

# CSC4730-Fall-2023

This document serves as both the course's syllabus and its main web page.

## Course Summary

In this course you will not only open the curtain to see the wizard, you will strip the wizard bare.

The biggest parts of this course include:

- We will cover nearly all of the great free textbook: OSTEP - Operating Systems in Three Easy Pieces. This book is written at a popular level and is very accessible.
- We will leverage the Unix programming environment in a deep and meaningful way. These are marketable skills.
- We will do up to 8 projects both inside an OS itself and in userland.

## Learning Outcomes

There are too many learning outcomes in this course to name them all.

These four learning outcomes are highlighted:

1. At the end of this course, students will be able to design and implement complex multi-process and threaded applications.
2. At the end of this course, students will be able to design and implement various synchronization schemes using mutexes, condition variables and semaphores.
3. At the end of this course, students will have mastered user level memory allocation schemes including free space management.
4. At the end of this course, students have designed and synthesized complex data structures and algorithms demonstrating mastery of the structure of at least one kind of file system.

Implicit in this course is learning the lessons of care, testing, debugging and commenting.

## Meetings

Days	Times	Room
TR	2:20 PM to 4:00 PM	CC 114

### **In the event of virtual classes**

Should we meet via Zoom. **All cameras must be on to the maximum degree possible.** If you are on a desktop, get a Webcam, they are cheap enough. Not having a Webcam is not an allowable excuse.

The link given below will be used for classes, should that be needed.

### **Office hours**

All office hours are held via ZOOM. This link will remain valid all term.

Day	Times
M	11 AM to 12 PM
T	11 AM to 12 PM
R	7 PM to 9 PM
F	12 PM to 1 PM

Office hours may be recorded if the content is deemed by me to be of general use to the class.

### **Meetings / Questions outside of office hours**

Due to abuse of the instructor's time during Spring 2023, help shall be largely limited to the above hours. Make sure you ask your questions during class or during posted office hours.

### **Meetings / Questions prior to project due dates**

With some exceptions, to be decided by me, help on projects will not be available within 24 hours of a project's due date. This prohibition is imposed to encourage good working habits including but not limited to starting work early / working steadily. This is not intended punitively.

### **Texts**

Primary: OSTEP This book is free.

Ancillary:

1. Required: Advanced Programming in the UNIX environment, third edition. I will refer to this as Stevens and Rago. If you have any questions about the Unix / Linux environment and APIs, consult this book before approaching me.
2. xv6 risc-v This book is free.

## Readings

There are two types of readings in this class.

1. Assigned readings will come from OSTEP.
2. Unassigned readings will come from the two ancillary texts. Being unassigned, you will **not** be told when to do them. Instead, review the table of contents in each of the two ancillary sources to get a general idea of that they contain. Then, later in the course, when you have a question, review the ancillary texts for details.

You expected to have consulted the texts before approaching me with questions. If I should answer something to the effect that the answer is in the book, you are expected to explain to me what you read and articulate what you found unclear. This step is take to encourage good working habits on your part and is not punitive.

## Topics Covered

The following are the topics which we can expect to be covered:

OSTEP	Topic	OSTEP	Topic
4	Processes	21	Swapping Mechanisms
5	Process API	22	Swapping Policies
6	Limited Direct Execution	23	Complete VM System
7	CPU Scheduling	26	Concurrency and Threads
8	Multi-level Feedback	27	Thread API
9	Lottery Scheduling	28	Locks
10	Multi-CPU Scheduling	29	Locked Data Structures
13	Address Spaces	30	Condition Variables
14	Memory API	31	Semaphores
15	Address Translation	32	Concurrency Bugs
16	Segmentation	33	Event-based Concurrency
17	Free Space Management	36	I/O Devices
18	Introduction to Paging	37	Hard Disk Drives
19	Translation Lookaside Buffers	38	RAID
20	Advanced Page Table	39	Files and Directories
		40	File System Implementation

There are 31 chapters listed above. While we may not get to all, you can be certain there will be weeks in which you will be responsible for reading more than one chapter. This is not a major burden as the chapters are short and the text is written at a popular level.

## Environment

All projects require use of WSL on Windows or the terminal on the Mac. If you're passionate about using Linux directly, this is also an option but no help will be provided.

The easiest way to install WSL (Ubuntu) on Windows 11 is to run a command prompt as Administrator and then:

```
wsl --install
```

If this doesn't work, use the Microsoft Store to install Ubuntu.

It is possible you'll have to turn on the Windows Subsystem for Linux in the Add Remove Windows Features settings.

I recommend you select a trivial user name and password. For example, for my WSL I use user name **a** and password **a**.

You'll then need to install the C and C++ environment.

```
sudo -i
apt update
apt upgrade
apt install build-essential gdb
```

By now you will likely have already installed the plugins in VS Code needed to work inside WSL. If not, ask a friend.

On the Mac, ensure you have **gcc** and **g++** available to you on the command line.

## Installing xv6

On WSL:

```
sudo -i
apt update
apt install crossbuild-essential-riscv64
exit
# decide where you want to load the original xv6
# and change directory to there.
git clone https://github.com/mit-pdos/xv6-riscv.git
```

Make a copy of the xv6 directory for each of the projects using it.

On Mac:

Ensure you have the latest **brew**.

