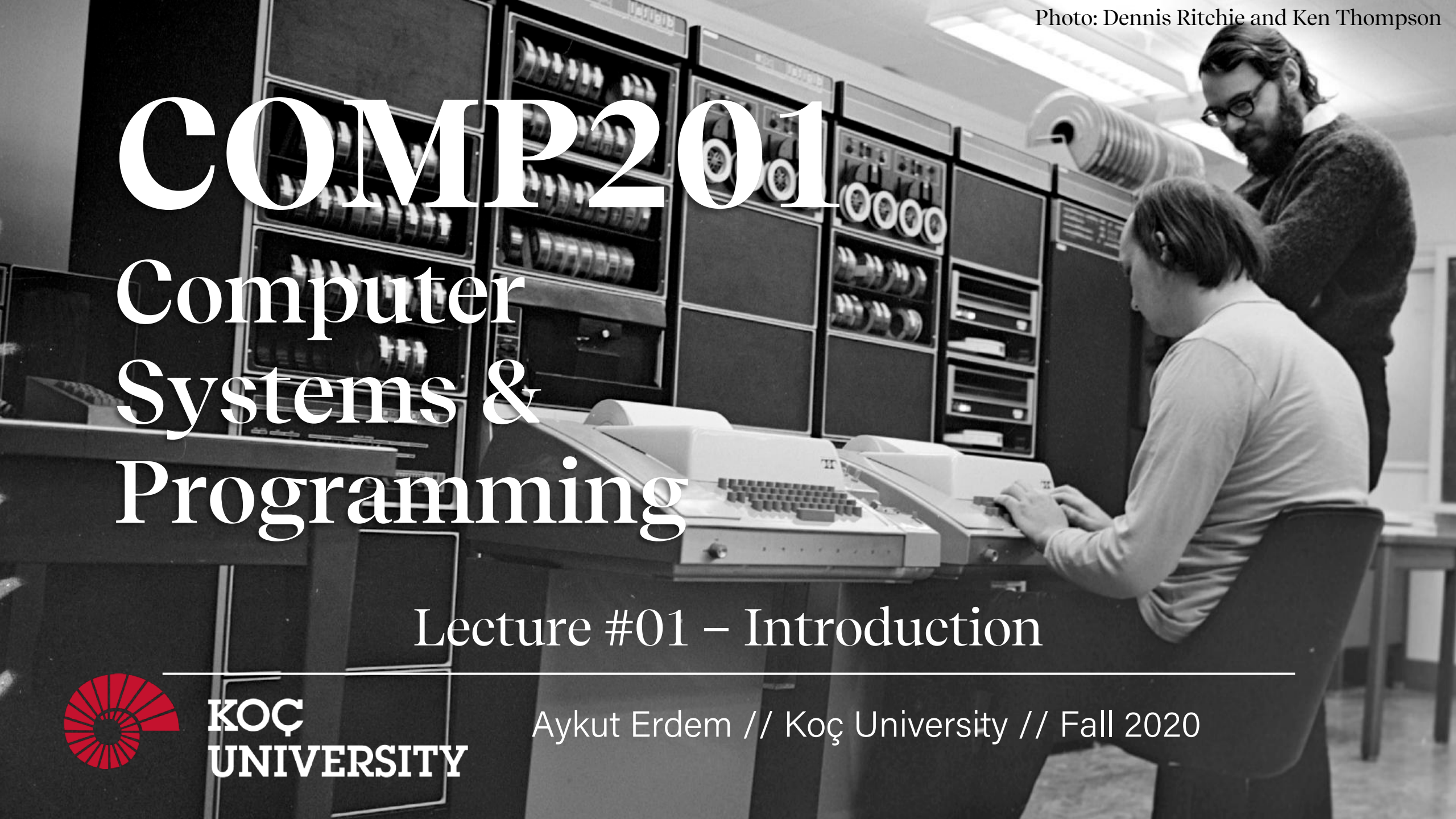


Photo: Dennis Ritchie and Ken Thompson

COMP201

Computer Systems & Programming

Lecture #01 – Introduction

KOÇ UNIVERSITY

Aykut Erdem // Koç University // Fall 2020

A little about me…

Koç University
Associate Professor
2020-now

Hacettepe University
Associate Professor
2010-2020

Universitá Ca' Foscari di Venezia
Post-doctoral Researcher
2008-2010

Middle East Technical University
1997-2008
Ph.D., 2008
M.Sc., 2003
B.Sc., 2001

MIT
Fall 2007
Visiting Student

VirginiaTech
Visiting Research Scholar
Summer 2006

joined Koç University this September

https://aykuterdem.github.io

I explore better ways to understand, interpret and manipulate visual data. My research interests span a diverse set of topics, ranging from image editing to visual saliency estimation, and to multimodal learning for integrated vision and language.

