Course Syllabus

✎ **Edit**

San José State University

Science/Computer Science
SE/CS 149, Operating Systems, Section 1, Fall 2019
Course and Contact Information

Instructor:	Ben Reed
Office Location:	MH 213
Telephone:	(408) 924-5174
Email:	ben.reed@sjsu.edu
Office Hours:	10-11AM, 3-4PM Monday 3-4PM Wednesday 6-7:30PM Tuesday & Thursday 10-11AM Wednesday in Chicanx/Latinx Student Success Center Diaz Compean Student Union 1340 (across from Jamba Juice)
Class Days/Time:	Tuesday & Thursday/ 4:30-5:45
Classroom:	SH 100
Prerequisites:	CS 146 or SE 146 (with a grade of "C-" or better)

Course Description

Fundamentals: Contiguous and non-contiguous memory management; processor scheduling and interrupts; concurrent, mutually exclusive, synchronized and deadlocked processes; files. Substantial programming project required.

Course Learning Outcomes (CLO)

Upon successful completion of this course, students will be able to:

1. Know the difference between kernel and user space.
2. Know how to manage processes.
3. Understand virtual memory management.
4. Understand CPU scheduling policies.
5. Know how to manage threads and the issues associated with them.
6. Understand and implement concurrent data structures.
7. Know how to observe kernel operation and configuration.
8. Understand the issues associated with different storage technologies.
9. Understand how virtual file systems work.
10. Identify and reason about ethical issues surrounding various operating system concepts.

Required Texts/Readings

Textbook

Operating Systems: Three Easy Pieces: http://pages.cs.wisc.edu/~remzi/OSTEP/

Other technology requirements / equipment / material

Programming assignments will be a significant part of this course, so access to a computer is required. Your computer must be able to run Oracle VM VirtualBox in 64-bit mode https://www.virtualbox.org/.

Course Requirements and Assignments

Homework will be given, but will not be graded. It is intended for self evaluation and will be the basis for future exams. I encourage students to work on homework in groups and discuss possible solutions together. We will take time at the beginning of each class to discuss any difficulties students have completing the homework.

Along with technical questions in the homework, we will also discuss ethical issues related to operating systems. We want you to understand that along with technical choices come moral implications, and we want to to be able to identify and reason about them. There will be 2 written (1 page) assignments to discuss contemporary ethical issues in operating systems today.

Because the class is so large, we will be using iClicker to make sure everyone is up to speed. To encourage participation 1% of your final grade will come from your participation. Each iClicker poll that you participate in will count for 1 point and each answer you get correct will be another point. At the end of the semester the points you will recieve 100% if you get at least 70% of the total possible points. Anything under 70% will be prorated.

I do not grade on a curve. The exams and projects measure what you are expected to have learned. There aren't many opportunities for extra credit, but there are bonus questions on exams.

We will be doing individual programming assignments. Late submissions less than 24 hours late will have 10 points deducted from the final score. Submissions over 24 hours late will have 20 points deducted. Submissions over 7 days late will not be accepted. **Individual programming assignments are not group projects.** If students get help on assignments, even to resolve a stupid problem, it must be documented in the code with the name of the person rendering the help and a brief description of the help provided. Extensive help on a project will result in a reduced grade. Failure to document help, or any other forms of cheating will result in a failing grade on the assignment at a minimum and may result in failure of the course. All incidents will be reported to the Office of Student Conduct & Ethical Development. See **http://info.sjsu.edu/static/schedules/integrity.html** **(http://info.sjsu.edu/static/schedules/integrity.html)** for more information. Even in open source, you cannot copy code from one open source project to another without attribution. Sharing solutions with other students, even if it is indirectly through public source repositories, falls under "aiding and abetting".

The University Policy S16-9 **(http://www.sjsu.edu/senate/docs/S16-9.pdf)** , Course Syllabi (http://www.sjsu.edu/senate/docs/S16-9.pdf) requires the following language to be included in the syllabus:

"Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of 45 hours over the length of the course (normally three hours per unit per week) for instruction, preparation/studying, or course related activities, including but not limited to internships, labs, and clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus."

Final Examination or Evaluation

This course will have a cumulative final exam given during exam week.

There will be three in class exams given in the semester (the last being the final exam :)). The second exam will have two questions derived from the previous exam, and the final exam will have two questions derived from the first exam and two questions derived from the second exam.

Grading Information

Determination of Grades

Grades will be calculated by averaging the percentages of the average of group project grades, the individual project grades, the two mid semester exams, and the final. Thus, the grade distribution is 23% individual projects, 21% exam 1, 21% exam 2, 24% final exam, 10% ethic projects, and 1% participation via (iClicker).

Percentage	Grade

97 and above	A+
92-96	A
90-91	A-
88-89	B+
82-87	B
80-81	B-
78-79	C+
72-77	C
70-71	C-
68-69	D+
62-67	D
60-61	D-
59 and below	F

Classroom Protocol

This is your class. Please ask questions. Please come prepared. Do not engage in activity that may distract other students.

I do not take attendance except for the first two classes. Students not attending either of the first two classes will be dropped to make room for students on the waiting list. Attempting to get marked as present (by have someone else attend in your place or using technological deceptions) will be considered academic dishonesty and at a minimum will result in you getting dropped from the course.

University Policies

Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and Undergraduate Programs' [Syllabus Information web page](http://www.sjsu.edu/gup/syllabusinfo/) [(http://www.sjsu.edu/gup/syllabusinfo/)](http://www.sjsu.edu/gup/syllabusinfo/) at http://www.sjsu.edu/gup/syllabusinfo/" Make sure to review these policies and resources.

