ECE327 - Introduction to Software Engineering – January 22, 2025

Lecture 1 – Overview, Logistics, & Digital Logic

BU Department of Electrical & Computer Engineering

BOSTON UNIVERSITY

Ed Solovey – Associate Professor of the Practice - esolovey@bu.edu

- MIT ‘01 – Computer Science, Bachelor’s of Engineering
- MIT ‘03 – Computer Science, Master’s of Engineering – Theory of Computation
- Twenty+ Years of Software Experience

- Oracle
- Adobe
- Brightcove
- Crashlytics
- Twitter
- Google
- Digits

Staff

Graduate TA
Bruce Jia - brucejia@bu.edu

Graduate TA
Jueqi Wang - jueqiw@bu.edu

Undergraduate TA
Adrian Pawlowski - ajp12@bu.edu

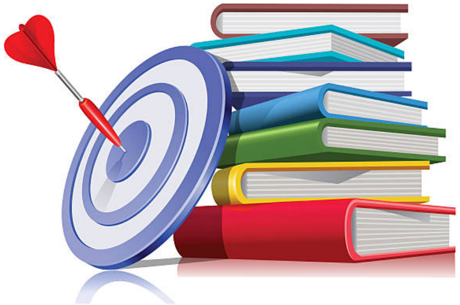
Undergraduate TA
Erika Hammond - eth@bu.edu

Undergraduate TA
Eric Chen - chene@bu.edu

Undergraduate TA
James Knee - jknee@bu.edu

Learning Objectives

- You will build a foundational, top to bottom, understanding of what it takes to execute a program on a modern day computer.
- You will be confident in your ability to write and analyze programs at various levels of abstraction.
- The depth of your comprehension will enable you to make trade offs on concepts ranging from low level memory management to polymorphism and inheritance.
- You will gain a baseline understanding of core data structures and algorithm performance analysis.
- You will appreciate the benefits and challenges of collaborating on software projects with others.



Why is covering this material relevant?

Business

Heartbleed bug: RCMP asked Revenue Canada to delay news of SIN thefts

Tax agency waited until Monday to reveal that Heartbleed bug led to 900 social insurance numbers being stolen

Pete Evans · CBC News · Posted: Apr 14, 2014 8:40 AM EDT | Last Updated: April 14, 2014

- **Heartbleed**
  - A bug in the OpenSSL library implementation of the Transport Layer Security (TLS) protocol.
  - TLS protocol is used to secure HTTPS traffic - HTTP with encryption enabled.
  - Root cause: buffer-overread - a low level bug, where more data could be read than should have been permissible.

Why is covering this material relevant?

Facebook outage: what went wrong and why did it take so long to fix after social platform went down?

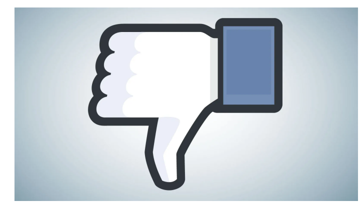
Billions of users were unable to access Facebook, Instagram and WhatsApp for hours while the social media giant scrambled to restore services

Facebook, Instagram and WhatsApp all went down, and reappeared online after a six-hour global outage. Photograph: Anadolu Agency/Getty Images

Facebook and its other platforms, including Instagram, WhatsApp and Messenger, went down globally for close to six hours on Monday and Tuesday, depending on your time zone. As services are being restored, questions are being asked about what caused the outage, and why it took so long to fix.

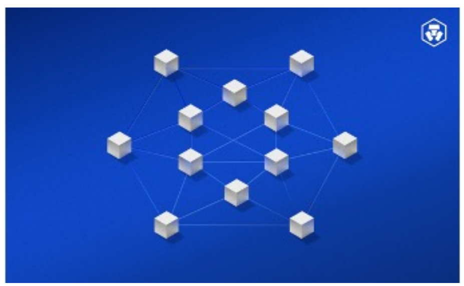
Facebook 2021 Outage

- Caused by configuration change to routers.
- Facebook's own internal systems run on same infrastructure, so their internal communications were impacted, adding to time-to-recovery



- Lessons Learned:
  - Fault Isolation is important
  - Fully understanding risks for each type of change
- **We will cover these ideas in this class too!**

All software systems have problems, but …



they have also enabled some amazing things!

