

Course Syllabus

- Classes
- Contact Info
- Texts
- Grading
- Letter Grade Assignment
- Academic Honesty
- Course Web Site
- Course Mailing List
- Course Setup
- Aims of Course
- Main Topics
- Main Topics Continued

Classes

Classroom

EB R15

Times

MW 7:30 - 9:00p. Extended to 9:30p if any classes missed. 2 - 3 minute break approximately halfway through each class.

Contact Info

Instructor

Zerksis D. Umrigar
Office: EB P14
Office Hours: Mon, Wed: 6:25 - 7:25p
Phone: (607) 777-4316 (during office hours)
Email: umrigar@binghamton.edu

Out of town on Thurs and Fri, but will read email.

Teaching Assistant

Adriano Garcia
Office: EB P17
Office Hours: Mon, Wed: 3:00 - 4:00p
Email: agarcia3@binghamton.edu

Texts

Michael Kerrisk, *The Linux Programming Interface*, No Starch Press, 2010. Required.

Samuel P. Harbison, Guy L. Steele, *C: A Reference Manual*, 5th Edition, Prentice Hall, 2002.
Recommended.

Grading

Pop Quizzes (see below):	15%
Programming projects (lowest dropped)	30%
Homeworks (lowest dropped)	20% (551), 25% (451)
Paper	5% (551 only)
Midterm:	15%
Final:	15%

- If there are n pop quizzes, then $\text{floor}(n/3)$ will be dropped.
- Pop quizzes will be closed-book. Midterm and final will be open-book, open-notes (but no electronic devices) unless specified otherwise.
- Assignments which are late upto 1 week will be accepted with a 15% penalty. Assignments which are later than 1 week will not be accepted.
- Flexible under exceptional circumstances.

Letter Grade Assignment

- Letter grades assigned strictly monotonically based on numeric course grade.
- A letter grade of A will be given only for consistent superior work.
- It should be relatively easy to get a grade around a B.
- You will get an F only if you miss turning in a lot of work or submit consistently very poor quality work or if you cheat.
- TA Grading Guidelines are available.

Academic Honesty

Cheating of any type will be penalized heavily.

- Minimal penalty: an F letter grade for entire course.
- Permissible to collaborate to understand course material, homework questions or project assignments. Not permissible to discuss solutions.
 - If you feel you may have inadvertently crossed the line, then let us know; will not be considered cheating.
 - If submitting an assignment late after solution has been posted, you should obviously not be looking at the solution.
- All registered students must sign and complete an Academic Honesty Statement.

Course Web Site

- All course material on course web site at <http://zdu.binghamton.edu/cs551s16>.
- Course web site mirrored at <http://cs.binghamton.edu/~umrigar/cs551s16>.
- Slides usually available by 6:30p before class.
- Course web site available via `git` repository at `ssh://user@remote.cs.binghamton.edu/~umrigar/cs551s16.git`.
Useful for tracking changes.

Course Mailing List

- All students registered for course should have been subscribed to the CS551 mailing list.
- To change the email address via which you are subscribed to the list or would like to edit your subscription options, please visit <https://www.cs.binghamton.edu/mailman/listinfo/cs551>.

Course Setup

- Will cover most of textbook (limited Linux-specific material).
- Pop quizzes will test on material covered recently.
- 5 projects all solving roughly the same problem using different techniques (the techniques used by a particular project may be contorted to fit the problem).
- 4-5 homeworks.
- Midterm and final (last during exam week).
- Paper required only for CS 551 students.

Aims of Course

- To apply theoretical concepts from other courses in a practical setting.
- To get a good understanding of programming to the API of a modern operating system.
- To develop expert proficiency in C-like languages.
- Largely an applied course; not particularly academic.

Main Topics

Concentration on Unix systems-programming API;
minimal coverage of kernel internals.

- Advanced C programming.
- Programming with C standard library.
- Unix systems introduction.
- File I/O.
- Processes and threads.

Main Topics Continued

- Inter-process communication.
- Network programming introduction.
- Signals.
- Concurrent programming.

Will jump around somewhat as projects are assigned.