SJSU SAN JOSÉ STATE UNIVERSITY

College of Science · Computer Science

Operating Systems

CS 149

Spring 2026 Section 01 In Person 3 Unit(s) 01/22/2026 to 05/11/2026 Modified 01/26/2026

Contact Information

Instructor(s): William “Bill” Andreopoulos

Office Location: MacQuarrie Hall 416

Telephone: (408) 924-5085

Email: william.andreopoulos@sjsu.edu

Office Hours: Friday 9:00-11:00am

Class Days/Time: Monday and Wednesday, 4:30pm-5:45pm

Classroom: Washington Square Hall 109 (in-person)

Course Information

Course Format

This course adopts an in-person classroom delivery format.

Faculty Web Page and MYSJSU Messaging

Course materials such as syllabus, handouts, notes, assignment instructions, etc. can be found on Canvas Learning Management System course login website at http://sjsu.instructure.com. You are responsible for regularly checking with the course messaging system to learn of any updates. You should modify the Canvas settings for notifications of announcements and messages to be sent to you.

Course Description and Requisites

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

Prerequisite(s): CS 47 or CMPE 102 (with a grade of "C-" or better), and CS 146 (with a grade of "C-" or better). Allowed Declared Majors: Computer Science, Applied and Computational Math, Forensic Science: Digital Evidence, or Software Engineering Majors only; or Instructor Consent.
Grading: Letter Graded

✱ Classroom Protocols

Communication with the instructor

As this is an in-person section, course-related communication should preferably be done in-person during the regular class meeting time or office hours. For online communication, students should use the course Discord channel. Rather than emailing redundant questions to the teaching staff, students should post questions on the course Discord channel where the entire class can read and benefit from the responses. The system is catered to getting students help efficiently from classmates, the TA, embedded tutor, and the instructor. *Private messages sent to the instructor's other email addresses may get lost due to the large volume of emails received.*

The professor responds primarily to the Discord channel. The professor will re-post questions that are of general interest (e.g. about homework) or discuss them in class. Students are responsible for everything said in class. It is students' responsibility to keep up with what is said in class and not re-post the same questions repeatedly.

When students use the course Discord channel, they are expected to be identifiable through their names. Anonymous postings are unacceptable. Students who use fake nicknames may be removed from the Discord channel.
The instructor does not write messages after normal business hours, on weekends or holidays.

Technical trouble-shooting should be done during the office hours.

Never email your entire code for an assignment to the instructor. Limit the code you post to 20 lines or less.

Announcements that concern everyone, such as reminders about due dates or class policy, will be posted.

Class Attendance

Attendance (in-person or via Zoom) is highly recommended. When technology allows, Zoom will be available. Students are responsible for following all material presented in class.

The polling questions in the slides are in the form of multiple-choice and true-false questions. Students should participate and follow the polling questions, either via Zoom polling or Zoom chat or ask in class.

Regrading Procedure

Grades assigned are final, unless there was an error in the grading. Special requests (e.g. grade changes or deadline extensions) should be done in-person; such special requests sent via electronic messages to the teaching staff will be disregarded, since this is an in-person section. To request a higher grade, students should first submit the Canvas "Regrade request" form so there is a record of the request, and afterwards speak with the professor. Grades may be reevaluated at anytime and may go down as a result of a regrade. There will be no regrades after the end of the semester (final exam).

At the end of the semester grade roundups (e.g. 89.95% to 90%) will only be considered if a student has pursued any extra credit opportunities offered and completed the SOTE evaluation.

Classroom Protocol

Students on Zoom should be muted when not speaking, and must be dressed appropriately when their camera is on.

Course material developed by the instructor is the intellectual property of the instructor. Students can not publicly share or upload instructor generated material for this course such as exam questions, lecture notes, hands-on exercises or homework solutions without instructor permission.

Program Information

Diversity Statement - At SJSU, it is important to create a safe learning environment where we can explore, learn, and grow together. We strive to build a diverse, equitable, inclusive culture that values, encourages, and supports students from all backgrounds and experiences.

Course Learning Outcomes (CLOs)

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

CLO 1 Understand the role that the operating system software plays in the management of the various hardware subsystems of the computer system.

