

Course Wrap-up & Feedback

Intro to Computer Systems, Fall 2022

Instructor: Travis McGaha

TAs:

Ali Krema

Andrew Rigas

Anisha Bhatia

Audrey Yang

Craig Lee

Daniel Duan

David LuoZhang

Eddy Yang

Ernest Ng

Heyi Liu

Janavi Chadha

Jason Hom

Katherine Wang

Kyrie Dowling

Mohamed Abaker

Noam Elul

Patricia Agnes

Patrick Kehinde Jr.

Ria Sharma

Sarah Luthra

Sofia Mouchtaris

Upcoming Due Dates

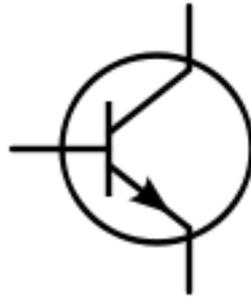
- ❖ HW10/11 (J compiler) is due Friday December 9th
 - HW10 & 11 make up a 2-part assignment that take a while to complete.
 - Can grant extensions on this, but there will be reduced office hours and Ed activity after a bit

- ❖ Final Exam: Thursday December 15th
 - Cumulative
 - In person exam
 - Two 8.5 x 11 inch cheat sheets
 - More info coming soon
 - Review In-lecture next week
 - TA led review during reading days

Lecture Outline

- ❖ **Course Wrap-up & Feedback**

Course Overview: First Half



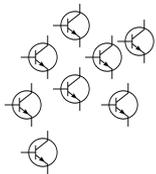
A transistor!



Course Overview: First Half

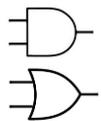


Course Overview: First Half

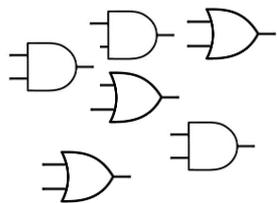


Course Overview: First Half

Logic Gates



Course Overview: First Half



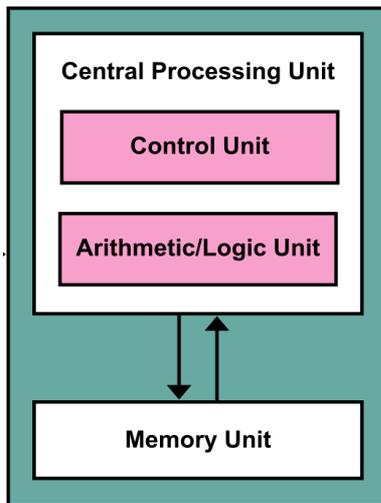
Course Overview: First Half

Adder

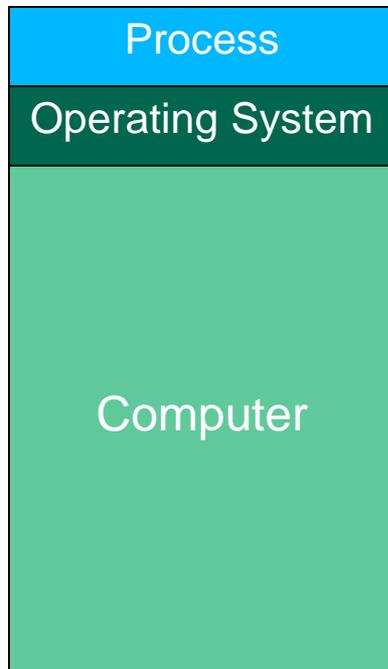
Mux/Demux

Latch/Flip-Flop

Course Overview: First Half



Course Overview: First Half



Course Overview: Second Half

```
1 #include <stdio.h>
2
3 int main() {
4     printf("hello world!\n");
5 }
```

C Programming 😊

```
4 .LC0:
5     .string "hello world!"
6     .text
7     .globl main
8     .type   main, @function
9 main:
10 .LFB0:
11     .cfi_startproc
12     pushq   %rbp
13     .cfi_def_cfa_offset 16
14     .cfi_offset 6, -16
15     movq   %rsp, %rbp
```

