CMSC 395
Special Topics: High Performance Computing
Spring 2021

http://marmorstein.org/~robert/Spring2021/cs395.html

Lecture/Lab: 11:00am – 12:15pm TR (Stevens 118)

Instructor: Robert Marmorstein (marmorsteinrm@longwood.edu) **Phone:** (434)395-2185
Office Hours: 3:15 – 4:30 MTWF (**held online)** **Office:** Rotunda 329

I am also available by appointment. My schedule is posted near my office door. To make an appointment, please check the schedule to see which times I am free, then contact me by e-mail and list some possible times we could meet. In general, I need at least 24 hours of notice to schedule an appointment.

Communications Policy
The best ways to get in touch with me are by Slack or by e-mail to marmorsteinrm@longwood.edu. Typically, I will reply within 24 hours (often sooner) on weekdays. I often reply much quicker – even on weekends.

If you are asking for help with a project or homework problem by e-mail, you should attach your code or your work to the e-mail or copy/paste the part you are working on into the body of the e-mail. **Do NOT attach screenshots or pictures taken on your phone.** They are hard to read and take up too much space in my inbox. In general, e-mails containing images will be deleted unread.

An even better way to get in touch with me is to use **Slack**. Slack is a chat utility with clients for mobile devices and desktop computers. It will allow you to easily send me code snippets. Also, since I get notifications when a slack message comes in, I am more likely to reply to your message quickly if you use Slack than if you send me e-mail.

Slack is also a good way to communicate with other members of the class. Feel free to ask for help on the course Slack channel – as long as you stick to general questions about topics and do not share large blocks of code.

Course description:
Selected topics in computer science. The topics vary from semester from semester. **(3 credits)**

Prerequisite:
CMSC 162 or permission of instructor.

Required Textbook:
The textbook for this course is “An Introduction to Parallel Programming” by Peter S. Pachecho, Morgan Kauffman Publishers, 2011, ISBN: 978-0-12-374260-5

Course Student Learning Outcomes:
By the end of the course, the successful student will be able to:
– Apply computer science knowledge to topics in a specific area of specialization

Section Student Learning Outcomes:
By the end of the course, the successful student will be able to:
– Compare different parallel and distributed computing architectures
– Use shared libraries and system calls in software
– Implement shared memory parallelism using POSIX threads and OpenMP

– Implement distributed parallelism across a supercomputing cluster using MPI

Course Structure and Student Expectations:
This course is heavily project-driven. There will be roughly four major projects and you will have roughly three weeks to work on each. In general, I will lecture on Tuesdays and Thursdays for two weeks, then give you a work week to ask me questions and complete each project. The projects will be non-trivial and you should start them the day I hand them out, not wait for the project work week.

In addition to the projects, I will expect you to read chapters of the textbook and complete exercises from the end of each chapter. Many of these exercises will require additional programming, perform mathematical calculations, or demonstrate comprehension of the topics in the chapter. While I will try to give a rough overview of the topics during lecture, the textbook covers additional material that is important to your understanding of the topics. I expect you to read each week's assigned reading before coming to class the next Tuesday.

Class sessions will take roughly **three hours** each week. In addition to lectures and in-class lab sessions, you should expect to spend at least **nine to twelve additional hours** each week reading the materials, working on homework and the laboratory projects, and preparing for exams.

Grading Scale:		100-91:	A	90:	A-
89:	B+	88-81:	B	80:	B-
79:	C+	78-71:	C	70:	C-
69:	D+	68-64:	D		
63 or lower: F		(There is no grade of D- in this course)			

Grading Policy:
Late work will not be accepted unless you have a serious medical or family emergency which prevents you from completing the assignment on time. In such cases, you do not need a doctor's note, but you must send me e-mail within twelve hours of the assignment due date to explain your circumstances and to make arrangements for the work to be completed.

Course Requirements and Major Assignments:
This class is heavily project driven and most of your grade will come from successful completion of the projects. However, there will also be a final exam, end-of-chapter homework assignments, and a final review packet (counted toward the homework grade).

Projects:

Projects are worth 50% of your grade. There will be four major projects. For tentative due dates, see the course schedule below.

Exams:

The final exam will be held on and is worth 15% of your grade.

Homework and Quizzes:

Homework assignments (largely based on the reading) will comprise 30% of your grade. In general, I will assign these at the beginning of the week and they will be due on Tuesday the next week, however, due dates may vary depending on the schedule so check the course web site for updated due dates. I also give unannounced pop quizzes, so make sure you have done the reading for each week by the beginning of that week!

Probably the largest part of your homework grade will be the final review packet at the end of the year which will be due on April 22nd.

Course Work: Your grade will be determined by your performance on the final exam (15%), participation (5%), homework and quizzes(30%) and programming projects (50%).

Attendance Policy:

Because this class will be taught both in-person and synchronously online, "attending" a class means either:

On days you are scheduled to be in person: showing up for class prepared to learn.
On days you are scheduled to be online: connecting to the stream with video enabled and actively listening to lectures or engaging in class discussions.

I will expect you to have video enabled when you are connected to the stream. If, for some reason, video is inadvisable, you must make arrangements with me in advance to reach an accommodation.