You'll also need the riscv toolchain plus qemu.

```
brew install qemu riscv-gnu-toolchain
```

And as per Windows, get the xv6 repo. It is likely there is a small change to be made to the `makefile` to specify the appropriate tool chain.

### **man Pages**

Since the early days, Unix has had the `man` command. For example:

```
man man
```

or

```
man 2 chdir
```

Consult the man pages before approaching me with questions. Prior to my answering a question whose answer is found in the man page, I will require you to state your understanding of the man page and articulate where your understanding is insufficient. This step is taken to encourage good working habits and is not punitive.

### **Attendance**

Attendance will be taken at nearly every class. The only exceptions will be when I am so excited to get down to business, that I forget to take it.

You get 6 free absences. After that each absence will remove 1 percent from your aggregate score at the end of the term. All absences are treated equally. That is, there are no “excused” absences.

The number of free absences equates to three weeks of missed class time so it is unlikely you’ll exceed this and incur a penalty.

Should a long-term absence due to uncontrollable circumstances become necessary, please speak to me as soon as possible.

### **Arriving Late**

Once attendance has been taken, anyone arriving after than will be counted as absent. Generally, attendance will end between five and ten minutes after the official start of class.

### **Leaving Early**

No, your coach cannot force you to leave class early to make practice.

### **Essay**

Here is a link to an essay assignment. You can look at it at your convenience but do it only when assigned. The essay is six percent of your grade and, at 500 words, is one third to one sixth the length of a typical college essay. Five

hundred words about life, death and your responsibility as a coder is not too much to ask.

## Projects

Below is a listing of 10 “projects” of which only 8 are truly projects. One is a tutorial and another is just to provide proof you’re ready to tackle the two kernel projects.

The *Weight* column provides a weighting of the project within the Project Grading Component. It suggests my anticipation of the relative difficulty of each project compared to the others.

The *OSTEP* column indicates how the projects correspond to course material. The indicated chapters will be most helpful.

The *S&R* column indicates which chapters of Stevens and Rago can be helpful.

The warmups each contribute one point to your final grade.

The projects together contribute 92 percent of your final grade.

#	Project	Space	Weight	OSTEP	S&R
-	<b>getopt</b>	User	N/A	N/A	N/A
1	Shell	User	70	4, 5	1, 3, 4, 5, 8, 15
-	Working xv6	N/A	N/A	N/A	N/A
2	System Call	Kernel	60	6	N/A
3	Schedulers	User	70	7, 8, 9	N/A
4	Guard Page	Kernel	60	13	N/A
5	Allocators	User	70	17	7
6	Semaphores	User	70	26, 27, 28, 29, 30, 31	8, 11, 12
7	Named Pipe	User	100	26, 27, 28, 29, 30, 31	8, 11, 12
8	FSCK	User	100	39, 40, 42	3, 4, 5

## Project Partners

- You are **required** to use a partner for all non-warmup projects.
- Your partner will be **assigned** to you by me.
- Members of the team must be specified by the partner handing in the project within the code they are handing in.
- The partner not handing in the project must submit a text file stating who their partner is.

- Both partners receive the same grade.
- Any partnership problems should be worked out by you but I can step in if necessary.

### Project Requirements

- Requirements are spelled out in each specification.
- Any deviation from the specification likely results in penalties.
- Generating even a single warning during building your own code will result in a 10 percent penalty.
- Commenting is required. Failure to adequately comment is a mandatory 5 percent penalty.

### Late work

Late work (factoring in one free grace day) will not be accepted. To state this differently, if an assignment is due on day  $n$ , you have until the end of day  $n + 1$  to turn it in. After that, it will not be accepted.

I award partial credit even for substantial but incomplete work. This is unlike the real world where incomplete work earns termination of your employment. Hand in what you have... it's better than a zero.

### Grading Criteria

Gradable Component	Weight	Notes
Projects	92%	
Essay	6%	
<code>getopt</code>	1%	warm up
<code>working xv6</code>	1%	warm up
Attendance	Deduction Only	

There is little likelihood a curve will be applied in computing your final grade.

### Accommodations

There are no exams in this course and projects already include a grace day. Therefore, no accommodations ought to be necessary.

### Academic Honesty

All work will be examined for originality. Should I find evidence of copied work, all parties shall receive a grade of zero. If any individuals are found to have

copied a second time, the individuals will be reported to the Provost's office for discipline and will receive a zero for the assignment. Should an individual be associated with a third instance, in addition to a second trip to the Provost, the individual shall receive an F in the course.

The decision of the instructor is final.

## **Bonus Materials**

See here for some very old slides I made describing a few topics in very succinct terms.

## **General statement about “help”**

Apart from lecture, I will not “just give you the answer.” Rather, you must demonstrate that you have exhausted any means available to you to learn on your own. This is not punitive. This will be a requirement for your future. You'll get in the habit now. This may be a lasting and meaningful “gift” to you from me.

## **Jishnu Mukerjee, Senior Systems Architect at Hewlett-Packard**

When I was an undergraduate, Jishnu Mukerjee was the graduate student who was in charge our department's PDP-11/60. It ran an early version of Unix, V6 to be exact... sound familiar?

The first time I had a question, I asked Jishnu and he answered it.

The second time I had a question, I asked Jishnu and he answered it.

The third time I had a question, I asked Jishnu and he handed me a print out of the entire OS. He said, “All your answers are in here.”

It was the greatest gift he could possibly give and I am thankful to him to this day.