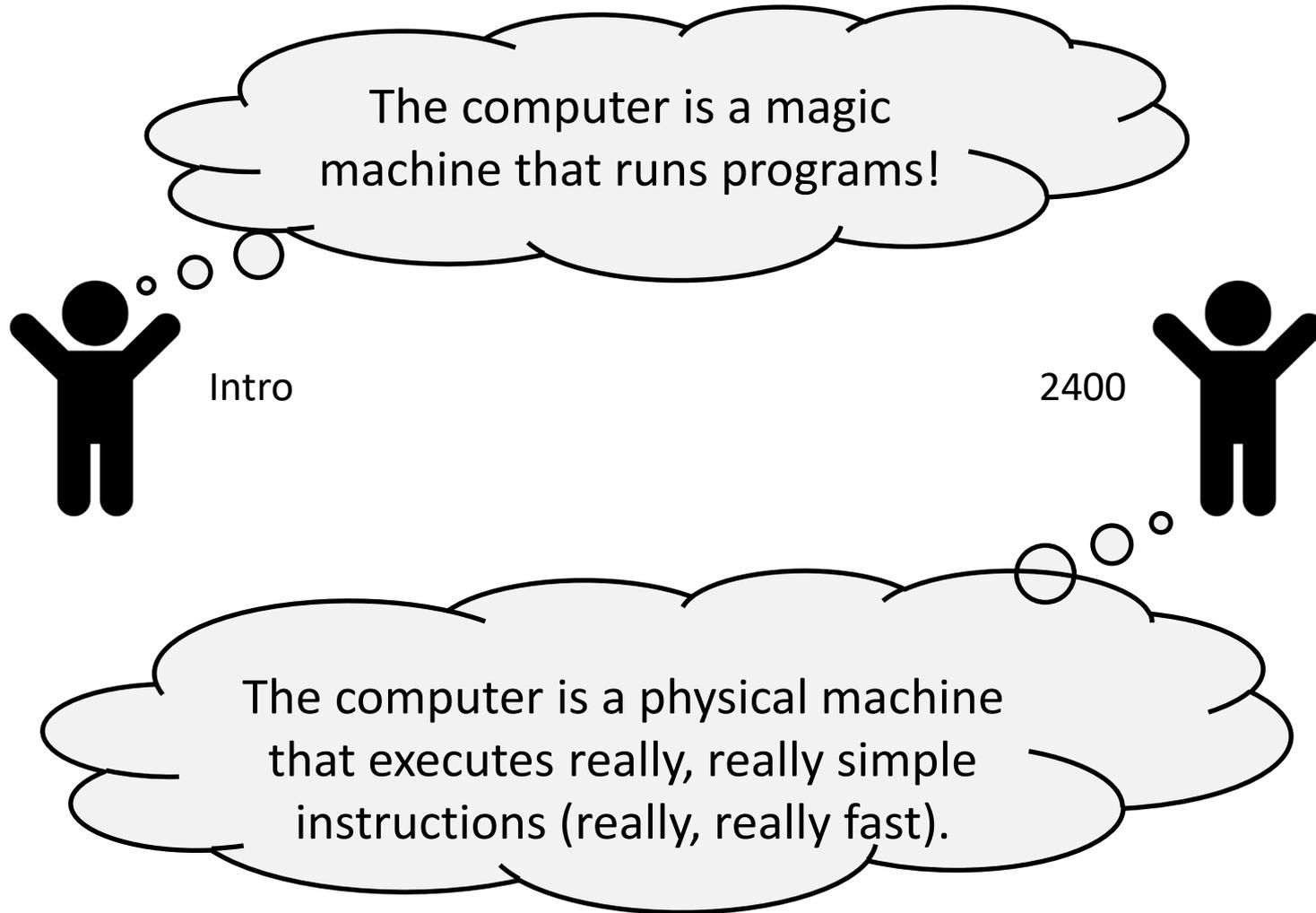
Assembly Translation

```
00000200: 52e5 7464 0400 0000 b80d 0000 0000 0000
00000210: b80d 2000 0000 0000 b80d 2000 0000 0000
00000220: 4802 0000 0000 0000 4802 0000 0000 0000
00000230: 0100 0000 0000 0000 2f6c 6962 3634 2f6c
00000240: 642d 6c69 6e75 782d 7838 362d 3634 2e73
00000250: 6f2e 3200 0400 0000 1000 0000 0100 0000
00000260: 474e 5500 0000 0000 0300 0000 0200 0000
```

