Version: 11/19/2015

ECE/ME/EMA/CS 759 – Fall 2015
High Performance Computing for Engineering Applications

Time: 2:30 – 3:45 PM Monday-Wednesday-Friday
Location: 1610EH

Instructor: Dan Negrut
Office: 4150ME
Phone: 608 265 6124
E-Mail: negrut@wisc.edu
Course Page: http://sbel.wisc.edu/Courses/ME964/2015
Grades Page: http://learnuw.wisc.edu
Forum: http://sbel.wisc.edu/Forum/viewforum.php?f=15
Grader/TA: Ang Li (ali28@wisc.edu)

Office Hours (held in 4150ME):
Tuesday, 9:00 – 10:00 AM
Thursday, 9:00 – 10:00 AM
Other times by appointment (please call or email to arrange)

Prerequisites: C Programming

Recommended Texts (see also lecture handout for other recommendations):
- R. Bryant and D. O'Hallaron, *Computer Systems: A Programmer's Perspective*, Prentice Hall, 2nd Edition, 2011
- D. Negrut, *Primer: Elements of Processor Architecture. The Hardware/Software Interplay*, available on the course website.
- Jason Sanders and Edward Kandrot: *CUDA by Example: An Introduction to General-Purpose GPU Programming*, Addison-Wesley Professional, 2010
- GPU Programming Guide, version 5.0
- David B. Kirk and Wen-mei W. Hwu: *Programming Massively Parallel Processors: A Hands-on Approach*, Morgan Kaufmann, 2010
- H. Nguyen (ed.), *GPU Gems 3*, Addison Wesley, 2007
- Peter Pacheco: *An Introduction to Parallel Programming*, Morgan Kaufmann, 2011
- T. Mattson, et al.: *Patterns for Parallel Programming*, Addison Wesley, 2005
- Michael J. Quinn: *Parallel Programming in C with MPI and OpenMP*, McGraw Hill, 2003

Course Objectives: The course will (a) provide an overview of existing High-Performance Computing (HPC) software and hardware, (b) present basic software design patterns for high performance parallel computing, (c) introduce CUDA for parallel computing on the Graphics Processing Unit (GPU), (d) introduce the OpenMP solution to enabling parallelism across multiple CPU cores, and (e) introduce the Message Passing Interface (MPI) standard for leveraging parallelism on a CPU cluster. The approach is hands-on, the students are expected to use the lecture information, a series of assignments and a final project to emerge at the end of the class with parallel programing knowledge that can be immediately applied to their research projects.

Hardware Used: The course is designed to leverage Euler, a cluster with about 1,200 CPU cores and 56 GPU cards. Each student will receive an individual account on Euler that will be used for GPU computing, MPI-enabled parallel computing and possibly OpenMP multi-core computing. Euler runs CentOS Linux release 7.1.1503 (Core).

Course Workload: The course will have one midterm exam and one final project.
NOTE: This is an accelerated pace course. Although the exam is listed as "midterm", it will essentially cover the entire material that will be presented in ME759.

Course Structure: There will be 28 lectures for this class. The Microsoft PowerPoint notes used in class will be posted online at http://sbel.wisc.edu/Courses/ME964/2015. Grades in ECE/ME/EMA/CS759 will be based on your performance on homework, midterm exam, one final project, and course participation. All homework and exam scores will be maintained on the Learn@UW course website. This will allow you to monitor your performance and see aggregate scores for the rest of the class, which can give you a continuous idea of your performance in relation to the rest of the class. Should you have questions about your score, please contact the instructor. Policies regarding grading and turning in your homework:

1. *Score-related questions about assignments and midterm exam must be raised prior to the next class period after receiving the score.*
2. *If homework that you turned in appears not to be graded (missing) on the Learn@UW course website please point that out to me within one week after the return of the corresponding set of graded assignments. It is a good practice to save your homework so that I will be able to update the grade to give you full credit for your work.*
3. *The two assignments with the lowest scores will be dropped when computing the final homework average*

Final Grade Policy: The final grade will be computed using the following weights:

Homework	**=**	**40%**
Midterm Exam 1	**=**	**15%**
Midterm Exam 2		**15**
Final Project	**=**	**25%**
Course Participation	**=**	**5%**

