

# Introduction to CS350

Lesley Istead, Kevin Lanctot

David R. Cheriton School of Computer Science  
University of Waterloo

Winter 2019

1 / 17

## Welcome!

# Welcome to CS350 - Operating Systems!



- Administrative Information
- Introduction to Operating Systems

2 / 17

## General Information

Important links:

- <http://www.student.cs.uwaterloo.ca/~cs350>  
Course personnel, office hours, readings, assignments, tutorials, previous midterms, review problems, etc.
- <https://piazza.com>  
Piazza will be used for announcements, extra notes, questions, corrections, etc. Please check piazza regularly. **Do not post your code in public piazza posts; use private posts when appropriate.**

3 / 17

## Course Readings

Course notes are **required**.

They are **NOT** designed to be standalone. Come to class, take notes. Notes are available online from the course website. You may also purchase a printed copy, if you desire.

Textbook is **NOT** required, but highly recommended.

Operating Systems: Three Easy Pieces

Remzi H. Arpaci-Dusseau and Andrea C. Arpaci-Dusseau

Textbook is available **FREE** on-line. Link to the text is available on course website. All recommended readings are linked on course website.

4 / 17

## Grading Scheme

A0, A1, A2a, A2b, A3: Assignment marks as a percentage

M: Midterm exam grades as percentages

F: Final exam grade as a percentage

$$\text{Normal} = (0.02 * A0 + 0.08 * A1 + 0.07 * A2a + 0.08 * A2b + 0.10 * A3) + (0.20 * M + 0.45 * F)$$

$$\text{Exams} = (0.20 * M + 0.45 * F) / 0.65$$

```
if (Exams < 50%) {  
    FAIL EXAMS, FAIL THE COURSE  
    Course Grade = min(Normal, Exams)  
} else {  
    Course Grade = Normal  
}
```

You **WILL FAIL** this course if you fail the weighted exam average, regardless of your assignment grades.

5 / 17

## Assignments

There are **5** assignments.

All assignments are to be done **individually**.

You will not be writing your own OS.

You will be adding/fixing features of an existing OS.

We use **OS/161** (~22,000 lines for kernel), which runs on **SYS/161** (MIPS simulator/VM)

Slip days:

- Allows flexibility in assignment deadlines
- Total of 5 slip days
- Can use maximum of 3 slip days per assignment (except A3)

6 / 17

READ AND UNDERSTAND INFO ON COURSE WEB PAGE

This course has extra requirements and ignorance is no excuse!

Do not use code from other sources:

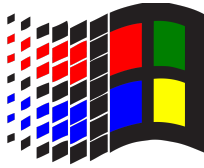
- Do not copy code from friends, web sites, or other sources
- Do not search for or look at other code for any reason
- Avoid blogs that provide instructions
- We use VERY GOOD cheat detection software
- Every term people are caught
- Often: 0 on assignment and -5% off final grade

**IF** you have taken this course before, you may reuse your previous code if:

- You ask your instructor for permission
- Your code was not subject to previous cheating penalties
- You understand it will be re-tested using our cheat detection software

# What is an Operating System?

Generally, an OS is a system that:



- manages resources
- creates execution environments
- loads programs
- provides common services and utilities

## Operating Systems

- originated 1951, 'LEO I' from J. Lyons and Co.
- started as simple I/O libraries, batch processors

9 / 17

# Three views of an Operating System



**Application View:** what services does it provide?

**System View:** what problems does it solve?

**Implementation View:** how is it built?

An operating system is part cop, part facilitator.

10 / 17

## Application View of an Operating System

The OS provides an execution environment for running programs.

The execution environment:

- provides a program with the processor time and memory space that it needs to run, and
- provides interfaces through which a program can use networks, storage, I/O devices, and other system hardware components. Interfaces provide a simplified, abstract view of hardware to application programs.
- isolates running programs from one another and prevents undesirable interactions among them.

11 / 17

## System View of an Operating System

The OS:

- manages the hardware resources of a computer system. Resources include processors, memory, disks and other storage devices, network interfaces, I/O devices such as keyboards, mice and monitors, etc.
- allocates resources among running programs.
- controls the sharing of resources among programs.

