CS111, Lecture 1
Welcome to CS111!

reading:

Course Syllabus

Honor Code and Collaboration Page

😷 masks required

Plan For Today

- Introduction
- CS111 Course Topics
- CS111 Course Policies
- Filesystems Introduction

Asking Questions

- Feel free to raise your hand at any time with a question

- If you are more comfortable, you can post a question in the Ed forum thread for each day's lecture (optionally anonymously)

- We will monitor the thread throughout the lecture for questions

Visit Ed (or access via Canvas):
https://edstem.org/us/courses/29668/discussion/

Today's thread:
https://edstem.org/us/courses/29668/discussion/1825527

Guiding Principles For In-Person Class

- We are likely not fully recovered or restored from the stresses of the past 36 months and now facing new uncertainties, responsibilities, and emotions.

- We will do everything we can to support you. We have designed the course to the best of our ability to provide flexibility.

- We will constantly evaluate and listen to ensure the class is going as smoothly as possible for everyone.

- Please communicate with us if any personal circumstances or issues arise! We are here to support you.

Guiding Principles For In-Person Class

- Stanford University is mandating the use of masks in classrooms for everyone, regardless of vaccination status. Please keep your mask on while in the lecture hall, including when asking questions.

- Some of us have health conditions precluding our ability to wear masks. Students in this situation should work with the **Office of Accessible Education**.

- Some of us might feel more comfortable wearing masks/social distancing even when not required. All of our preferences are reasonable, and it is important that we treat each others' preferences with respect and care.

Plan For Today

- **Introduction**
- CS111 Course Topics
- CS111 Course Policies
- Filesystems Introduction

Teaching Team

Nick Troccoli

Bharat Khandelwal

Briana Berger

Parthiv Krishna
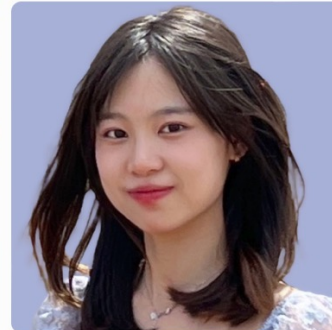
Pratyush Agarwal

Sumer Kohli

Keegan Mehall

Michela Marchini

Nikhil Raghuraman

Swayam Parida

Xiyu Zhang

About Nick Troccoli (troccoli@stanford.edu):

- Lecturer in CS, taught CS106X, CS107, CS110
- Stanford BS/MS (coterm) in CS
- Systems track undergrad, AI track grad

Companion Class: CS111A

- **CS111A** is an extra 1-unit "Pathfinders" or "ACE" section for undergraduates with additional course support, practice and instruction.

- Meets for an additional weekly section and has additional review sessions

- Section Tues/Thurs 10:30-11:50AM in 160-124

- Entry by application

- see the course homepage for more details and the link to apply: **cs111.stanford.edu**

Trip Master

Enroll in CS111A!

- The ACE program seeks to provide strong supplemental support in technical classes, particularly for students from under resourced and/or minoritized backgrounds.

- ACE is a <u>great</u> way to get extra practice and build a strong conceptual foundation in CS111.

- Applications for CS111A are still open! If you apply, please come to our first class **Tuesday 9/27 (tomorrow!)** at 10:30AM in 160-124. Before our Thursday class, I will go through applications and begin to send out access codes to accepted students. The final deadline for applications is Friday 9/30 at 5pm!

Trip Master

Course Website

cs111.stanford.edu

*lecture videos on Canvas

Plan For Today

- Introduction
- **CS111 Course Topics**
- CS111 Course Policies
- Filesystems Introduction

What is CS111?

CS107 built up and expanded your breadth and depth of programming experience and techniques and showed you how machines really work.

CS111 leverages this programming experience to introduce operating systems and how they work.

What is an operating system?

What is an Operating System?

An operating system ("OS") is software that allows people to run programs on a computer.

- Examples: iOS, Android, Windows, macOS, Linux

You may think mostly of the *user interface* of the operating system, but an operating system does so much more!