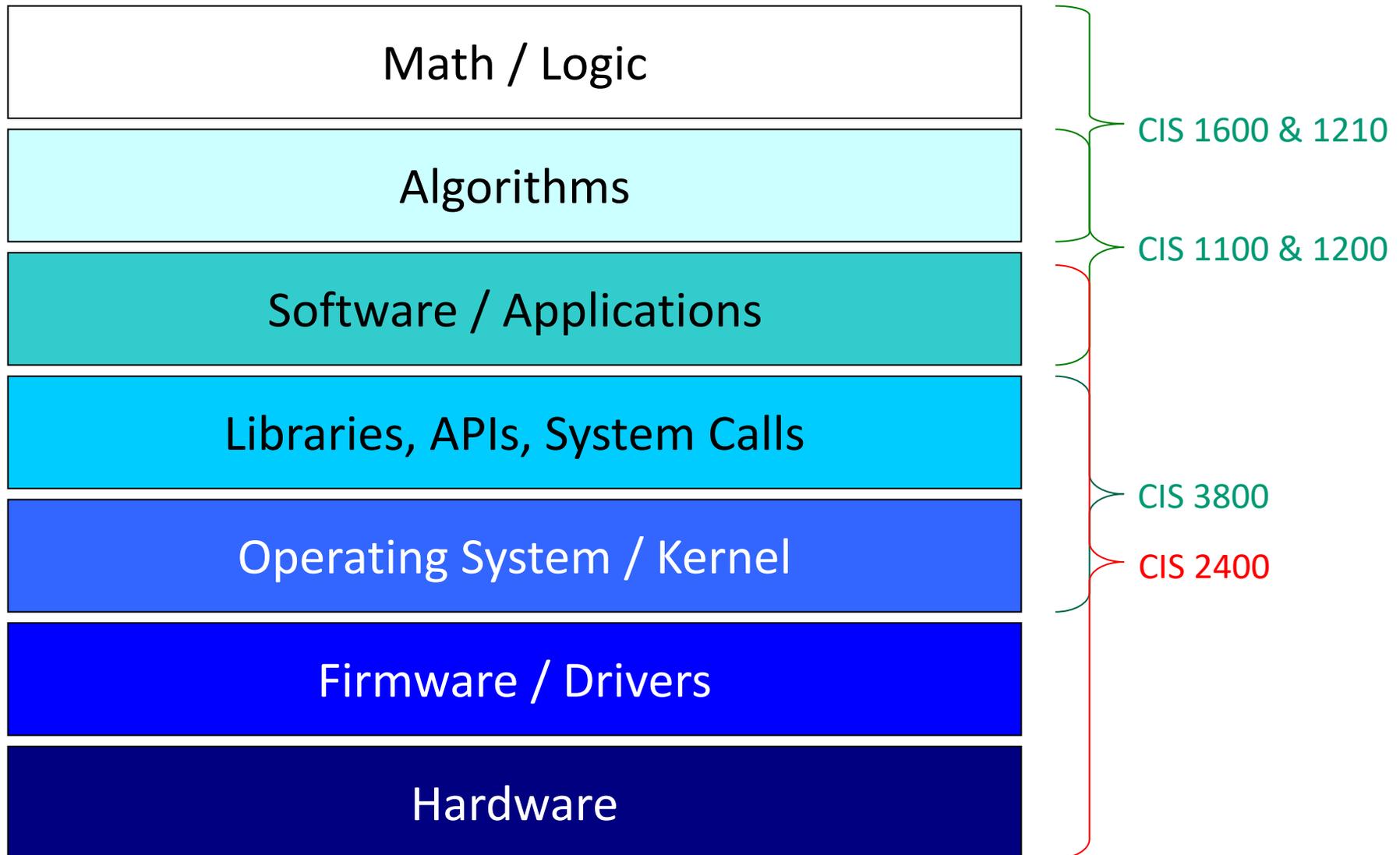
Machine-runnable code

Course Goals: Computers are Physical

❖ Go from:



Course Goals: Abstractions and Interfaces



Course Goal: Binary Representation

- ❖ All digital computer represent EVERYTHING with a combination of 0s and 1s.