In general, I expect you to attend class unless you are sick or engaged in an approved extracurricular activity.

Please do NOT come to class if you are sick. Instead, contact me within 12 hours of the absence to report your illness and make arrangements for completing any missed work. You should also check the course web site for announcements, new assignments, and other important updates. It is your responsibility to make up any missed work and get notes on any material you have missed.

If you are well enough to stream, I may allow you to do so, but you should give me at least 24 hours prior notice if you need to switch to the online section.

I will rely primarily on your honor for enforcement of the attendance policy. However, I will keep a record of your attendance as required by Longwood policy. In accordance with that policy, I may (at my discretion) penalize you for missing more than 10% of scheduled class time (about 5 class sessions) to unexcused absences. If you miss 25% or more of scheduled class meetings (about 14 sessions), you will automatically fail this course.

Honor Code/Collaboration:
Exams and quizzes are to be completed entirely on your own. You may discuss the homework and lab projects with other students subject to these restrictions:

1. Only turn in work which YOU have typed or written.
The work you submit should, in general, be either your own original work or modifications of material which I have provided. You MAY assist other students or get assistance with simple problems like syntax errors, but you may NOT copy large blocks of code from each other. A good guideline of what "large" means is that changes that involve one or two lines of code are usually okay, but copying more than three complete statements is usually too much.

2. You may NOT copy code electronically from other students or the Internet.
This doesn't mean you can't look online for help with a project. It just means that you shouldn't copy/paste or download code and turn it in as your own. You must re-type any code you find. You should also not be using large blocks of code from the Internet (again, the three line limit is a good rule of thumb).
*You may not share code with other students using flash drives, cell phones, e-mail, web sites, floppy disks, CDs, or **any other** electronic storage or communication device. You may not print out copies of your code to share with other students (personal copies are fine).*

3. You must give proper attribution.
Whenever you receive help or use an online resource, you should comment your code to give proper credit. A simple comment like "/ based on* http://codewarrior.com **/" is fine. This comment should go directly above or directly after the place that you used the resource or received help to make it clear which parts of your program are not entirely original.*

4. You are responsible for securing your code.
Helping other students to cheat is also cheating. Furthermore, it is your responsibility to make sure that other students do not use your work to cheat. Be careful with who you allow to access your computer or account. Report any missing files, flash drives, or other devices that contain your work to me promptly.

Infractions of these policies will be dealt with harshly under the Longwood Honor Code. Any student convicted of an honor offense involving this class will automatically receive a final course grade of **F** in addition to any penalties imposed by the Honor Board. You should consider all work in this class to be pledged work, whether or not the pledge appears on the assignment.

Food and Drink:
You may bring non-alcoholic beverages, including soft drinks, to class. However, please do not eat in class (it distracts me and the other students). Violations of this policy will be considered an unexcused absence. I occasionally grant exceptions to this rule for students who must otherwise forgo lunch or have medical needs that require them to eat in class. If you feel that you need such an exception, you must make arrangements with me in advance (that is, before bringing food to class).

Cell Phones and Laptops:
Cell phones, music players, and laptops are to be turned off and put away during class, except as needed for the lab sessions. Violations of this policy will be considered an unexcused absence.

Tentative Class Schedule:

Jan. 14	Introduction to High Performance Computing Programming in C Using library functions and system calls **Read Chapter 1**
Jan. 19–21	Parallel Architectures Processes and Threads Memory allocation and pointers **Read Sections 2.1 – 2.5**
Jan. 21	**Last day of Add/Drop (by 5pm)**
Jan. 26–28	Measuring Performance Foster's Methodology/Parallel Design Compiler Flags and Makefiles **Read Sections 2.6 – 2.10**
Feb. 2–4	**Project Work Week** Project 1: Measuring Performance (Due: Feb. 4, 11:59pm)
Feb. 9–11	Introduction to MPI Applications of HPC: Monte Carlo Simulation **Read Sections 3.1 – 3.3**

Feb. 16–18	Intermediate MPI: Map/Reduce and Scatter/Gather Applications of HPC: Sorting **Read Sections 3.4 – 3.8**
Feb. 23–25	Project Work Week **Project 2: Using MPI (Due: Feb. 25, 11:59pm)**
Mar. 2–4	Shared Memory Parallelism POSIX Threads Applications of HPC: Matrix Multiplication **Read Sections 4.1 – 4.3**
Mar. 9–11	Synchronization Race Conditions and Mutual Exclusion Barriers, Conditions, and Locks **Read Sections 4.4 – 4.9**
Mar. 16–18	Project Work Week **Project 3: Using Pthreads (Due: Mar. 21, 11:59pm)**
Mar. 23–25	Programming with OpenMP Pragmas and Directives Parallel For Loops **Read Sections 5.1 – 5.5**
Mar. 30	Catch up and Review
Mar. 31	**Deadline to Withdraw without an F (5pm)**
Apr. 6–8	GPU Programming with OpenCL Applications of High Performance Computing: Genetic Algorithms
Apr. 13–15	Project Work Week **Project 4: Using OpenMP and OpenCL (Due: Apr. 15, 11:59pm)**
Apr. 20–22	Catchup and Review
Apr. 30	**Final Exam (Friday, 3pm)**