Plan For Today

- Course Introduction
- COMP201 Course Policies
- Unix and the Command Line

Disclaimer: Slides for this lecture were borrowed from

—Nick Troccoli's Stanford CS107 class

COMP201 on Zoom

- You are encouraged to turn on your cameras in order to keep human touch with others and not to feel alone, but this is clearly not a must!
- Before the class, you can post your questions or comments to the discussion forum thread on Blackboard for that day's lecture. We'll take regular question breaks to address your questions.
- In Zoom, everyone is muted by default. However, you can unmute yourself or alternatively use Zoom chat to ask your questions during the question breaks.

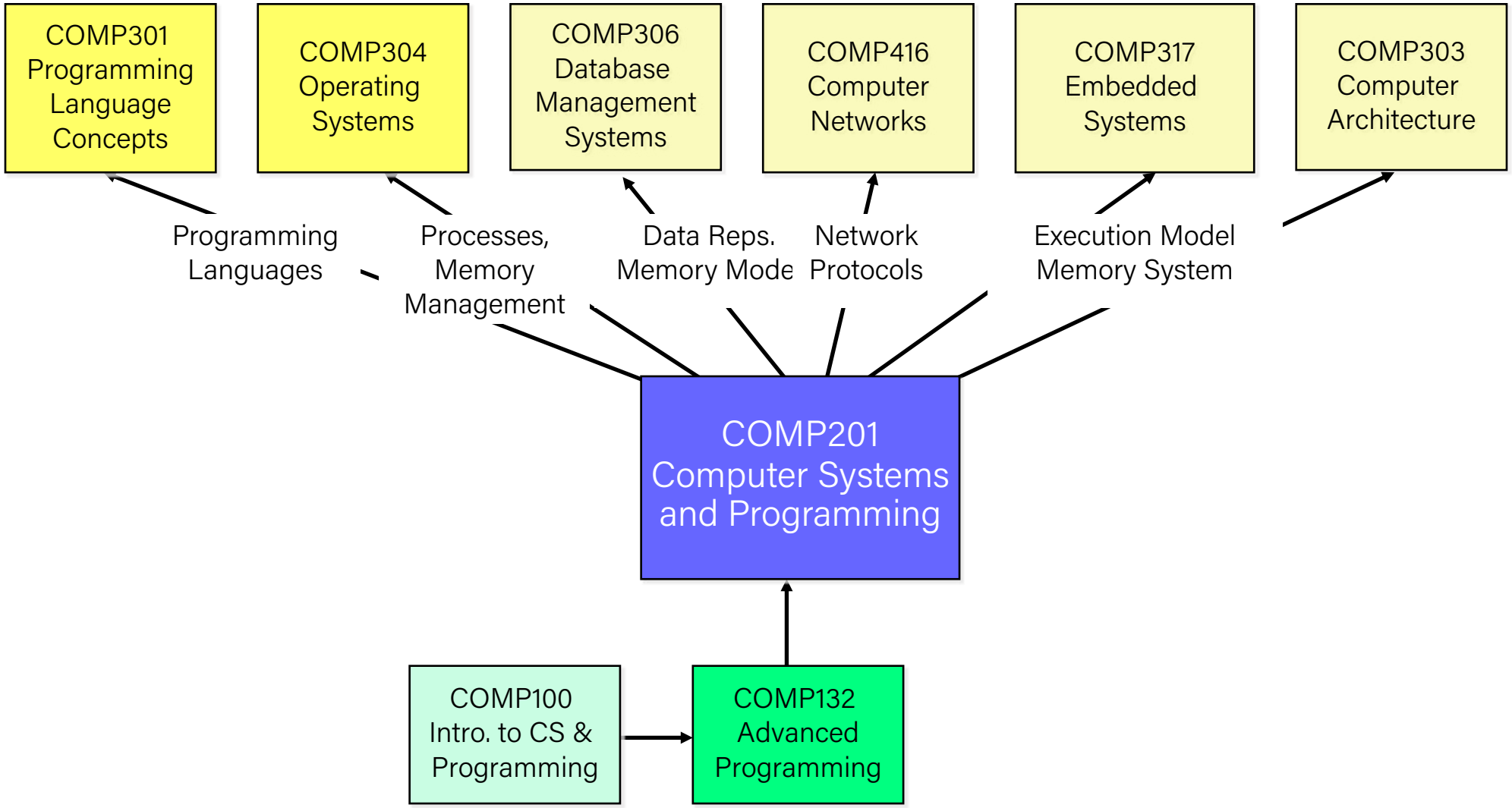
Plan For Today

- Course Introduction
- COMP201 Course Policies
- Unix and the Command Line

What is COMP201?

