Course Introduction

COS 417: Operating Systems

Spring 2025, Princeton University

Faculty

Prof. Mae Milano

Prof. Amit Levy

TAs

Alan Zhang

Angie Zhao

Phyllis Wang

Today

1. What is an operating system?
2. OSs and Computers focus of this course
3. Some course administrativia
   - First assignments

What is an Operating System?

Some examples of Operating Systems

- Windows
- Mac OS
- iOS
- Android
- Linux
- FreeBSD
- Fuschia
- sel4

Some examples of Operating Systems

- Windows
- Mac OS
- iOS
- Android
- Linux
- FreeBSD
- Fuschia
- sel4

- Windows NT
- Darwin
- Linux kernel

Some examples of Operating Systems

- Windows
- Mac OS
- iOS
- Android
- Linux
- FreeBSD
- Fuschia
- sel4

- Windows NT
- Darwin
- Linux kernel

- Browser
- “The Cloud”
- JVM

No well-accepted definition

Strawman definitions:

No well-accepted definition

Strawman definitions:

1. “Whatever an OS vendor ships with the OS”

No well-accepted definition

Strawman definitions:

1. “Whatever an OS vendor ships with the OS”

- Garage band?
- Calculator?

No well-accepted definition

Strawman definitions:

1. “Whatever an OS vendor ships with the OS”

- Garage band?
- Calculator?

2. “The program that runs first and always, spawns all other programs.” Usually referring to the kernel.

No well-accepted definition

Strawman definitions:

1. “Whatever an OS vendor ships with the OS”

- Garage band?
- Calculator?

2. “The program that runs first and always, spawns all other programs.” Usually referring to the kernel.

- Board management controller
- UEFI
- Bootloader

Why do we need an Operating System?

1950s-1960s: Mainframes & human operators

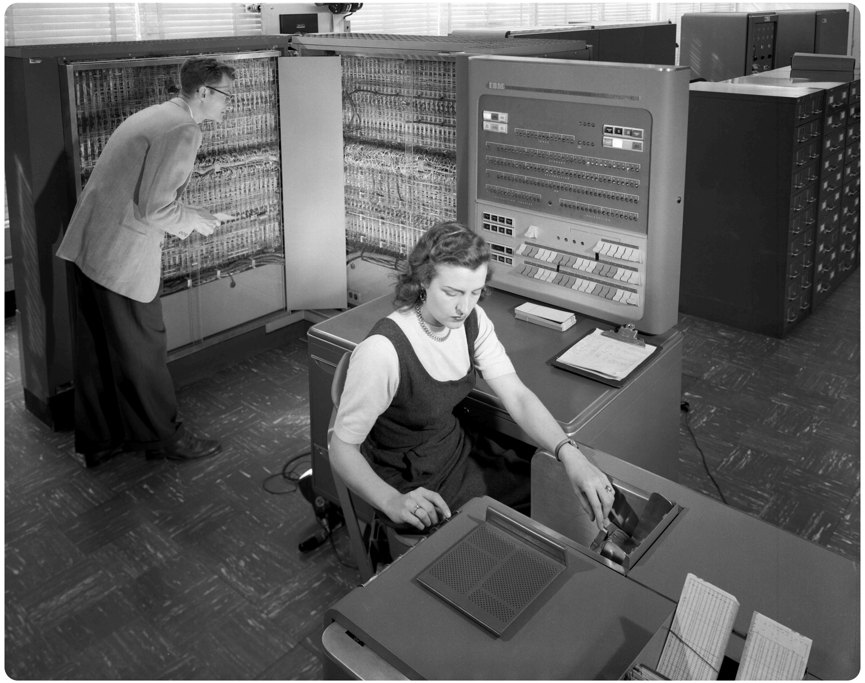

Figure 1: IBM type 704 data processing machine used for making computations for aeronautical research.

1950s-1960s: Mainframes & human operators

Human operator responsible:

- How to share this expensive resource
- Make sure programs run correctly,
- Keeping the machine running efficiently

1960s-1970s: Minicomputers & First OSs

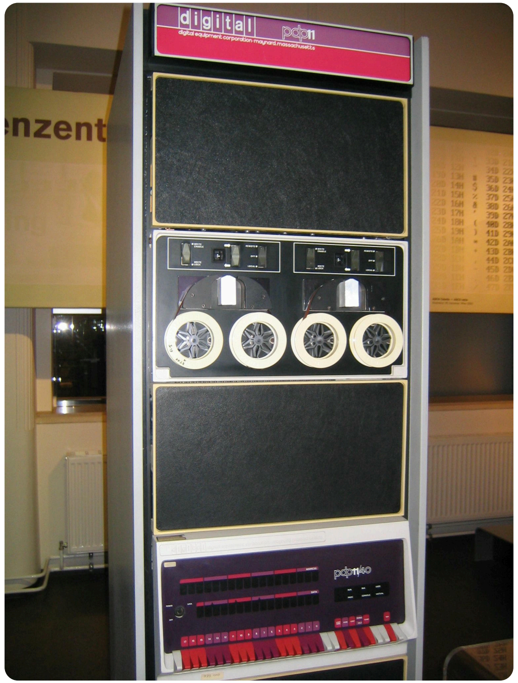

Figure 2: PDP-11, an early “minicomputers”. $1M (in 2025 $s), 10MHz CPU, 100s KB of memory.

1960s-1970s: Minicomputers & First OSs

Need software operators. An Operating System!

- Mediate access to the computer’s resources.
- Ensure concurrent programs don’t affect each other’s correctness
- Consistent interface to the hardware.
- Retain reasonable performance

Today: Dozens of supercomputers per person

Figure 3: Desktops, laptops, phones, cars, buildings, medical devices...

Today: Dozens of supercomputers per person

Much more powerful, and typically only one user, but…

Still run more than one program at a time. Still use operating systems for:

- Resource sharing
- Safety & correctness
- Performance
- Portability

UNIX 1969-Today

Darwin (Mac OS, iOS), Linux, among most other OSs you've heard of, still include nearly unchanged:

- Process model
- Core process API
- File system
- Kernel architecture

Figure 4: Ken Thompson & Dennis Ritchie building UNIX on a PDP-11, 1972

What resources are there to manage?

What resources are there to manage?

- CPU
  - Processor(s)
  - Independent and shared caches
- Memory
- Storage
- Network
- Peripheral computation units
  - GPUs
  - TPUs
  - TPMs
- “HW” root-of-trust, etc
- Human-Machine-Interfaces
- Sensors
- Actuators

Course Administrativia

- up-to-date syllabus on the website: https://www.cs.princeton.edu/courses/cos417

1/27	Course Intro		Process 1	2,4	Warmup
2/3	Process 2	5	Alternatives	-	Shell
2/10	Scheduling	6-7	Scheduling	8-10	
2/17	Memory	13-15	Paged VM	13-15	Scheduler
2/24	Swapping	18-20	Alternatives	16	
3/3	Alternatives	-	Midterm		Virt. Mem.

- For Thursday, read chapters 2 & 4
- Get started on warmup, and make sure you can SSH to courselab

What is an Operating System?

A set of choices about…

- Resources to expose to programs
- Abstractions used to present resources
- How to guarantee abstractions contracts

This semester we'll learn

- Choices made by predominating operating systems
- How to implement them
- Alternative choices and their implications