CLO 2 Understand locality of memory reference and how it is used to perform effective memory hierarchy management.

CLO 3 Understand the various mapping, replacement, and dynamic allocation algorithms for cache and virtual memory management.

CLO 4 Understand the alternative CPU scheduling schemes, their tradeoffs, and their applications to other queue processing situations.

CLO 5 Appreciate the difficult tradeoffs faced when attempting to deal with the resource deadlock problem and distinguish between the different deadlock prevention and avoidance schemes and understand why and how deadlocks can still happen today.

CLO 6 Understand software race conditions, their origin and the problems they can cause, along with knowing how to apply semaphores in software design to solve the race condition problem.

CLO 7 Understand the various issues associated with the operating system's role in performing I/O and file management.
Edit Course Materials item

Course Materials

Textbooks

Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau. Operating Systems: Three Easy Pieces. (*OSTEP*)

This book is available online: http://pages.cs.wisc.edu/~remzi/OSTEP/

Other Readings

- Richard Stevens, Stephen A. Rago. Advanced Programming in the UNIX Environment - 3rd Edition, 2013, Addison-Wesley. (APUE)
- Robert Love. Linux Kernel Development - 3rd Edition, 2010, Addison-Wesley. (LKD)
- Silberschatz, P. Galvin, and G. Gagne. Operating System Concepts - 9th Edition, 2012, Wiley. http://www.os-book.com/
- Handouts through Canvas.

Other technology requirements / equipment / material

In this class we will use Linux and zyBooks as our C programming environment for homework assignments and worksheets. See Canvas for details to subscribe to zyBooks or install Linux. We will use the provided C compiler for programming assignments and worksheets. Unless otherwise stated, all homework assignments and worksheets should compile and run on the particular Ubuntu and C version, which is explained on a Canvas page.

Integrated Development Environment for C - different students prefer to use different IDEs or even text editors. You can choose from visual studio, eclipse, or cLion. You can also work in vi or nano and do the compilation on the command line.

Course Requirements and Assignments

SJSU classes are designed such that in order to be successful, it is expected that students will spend a minimum of forty-five hours for each unit of credit (normally three hours per unit per week), including preparing for class, participating in course activities, completing assignments, and so on.

Reading assignments: Readings will regularly be assigned for the next class (see schedule). Slides will be posted under the Canvas modules before the next class.

Homework assignments: Programming assignments will be assigned. More information will be given at the time of the first programming assignment.

Students can work with a partner (in groups of two) on the assignments. If two students form a group, the pair of both group members will get the same grade for any assignment they submit together. *If you work with a partner, you must put your group members' names in a spreadsheet that will be provided*.

While it is fine to discuss the homework solutions with others, code solutions submitted should reflect the students' own efforts in writing the code. *Do not write the code for anyone else. Never copy any code you find on another source, such as a website or AI tool. Canvas and zyBooks automatically checks submissions for plagiarism from multiple online sources and such instances will be reported to the .* Oral examination might be requested.

All homework should be submitted online. Homework sent via an email or message will not be graded. Homework cannot be graded after it has been reviewed in class or a solution has been posted. All homework is due on the last day of class.

Worksheets: There will be worksheets with problem solving. These will generally involve coding problems (in C or bash) from the reading assignment or similar to the homework. The worksheets are a tool for you to learn the material, prepare for exams and practice coding for your future job interviews.

The worksheets are graded based on effort and get graded "complete" if a reasonable solution is proposed for each problem. *Unlike the assignments, some or all of the worksheets should be done individually (as shown on the particular worksheet).*

A worksheet submission is due approximately every week. Please refer to Canvas for detailed instructions and deadlines. As this is a fast-paced course, it is essential that you submit your worksheet homework in a timely fashion in order to keep up.

We will take time at the beginning of each class to discuss any difficulties students have in completing the worksheets from previous classes. We will also do code reviews.

Late policy: Late penalty is 2% per day up to 14 days. After 14 days (or after the last day of classes if it is sooner) the submission page will be closed and will not be re-opened. No submission will be accepted after the closing deadline.

Participation during class: 1% of the grade consists of participation points, given for answering at least 40 Zoom polling questions during class throughout the semester. The polling questions are in the form of multiple-choice and true-false questions. It is not necessary to get the correct answer in a polling question to get participation credit.