- A new addition to the COMP curriculum, designed from ground-up.
- The third course in the line of COMP's introductory to programming courses (COMP100, COMP132, and COMP201)
  - COMP100 teaches you the notion of computational thinking and how to solve problems as a programmer (using Python)
  - COMP132 introduces you object-oriented programming paradigm (using Java)
- COMP201 takes you **behind the scenes**:
  - Not quite down to hardware or physics/electromagnetism (that's for later…)
  - It's how things work **inside C++/Python/Java**, and how your programs map onto the components of computer systems
  - Not only does it just feel good to know how these work, it can also inform projects you work on in the future.

Role within COMP Curriculum



What is COMP201?

Computer Systems and Programming

- How languages like C++ and Java **represent data** under the hood
- How programming structures are encoded in **bits and bytes**
- How to efficiently **manipulate and manage memory**
- How computers **compile** programs
- How **cache memories work** and **how to exploit them** to improve the performance of your programs
- Uses the **C** programming language
- Programming **style** and software development practices

COMP201 Learning Goals

The goals for COMP201 are for students to gain **mastery** of

- writing C programs with complex use of memory and pointers
- an accurate model of the address space and compile/runtime behavior of C programs

to achieve **competence** in

- translating C to/from assembly
- writing programs that respect the limitations of computer arithmetic
- finding bottlenecks and improving runtime performance
- working effectively in a Unix development environment

and have **exposure** to

- a working understanding of the basics of cache memories

Course Overview

1. **Bits and Bytes -** *How can a computer represent integer numbers?*
2. **Chars and C-Strings -** *How can a computer represent and manipulate more complex data like text?*
3. **Pointers, Stack and Heap –** *How can we effectively manage all types of memory in our programs?*
4. **Generics -** *How can we use our knowledge of memory and data representation to write code that works with any data type?*
5. **Assembly -** *How does a computer interpret and execute C programs?*
6. **The Memory Hierarchy -** *How does the memory system is organized as a hierarchy of different storage devices with unique capacities*
7. **The Heap Allocators -** *How do core memory-allocation operations like malloc and free work?*

COMP201 - The Online Edition

- This quarter, we will try our best to make your COMP201 experience as close as the face-to-face teaching version.
- We are working to emphasize community and connection.
- We understand that the pandemic provides unique challenges for each and every one of us involved.
- We will constantly evaluate and listen to ensure the class is going as smoothly as possible for everyone.
- Please don't hesitate to reach out if you want our support. We are always here for you.

Teaching Team

Aykut Erdem

Ahmed Imam Shah

Amir Mohamad Akhlaghi Gharelar

Farzin Negahbani

Ilker Kesen

Mandana Bagheri Marzijarani

Muhammad Aditya Sasongkoand

Samet Demir

Course Website

https://aykuterdem.github.io/classes/comp201/

*lecture videos on Panopto – can be accesses through Blackboard or from the course webpage

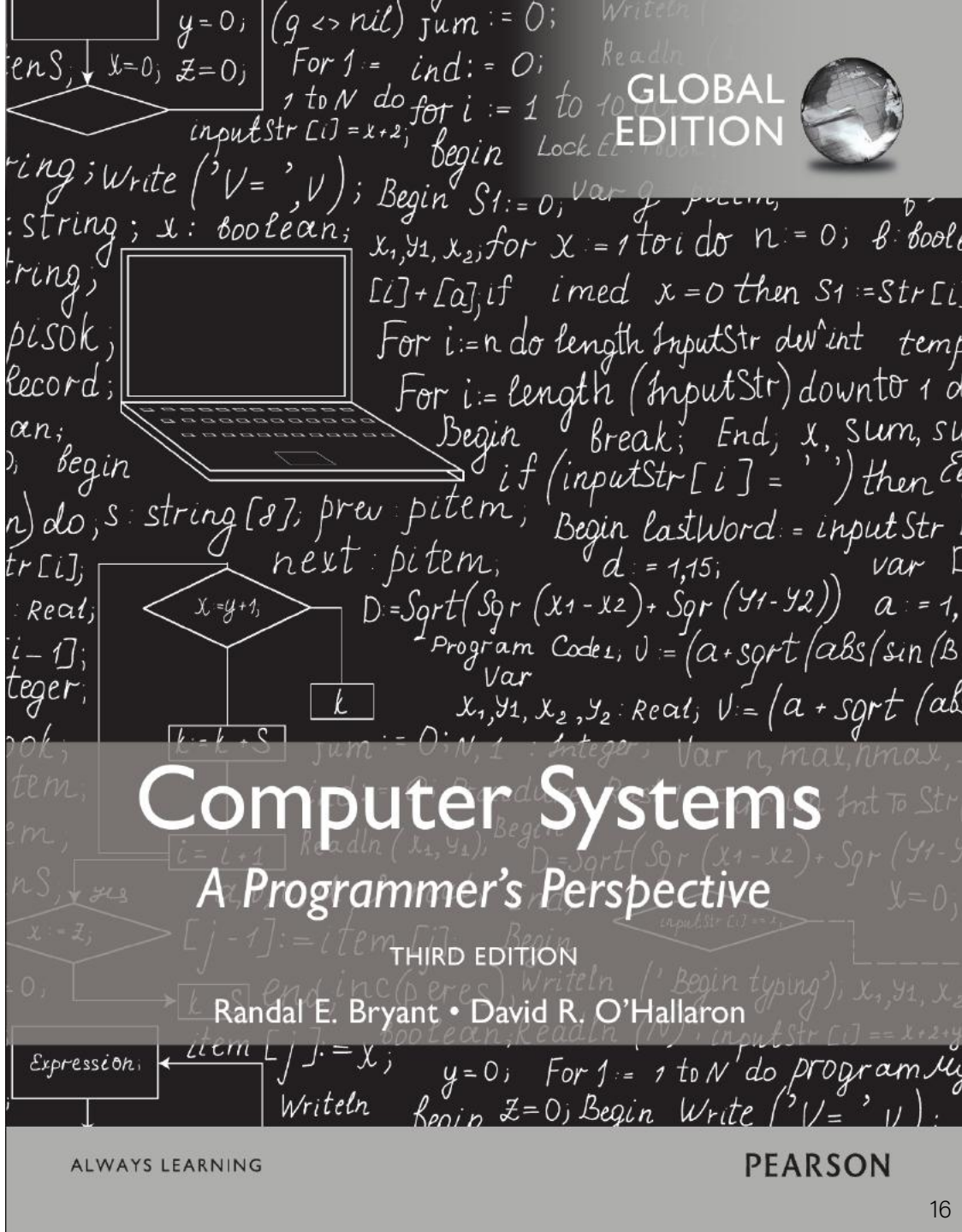
Question Break!