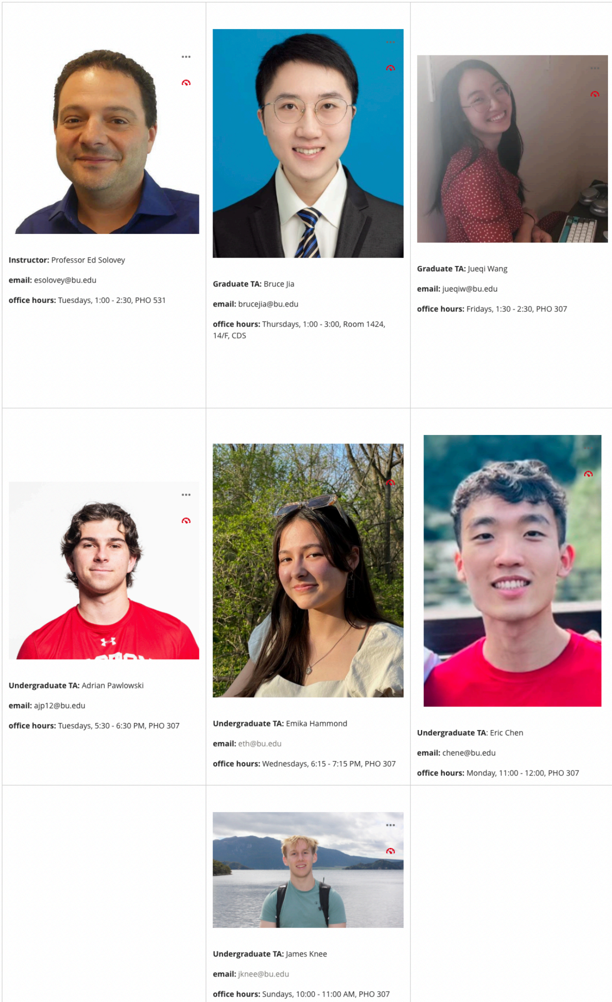
Instruction Plan

- **Lectures**
  - prepare by doing readings in **Horstmann** book - chapters assigned in Blackboard
  - made more interactive by PollEverywhere questions with opportunity for you to explain your answers to others
- **Lab Sessions and Assignments**
  - exercise and internalize material covered in lectures and book
  - TA support during your assigned lab sessions
  - submitted via Gradescope and auto-grader feedback provided
  - unlimited submissions up to deadline
  - no submissions post deadline
  - single, lowest lab grade dropped

Instruction Plan (cont)

- **Homework Assignments**
  - similar topics as lab assignments
  - submitted via Gradescope with no/limited auto-grader feedback - up to your to write test cases to gain confidence in your submissions
  - unlimited submissions up to deadline
  - no submissions post deadline
  - single, lowest homework grade dropped
- **Office Hours**
  - please come early and often with your questions!

Office Hours



- Please consult **Course Logistics** document in Blackboard. Our goal is to make ourselves available evenly through out the week.

Evaluation

- **10%** - in class responses to interactive PollEverywhere questions
- **15%** - lab assignments
- **20%** - homework assignments
- **20%** - closed book midterm
- **15%** - team based final project (completed during second half of the class)
- **20%** - closed book final exam

Interactive Questions

Poll Everywhere

https://pollev.com/edsolovey024

- **Goals:**
  - Make lectures more interactive - challenging with class of this size
  - Gives all of you a chance to participate more
  - Gives those interested a chance to share their thoughts with the class
- **Details:**
  - 2 or 3 times per lecture, 2 or 3 questions each time
  - All questions multiple choice
  - Between 1 and 2 minutes per question
  - All questions on subject matter of the lecture that they are a part of

Interactive Questions

Poll Everywhere

https://pollev.com/edsolovey024

- **Grading:**
  - Total of 100 questions throughout the course
  - Each correct answer is 1 point
  - You start with 25 points
  - No extra credit for going over 100 points
  - 10% of your final grade will be based on
    - (25 + # of questions you got correct) / 100
- **Not Part of Grading:**
  - Those who are interested, can come up and explain their correct answer
- **Will do some later today!**

Class Communication

- If you have not yet, please join our Slack workspace via this link!
- Install the desktop/mobile app for best experience.