- ❖ "Everything" includes:
 - Integers
 - Floating point numbers
 - Characters
 - Code (instructions)
 - Pointers
 - File contents, Images, Audio, Videos, etc.

Course Goal: Memory

- ❖ Data must be stored somewhere on a computer, and that place is usually in memory

- ❖ How does a computer organize different types of data?
 - Dynamic data (Heap)
 - Local Data (Stack)
 - Globals (Global segment of memory)

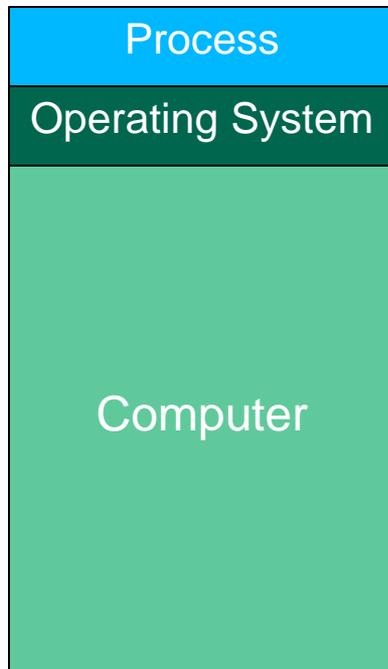
Why do these themes matter?

- ❖ Helps Programmers & Engineers develop a better mental model of how a computer works
- ❖ Understanding how your code translates & runs on hardware may allow you to write better code
- ❖ Many of the logic/themes/approaches to solving problems in this course apply to many other concepts

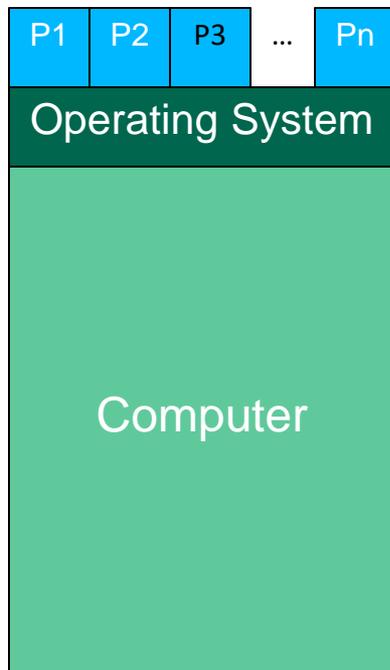
Course Goals: Survey of Computer Systems

- ❖ There is a lot more detail to all of the course topics than what we could cover.
- ❖ There are many topics that we did not really touch on at all
- ❖ This course is designed to be a survey, an introduction. There are many other courses that build on top of what we introduced in this course.

Missing Details



Missing Details: Multiple Processes



Missing Details: Networks



Future Courses

❖ Some Courses Related to CIS 2400:

- CIS 4710/5710
- CIS 3800/5480
- CIS 5050
- CIT 5950
- CIS 3410
- CIS 1910
- CIS 3310
- CIS 5530
- CIS 4410
- CIS 4600
- CIS 1900

CIS 4710/710: Computer Org and Design

- ❖ Pre-requisites: CIS 2400
- ❖ Direct continuation on CIS 2400's discussion of hardware
 - Instruction set design
 - Implementing ALU components
 - Building on top of the single-cycle processor design
- ❖ Uses Verilog (A Hardware Descriptive Language HDL)

CIS 3800/5480: Intro to Operating Systems

- ❖ Pre-requisites: CIS 2400
- ❖ How does a computer manage multiple programs running on it at the same time? How do you share the computer's resources?
 - Must do this while being efficient, portable, “fair”, etc.
- ❖ C programming (and lots of it)
 - Projects come with a LOT of code to build on top of
 - A lot goes into designing what you will do