Plan For Today

- Introduction
- COMP201 Course Policies
- Unix and the Command Line

Textbooks

- *Computer Systems: A Programmer's Perspective* by Bryant & O'Hallaron, **3rd Edition**
  - **3rd edition matters** – important updates to course materials
- A C programming reference of your choice
  - *The C Programming Language* by Kernighan and Ritchie
  - Other C programming books, websites, or reference sheets



Course Structure

- **Lectures:** understand concepts, see demos
- **Labs:** learn tools, study code, discuss with peers
- **Assignments:** build programming skills, synthesize lecture/lab content

Monday	Wednesday	Thursday	Friday
Lecture	Lecture		Lecture
Lab-A	Lab-B	Lab-C	

- **assg0**: out later on Wednesday, due Oct 16 (covers Wed. lecture)
- **C bootcamp**: this week (3 different sessions during lab hours)

Grading

32%	Programming assignments
16%	Labs
10%	Quizzes
18%	Midterm exam
24%	Final exam

Grading

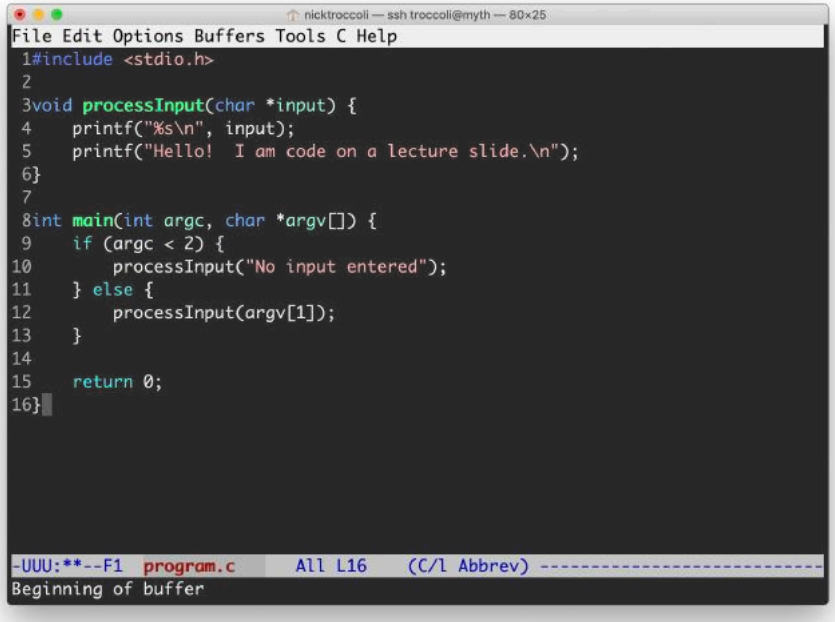
32%	Programming assignments
16%	Labs
10%	Quizzes
18%	Midterm exam
24%	Final exam

Assignments

- 7 programming assignments completed individually using **Unix command line tools**
  - Free software, pre-installed on a server dedicated to COMP students (*later*)
  - GitHub Classroom + Repl.it (*for now*)
  - We will give out starter projects for each assignment
- Graded on **functionality** (*behavior*) and **style** (*elegance*)
  - Functionality graded using *automated tools,* given as point score
  - Style graded via *automated tests* and TA code review,
  - Grades returned via Blackboard

GitHub Classroom

repl.it



Late Policy

- **Start out with 5 *grace days***: each late day allows you to submit an assignment up to 24 additional hours late without penalty.
- **Hard deadline:** No submissions will be accepted **48 hours** after the original due date
- Penalty per day after grace days are exhausted
  - 1 day: 20% off
  - 2 days: 40% off

Question Break!