What is an Operating System?

The operating system sits between the hardware and user programs. It manages shared resources and provides functionality for programs to run.

It manages things like:

- Processor (CPU): decides what program gets to do work and for how long
- Memory (RAM): decides what programs get to use what areas of memory
- Hard Drive: decides how the disk is used to store files

User Programs
Operating System
Hardware (memory, hard drive, processor, etc.)

Key Operating System Responsibilities

- **Concurrency**: allows several different tasks to share processors
- **Memory (RAM)**: allows system memory be shared among several tasks
- **Files:** allows many files, for many different users, to share space on disk
- **I/O devices:** allows many devices to operate concurrently and be shared
- **Networks:** allows groups of computers to work together
- **Security:** allows interactions while protecting participants from each other

User Programs
Operating System
Hardware (memory, hard drive, processor, etc.)

What is an Operating System?

- So far, when you've written programs, you haven't had to think about any of this. That's the point! The OS is doing its job – it abstracts away complexity from programs.
  - Don't have to coordinate with other programs for who gets to use what memory
  - Don't have to coordinate with other programs for who gets to run when
- OSes work behind the scenes, but are extremely powerful
  - **Example:** devices with 1 CPU core (common through early 2000s) could really only execute 1 program at a time! OSes switch *very* quickly between different tasks to simulate appearance of multitasking.
  - **Example:** how can every program think it can use every address from NULL to 0xfff…? OSes tell programs *fake ("virtual") addresses* and behind the scenes it maps them to the actual ("physical") addresses that it organizes itself.

What is an Operating System?

- We can directly leverage operating system functionality in our programs.
- **System calls** are functions the operating system provides that we can call in our code. For example, in Linux:
  - the **open()** function lets us open a file on disk
  - the **fork()** function lets us spawn a new program (!)

What is CS111?

In CS111 we are going to explore both "sides" of operating systems:

- We'll learn what functionality is exported by operating systems to make the programs that we write more powerful.

- We'll learn how the operating system provides that functionality and how it acts as an **interface** to the computer hardware.

User Programs

Operating System

Hardware (memory, hard drive, processor, etc.)

Course Overview

1. **Filesystems** - *How can we design filesystems to manage files on disk, and what are the tradeoffs inherent in designing them? How can we interact with the filesystem in our programs?*

2. **Multiprocessing** - *How can our program create and interact with other programs? How does the operating system manage processes?*

3. **Multithreading** - *How can we have concurrency within a single process? How does the operating system manage concurrency?*

4. **Virtual Memory** - *How can one set of memory be shared among several processes? How can the operating system manage access to a limited amount of system memory?*

5. **Additional Topics:** Virtual Machines and Linkers

Course Overview

Why is it useful to know about operating systems?

- Understanding computing at this level demystifies how these seemingly-complex systems work and can aid future projects you work on.

- OSes contain many examples of elegant ideas in computing (concurrency, virtualization) that apply well beyond OSes, and pull together ideas like data structures, algorithms, languages, etc.

- We can learn how we can maximally take advantage of the hardware and operating system software available to us in our programs.

- Operating Systems are constantly evolving and encountering new applications (e.g., large datacenters) and new challenges

Operating Systems - A Brief History

- Initially, computers were just one user a time, working directly at a computer. "OSes" were just I/O libraries shared by users for convenience and efficiency.

- Over time, computers became more *shared* (jobs were batched, users didn't need to be in front of the machine) and *concurrent* (e.g., reading in next job while a job is running), and OSes took on more responsibility.

- Later computers supported *timesharing* (a "terminal" was a screen and keyboard plugged into a machine, and there could be multiple terminals for multiple users). This introduced complexities with filesystems and systems getting bogged down with concurrent users.

