CARTHAGE COLLEGE

Fall 2022 – CSC 4730 Section 1

Professor:	Perry Kivolowitz
Office:	Straz 090 – I am not likely to be there – Use Google Calendar
Office Hours:	Very available By Appointment - Use Google Calendar No close contact – all office hours via zoom
Contact Info:	pkivolowitz@carthage.edu
Meeting pattern:	MWF 1450 to 1555
Texts (Required):	(1) Operating Systems: Three Easy Pieces Remzi Arpaci-Dusseau, Andrea Arpaci-Dusseau http://pages.cs.wisc.edu/~remzi/OSTEP/ *Please consider purchasing the book* (2) https://pdos.csail.mit.edu/6.828/2012/xv6/book-rev7.pdf Note: this book might refer to a newer xv6 than what we use so there may be slight differences.
Text (Suggested):	Advanced Programming in the Unix Environment (any edition) Stevens and Rago

Syllabus and Policies (Subject to change)

At the end of this course, you will be able to…

At the end of this course, you will be able to:

- Understand how an OS initializes an application and causes it to run
- Understand how OSs virtualize the CPU
- Understand how OSs virtualize memory
- Understand how OSs virtualize persistent storage
- Deeply understand multithreaded programming and synchronization
- More easily identify and fix software artifacts (bugs) in your applications
- Add fundamental features to the internal implementation of an OS
- Use the Linux command line environment
- Utilize `gcc` /g++ and `make` in a development tool chain

Topics

My friends, the Professors Arpaci-Dusseau, divide operating systems into the three (easy) pieces: Virtualization, Concurrency and Persistence. They recently added another piece (Security) but their text is still called OSTEP (TEP ☞ Three Easy Pieces).

Their text is divided into (more than) 57 relatively short chapters. While we will not cover every chapter, *each of our weeks will cover from one to three chapters.* Chapters to be skipped are primarily in the third part of the text (Persistence) and the new fourth section (Security).

Topics we will cover:

Process API	Address Translation	Concurrency	I/O Devices
Direct Execution	Segmentation	Threads	Spinning Disks
CPU Scheduling:	Free Space Management	Locks	RAID
• MLFQ	Paging	Locked Data Structures	File Systems
• Lottery	TLB's	Condition Variables	File System Checking
Address Spaces	Hierarchical Paging	Semaphores	Flash & SSD
Memory API	Swapping	Concurrency Bugs	

Difficulty of this course

This course is known at UW Madison as one of the most demanding in the CS curriculum. One false step and it isn't your program that crashes… your "machine" crashes. Often, no part of your project will work until it *all* works.

The most difficult part of this course will be concurrency. The frustration you will experience with concurrency is well spent because your future is multithreaded and multicore. These are skills you must master to become superior to your peers graduating from other colleges.

The projects in this course have occasionally been likened to death marches. This demeans and disrespects true death marches such as Bataan or the Trail of Tears[1]. With that said, the projects do have a reputation sufficient to justify the

CS 537 (as this course is known at UW-Madison) shirt shown at right.

Here at Carthage College, this course is also known as one of the most demanding in our CS curriculum. You may object to the long hours that go into this class's projects but almost without exception, students come out the other side being much more confident in their abilities and appreciate the lessons taught in this course for years to come.

Projects

Projects used to come in two varieties – userland and kerneland. Some will be user programs that execute like all the others you have written. Some used to be inside a Unix kernel square in

[1] If you are not familiar with these, remedy this gap in your education immediately.

the belly of the beast. You would have been implementing things that seemed like magic to you during your CS childhood. The magician's tricks would be laid bare. Unfortunately, I can no longer find an environment that will work for you. So, all projects will be userland and I will only be able to describe implementing features within an OS. In a way, this makes things easier for you. Or not.

Every project provides one grace day. After that, no credit will be given.

Depending on scheduling you can assume a minimum of six projects and a maximum of eight though this many is unlikely.

All projects will last nine days from date of assignment to due date. Projects may be assigned every seven days so there may be weeks where you have two current projects simultaneously. Using nine days (ten including the grace day) will put you behind. You are advised to finish your projects in seven.

