

CS 241 Honors Class Introduction

CS 296-41 Course Staff

University of Illinois Urbana-Champaign

September 6, 2017

The Goals

- To learn about systems programming at a deeper level than in CS 241 through occasional lectures about various systems topics throughout the semester
- To explore systems programming by working on a semester-long project of your choice

About This Course

Lectures

- Wednesday, 7-7:50pm, 1304 Siebel
- We'll average around one lecture every other week
- Full lecture schedule will be available on the honors course website and on piazza
- There is an attendance grade, but we will drop 1 lecture attendance without question
- If you must miss another lecture, inform course staff and we will determine the validity of the excuse

Projects

- Teams consist of 2-3 people (see us for anything else)
- Each team is assigned a mentor from the course staff
- Every team will meet with their mentor once a week
- Must be technically rigorous
- Must relate to systems programming
- Games are highly discouraged

Grading

- 15% – Lecture attendance
- 85% – Semester project
 - 5% – Project Proposal
 - 25% – Weekly check-ins with mentor
 - 35% – Final deliverables (code, project website, etc.)
 - 20% – Final presentation

The minimum grade thresholds for this class are the same as CS 241

Course Staff

Name	NetID	Interests
Aneesh Durg	durg2	Functional Programming, TextUls
Kevin Hong	khong18	Concurrency, IPC, Statistics, Machine Learning
Ben Kurtovic	kurtovc2	Operating Systems, Networking, Security
Yasha Mostofi	mostofi2	Go, Algorithms
Jonathan Wexler	jwexler2	Parallelism, Networking
Brian Zhou	bwzhou2	Full-stack development, Network- ing

If there is someone you are interested in working with, let us know!

Project Ideas

What is systems programming?

The big topics in systems programming include...

- Process Control
- Memory Management
- Parallelism
- Networking
- Security
- Filesystems

Project Ideas

Process Control

- Managing the execution of programs on one (or multiple) machines
- Past Projects: Raspberry Pi Compute Cluster, Distributed Server Manager, earliest-deadline-first process scheduler

Memory Management

- Managing/Abstracting process use of memory and resources
- Past Projects: Custom C Garbage Collector

Parallelism

- Using multiple processors to work together on a problem
- Past Projects: Chess Engine, GPU Based Fluid Simulator

Project Ideas

Networking

- Having remote computers talk to each other
- Past Projects: Remote Desktop, multi-user real-time text editor

Security

- Keeping undesired entities from accessing protected resources
- Past Projects: Custom Network Encryption tool, Build a Proxy Server, Steganography Toolset

Filesystems

- Organizing bits and bytes into readable data
- Past Projects: Distributed FUZE Filesystem

Brainstorming Time!

Some Other Stuff

A couple other things about your projects...

Writing Readable Code

- When you write solutions for an MP, the only other entity looking at your code is a machine
- Now, you are writing code that *other people* (including those grading your project) have to read!
- Low-Level languages like C can get especially hard to comprehend when written badly
- Some general guidelines:
 - Name your variables and functions so that it's obvious what they're used for
 - If someone else can't tell what your code is doing at a glance, write comments to explain it
 - Code in whatever style you want, but make sure your style is consistent
- Keep in mind that your teammates are not mind-readers; they cannot guess what you were thinking when you were writing your code

Team Communication

- If you are working on a team, you will likely need to meet as a team (outside of your mentoring meetings) to determine who is doing what
- Consider managing units of work on your project using the issue tracker on GitHub
 - Each logical unit of work should have its own ticket
 - Using an issue tracker allows you to keep track of any conversation pertaining to a particular feature
 - Assigning team members individual tickets can help organize collaboration
- If two members are developing features that interact with one another, make sure you are coordinating your work so that your features can work together properly

- These projects are long-term and fundamentally different from most other assignments you've done
- It's very easy to fall behind, so make sure you're making progress every week
- Suggestion: make deadlines for yourself over the course of the semester! (or ask your mentor for deadline suggestions)
- It's much easier to make a few hours of progress every week instead of trying to make 40+ hours of progress in a single week

Next Week

- Next week we will be having a guest lecture by Jeff Challen
- He will be discussing GoLang, another systems programming language developed by Google
- This is not an optional lecture