

[CS 25200: Systems Programming](#)

Spring 2018

---

**Class:**

LE1: MWF 12:30pm – 1:20pm, Room MATH 175

LE2: MWF 4:30pm – 5:20pm, Room MATH 175

**Course Web Page:**

<https://www.cs.purdue.edu/homes/cs252/>

**Course Newsgroup:**

<https://piazza.com/purdue/spring2018/cs252>

**Instructors:**

Prof. Gustavo Rodriguez-Rivera, [grr@cs.purdue.edu](mailto:grr@cs.purdue.edu), LWSN 1210

Prof. Jeff Turkstra, [jeff@purdue.edu](mailto:jeff@purdue.edu), HAAS 128, 49-63088.

**Office Hours:**

Prof. Rodriguez-Rivera: MWF 10:00am – 11:00am

Prof. Turkstra: T 10:00am – 12:00pm, Th 2:00pm – 4:00pm

**Teaching Assistants:**

This course has eight graduate teaching assistants as well as a number of undergraduate teaching assistants. The names and email addresses for the GTAs are given below.

---

Evan M. Hanau [ehanau@purdue.edu](mailto:ehanau@purdue.edu)

Brian P. Hays [hays1@purdue.edu](mailto:hays1@purdue.edu)

Caluadewa Dharshaka T. Matthew [mathewc@purdue.edu](mailto:mathewc@purdue.edu)

Vedant Mishra [mishra63@purdue.edu](mailto:mishra63@purdue.edu)

Elkindi Rezig [erezig@purdue.edu](mailto:erezig@purdue.edu)

Guannan Wei [wei220@purdue.edu](mailto:wei220@purdue.edu)

Xilun Wu [wu636@purdue.edu](mailto:wu636@purdue.edu)

Sam K. Yap [syap@purdue.edu](mailto:syap@purdue.edu)

---

**Text:**

Book Online: Introduction to Systems Programming; Rodriguez-Rivera and Ennen; 2014  
<https://www.cs.purdue.edu/homes/grr/SystemsProgrammingBook/>

*Recommended*

Advanced Programming in the UNIX Environment; Stevens and Rago; Addison-Wesley Professional, 2013  
 ISBN-13: 978-0321637734

**Prerequisites:**

Computer Architecture – CS 25000  
 Programming in C – CS 24000  
 Foundations of Computer Science – CS 18200  
 Problem Solving and Object-Oriented Programming – CS 18000  
 Data Structures and Algorithms – CS 25100 (Corequisite)  
 Programming proficiency is *absolutely* required

**Course Outcomes:**

A student who successfully fulfills the course requirements will:

1. have consolidated their programming skills from previous core courses
2. understand how programs run in userland and interact with the OS
3. be able to write “large” programs with over 1,000 lines of code
4. be able to use tools---including IDEs, debuggers, profilers, and revision control---to create quality, maintainable code
5. learn how to work in teams
6. be able to use and write scripts
7. learn how to write multi-process and multi-threaded programs

**Class Attendance:**

You are expected to attend all classes. Attendance will be recorded for randomly selected class sessions. If you choose to attend class, please arrive in the classroom on time. You are expected to be quiet in class. If you must miss a class, you are responsible for procuring any material, information, handouts, announcements, etc., that you missed.

**Preparation for Lectures:**

You should try to read over the relevant pages in the course text before arrival. Additionally, you are expected to check your email and the course website regularly. Here is the *tentative* lecture schedule:

<b>Wk</b>	<b>Lec</b>	<b>Subject</b>
1	1	Introduction, Program Structure
	2	Review of Pointers in C
	3	Memory Allocation
2	X	Martin Luther King, Jr. Day (No Class)
	4	More Memory Allocation (malloc())
	5	Memory Leaks and Bugs
3	6	Debugging with gdb, Introduction to OS
	7	File Systems
	8	File Systems and Shell Environments
4	9	I/O Redirection and Pipes, Basic UNIX Commands

	10	Basic UNIX Commands cont.
	11	Scripting Languages and Scripts
5	12	Shell Internals
	13	Shells cont.
	14	Shells, Introduction to System Calls
6	15	Shell Implementation
	16	Shell Implementation cont.
	17	Computer Architecture Review, Kernel vs. User Mode
7	18	Interrupts
	19	System Call Internals
	20	Midterm Exam Review
8	21	More System Calls
	X	<b>MIDTERM EXAM</b>
	22	Scheduling and Threads
9	23	Programming with Threads
	24	Concurrency and Synchronization
	25	Concurrency and Synchronization
10	X	Spring Break Week
11	26	Concurrency and Synchronization
	27	HTTP Protocol and Server
	28	More HTTP (Concurrency)
12	29	Bounded Buffer Problem
	30	Types of Computer Systems
	31	Networking
13	32	Networking cont.
	33	Networking cont.
	34	Networking cont.
14	35	Sockets Programming
	36	Databases
	37	Software Development Models
15	38	Revision Control
	39	Software Patterns
	40	Program Optimization

### Quizzes:

There will be a number of unannounced, 5 to 10 minute quizzes in class. A score of zero will be given in case of absence.