- Now, we have personal computers – OSes manage concurrency, filesystems, networking, and more

- OSes popping up in even more places – very small devices, and very large (datacenters / cloud)

CS111 vs. CS110

- CS111 focuses more specifically on operating systems and how they work
- Topics like filesystems, multiprocessing, and multithreading are similar, but covered in slightly different ways, along with some new assignments
- New topics like Virtual Memory and Virtual Machines
- CS111 is a relatively new class, and we're continuously working to make it the best it can be. We appreciate any and all feedback!

Plan For Today

- Introduction
- CS111 Course Topics
- **CS111 Course Policies**
- Filesystems Introduction

Prerequisites

The prerequisite for CS111 is CS107 (or equivalent).   You want to have:

- practical C/C++ skills and be able to write programs with complex use of memory and pointers and dynamic memory allocation (malloc/realloc/free/new/delete).

- an understanding of C++ classes, methods, and be able to work with appropriate data structures (arrays, maps, etc.) and standard algorithms (searching, sorting, hashing).

- familiarity with programming in a Unix/Linux environment using tools such as make, gcc/g++, valgrind, gdb, etc. and have a working understanding of the basics of computer architecture (x86-64 from 107, or other)

Course Syllabus and Calendar

cs111.stanford.edu/syllabus

cs111.stanford.edu/calendar

Getting Started Guide

cs111.stanford.edu/getting-started.html

(on course website under "handouts")

Textbook(s)

- There's no required textbook for the course, but if you're looking for additional reading, we recommend **Operating Systems: Principles and Practice (2nd Edition)** by Thomas Anderson and Michael Dahlin

- C and C++ references will also come in handy throughout the quarter; use the manual pages, visit cppreference.com or cplusplus.com as needed, etc.

Grading

***** 55% Assignments

* 5% Section Participation

* 5% Lecture Participation

** 15% Midterm Exam

** 20% Final Exam

Grading

****** 55% Assignments

* 5% Section Participation

* 5% Lecture Participation

** 15% Midterm Exam

** 20% Final Exam

Assignments

- ~6 programming assignments in C/C++ completed individually using **Unix command line tools**
  - Free software, pre-installed on Myth machines / available on course website
  - We will give out starter projects for each assignment
- Graded on **functionality** (behavior)
  - Functionality graded using *automated tools,* given as point score
  - May be some additional manual review for functionality issues for certain assignments.
  - **Assignments are not graded/reviewed for style**
  - Grades returned via course website

Assignment Late Policy

- **Start out with 5 "free late days"**: each late day allows you to submit an assignment up to 24 additional hours late without penalty. (No late days permitted for the last assignment)

- **Hard deadline 48 hours** after original due date

- Penalty per day after late days are exhausted (1 day: 80% cap; 2 days: 60% cap)

- Late days are "pre-granted extensions" – additional extensions for exceptional circumstances must be approved by the **instructor**. Please communicate with us! We are here to accommodate you as much as possible.

Question Break!

What questions do you have about the overall course goals, textbook or assignments?

Grading

***** 55% Assignments

* 5% Section Participation

* 5% Lecture Participation

** 15% Midterm Exam

** 20% Final Exam

Weekly Sections

- Weekly 50-minute in-person sections led by a CA, starting *next* week, offered on Wednesdays, Thursdays and Fridays.

- Hands-on practice in small groups with lecture material and course concepts. Great preview of homework!

- Graded on attendance + participation

- Section preference submissions open **Tuesday 9/27 at 5PM PST** and **are not first-come first-serve**. You may submit your preferences anytime until **Saturday 10/1 at 5PM PST**. Sign up on the course website.

Grading

***** 55%   Assignments

* 5%   Section Participation

* **5%   Lecture Participation**

** 15%   Midterm Exam

** 20%   Final Exam

Lecture Recordings

- Because CS111 is not on SCPD (for professional development students) this quarter, the course is not officially recorded.

- However, we have recording equipment provided in the lecture hall and will do our best to record each lecture ourselves. There may be quality issues and some content may be lost, but we'll do our best!