Homework: Assignments will be handed out roughly on a weekly basis for the first part of the class. The homework solution will be returned using the Learn@UW drop-box system. Homework solutions should be *neat, well organized, and well commented.* Your score for each assignment will be between 0-100. No late homework will be accepted. The homework with the two lowest scores will be dropped when computing final score. You can use this mechanism to basically not turn in two assignments.
Turning in your assignment should boil down to uploading at Learn@UW a zipped (tar.gz) file. Upon unzipping, in the top directory, which should be called HW07 (for the seventh assignment, for instance), one should be able to find a **`hw07ReadMe.pdf`** file with your text, a `makefile` that when run (by typing "`make homework`") will generate an executable called **`hw07.exe`**. The grader will run this executable and expect that all results you report in `hw07ReadMe.pdf` will be output in a file or to the screen.

Exams: There will be two "open-book" exams. The first one will focus on material presented in the first part of the class; i.e., general concepts and GPU computing. The second exam will cover the

entire material discussed in the course. You can bring along annotated copies of the documents that you have been asked to read (reading assignments). There will be no need for a computer for these tests. The best way to prepare for exams is to participate in class, learn the fundamental concepts, and work on the assignments diligently. The exams will be scored on a scale of 0 to 100. The first exam is scheduled for Friday, 10/09/2015 during the regular lecture hours. The second exam is scheduled for Monday, 11/23/2015 at 7:15 PM. A review session will be organized for each of these exams.

Final Project: The projects can be individual or team projects. The number of students allowed on a team depends on the complexity of the project. The topic of the project could be related to your research. A two-page or shorter PDF doc describing your final project is due on 11/13/2015 at 11:59 PM. Once I get a chance to read your final project proposals I might provide feedback and adjust the topic you suggested. In case you don't have a research project to work on, I will provide two default projects that are challenging enough to lead at least to a conference submission.
During the Finals week, each student or student team will make 15 min long presentations outlining results/accomplishments related to their final project. I will come up with additional time slots during the finals' week for students/teams to report on their work. For all students the final project report is due at 11:59 PM on Monday, December 21.

Course Participation: You are expected to participate actively in class discussions and to pose questions. Beyond this and in order to earn the 5% assigned to this category you will have to post at least five answers by the end of the semesters to the questions posted on the ECE/ME/EMA/CS759 forum (http://sbel.wisc.edu/Forum/viewforum.php?f=15).

Disability requests: I must hear from anyone who has a disability that may require some modification of seating, testing or other class requirements so that appropriate arrangements may be made. Please see me after class or during my office hours.

Complaints: If you have a complaint regarding the course and if you are unsatisfied with the response of the instructor, then you should contact the Chair of the Department of Mechanical Engineering. The Chair's office is in 3107ME, and an appointment to see the Chair can be made by contacting the Department Office at 608 263-5372.

Letter Grades: Final letter grades will be based on the total score accumulated on homework and exams throughout the semester using the following scale:

Score	**Grade**
≥92	A
86-91	AB
78-85	B
70-77	BC
60-69	C
50-59	D

ECE/ME/EMA/CS 759 High Performance Computing for Engineering Applications
Fall 2015– Syllabus [draft, subject to change]