Grading

32%	Programming assignments
16%	Labs
10%	Quizzes
18%	Midterm exam
24%	Final exam

Lab Sections

- Weekly 60-minute labs led by a TA, starting *next* week, offered on Mondays, Wednesdays and Thursdays.
- Hands-on practice with lecture material and course topics.

  GitHub Classroom + repl.it
- Graded on attendance + participation *(verified by submitting work at the end)*
  - Your lowest 3 scores will be dropped, hence there will be no make-up
- Lab preference submissions will be open **Tuesday 10/6 at 5PM** and **are not first-come first-serve.** You may submit your preferences anytime until **Friday 10/9 at 5PM.** Sign up info will be posted on Blackboard

Question Break!

Grading

32%	Programming assignments
16%	Labs
10%	Quizzes
18%	Midterm exam
24%	Final exam

Midterm and Final Exams

- Online exams through Blackboard
  - Midterm: At week 9, date and time will be announced later
  - Final: Date and time will be announced later
- The exams will include multiple-choice questions as well as open-ended fill-in-the-blanks or short answer type questions.
- The exams will be released in multiple sessions, in which each student is required to complete each session in a limited time.

Question Break!

Grading

32%	Programming assignments
16%	Labs
10%	Quizzes
18%	Midterm exam
24%	Final exam

Quizzes

- Short quizzes after every Friday class to help you stay on the topics covered in the class.
- They will be relatively simpler but you need to attend the lectures at each week in order to answer the given questions.
- Each quiz will be weighted the same.
- Your lowest 2 scores will be dropped, hence there will be no make-up.

Grading

32%	Programming assignments
16%	Labs
10%	Quizzes
18%	Midterm exam
24%	Final exam

NOTE: Once the final letter grades are determined, a student can choose to be graded with “S” grade if the final grade is "D" or above.

Question Break!

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 or general assignment questions (DON'T POST ASSIGNMENT CODE!)
- Visit **Online Office Hours**
  - More info to come soon!
- **Email** the Course Staff
  - Best for **private matters** (e.g. grading questions).

Koç University Honor Code

- For assignments/quizzes/exams students should be required to digitally add and approve a version of the agreement below.

> *I hereby declare that I have completed this examination individually, without support from anyone else.*
>
> *I hereby accept that only the below listed sources are approved to be used during this open-source examination:*
>
> *(i) Coursebook,*
>
> *(ii) All material that is made available to students via Blackboard for this course, and*
>
> *(iii) Notes taken by me during lectures.*
>
> *I have not used, accessed or taken any unpermitted information from any other source. Hence, all effort belongs to me.*

Honor Code and COMP201

- 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 to the forum.
  - Report any inappropriate activity you see performed by others.
- Assignments are checked regularly for similarity with help of automated software tools.
- If you realize that you have made a mistake, you may retract your submission to any assignment at any time, no questions asked. Come to use before we come for you.
- 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.

Question Break!