High Level Tenets…

- Growth Mindset
- Feedback
- Inclusivity, Belonging, & Safety
- Tools for Evaluating Trade Offs

Growth Mindset

- Growth mindset: belief that abilities, intelligence, and talent can be **developed** through:
  - effort
  - learning
  - perseverance in face of setbacks
  - seeking feedback
  - asking questions
- Fixed mindset:
  - abilities
  - intelligence
  - and talent
    - **are all static.**

Growth Mindset

- I am fully bought into this and will foster this environment in the classroom.
- The best software companies also believe in this and have designed parts of their interview process to evaluate the growth mindset of potential candidates.

Ask Questions!

- We are going to cover a lot of ground in this class.
- You are likely not to be familiar with many of the topics… **yet**!!
- Please ask questions!
- No such thing as a bad question.
- **Asking a question is not an admission of defeat!**
- Admitting that something is not clear to you (**yet**),and asking a question about it is a huge part of having a growth mindset.
- Others (including myself) will benefit from you asking questions.

Ask Questions!

- In ALL venues:
  - these lectures
  - in your lab sessions
  - in office hours
  - in Slack

Feedback

- Feedback is an indispensable part of the Growth Mindset.
- A prerequisite for constant growth is identifying growth areas; feedback is a great way to do so.
- Not surprisingly, the best software companies interview for this trait as well.

Feedback

- Two types of feedback:
  - Reenforcement
  - Constructive
- Examples:
  - "I appreciate that you took the time to explain the root cause of that production incident to me, rather than just glossing over the complex pieces."
  - "In the future, it would be really helpful, if instead of giving me the answers, you took the time to help me figure out how to arrive at them myself."

Feedback

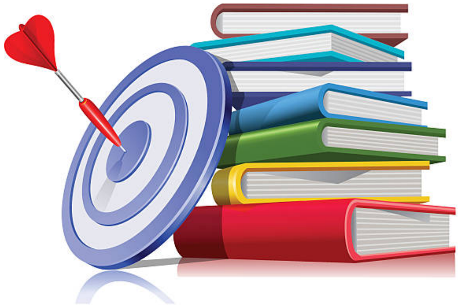
- I am open to any feedback at any time and will always be grateful for it.
  - Please don't hold back!
- Hopefully, this environment and class helps you grow an appreciation for it as well.
- You'll have weekly opportunity to provide that feedback
  - should take at most a few minutes
  - will get you an extra point on your PollEverywhere total
  - will allow me to make adjustments and make the class better for you

Inclusivity, Belonging, & Safety

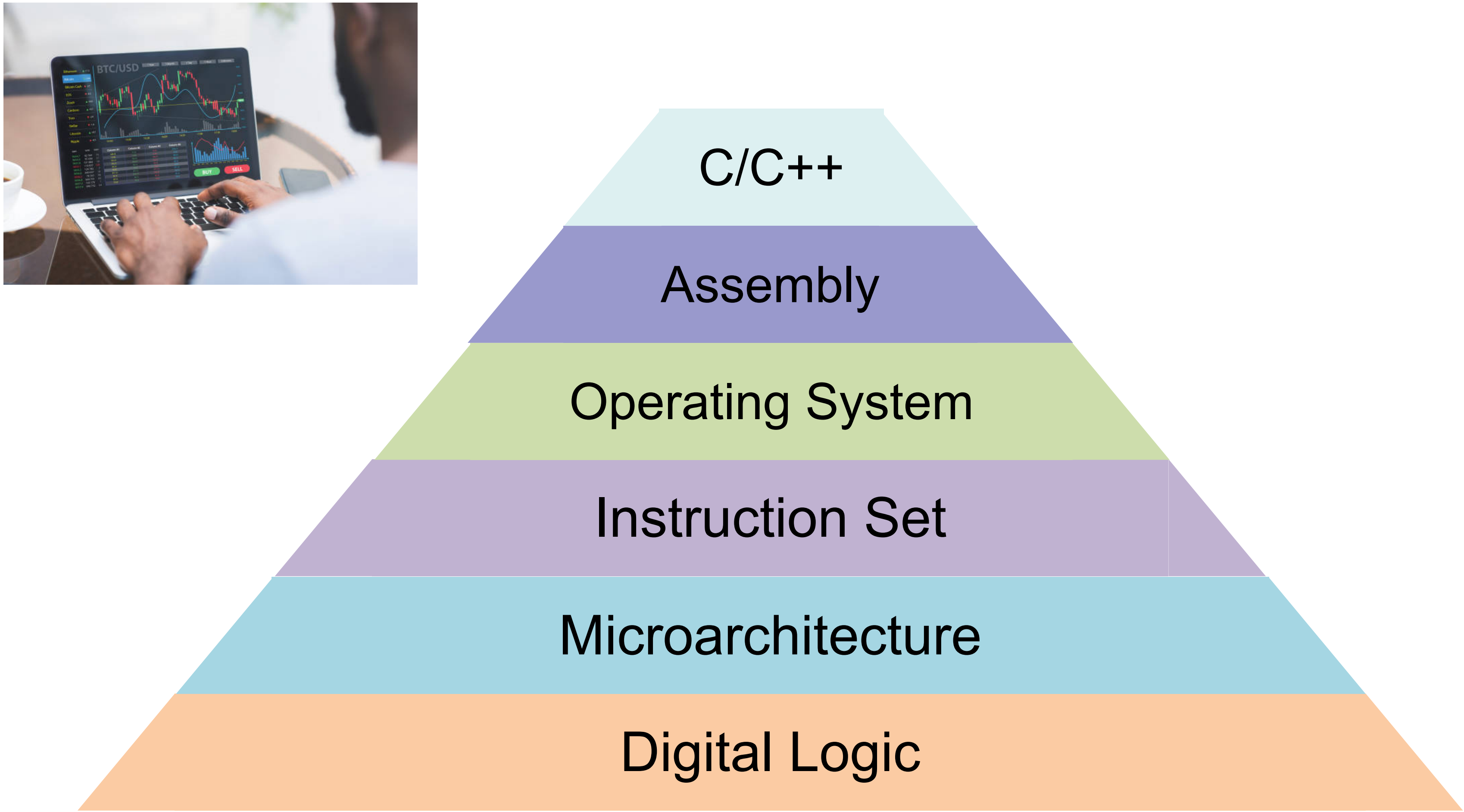
- Much like the Growth Mindset, Inclusivity is a tenet that is
  - valued and assessed during interviews in industry
  - is super important to me
  - allows groups to reach their full potential
- Goals
  - make everyone feel comfortable, confident, and supported in voicing their opinion
  - create an environment where it is clear that all of us are on the same team