The OS itself also uses resources, which it must share with application programs.

12 / 17

The OS is a concurrent, real-time program.

- Concurrency arises naturally in an OS when it supports concurrent applications, and because it must interact directly with the hardware.
- Hardware interactions also impose timing constraints.

How does the OS implement concurrency and hardware interactions?

## Concurrency

... multiple programs or sequences of instructions running, or appearing to run, at the same time.

13 / 17

## Definitions

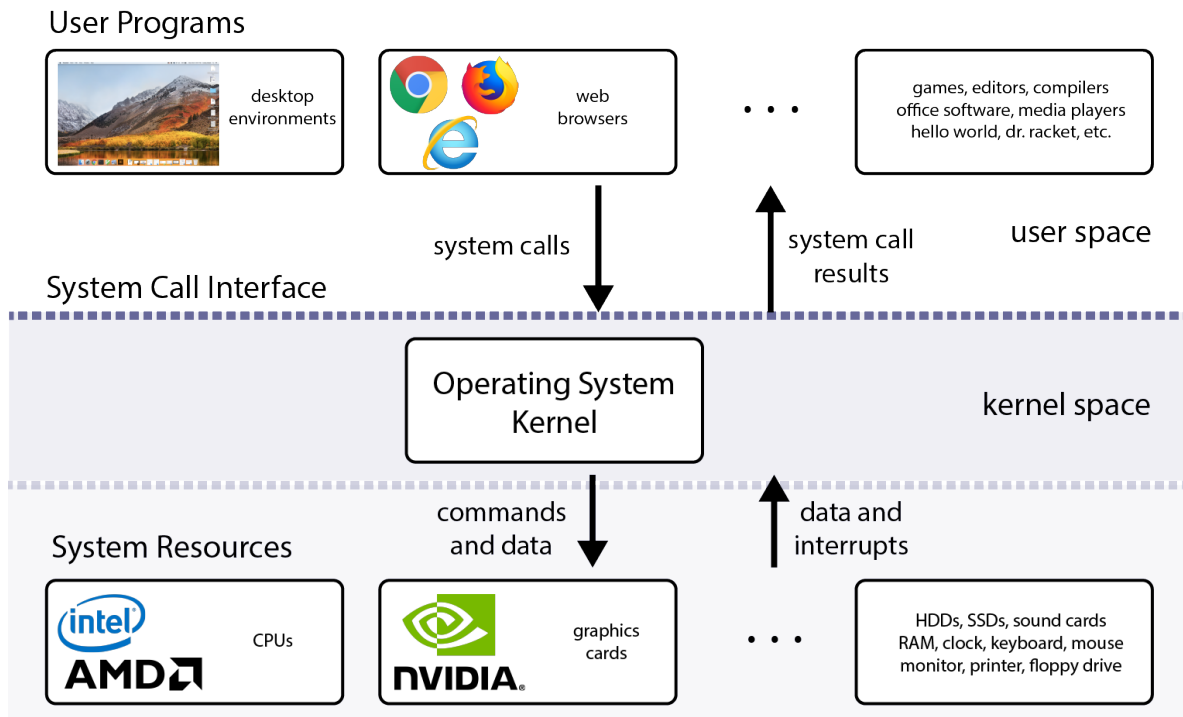
**kernel:** The operating system kernel is the part of the operating system that responds to system calls, interrupts and exceptions.

**operating system:** The operating system as a whole includes the kernel, and may include other related programs that provide services for applications. This may include things like:

- utility programs
- command interpreters
- programming libraries

14 / 17

# Schematic View of an Operating System



15 / 17

## Operating System Abstractions

The **execution environment** provided by the OS includes a variety of **abstract entities** that can be manipulated by a running program. Examples of these abstractions:

**files and file systems** → secondary storage

**address spaces** → primary memory (RAM)

**processes, threads** → program execution

**sockets, pipes** → network or other message channels

This course will cover why and how these abstractions are:

- designed the way they are
- manipulated by application programs
- implemented by the OS

16 / 17



- Introduction
- Threads and Concurrency
- Synchronization
- Processes and the Kernel
- Virtual Memory
- Scheduling
- Devices and Device Management
- File Systems
- Virtual Machines