

CPS 444/544: Systems Programming I/Fall 2006

CPS 444/544 (3 sem hrs) provides an introduction to systems programming in UNIX and C. Topics include operating system structures, system calls and libraries, interprocess communication, pipes, and signals. Students can also expect a survey of various software tools supporting systems programming, including gcc, gdb, make, sed and awk, and lex and yacc. The course does not aim to be comprehensive, but rather focuses on thematic issues. Assignments are designed to provide students with a pragmatic exposure to these tools as well as issues faced by modern practitioners. CPS 444/544 is a programming-intensive course and assumes no prior experience with UNIX.

- Pre-requisite:** CPS 350 (Data Structures and Algorithms) and CPS 346 (Operating Systems) with a minimum grade of C in both for students enrolled in CPS 544.
- Meeting times:** T Th 4:30pm–5:45pm, MH 205
- Instructor:** Dr. Saverio Perugini, 229–4079, AN 145, perugisa@udayton.edu
- Office hours:** M T W Th 5:45pm–6:45pm and by appointment.
- Teaching assistant:** John Cresencia, 229–2364, AN 131/5, cresenjv@notes.udayton.edu
- TA office hours:** T Th 12:00pm–2:00pm, F 12:00pm–1:00pm, and by appointment.
- Required textbook:** *The UNIX Programming Environment* by B.W. Kernighan and R. Pike. Prentice Hall, Second edition, 1984. ISBN: 0-13-937681-X.
- Recmd'd textbook:** *The C Programming Language* by B.W. Kernighan and D.M. Ritchie. Prentice Hall, Second edition, 1988. ISBN: 0-13-110362-8.
- Final exam:** 12/12, 4:30pm–6:20pm, MH 205
- Course webpage:** <http://academic.udayton.edu/SaverioPerugini/courses/Fall2006/cps444/>

Objectives:

- Develop a proficiency in UNIX and C as a systems programming language/environment.
- Survey various system-oriented software tools, including debuggers, and compilation and configuration managers.
- Establish an understanding of the design and development of systems software, such as command interpreters and compilers, through the study of pattern matching and filters, inter-process communication, system libraries, signals, and automatic program generation.
- Explore UNIX internals and establish an understanding of UNIX system calls.

Evaluation:

Component	Qty	Pts per	Tot pts
Homeworks (lowest dropped)	8	71	497
CPS 544 Project	1	200	200
Exams (9/14, 10/17, 11/14)	3	100	300
Final exam (comprehensive)	1	203	203
Total (CPS 444):			1,000
Total (CPS 544):			1,200

Each homework requires a fair amount of critical thought and design, and approximately 100–500 lines of code. Assignments are due at 4:30pm in class. Any assignment not received at 4:30pm in

class or earlier will be assessed a 10% penalty. No assignment will be accepted after 6:45pm on its due date. All exams are in-class, closed-book, and closed-notes. Attendance is mandatory at all examinations; make-ups will not be given. Any missed exam will result in a zero. Make no assumptions about anything; always consult the instructor first. Final letter grades of A, A-, B+, B, B-, C+, C, C-, and D start at 93, 90, 87, 83, 80, 77, 73, 70, and 60, respectively.

Workload: CPS 444/544 is a challenging course and moves at a very fast pace. Spending a minimum of 9 hours outside of class each week programming is required. I advise you to see me to discuss any problems you may have before you are evaluated.

Classroom policies: Students are expected to conduct themselves in a professional manner. Keep cell phones or similar devices in a silent mode during class. The use of laptop computers or similar devices is not permitted in class.

Ethics: To achieve the course objectives, assignments must be a sole result of your individual work and must not be shared with classmates. Evidence indicating a violation of this policy will be handled according to the student handbook (Academic Information, pp. 2–4) and result in a doubly-weighted zero which will not be dropped. Make no assumptions about this policy; always consult the instructor first. No student should ever feel that they must resort to academic dishonesty. Please see me if you are struggling with the course or an assignment. No grade is worth your integrity. Honesty in your academic work will develop into professional integrity. The faculty and students of the University of Dayton will not tolerate any form of academic dishonesty.

Course outline (subject to minor shifts):

Date	Topic(s)
	<i>Introduction to Systems Programming in UNIX and C</i>
8/22	Introduction and course outline
8/24	System libraries; input/output
8/29	Files and directories
8/31	Files and directories (inodes, hard and symbolic links)
9/ 5	Using the shell
9/ 7	Using the shell (<i>cont'd</i>)
	<i>Pattern Matching and Filters</i>
9/12	Regular expressions and grep
9/19	Extended regular expressions and egrep
9/21	Filters, interprocess communication, and sed
9/26	awk
	<i>Shell Programming</i>
9/28	Essential elements
10/ 3	Conditionals
10/ 5	Repetition
10/12	Putting it all together
	<i>Automatic Program Generation</i>
10/19	Makefiles and RCS
10/24	Finite state automata and lex
10/26	Backus-Naur form and yacc
10/31	lex and yacc
	<i>UNIX System Calls</i>
11/ 2	I/O (read and write)
11/ 2	Processes (fork)
11/ 9	Processes (wait and exec)
11/16	Special files (pipes and FIFOs)
11/21	Token ring of processes
11/28	Signals
12/ 5	Signals (<i>cont'd</i>)