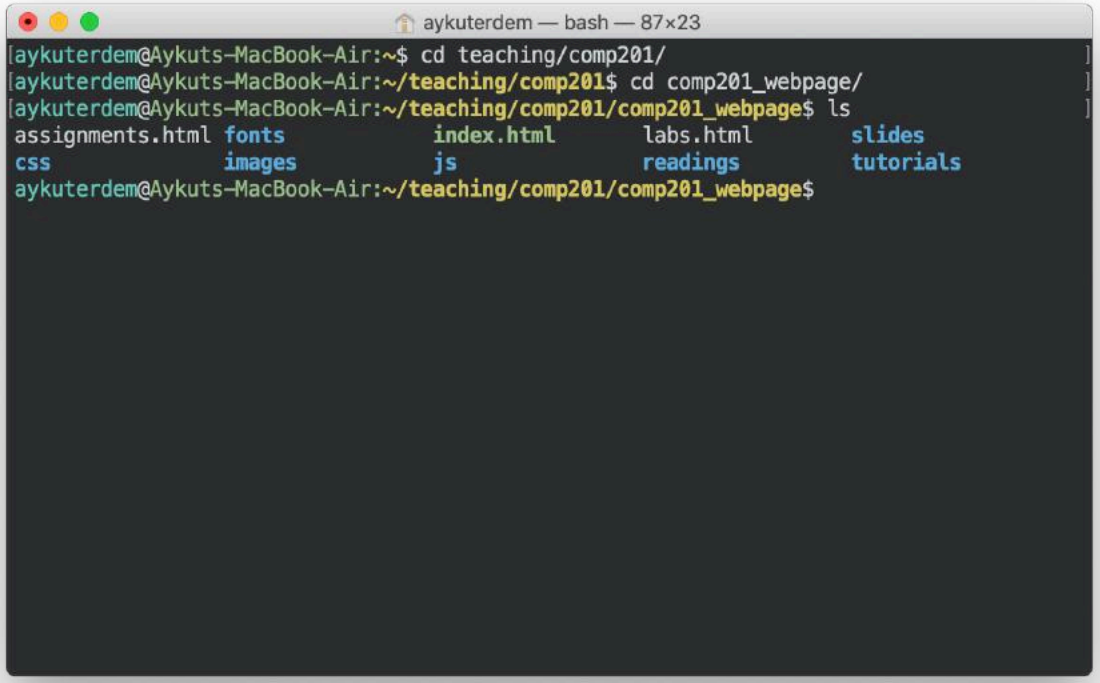
Poll Time

Plan For Today

- Introduction
- COMP201 Course Policies
- Unix and the Command Line

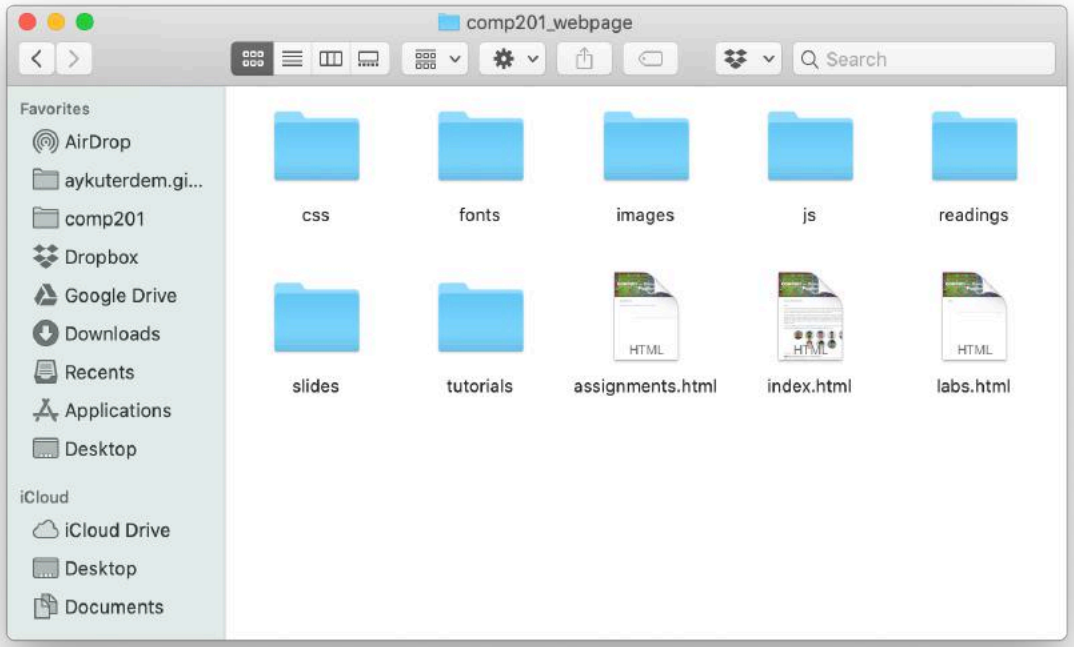
What is Unix?

- **Unix:** a set of standards and tools commonly used in software development.
  - **macOS** and **Linux** are operating systems built on top of Unix
- You can navigate a Unix system using the **command line** (“terminal”)
- Every Unix system works with the same tools and commands