Make sure you understand the specification.

Make sure you clear up any questions by asking them in class preferably, but I am also available to help outside of class.

You should work with a partner when authorized (per project basis).

Each project can be expected to take from 10 to 20 person-hours.

You can meet in groups to discuss and share ideas, but you cannot share code.

Grading

Projects	100 percent

COVID related requirements

As campus policy seems to change, I state mine. These are non-negotiable:

- Properly wearing a mask *may be* required in our classroom at *my* discretion.
- I will not do face-to-face office hours. Technology-based alternatives exist. See below: My office hours.

You are free to choose whether to comply with my requirements. If you choose not to, drop the class. Unfortunately, this is a required class, and I am the only instructor presenting it so dropping the class would have serious repercussions.

My office hours

I have always been very available to students. In the era of COVID I am even more available. Just make an appointment when it is convenient for you, using Google Calendar. By adding my calendar to yours, you'll see when I am available.

Select show co-worker's schedule and enter "pkivolowitz". When my schedule is visible, pick a blank spot. Click on the blank spot you want and enter an informative title and adjust the time carefully. THEN MAKE SURE YOU SEND ME AN INVITATION. Do this by entering my email address where guests are specified. Make sure you click the "send invitation" button shown to you when you've saved the meeting.

No office hour will be offered face-to-face in person. All contact will be by digital means, even if it means we're separated by only my office wall.

Advising and accommodations

The Carthage Advising Center offers a variety of services and accommodations to students with disabilities, based on appropriate documentation, nature of disability, and academic need. In order to initiate services, students should meet with Diane Schowalter at the start of the semester to discuss reasonable accommodation. After meeting with Diane Schowalter, students in need of accommodations should also speak with individual faculty members from whom accommodations are sought to communicate their needs and make requests in a timely manner. If a student does not request accommodation or provide documentation, the faculty member is under no obligation to provide accommodations. You may contact Diane Schowalter at ext. 5802 or via e-mail at dschowalter1@carthage.edu.

Attendance policy

Miss class at your peril. You are expected to attend every class in its entirety. If you must miss class:

1. **Be sure you connect with someone who was there for notes and discussion.**
2. **Let me know in advance.**

For every five classes missed, your final grade will be reduced by a full letter. Absences due to sports, music, or any other campus activity count as an unexcused absence. To be clear: A full letter grade means exactly that. For example, a B+ will be reduced to a C+.

We meet for 65 minutes. There is little room to "leave early." Therefore, leaving a class early will be counted as a half class missed.

We meet for 65 minutes. There is little room to "arrive late." Therefore, arriving late to a class will be counted as a half class missed.

Illness and family crises count as *excused* absence *only if you let me know prior to classes missed*.

Academic Honesty Guidelines

Carthage College Academic Honesty Guidelines are found at:

https://www.carthage.edu/community-code/academic-concerns/academic-honesty-guidelines/

A high-level summary follows but you are encouraged to consult the original document.

Topic	*Summary*
Plagiarism	Don't do it
Cheating on a Test	Don't do it
False Citation	Don't do it
Multiple Submissions	Don't do it
False Data	Don't do it

Folks, it is really easy to spot copied code. Just. Don't.

Do not get stuck (for long)

There is no honor or glory in bashing your head against a wall for tens of hours with little or no progress. Rather, bash your head for an hour or two then ask questions in class, talk to a classmate (talk means talk – don't look at code), via email to me or via office visit.

Possibility of changes made to the course

Please note the above schedule, policies, procedures and assignments in the course are subject to change in the event of extenuating circumstances, by mutual agreement and / or to ensure better student learning.

Concerning learning outcomes

Students may vary in their competency levels of any stated outcomes. Students can expect to achieve stated outcomes *only if* they honor all course policies, attend classes regularly, complete all assigned work in good faith and on time and meet all other course expectations of them as students.

Suggestions for how to do well in CS courses

Start Early – Work Steadily

Ask For Help