Learning Objectives for Lecture 1

- You will be exposed to the six level computer architecture
- You will gain an appreciation for how each level of the six level architecture builds on the previous ones
- You will understand the components of the Digital Logic level - binary data storage and core logic gates.
- You will become comfortable manipulating binary numbers.
- You will become comfortable reasoning through digital logic circuits.

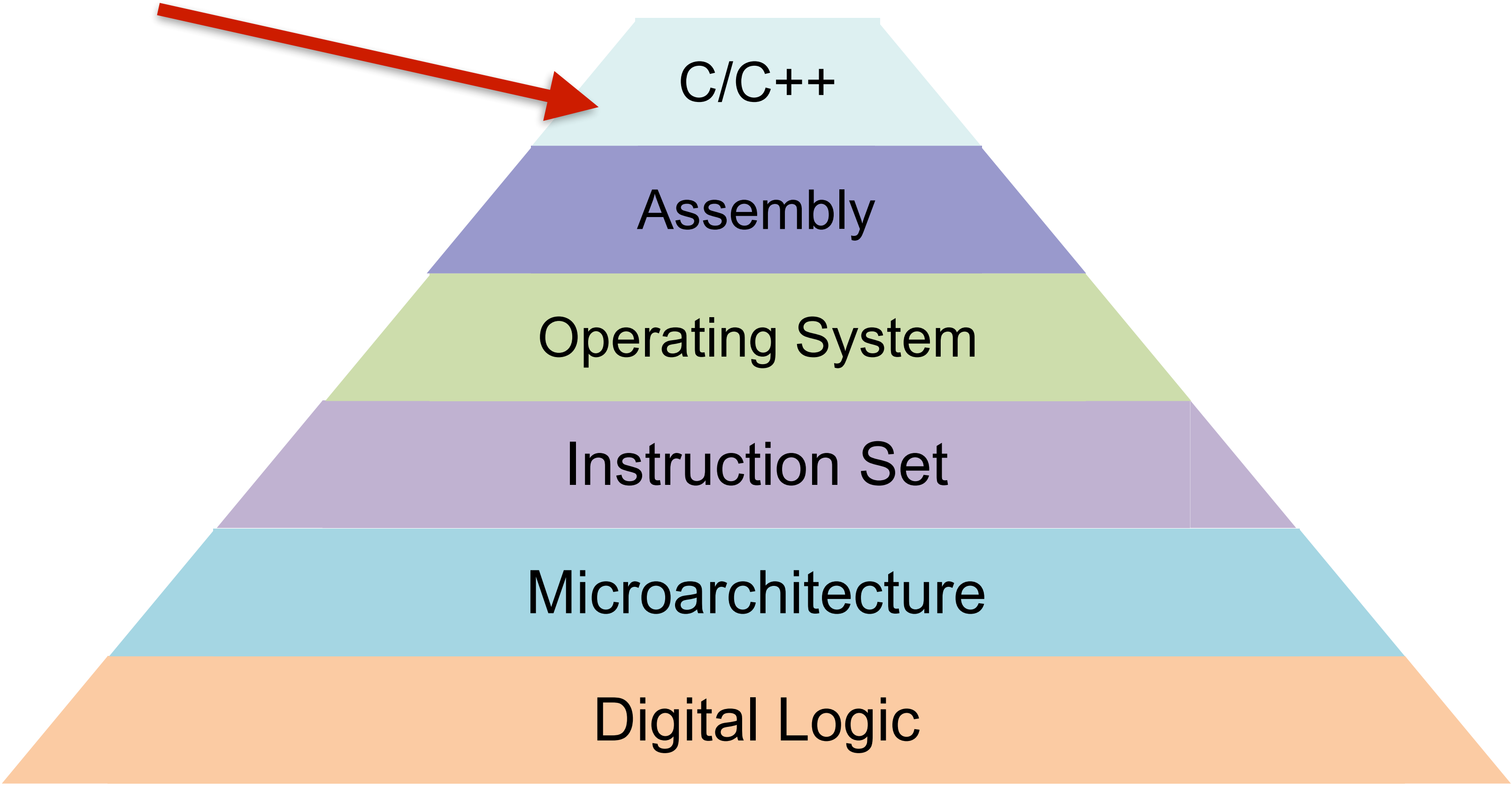


Six Level Computer Architecture

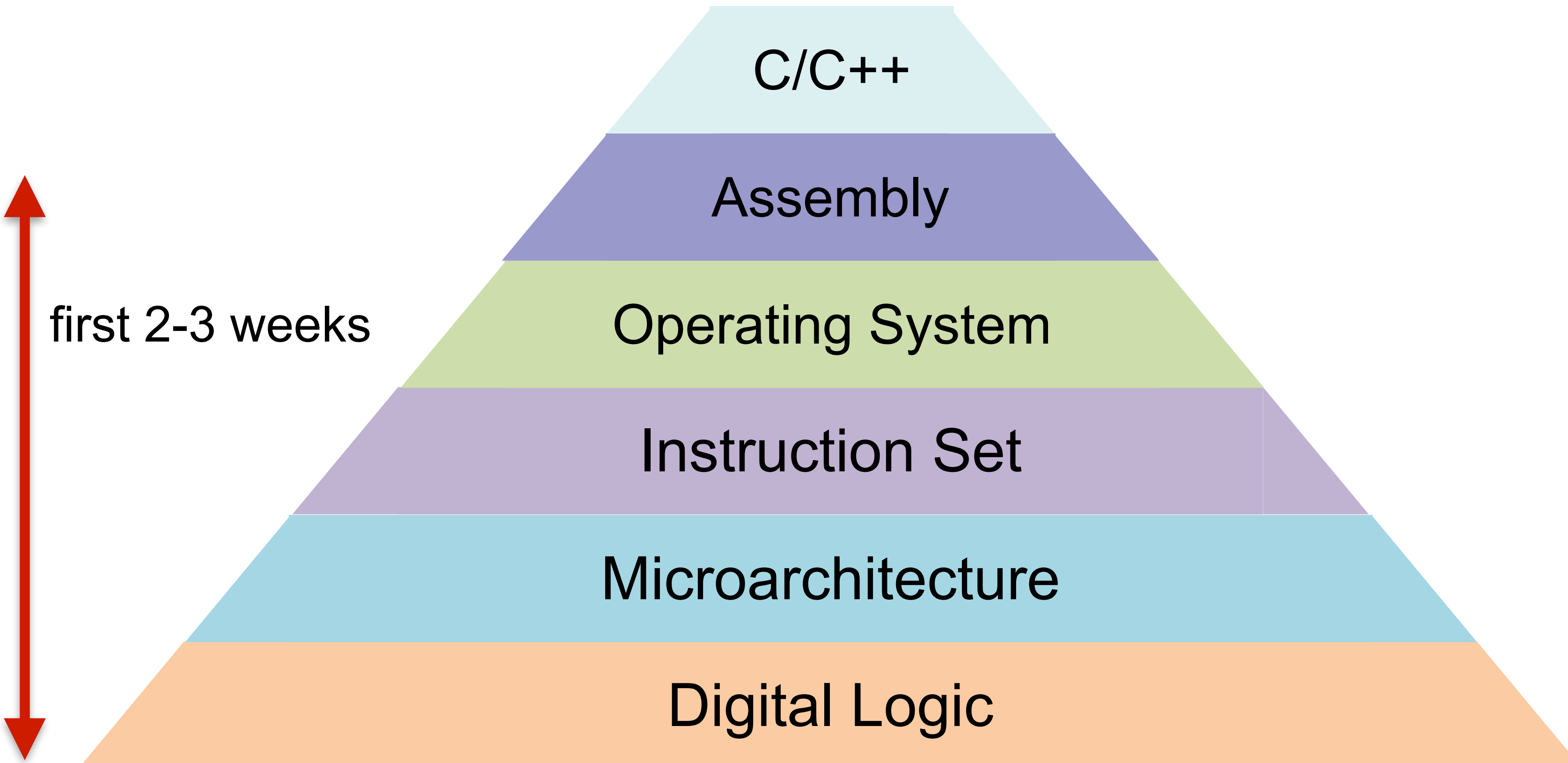


Six Level Computer Architecture

primary focus of the class

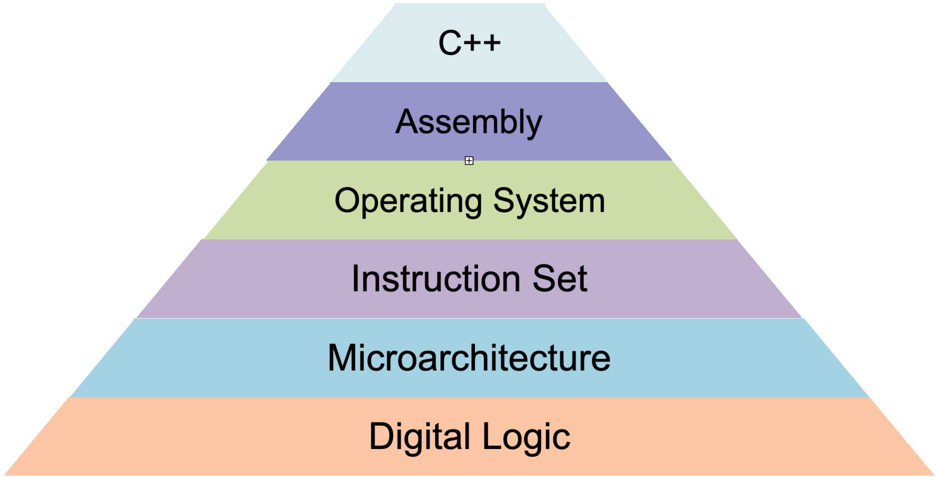


Six Level Computer Architecture



Six Level Computer Architecture

- **Why spend 2-3 weeks at the base of this pyramid if the primary focus of the class is the top?**
- Getting an understanding of the base levels will
  - help you understand the motivation for some of the design decisions at the top
  - give you an appreciation of engineer productivity gains at each level
  - provide you with history and background that will help you reason through relevant conversations

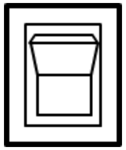


Digital Logic - Transistor Building Block

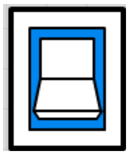
A transistor is a semiconductor device that acts as a switch or an amplifier for electrical signals. It has three terminals: the **source** (or emitter), **drain** (or collector), and **gate** (or base). By applying a small voltage to the gate, the transistor controls the flow of current between the source and drain. This on/off switching behavior is the basis for binary logic (1s and 0s).