Examinations

Midterm exams: There will be two Midterm exams during the semester.

Final exam: One final cumulative exam.

The exams will contain multiple choice questions, true/false and short answer questions. Exams are closed book, closed notes, and comprehensive. Exams are in-person. The exams should be done individually. No make-up exams except in case of verifiable medical reasons.

Extra credit opportunities

Extra credit of 1% is given to a student who volunteers to review his/her code solution for an entire assignment or a worksheet in-class (either via Zoom or in person). A code review lasts for 10 minutes max. These will take the form of code reviews, where the student walks us through his/her code solution for an assignment or a worksheet, we discuss the proposed solution and if there are better ways to solve the problem. Students have to add their name to a code review worksheet to reserve a code review timeslot. An assignment or worksheet can only be reviewed once. A student may reserve one timeslot at a time. If, after presenting, there are other timeslots available, a student may reserve another timeslot.

If you attend one 30' of tutoring session with the embedded tutor before the Thanksgiving break you will receive 1% extra credit. And after Thanksgiving break you will receive 0.5% extra credit.

There may also be a bonus assignment worth 1%.

Embedded Tutor

Martin Tran vietnhatminh.tran@sjsu.edu

Graders/TAs

To be announced

Use of generative AI tools

All assignments and worksheets submitted are expected to be the students' own original work. The instructor may, at any time, ask a student to explain the meaning of any part of any answer that they submit. If the student can't explain the answer to a question sufficiently, the penalty for such incidents will be zero points on the homework and a report to the Office of Student and Ethical Conduct.

✔ Grading Information

Final Grade is based on:

40% Assignments

25% Midterms

25% Final

9% Worksheets

1% Participation (Polling questions)

Grade	*Points*	*Percentage*	*Interpretation*
A plus	*960 to 1000*	*96 to 100%*	*Exceptional*
A	*930 to 959*	*93 to 95%*	*Excellent*
A minus	*900 to 929*	*90 to 92%*	*Very good*
B plus	*860 to 899*	*86 to 89 %*	*Good*
B	*830 to 859*	*83 to 85%*	*Fair*
B minus	*800 to 829*	*80 to 82%*	*Fair*
C plus	*760 to 799*	*76 to 79%*	*Passed*
C	*730 to 759*	*73 to 75%*	*Passed*
C minus	*700 to 729*	*70 to 72%*	*Barely passed*
D plus	*660 to 699*	*66 to 69%*	*Fail*
D	*630 to 659*	*63 to 65%*	*Fail*
D minus	*600 to 629*	*60 to 62%*	*Fail*

University Policies

Per University Policy S16-9 (PDF) (http://www.sjsu.edu/senate/docs/S16-9.pdf), relevant university policy concerning all courses, such as student responsibilities, academic integrity, accommodations, dropping and adding, consent for recording of class, etc. and available student services (e.g. learning assistance, counseling, and other resources) are listed on the Syllabus Information (https://www.sjsu.edu/curriculum/courses/syllabus-info.php) web page. Make sure to visit this page to review and be aware of these university policies and resources.

Course Schedule

Week/class	Topic
01/26-01/30	Introduction, Review C and the command line
02/02-02/06	Review C and the command line
02/09-02/13	Processes / Process API
02/16-02/20	Interprocess Communication, Sockets, Pipes / System calls with File I/O
02/23-02/27	Review of Processes / Signals
03/02-03/06	**Midterm 1** / CPU Scheduling
03/09-03/13	Multilevel CPU Scheduling / Address Space
03/16-03/20	Memory API / Free-Space Management
03/23-03/27	Paging / Swapping Policies, Memory Review
03/30-04/03	*Spring recess*
04/06-04/10	**Midterm 2** / Thread API, Locks
04/13-04/17	Lock-based concurrent Data Structures / Lock and Concurrency Bugs
04/20-04/24	Condition Variables and Semaphores / Advanced Locks

04/27-05/01	Review of Threads vs. Processes, Hard Disks / Files and Directories
05/04-05/08	File System Implementations / Direct Execution, Security Topics
05/11	Review
	Final exam on Monday, May 18, 3:15-5:15pm

The schedule is subject to change with fair notice.