



California State University, Sacramento
College of Engineering and Computer Science

**Computer Science 60:
Introduction to System Programming in UNIX.**

Fall 2019 Syllabus

Instructor and Contact Information

Sankar Srivatsa

E-Mail	sankarsrivatsa@gmail.com
Office	Monday: RVR 3024 , 4:30PM-5:20PM

Class Times:

	M-W	5:30pm - 6:45pm	Section 5
--	-----	--------------------	--------------

Final Exams:

See Schedule published by the university
--

(Those of you, who travel after the semester, **please** make your travel plans for **AFTER** all your finals are over.)

TEXTBOOK & COURSE MATERIALS

- **Required Textbook:** The Linux Programming Interface: A Linux and UNIX System Programming Handbook, by Michael Kerrisk.
- **Recommended Text:** by Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language (ANSI C), Second Edition, Pearson/Prentice Hall, 1988.

GENERAL: SOFTWARE & HARDWARE

Many of you will do your class or assignment at the ECS Lab computers, using an open-source free compiler in a UNIX environment. We will use the ECS computer "athena". We also will be able to log into campus from home to do class work.

COURSE DESCRIPTION

Features of the C language commonly used in systems programming, application to systems programming in a UNIX environment. Topics include C preprocessor macros, I/O, bit-manipulation facilities, timesharing system concepts, file permissions, shell script programming, make files and source code control, basic system calls like fork and exec, pointers and dynamic memory allocation, libraries and relocation and linking concepts including

assembler handling of symbol tables. Prior knowledge of a C like programming language is presumed.

Prerequisite: Required satisfactory completion of CSC 20, CSC 35 or their equivalents.

TOPIC OUTLINE/SCHEDULE

Brief List of Topics:

1. UNIX file system and Shell Script and Programming
2. C functions, variables, operators, and expressions
3. Pointers.
4. Standard IO and system-level IO, string handling
5. Dynamic memory management.
6. C preprocessor.
7. Make Utility.
8. Static and Dynamic libraries
9. System calls, error handling and recovery.
10. Debugger.
11. Source code control.
12. Signals.
13. Multiprogramming in UNIX, process creation and termination.
14. Pipes.
15. Semaphores.
16. Shared memory.
17. POSIX threads.

TOPICS COVERED

Students completing this course will be able to

1. Write well-structured, procedurally oriented programs in C,
2. Use a multi-user, time-sharing operating system -UNIX/Linux; and
3. Write systems programs in UNIX/Linux.
4. Understand principles of concurrency
5. Understand and apply synchronization mechanisms to the critical section problem

Course Requirements: Internet connection (DSL, LAN, or cable connection desirable)

ATTENDANCE

You must take the mid-term and the final at the scheduled time unless you have a good reason. If you cannot attend the mid-term or the final for any reason, you are expected to arrange with me **in advance**. You have to discuss your situation with me.

Class attendance will be considered in your grade. If you miss a class, you are responsible for knowing all the material that has been covered in class. In addition, you are responsible for knowing any important information announced in class; such information includes (but is not confined to) the date of the mid-term, what material will be covered on the mid-term and on the final exam. In short, absence from class is no excuse for failure to know any material or for not knowing when a mid-term is given. I will create an email list for this class where I will post test dates.

BASIS OF GRADE

The dates of the mid-terms are mentioned with an approximate date in this handout. This handout does not list the material to be covered on the mid-term and on the final exam. I will announce the mid-term's date in class at least a week in advance. The material to be covered on each mid-term and on the final will also be announced in class. I will not give incompletes under ordinary circumstances.

Your grade will **approximate** the following scheme.

Class Attendance, in-class and online quizzes	10%
Assignments Work	30%
Exams:	
Mid-Term-1	15%
Mid-term-2	20%
Final	25%

WRITING YOUR OWN PROGRAMS

Although you may engage in general discussions with other students, you are expected to write your own programs. The word "general" implies that the discussions are not at the level of detail of C statements. "Taking someone else's program, copying it, and putting your name on it" is considered cheating. Loaning the program makes you part of the deception.