Digital Logic

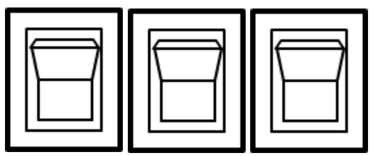
Digital Logic - Binary



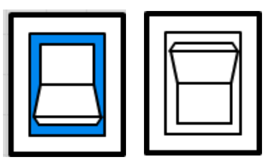
switch / transistor / bit in the **off** position : **0**



switch / transistor / bit in the **on** position : **1**



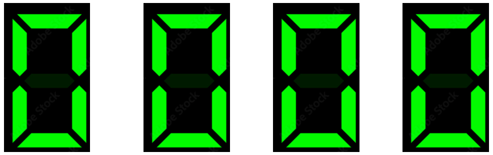
multiple switches/bits next to each other represent a multi (binary) digit number



what number does this represent?

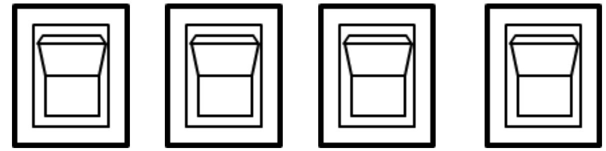
Digital Logic

Digital Logic - Binary



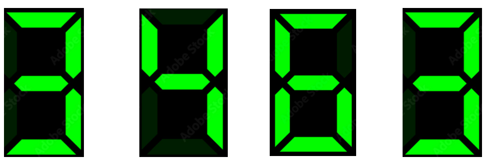
10^3 10^2 10^1 10^0

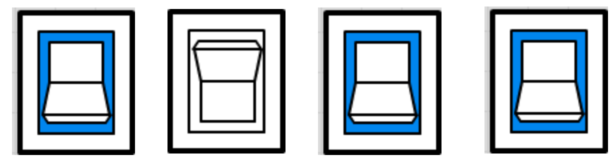
decimal - base 10



2^3 2^2 2^1 2^0

binary - base 2





1 0 1 1

$3 * 10^3 +$
$4 * 10^2 +$
$6 * 10^1 +$
$3 * 10^0$
$= 3000 + 400 + 60 + 3 =$
3463

$1 * 2^3 +$
$0 * 2^2 +$
$1 * 2^1 +$
$1 * 2^0 =$
$8 + 0 + 2 + 1 = 11$

Digital Logic

Digital Logic - Number System Conventions

- Use subscript to denote number base:
  - 111_{10} is a base 10 (decimal number)
  - 111_{2} is a base 2 (binary number)
  - without subscript, they would look the same but are very different!
    - 111_{10} and 7_{10}
- Group binary bits into fours:
  - $1001\ 0011_{2}$ is 147_{10}
  - Each group of 4 is single Hexadecimal (base 16) digit
  - 8 bits, 2 groups of 4, is a Byte

Digital Logic

Digital Logic - Binary Examples

1111_2	15_{10}
$0001\ 1101_2$	29_{10}
$1000\ 0001_2$	129_{10}
$0011\ 0011_2$	51_{10}

Digital Logic

Interactive Questions Setup

- Go to https://pollev.com/edsolovey024
- When asked to **Register for Credit** choose **Skip for Now**
- When prompted to enter a screen name, create one according to the template below:
- **<first-name>_<last-name>-n/y**
- that is
    - your first name
    - an underscore
    - your last name
        - **-y** if you are excited to share an explanation of your correct responses with the class
        - **-n** if you would rather not be called on to share an explanation of your correct responses with the class
    - choosing **-y** or **-n** will not impact your grade
- so user names should look like **Ed_Solovey-y** or **Ed_Solovey-n**

Interactive Questions

Poll Everywhere

https://pollev.com/edsolovey024

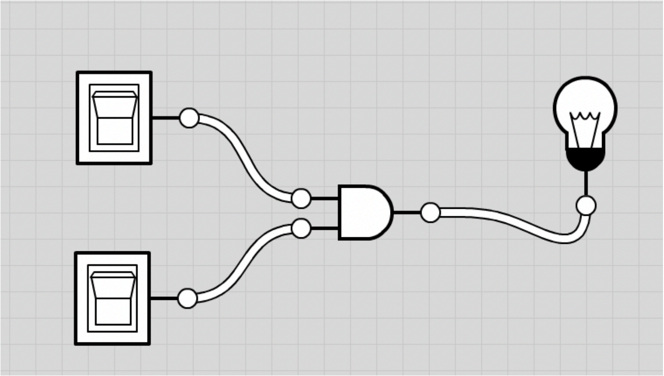
questions 1 & 2

Digital Logic

Six Level Computer Architecture

- In addition to storing binary numbers, the Digital Logic layer consists of Logic Gates.
- Logic gates perform simple, low-level operations on binary data (bits) and are the foundation for more complex computational tasks in higher levels of the architecture.