### Projects:

There will be a number of intensive projects that will be assigned throughout the semester. The goal of these assignments is to reinforce material covered in lecture as well as to provide practical experience through the exploration of low-level implementation details. Projects will be introduced and explained during your scheduled lab session. Unless otherwise specified, you should work alone. For team-based projects, you may collaborate and work closely *only* with your teammates. The standard academic honesty policies apply to any inter-team communication and sharing.

**Labs:**

Attendance is mandatory and counts toward the 10% attendance portion of your final grade. Projects will be introduced and explained during your scheduled lab session. This is also a time to ask questions and receive help.

**Examinations:**

Exams will be closed book and closed notes. You must solve the exam problems yourself, without any help (knowing or unknowing) from any other student. You must not seek any knowledge in advance of the test questions (beyond that given in class) and must report any advance knowledge of the test questions by any student that you are aware of. You must not allow any other student access to your solutions during the exam. If the seating situation makes this difficult, please inform the instructor or TAs.

Midterm Exam: Wednesday, February 28 8:00pm

Final: TBD

**Regrades:**

Problems regarding grading of assignments and the exam must be resolved within **one week** after the graded work has been returned to you. It is your responsibility to pick up the graded work on time. Grades will not be modified after the one week period.

**Late Submissions:**

Project solutions that are submitted after the specified deadline will incur a 10% reduction in score per day late.

**Make-up Examination Policy:**

Make-up exams will be given only in the **most extreme** circumstances and require certification for such circumstances. Eg, a medical doctor's statement certifying that the student is **unable** to attend the scheduled exam. Any travel (including interview trips), load from work or from other classes, failed alarm clocks, or simply not being able to make it to the exam will **not** be grounds for a make-up. If you have any recurring medical problems that may unexpectedly prevent you from making it to class or exams, please obtain a doctor's statement certifying your circumstance.

**Academic Integrity:**

As a student at Purdue you are subject to the [Purdue University Student Code of Conduct](#), which enjoins you to respect the highest standards of honesty and integrity. All work that you submit in this course must be your own; unauthorized group efforts are considered academic dishonesty. See the online brochure [Academic Integrity: A Guide for Students](#) for definitions and sanctions. Academic dishonesty is a serious offense which may result in suspension or expulsion from the University. In addition to any other action taken, such as suspension or expulsion, a **grade of F** will normally be recorded on the transcripts of students found responsible for acts of academic dishonesty. Students are encouraged to report academic dishonesty to the instructor directly, or to the Office of the Dean of Students.

You may discuss assignments in a general way with other students, but you may not consult anyone else's work. Among other ways to get an F, you are guilty of academic dishonesty if:

- You examine another student's solution to an assignment
- You allow another student to examine your solution to an assignment
- You fail to take reasonable care to prevent another student from examining your solution to an assignment and that student does examine your solution. For example, if you allow another student to check his/her email from your terminal while you step out of the room, you have failed to take reasonable care to prevent him/her from accessing your files.
- You submit an assignment that is not completely your own work
- You share results or notes during quizzes or exams

All work is subject to computer-based comparison and analysis. Do not con yourself into thinking that you can hide any collaboration. The risk of getting caught is too high, and the standard penalty is way too high.

If we find reason to believe that a student or team has cheated on any assignment, we may inform the student or team promptly, or we may decide to silently accumulate evidence against the student or team on later assignments.

**Grading:**

Final grades will be assigned according to the following *approximate* weighting:

Projects – 40%

Exams – 50%

Attendance – 10%

**Questions and Answers:**

Questions of general interest should be posted on the course piazza site. Answers will be posted as soon as possible. Project questions should be directed to the appropriate project coordinator via email. Answers will be sent to you directly. If you need to contact a specific TA or instructor, send email to that individual or go see him/her during office hours.

**Modifications:**

This syllabus may be modified at any time with notification.

*\*\*\* As an interesting side note, a significant portion of this syllabus is copied from Dr. Rodriguez-Rivera's, Dr. Dunsmore's, Dr. Hosking's, Dr. Brylow's, and Dr. Hu's policy pages from previous semesters. One of the major differences between plagiarism and proper reuse is giving credit where credit is due. \*\*\**