Date	Title	Lecturer	HW Assigned	Reading Assignments Other Observations
09/02 [L01]	Syllabus related issues. Brief course overview	Negrut	HW01 (due 09/09) C programming-related	
09/04 [L02]	Information about job execution on Euler. Instructions. Transistors. Moore's Law. Dennard's scaling	Negrut		Read Chapter 5 of Brian W. Kernighan and Dennis M. Ritchie "The C Programming Language" book
09/07	**Labor Day, No Class**			
09/09 [L03]	Pipelining. Measuring Computer Performance. Memory Aspects.	Negrut	HW02 (due 09/16) C programming-related	Additional information available in Negrut's Primer, see link on class website
09/11 [L04]	Caches. Virtual Memory.	Negrut		Read the articles of Manferdelli and Amdhal's (link on class website)
09/14 [L05]	High Performance Computing: Why, and why now?	Negrut		Read 2005 article of Dongarra et al. for an overview of HPC (link on class website)
09/16 [L06]	Big Iron HPC.	Negrut	HW03 (due 09/30) HW: getting started with CUDA, understanding thread/block index issues.	
09/18 [L07]	GPU Computing Intro. The CUDA Programming Model.	Negrut		
09/21	**NO CLASS**			Read article on GPU computing evolution of Nickolls & Dally (link on class website)
09/23	**NO CLASS**			
09/25 [L08]	CUDA Execution Configuration. Thread Indexing	Negrut		
09/28 [L09]	CUDA Execution Model. CUDA API	Negrut		
09/30 [L10]	GPU Memory Spaces	Negrut	HW04 (due 10/07) CUDA matrix-vector multiplication & matrix addition	
10/02 [L11]	GPU Scheduling Issues	Negrut		Read 2009 paper (or tech report) of Bell and Garland on Sparse Linear Algebra on the GPU (link on class website).
10/05 [L12]	CUDA Scheduling Issues. Global Memory Access Patterns and Implications. Control Flow in CUDA	Negrut		Read Chapters 4 & 5 of the CUDA Programming Guide 5.0 (link on class website)
10/07 [L13]	CUDA Shared Memory Issues. Atomic operations in CUDA.	Negrut Mazhar	HW05 (due 10/21) CUDA tiled matrix multiplication Simple vector reduction Kernel Call Overhead	Read 2008 article of Volkov and Demmel on GPU benchmarking (link on class website)
10/09	1st MIDTERM EXAM	Negrut	Review on Th evening, 7:15 PM (the evening before). Room 1153ME	Browse the Appendices of the CUDA Programming Guide 7.0 (link on class website) Read Chapters 1 through 3 of CUDA Programming Guide 7.0

				(link on class website)
10/12	**NO CLASS**			Read 2011 GPU Gems 4 paper of Nathan Bell and Jared Hoberock (link on class website)
10/14	**NO CLASS**			Read 2010 GTC talk of Volkov (links on class website)
10/16 [L14]	Using the CUDA profiler. Example: 1D Stencil Operation	Negrut		Read "CUDA C Best Practices Guide".
10/19 [L15]	Tiling as a Programing Pattern in CUDA Example: Vector Reduction in CUDA	Negrut		
10/21 [L16]	CUDA Optimization Issues. Resource Utilization Issues. Parallel Prefix Scan on the GPU. Using Multiple Streams in CUDA.	Negrut	HW06 (due 10/28): CUDA matrix 2D convolution	Read 2010 paper written by Intel on debunking GPU performance. Read paper of Vuduc et al. on same topic (link on class website)
10/23 [L17]	Streams, and overlapping data copy with execution.	Negrut		
10/26 [L18]	Wrap up, streams. GPU Computing with `thrust`	Negrut		
10/28 [L19]	Wrap up, GPU computing.	Negrut	HW07 (due 11/04) CUDA parallel reduce operation & cudaMemcpy overhead	
10/30 [L20]	Parallel Programming with OpenMP: Work Sharing (Sections, Tasks)	Negrut		
11/02 [L21]	Parallel Programming with OpenMP: Variable Scoping	Negrut		
11/04 [L22]	OpenMP Scoping OpenMP Synchronization	Negrut	HW08 (due 11/11): Parallel scan operation in CUDA.	
11/06 [L23]	Advanced OpenMP Features	Negrut		
11/09 [L24]	MPI Parallel Programming General Introduction, Point-to-Point Communication	Negrut		
11/11 [L25]	MPI Parallel Programming Point-to-Point communication: Blocking vs. Non-blocking sends	Negrut	HW09 (due 11/18): Related to `thrust` and OpenMP	
11/13 [L26]	MPI Parallel Programming: MPI Collectives	Negrut	**Two page Final Project Proposal due at 11:59 PM**	
11/16 [L27]	Overview of C++11, TBB, cilk, Chapel	Negrut		
11/18 [L28]	Overview of Charm++		HW10 (due 11/25):Related to OpenMP	
11/20				
11/23	Review for Comprehensive Exam		**Second Exam is at 7:15 PM. Room 1610EH.**	
11/25	NO CLASS		HW11 (due 12/02):Related to MPI	
THANKSGIVING HOLIDAY				
11/30	NO CLASS			
12/02	NO CLASS			
12/04	NO CLASS			
12/07	NO CLASS			
12/09	NO CLASS			
12/11	NO CLASS			
12/14	NO CLASS			

This class has no final exam. Final Project Presentations time slots available during finals week. Scheduled through doodle.

Final Project due on Monday, Dec. 21 at 11:59 PM (submitted through Learn@UW)