SJSU SAN JOSÉ STATE UNIVERSITY

College of Science · Computer Science

Operating Systems Section 02

CS 149

Fall 2023 3 Unit(s) 08/21/2023 to 12/06/2023 Modified 08/20/2023

Contact Information

Instructor(s): William “Bill” Andreopoulos

Office Location: MacQuarrie Hall 416

Telephone: (408) 924-5085

Email: william.andreopoulos@sjsu.edu

Office Hours: Friday 12:00-14:00 (Zoom)

Class Days/Time: Monday and Wednesday, 12:00pm-13:15pm

Classroom: Boccardo Business Center (BBC 204)

Prerequisites: CS146 (Data Structures and Algorithms) and CS47 or CMPE102 with a grade of C- or better

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 Â¿C-Â¿ or better), and CS 146 (with a grade of Â¿C-Â¿ or better). Allowed Declared Majors: Computer Science, Applied and Computational Math, Forensic Science: Digital Evidence, or Software Engineering Majors only; or Instructor Consent.

Letter Graded

Classroom Protocols

Communication with the instructor

Students should use the correct channels for course-related communication. Questions can be done during the regular class meeting time (in-person or via Zoom) or office hours. For online communication students should use Canvas messaging and the discussion forums:

1) We will be using the Canvas discussion forums for class discussion. The system is catered to getting you help efficiently from classmates, the TA, embedded tutor, and the instructor. Rather than emailing redundant questions to the teaching staff, students should post questions on the Canvas discussion forums where the entire class can read and benefit from the responses. The professor may re-post questions that are of general interest or discuss them in class.

2) Students are invited to join the office hours.

Private messages sent to the instructor’s other email addresses get lost due to the large volume of emails received.

The instructor does not write messages after normal business hours, on weekends or holidays.

Reviewing code for the homework and technical trouble-shooting should be done during the office hours.

Never email your entire code for an assignment to the instructor. The instructor will not fix all the bugs in your code. 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.

Embedded Tutor

Lilou Sicard-Noel lilou.sicard-noel@sjsu.edu

Graders/TAs

Likhith Nemani likhith.nemani@sjsu.edu

Sirisha Murthy sirisha.krishnamurthy@sjsu.edu

Class Attendance

Attendance (in-person or via Zoom) is highly recommended. Classes will be recorded as Zoom screencasts and posted on Canvas. Students are responsible for 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. There will be no grade change through sending electronic messages to the teaching staff. If a student wants to request a higher grade for homework, they must follow instructions on the "Regrade request" page on Canvas. After submitting a regrade request, please speak with the professor during office hours or after class. A request for a regrade is not a technique to drum up a few more points. If the course instructor thinks a component was scored too generously the first time, it may be lowered in a regrade. Thus, regrading may result in a lower grade overall. Students who request a regrade for a higher grade are expected to also be pursuing the extra credit opportunities offered.

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.

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 Ubuntu as our programming environment for homework assignments – see Canvas for details to install it. We will use a C compiler for programming assignments. Unless otherwise stated, all homework assignments 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.

zyBooks – We will also use zyBooks for practicing C programming in-class. You can follow 3 steps to subscribe, as described on Canvas.

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.

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. It is understood that a worksheet solution might be imperfect.

A worksheet submission is due approximately every week. Please refer to Canvas for detailed instructions and deadlines. The worksheet submission page on Canvas closes after it is due. You need to submit the worksheets by their closing time on the due date. A worksheet will not be re-opened after its closing date. Late worksheets will not be accepted. 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.

Homework assignments: Programming assignments will be assigned. More information will be given at the time of the first programming assignment. Late penalty is 2% per day up to 15 days; after 15 days no submission will be accepted and the submission page will be closed.

Students can work with a partner (in groups of two) on the homework. If two students form a group, the pair of both group members will get the same grade for any worksheet/assignment they submit together. *Only one group member should submit a worksheet/assignment to avoid double grading. If you work with a partner, you must put your group members' names in a spreadsheet that will be given. Also, for each worksheet or assignment submission partners must type both of their names in the Canvas submission comment box. 5% penalty if you forget to show both names in the submission comment box and the graders have to do additional work to fix the grade.* Students may leave a group, or work on their own if they prefer. Note it is optional to work with a partner.

While it is fine to discuss the worksheet/assignment solutions with your partner within your group, code solutions submitted on Canvas 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. Canvas automatically checks submissions for plagiarism from multiple online sources.* Oral examination might be requested.

All homework should be submitted on Canvas. Homework sent via an email or message will not be graded.

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 unless there are extraordinary circumstances, in which case they will require access to the internet, Canvas, Lockdown Browser (on Windows or macOS machine), and Respondus Monitor (web camera). The exams should be done individually. No make-up exams except in case of verifiable emergency circumstances.

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 you will receive 1% extra credit.

There may also be a bonus assignment worth 1%.

✔ Grading Information

Final Grade is based on:

50% Assignments

20% Midterms (10% each)

20% Final

9% Worksheets

1% Participation (Canvas discussion forum)

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	Topic
08/21	Introduction
08/23	Review C and the command line

08/28	Review C and the command line
08/30	Review C and the command line
09/06	Review C and the command line
09/11	Processes
09/13	Process API
09/18	Interprocess Communication, Sockets, Pipes

09/20	System calls with File I/O
09/25	**Midterm 1**
09/27	Signals
10/02	CPU Scheduling
10/04	Multilevel CPU Scheduling
10/09	Direct Execution
10/11	Address Space
10/16	Memory API
10/18	Free-Space Management
10/23	Paging
10/25	Swapping Policies
10/30	Thread API
11/01	Locks
11/06	Lock-based concurrent Data Structures
11/08	**Midterm 2**
11/13	Condition Variables and Semaphores

11/15	Concurrency Bugs
11/20	Advanced Locks
11/27	Hard Disks
11/29	Files and Directories
12/04	File System Implementations
12/06	Review, wrap-up
	Final exam on Thursday, December 14, 9:45am-12:00pm

The schedule is subject to change with fair notice.