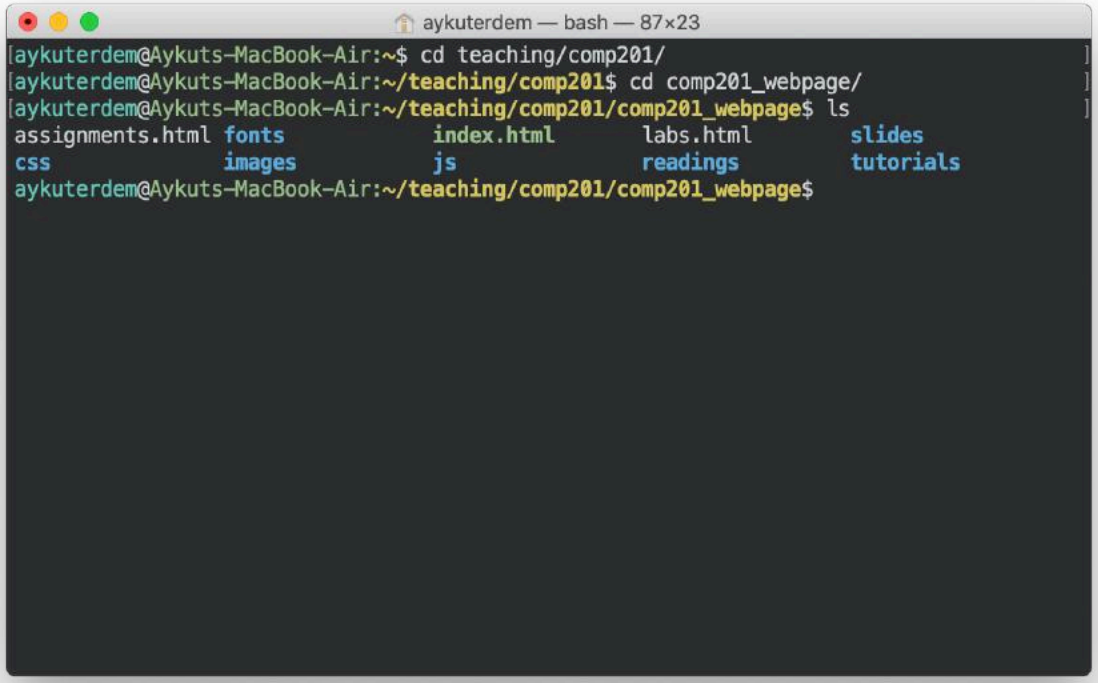


What is the Command Line?

- The **command-line** is a text-based interface (i.e., **terminal** interface) to navigate a computer, instead of a Graphical User Interface (GUI).



Graphical User Interface

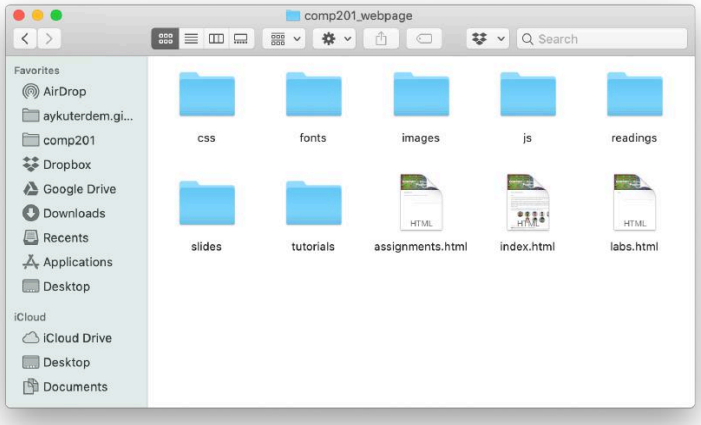


Text-based interface

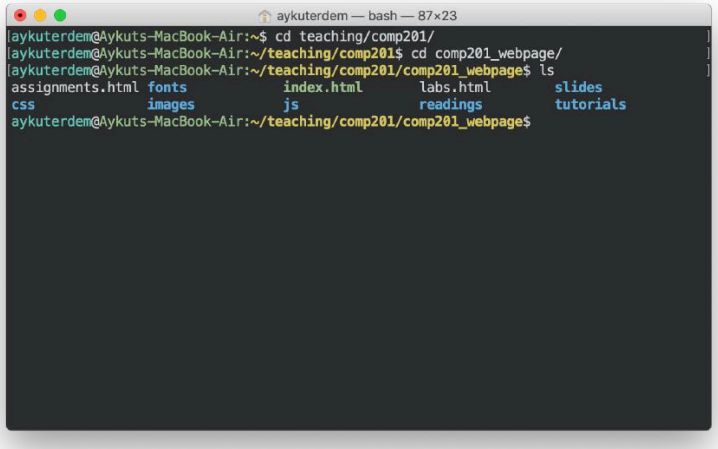
Command Line vs. GUI

Just like a GUI file explorer interface, a terminal interface:

- shows you a **specific place** on your computer at any given time.
- lets you go **into folders** and **out of folders**.
- lets you **create new** files and **edit** files.
- lets you **execute programs**.



Graphical User Interface



Command-line interface

Why Use Unix / the Command Line?

- You can navigate almost any device using the same tools and commands:
  - Servers
  - Laptops and desktops
  - Embedded devices (Raspberry Pi, etc.)
  - Mobile Devices (Android, etc.)
- Used frequently by software engineers:
  - **Web development:** running servers and web tools on servers
  - **Machine learning:** processing data on servers, running algorithms
  - **Systems**: writing operating systems, networking code and embedded software
  - **Mobile Development**: running tools, managing libraries
  - And more…
- We'll use Unix and the command line to implement and execute our programs.

Demo: Using Unix and the Command Line

Unix Commands Recap

- **`cd`** – change directories (..)
- **`ls`** – list directory contents
- **`mkdir`** – make directory
- **`emacs`** – open text editor
- **`vi`** – open text editor
- **`rm`** – remove file or folder
- **`man`** – view manual pages

Lab 1:
The Linux Shell
(*next week*)

See the Resources page of the course website for more commands, and a complete reference.