- See the calendar page (or lecture dropdown) on the course website for slides and lecture code

Lecture Participation

- At the same time, lecture is an important part of the course and we've found it most effective when students participate in person.

- In each lecture (starting Fri), we'll use Poll Everywhere to take polls and do practice questions. If you submit answers to all questions in a lecture (regardless of correctness), you get credit for that lecture.

- We will provide **5 pre-excused absences** for when you are ill, in COVID-19 isolation, or have other extenuating circumstances. They are intended only for these scenarios!

- Further excused absences are granted by the **instructor** only in cases where you have already used your 5 excused absences for extenuating circumstances and further extenuating circumstances necessitate additional accommodations.

Lecture Feedback

- A small part of the lecture participation grade also comes from a very short course feedback survey that we ask you to fill out twice over the course of the quarter (takes only a few minutes).

- An automated system will email you at the start of two different weeks during the quarter asking for this short feedback to be submitted sometime by Friday of that week.

- Each student will only receive 2 requests during the quarter.

- Starting week 2

Grading

***** 55%   Assignments

*   5%   Section Participation

*   5%   Lecture Participation

**   15%   Midterm Exam

**   20%   Final Exam

Exams

- **Midterm exam** – Tuesday, November 1, 7-9PM outside of class
  - Contact the course staff by 11:59PM on Wednesday, October 26 if you have an academic or University conflict with this time, and absolutely cannot make the regularly scheduled midterm
- **Final exam** – Friday, December 16, 8:30AM-11:30AM
  - No alternate final! You **MUST** be able to take the final exam at the scheduled time (except for university athletics or OAE accommodations or other last-minute emergencies)

- Both exams are closed-book, closed-notes, but you may bring in 1 double-sided page of notes. You will also be provided with a syntax reference sheet.
- Exams are administered electronically

Grading

***** 55% Assignments

* 5% Section Participation

* 5% Lecture Participation

** 15% Midterm Exam

** 20% Final Exam

Read our full course policies document:
https://cs111.stanford.edu/syllabus.html

Getting Help

- Post on the **Discussion Forum**
  - Online discussion forum for students; post questions, answer other students' questions
  - Best for course material discussions, course policy questions, short debugging questions or general assignment questions (DON'T POST ASSIGNMENT CODE!)
- Visit **Helper Hours**
  - Sign up in a queue for 1:1 TA help; schedule will be posted on course website tomorrow.
  - Mix of in-person-only and online-only helper hours
  - Except for assign1 if needed, course staff cannot look at assignment code
  - Best for **group work**, **in-depth code questions (with TAs only!)** or **longer course material discussions**
- **Email** the Course Staff
  - Email instructor for **private matters** (e.g. OAE accommodations, extension requests, other personal matters).
  - Email your grader for grading questions about a particular assignment

Question Break!

What questions do you have about section, lecture or exams?

OAE Accommodations

- Please email the instructor as soon as possible with any accommodations you may need for the course.
- We are eager to do everything we can to support you and make you successful in CS111!

Course Flexibility

If you are ever sick or encounter an emergency or other exceptional circumstance, we have a variety of accommodation mechanisms, including:

- Assignment late days
- Makeup sections or excused absences
- Lecture excused absences
- Exam accommodations for emergencies/illness
- Ability to attend all helper hours remotely with instructor permission

If you feel ill or are sick, **please stay home and take care of yourself**. We never want you to feel that you must attend class or helper hours if you are not feeling well. And if you are ill or have another emergency or exceptional circumstance, please reach out to us so that we can help!

Stanford Honor Code

- The **Honor Code** is an undertaking of the students, individually and collectively:
  - that they will not give or receive aid in examinations; that they will not give or receive unpermitted aid in class work, in the preparation of reports, or in any other work that is to be used by the instructor as the basis of grading;
  - that they will do their share and take an active part in seeing to it that others as well as themselves uphold the spirit and letter of the Honor Code.

