7799 US (Houston)

Meeting ID: 917 5359 7702
Passcode: 8097186410
Find your local number: https://notredame.zoom.us/u/abQ2tZRqPR

Join by SIP
91753597702@zoomcrc.com

Join by H.323
162.255.37.11 (US West)
162.255.36.11 (US East)
115.114.131.7 (India Mumbai)
115.114.115.7 (India Hyderabad)
213.19.144.110 (Amsterdam Netherlands)
213.244.140.110 (Germany)
103.122.166.55 (Australia Sydney)
103.122.167.55 (Australia Melbourne)
64.211.144.160 (Brazil)
69.174.57.160 (Canada Toronto)
65.39.152.160 (Canada Vancouver)
207.226.132.110 (Japan Tokyo)
149.137.24.110 (Japan Osaka)
Meeting ID: 917 5359 7702
Passcode: 8097186410