CSCI 338: Parallel Processing

Syllabus for Spring 2025

General Info

Instructor:	Kelly Shaw
Email:	kshaw@cs.williams.edu
Office:	TCL 309
Lectures:	TuTh 9:55am – 11:10am, Schow Library 030A
Help hours:	TBD
Webpage:	http://www.cs.williams.edu/~kshaw/cs338/index.html
Textbooks:	*Unix Systems Programming: Communications, Concurrency, and Threads* (2nd Edition) by Kay Robbins and Steve Robbins *An Introduction to Parallel Programming* (2nd Edition) by Peter S. Pacheco and Matthew Malensek *Recommended:Programming Massively Parallel Processors: A Hands-on Approach* (4th Edition) by David B. Kirk, Wen-mei W. Hwu, and Izzat El Hajj

Course Description

This course explores different parallel programming paradigms used for writing applications on today's parallel computer systems. The course will introduce concurrency (i.e., multiple simultaneous computations) and the synchronization primitives that allow for the creation of correct concurrent applications. It will examine how a variety of systems organize parallel processing resources and enable users to write parallel programs for these systems. The primary objective of the class is to learn how to map algorithms to different types of systems and to assess applications' performance on different systems based on applications' resource demands and systems' characteristics. Throughout, students will explore these issues using several forms of graph algorithms.

Grading Details

Grades will be computed as follows:

5%	Class participation
5%	Reading assignments
40%	Programming assignments
30%	Midterm
20%	Final project

Each of these items are explained in detail in the following sections. **In general, beyond the 3 hours we spend together during our class meeting time, you should expect to spend (on average) at least 10 hours per week on work related to class.**

Class Participation

Lectures are mandatory, and you are expected to attend regularly. One goal of this course is to promote discussion of the assigned topics among all class members. As such, you are encouraged to ask questions, point out problems, and make observations during class.

Reading Assignments

For each assigned reading, students will be asked to answer a small number of questions about the reading before lecture. **Answers for reading assignments will be due before the beginning of the corresponding lecture. Late submissions will not receive credit.** The two lowest scores on reading assignments will be dropped.

Programming Assignments

There will be multiple assignments in this course that entail writing code, evaluating code performance, and/or communicating results and insights (e.g., write-ups and presentations). Assignments should be turned in by the posted due date. Each student may use a maximum of **three late days** on non-presentations components during the course of the semester. A late day permits you to hand in an assignment up to 24 hours late, without penalty. Once those late days are exhausted, late programming assignments will be penalized 20% per day. To use a late day, you must email the instructor in advance and include a note in the README file accompanying your submission. You may not use late days on the final project or on reading assignments.

We will primarily use the Computer Science Department's Linux computers for our programming assignments, but we will also use some cloud resources. Due to the nature of the course material, different libraries and compilation environments will be used for different assignments, but most assignments will be completed using the C/C++ programming languages. More complex programming assignments will be worth more points towards your final grade than simpler programming assignments.

Midterm Exam

There is a written midterm exam in this course. It will stress conceptual understanding of the material. Details regarding the specific format of the exam will be discussed in class. **The midterm exam will be Thursday, March 13, in class.**

Final Project

In lieu of a final exam (and in addition to the programming projects described above), you will complete a final project that builds on the knowledge gained throughout the semester. You will have a choice to evaluate a new-to-you graph algorithm across multiple paradigms discussed in class or to evaluate a familiar graph algorithm on a new specified platform. In addition to completing the programming and evaluation aspect of the project, you will present your results in class and write a paper describing your work.

The final project paper and project code will be due during Reading Period/Finals Week (at 5pm on Saturday, May 24). No extensions are possible as this is a college policy.

Collaboration

I want this class to be collaborative and for us to learn from each other. I encourage collaboration and assistance in understanding material but not in developing solutions; for developing solutions, you will only be allowed to work with an official partner if the assignment permits. When solving programming assignments, discussing how to use different libraries (i.e. syntax) and looking up online how to use different libraries are allowed. Discussing or searching online for solutions to specific questions on reading assignments, programming assignments, and exams is explicitly not allowed. **Please be sure to give explicit credit for any help received.** If you have any doubts about this, ask me whether or not collaboration is appropriate. Uncredited collaborations will be considered a violation of the honor code. The Computer Science honor code and computer usage policy applies to all material in this class. Please review this info at `https://csci.williams.edu/the-cs-honor-code-and-computer-usage-policy/`.

Help

Help. We all need it. There are many resources available when {\em you} need it. You are encouraged to discuss any questions, concerns, difficulties, or thoughts about the course with me.. If you find yourself facing challenges beyond the typical, I encourage you to reach out. Talk to me, a friendly face from the Dean's Office, or some of the many professionals across campus who stand ready to help.

Students with disabilities or disabling conditions who experience barriers in this course are encouraged to contact me to discuss options for access and full course participation. The Office of Accessible Education is also available to facilitate the removal of barriers and to ensure access and reasonable accommodations. Students with documented disabilities or disabling conditions of any kind who may need accommodations for this course or who have questions about appropriate resources are encouraged to contact the Office of Accessible Education at oaestaff@williams.edu.

Students experiencing mental or physical health challenges that are significantly affecting their academic work or well-being are encouraged to contact me and/or speak with a dean so we can help you find the right resources. The deans can be reached at 597-4171.

Some other campus resources:

- The Health Center: Sometimes your challenges are not course-related. The Health Center provides a range of medical, psychological, and health/wellness services. https://health.williams.edu

Inclusivity

The Williams community embraces diversity of age, background, beliefs, ethnicity, gender, gender identity, gender expression, national origin, religious affiliation, sexual orientation, and

other visible and non visible categories. I welcome all students in this course and expect that all students contribute to a respectful, welcoming, and inclusive environment. If you have any concerns about classroom climate, please come to me to share your concerns.

In this class, we use the name and gender pronouns that individuals ask us to use as a sign of mutual respect. I will use the pronouns you have indicated on GLOW unless you alert me to a different pronoun. That said, everyone makes mistakes—in general, should you use an incorrect pronoun or name, the best course of action is to make a quick correction and move on, rather than dwelling on it.

Public health and COVID-19 Safety
\noindent If you are sick (especially if you have Covid symptoms or have tested positive for Covid), please do not come to class or to in-person office hours.

Recordings and class materials
As per College policy, no part of this course may be reproduced and/or distributed. In particular, no videos recorded as part of this class may be shared with anyone external to the course. Please see the Williams College policy online.

Topics

The following is an outline of topics that we will cover in this class. Most of this class will use graph algorithms for exploration of these topics.

- Quantifying application resource demands
- Modeling application performance on specific platforms
- Concurrency and parallelism
- Multiprogramming with processes
- Message passing
- Threads and synchronization
- Graphics and vector processing
- Artificial intelligence / machine learning processing
- Cluster computing