CS 149, Operating Systems, Section 2, Spring 2019 Course Schedule

Week	Date	Topics, Readings, Assignments, Deadlines
1	8/22/2019	intro to OS booting: **https://www.ibm.com/developerworks/library/l-linuxboot/index.html (https://www.ibm.com/developerworks/library/l-linuxboot/index.html)** crash course on C part 1 **https://learning.oreilly.com/library/view/the-c-programming/9780133086249/ch01.html (https://learning.oreilly.com/library/view/the-c-programming/9780133086249/ch01.html)**
2	8/27/2019	crash course on C part 2 **https://learning.oreilly.com/library/view/the-c-programming/9780133086249/ch05.html (https://learning.oreilly.com/library/view/the-c-programming/9780133086249/ch05.html) (http://pages.cs.wisc.edu/~remzi/OSTEP/intro.pdf)**
2	8/29/2019	processes: **http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-intro.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-intro.pdf)**

3	9/3/2019	process API: **http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-api.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-api.pdf)**
3	9/5/2019	direct execution: **http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-mechanisms.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-mechanisms.pdf)**
4	9/19/2019	scheduling: **http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-sched.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-sched.pdf)**
4	9/21/2019	multi-level scheduling: **http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-sched-mlfq.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/cpu-sched-mlfq.pdf)**
5	9/26/2019	files & directories: **http://pages.cs.wisc.edu/~remzi/OSTEP/file-intro.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/file-intro.pdf)**
5	9/28/2019	exam 1
6	10/1/2019	signals **(http://pages.cs.wisc.edu/~remzi/OSTEP/vm-intro.pdf)**
6	10/3/2019	memory API: **http://pages.cs.wisc.edu/~remzi/OSTEP/vm-api.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/vm-api.pdf)**
7	10/8/2019	free space management: **http://pages.cs.wisc.edu/~remzi/OSTEP/vm-**

		freespace.pdf **(http://pages.cs.wisc.edu/~remzi/OSTEP/vm-freespace.pdf)**
7	10/10/2019	paging: **http://pages.cs.wisc.edu/~remzi/OSTEP/vm-paging.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/vm-paging.pdf)**
8	10/15/2019	TLP & paging mechanisms: **http://pages.cs.wisc.edu/~remzi/OSTEP/vm-tlbs.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/vm-tlbs.pdf)** & **http://pages.cs.wisc.edu/~remzi/OSTEP/vm-beyondphys.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/vm-beyondphys.pdf)** COW **http://pages.cs.wisc.edu/~remzi/OSTEP/vm-complete.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/vm-complete.pdf)** section 23.1 in "Other Neat Tricks"
8	10/17/2019	replacement policies: **http://pages.cs.wisc.edu/~remzi/OSTEP/vm-beyondphys-policy.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/vm-beyondphys-policy.pdf)**
9	10/22/2019	threads & thread API: **http://pages.cs.wisc.edu/~remzi/OSTEP/threads-intro.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/threads-intro.pdf)** & **http://pages.cs.wisc.edu/~remzi/OSTEP/threads-api.pdf** **(http://pages.cs.wisc.edu/~remzi/OSTEP/threads-api.pdf)**
9	10/24/2019	locks: **http://pages.cs.wisc.edu/~remzi/OSTEP/threads-**

		locks.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/threads-locks.pdf)
10	10/29/2019	exam 2
10	10/31/2019	concurrent data structures: http://pages.cs.wisc.edu/~remzi/OSTEP/threads-locks-usage.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/threads-locks-usage.pdf)
11	11/5/2019	condition variables & semaphores: http://pages.cs.wisc.edu/~remzi/OSTEP/threads-cv.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/threads-cv.pdf) & http://pages.cs.wisc.edu/~remzi/OSTEP/threads-sema.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/threads-sema.pdf)
11	11/7/2019	concurrency bugs: http://pages.cs.wisc.edu/~remzi/OSTEP/threads-bugs.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/threads-bugs.pdf)
12	11/12/2019	atomic integers
12	11/14/2019	I/O devices: http://pages.cs.wisc.edu/~remzi/OSTEP/file-devices.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/file-devices.pdf)
13	11/19/2019	hard disks: http://pages.cs.wisc.edu/~remzi/OSTEP/file-disks.pdf

		(http://pages.cs.wisc.edu/~remzi/OSTEP/file-disks.pdf)
13	11/21/2019	RAID: http://pages.cs.wisc.edu/~remzi/OSTEP/file-raid.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/file-raid.pdf)
14	11/26/2019	file system implementation: http://pages.cs.wisc.edu/~remzi/OSTEP/file-implementation.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/file-implementation.pdf)
15	12/3/2019	file system implementation: http://pages.cs.wisc.edu/~remzi/OSTEP/file-implementation.pdf (http://pages.cs.wisc.edu/~remzi/OSTEP/file-implementation.pdf)
15	12/5/2019	review
Final Exam	12/17/2019	from 14:45-17:00

Course Summary:

Date	Details	
Sun Sep 1, 2019	📝 **add them up** (https://sjsu.instructure.com/courses/1322173/assignments/5098974)	due by 11:59pm
	📝 **prereq proof** (https://sjsu.instructure.com/courses/1322173/assignments/5104556)	due by 11:59pm
Tue Sep 3, 2019	📝 **process** (https://sjsu.instructure.com/courses/1322173/assignments/5111945)	due by 11:59pm