CIS 5050: Software Systems

- ❖ Pre-requisite: OS & Networking Experience
- ❖ How do you get large collections of computers to collaborate (correctly)?
 - Reliably, efficiently, to scale, with high availability
 - Fundamentally must deal with concurrency between nodes
- ❖ C or C++ programming

CIT 5950

- ❖ Pre-requisite: CIT 5930 (very similar to 2400)
- ❖ "Intermediate" course between 2400 and OS Courses
 - Reinforce C programming experience & introduces C++
 - Goes over many OS concepts (processes, threads, networks, etc.)
- ❖ C and C++ programming
- ❖ I'm teaching it 😊
- ❖ Will have to use waitlist to register
- ❖ May not count for anything credit-wise

CIS 3410: Compilers & Interpreters

- ❖ Pre-requisites: Two of CIS 1200, 1210, 2400
- ❖ How does a compiler work?
 - Scan and parse source code, generate a symbol tree, check semantics, optimize, and output assembly
- ❖ OCAML programming
 - Project creates a working compiler
- ❖ Theory meets programming meets systems
 - Will always be relevant as new languages and new architectures arise

```

1      .data
2 DisplayOverloading2$$$$:      .quad 0
3      .quad DisplayOverloading2$disp$
4      .quad DisplayOverloading2$disp$
5
6 Overloading$$$$:      .quad 0
7      .quad Overloading$run$0$$$
8
9 DisplayOverloading3$$$$:      .quad D
10     .quad DisplayOverloading2$disp$
11     .quad DisplayOverloading2$disp$
12     .quad DisplayOverloading3$disp$
13     .quad DisplayOverloading3$disp$
14
15     .text
16 DisplayOverloading2$disp$0$$$:
17     pushq   %rbp
    
```

CIS 1910: Using Unix and Linux

- ❖ Pre-requisite: CIS 1100
- ❖ Mini-course to teach you how to use the terminal and many of the tools associated with it
- ❖ Will likely cover some slight OS/Systems level concepts
- ❖ Very useful to learn the terminal (in my opinion)

Networks Courses

- ❖ CIS 3310: Introduction to Networks and Security
 - Pre-requisite: CIS 1600 and CIS 2400

- ❖ CIS 553: Networked Systems
 - Pre-requisite: CIS 1210

- ❖ Both introduced networking and the various abstraction layers & protocols that support it.

- ❖ At a glance:
 - 3310 has a lot more discussion on security
 - 553 is more focused on the network and its details

Courses in C/C++

- ❖ CIS 4410: Embedded Software for Life-Critical Applications
 - How to interact with computers with limited resources (*e.g.*, RAM) and “real time” requirements
 - Usually in C
- ❖ CIS 1900: C++ Programming
 - Min-course to teach people C++ programming.
- ❖ CIS 4600: Computer Graphics
 - Theory- and coding-heavy course on creating digital art
 - Graphics almost always use C++

Thanks for a great semester!

- ❖ Special thanks to all the instructors before me (Both at UPenn and UW) who have influenced me to make the course what it is



CJ Taylor



Tom Farmer

- ❖ Huge thanks to the course TA's!



Ali



Andrew



Audrey



Craig



Daniel



David



Eddy



Ernest



Janavi



Jason



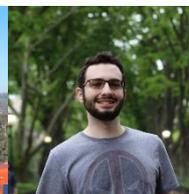
Kat



Kyrie



Mohammed



Noam



Patrick



Patricia



Sarah



Sofia

Not Pictured: Anisha, Heyi & Ria

Thanks for a great semester!

- ❖ Thanks to you!
 - It has been another tough semester. Still not completely out of the pandemic, Zoom fatigue, faltering motivation, etc
 - First time instructor for this course. Many assignments and infrastructure are recently developed.
 - You've made it through so far, be proud that you've made it and what you've accomplished!

- ❖ **Please take care of yourselves, your friends, and your community – a lot problems still remain and we all need to be a part of the solution**