RULES CONCERNING PROGRAMS

The first line of each program will be a Comment with your name.

The second line of each program will be a Comment with the assignment name.

(Example: Lab 2, or Program 3).

You will submit programs to Canvas, following the explicit directions given for each assignment.

Each program will be graded according to several criteria:

- Does the program produce the correct output?
- Were all required techniques used?
- Were all required comments included?

DROP INFORMATION

Last day – Please check university website -

<http://catalog.csus.edu/academic-calendar/>

It is **your** responsibility to ensure that your idea of your class schedule agrees with the on-line version.

If you just disappear in the middle of the semester without withdrawing properly, I am required to assign you a grade of "WU" which then converts to an "F" on your transcript. If you have major problems in your life, I may be able to work with you to finish this class, but only if you talk with me about arrangements.

INFORMATION ON ASSIGNMENTS & TESTS

Tests, the midterm and the final, will be open book and open note. Sharing with classmates will not be allowed during a test. No compilers will be allowed either. If you forget a detail during a test and want to look it up, that is rather like a real-life situation. If you expect to look it ALL up, I can tell you students before you have tried that, and they have gotten test grades of about 40%. There will not be enough time to learn it and/or look it all up during the test period.

You will not need bubble sheets or blue books for any test in this class. I will copy the tests for you and you may write your answers on the paper. Having extra pens, pencils, paper, and erasers are always a good idea for any test.

STUDENTS WITH DISABILITIES

If you have a disability and require accommodations, you need to provide disability documentation to SSWD (Services to Students with Disabilities), Lassen Hall 1008, (916) 278-6955, <http://www.csus.edu/sswd/>. Please discuss your accommodation needs with me after class or my office early in the semester.

THE UNIVERSITY READING AND WRITING CENTER

For free, one-on-one help with reading or writing in any class, visit the University Reading and Writing Center (URWC) in Calaveras 128. The URWC can help you at any stage in your reading and writing processes: coming up with a topic, developing and organizing a draft, understanding difficult texts, or developing strategies to become a better editor. Students can sign up for

half hour and hour sessions a week with a tutor by visiting the URWC in CLV 128—visit the URWC during week two of the semester to sign up for regular weekly tutoring. Students may also drop in at any time, but the best time to drop is during our “drop-in only” hours, Monday-Thursday 1:00pm-4:00pm. We also offer tutoring for one unit of academic credit through ENGL121. For more information, visit the website at <http://www.csus.edu/writingcenter>.

Tentative Schedule: Week	C Programming Language Topics	Unix Topics
1	C Types, operators & expressions (KR Chapter 2)	Introduction: History (LPI Chapter 1), Introduction: Login, Basic commands
2	Control structures , The C preprocessor, Program structure (KR Chapters 3, 4)	Basic commands (Cont), Programming Tools: Editor, Compiler, Debugger, Source Code Control, Make Utilities
3	Basic I/O, functions (KR Chapters 4, 7)	I/O (LPI Chapters 4, 5), Redirection
4	Arrays, pointers, structures (KR Chapter 5)	Shared library (LPI Chapters 40, 41)
6	Advanced I/O	Unix file system
7	Function pointers (KR Chapter 5)	Unix shell programming
8	Systems programming (LPI Chapter 3)	
9 Midterm subject to change of week →	*****Midterm Exam *****	*****Midterm Exam *****
10	Processes & Multiprogramming (LPI Chapters 6, 24, 25, 26)	
11	Forks, pipes (LPI Chapters 20, 21, 22)	
12	Semaphores, shared memory (LPI Chapters 45, 46, 47, 48)	
13	Threads , Sockets	
14	Signals	
15	Review	Review
16	***** Final Exam *****	***** Final Exam *****