Learning Unix and the Command Line

- Using Unix and the command line can be intimidating at first:
  - It looks retro!
  - How do I know what to type?

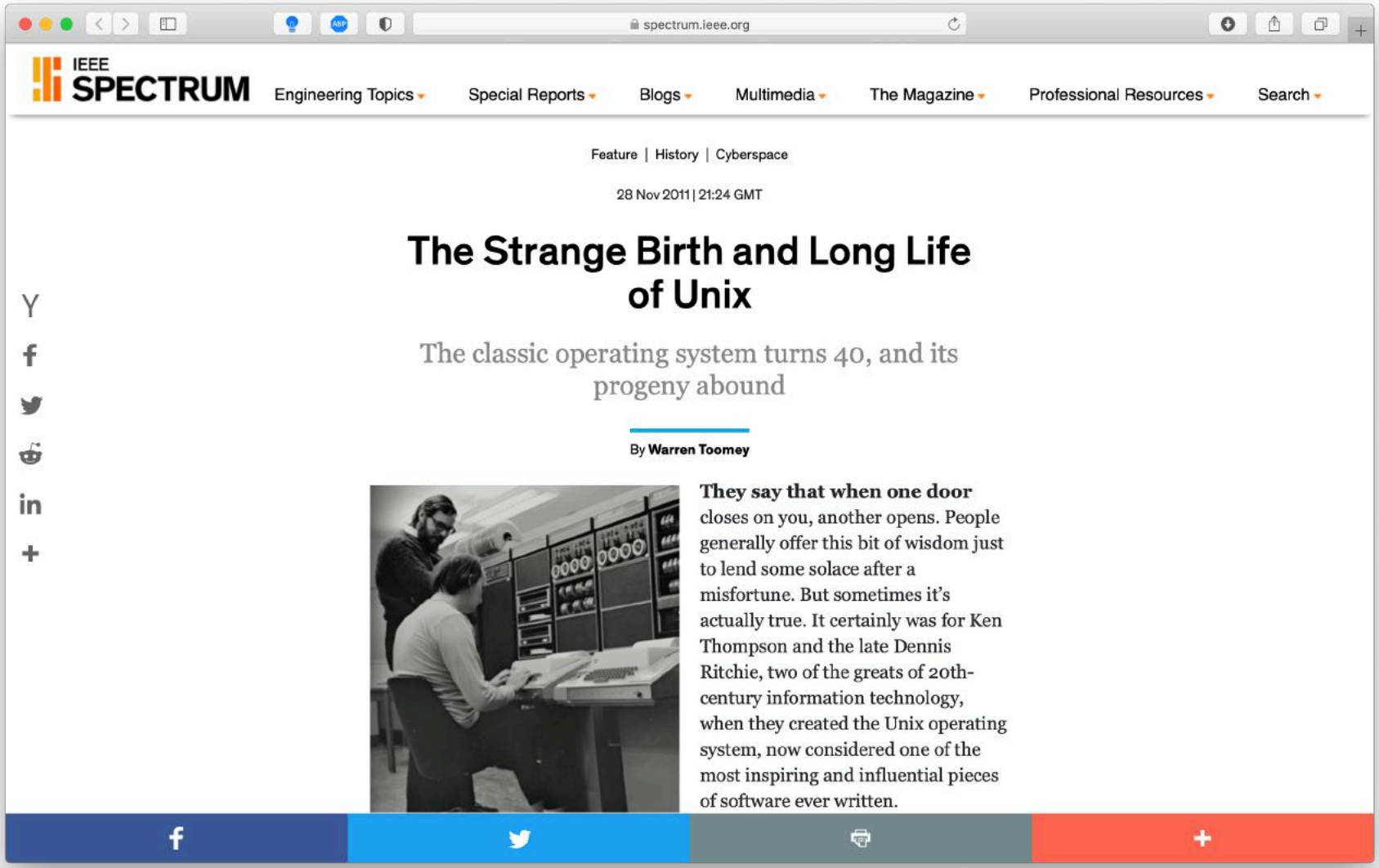
- It's like learning a new language:
  - At first, you may have to constantly look things up (**Resources** page on course website!)
  - It's important to spend as much time as possible (during labs and assignments) building muscle memory with the tools

Question Break!

Recap

- COMP201 is a programming class, which uses C to teach you about what goes on under the hood of programming languages and software.
- We'll use Unix and command line tools to write, debug and run our programs.
- Please regularly visit the course website, https://aykuterdem.github.io/classes/comp201
- **We're looking forward to an awesome quarter!**

Additional Reading



https://spectrum.ieee.org/tech-history/cyberspace/the-strange-birth-and-long-life-of-unix

Next Time on COMP201