- The faculty on its part manifests its confidence in the honor of its students by refraining from proctoring examinations and from taking unusual and unreasonable precautions to prevent the forms of dishonesty mentioned above. The faculty will also avoid, as far as practicable, academic procedures that create temptations to violate the Honor Code.

- While the faculty alone has the right and obligation to set academic requirements, the students and faculty will work together to establish optimal conditions for honorable academic work.

  see also: http://honorcode.stanford.edu/

It is your responsibility to ensure you have read and are familiar with the honor code guidelines posted on the main page of the CS111 course website. Please read them and come talk to us if you have any questions or concerns.

Honor Code and CS111

- Please help us ensure academic integrity:
  - Indicate any assistance received on HW (books, friends, etc.).
  - Do not look at other people's solution code or answers
  - Do not give your solutions to others or post them on the web or our Ed forum.
  - Report any inappropriate activity you see performed by others.
  - Tutoring is not appropriate for help with work that will be submitted for a grade.
- Assignments are checked regularly for similarity with help of software tools.
- If you need help, please contact us and we will help you.
  - We do not want you to feel any pressure to violate the Honor Code in order to succeed in this course.
  - If you realize that you have made a mistake, you may retract your submission to any assignment at any time, no questions asked, up to the start of the final exam.
    https://cs111.stanford.edu/collaboration
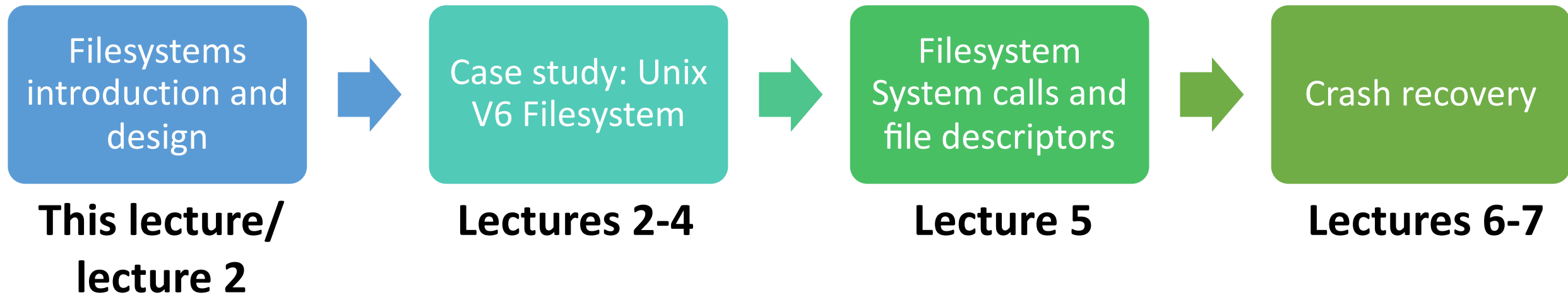
Question Break!

What questions do you have about course support or the honor code?

Plan For Today

- Introduction
- CS111 Course Topics
- CS111 Course Policies
- **Filesystems Introduction**

Topic 1: Filesystems - How can we design filesystems to manage files on disk, and what are the tradeoffs inherent in designing them? How can we interact with the filesystem in our programs?

CS111 Topic 1: Filesystems

Filesystems introduction and design → Case study: Unix V6 Filesystem → Filesystem System calls and file descriptors → Crash recovery

This lecture/ lecture 2 **Lectures 2-4** **Lecture 5** **Lectures 6-7**

assign1: implement portions of the Unix v6 filesystem!

Recap

- Introduction
- CS111 Course Topics
- CS111 Course Policies
- Filesystems Introduction

Recap

- CS111 is a class in C/C++ that teaches you about operating systems and how they work.
- Please visit the course website, cs111.stanford.edu, where you can read the General Information page, information about the Honor Code in CS111, and more about CS111 course policies and logistics.
- Check out our [getting started guide](#) for getting setup to work on assignments for the quarter.
- Our first topic is filesystems – understanding how we can store files on disk.

We're looking forward to an awesome quarter!

Next time: more about filesystems