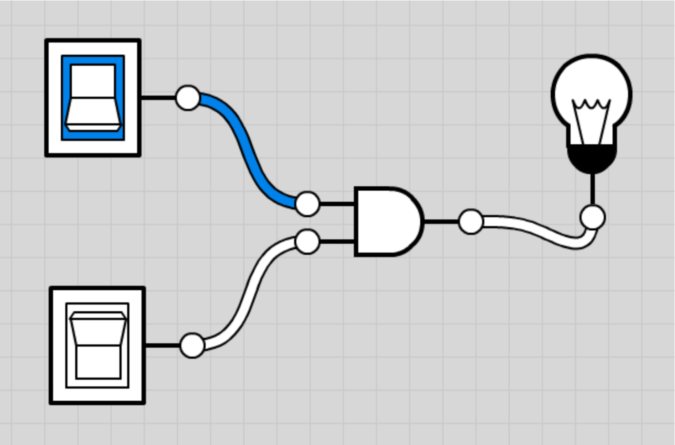
And Gate



- The light bulb turns on only when both of the input bits are on (are 1).
- The light bulb here is itself a bit that is either 0 (off) or 1 (on).
- Both inputs are off - the lightbulb is off.

Digital Logic

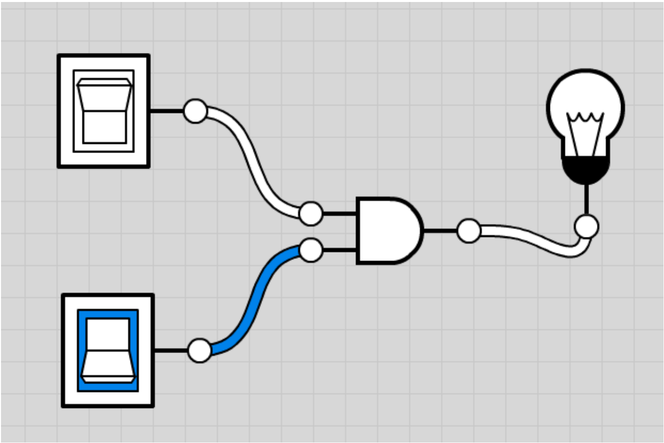
And Gate



- One input is on, but the lightbulb is still off…

Digital Logic

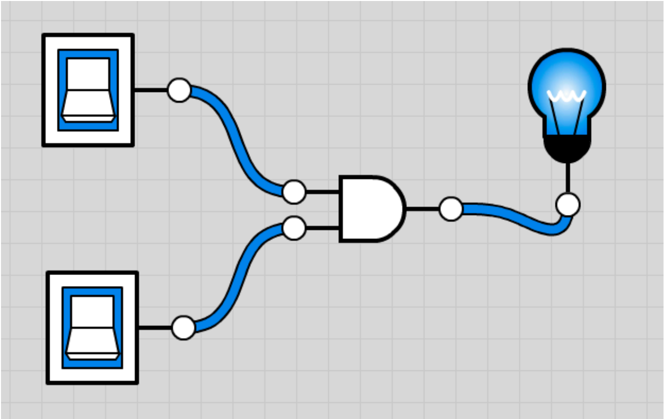
And Gate



- The other input is on, but the lightbulb is still off…

Digital Logic

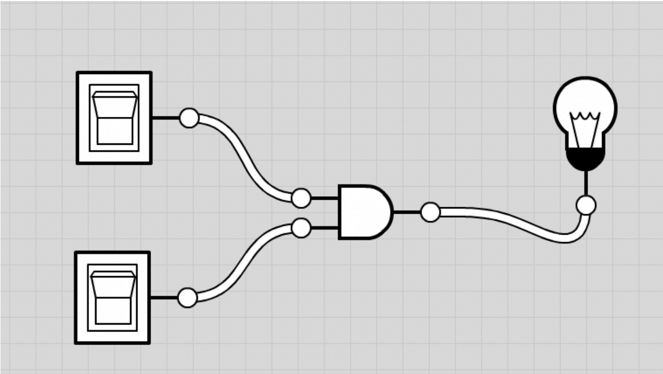
And Gate



- Both inputs are on - the lightbulb is on!

Digital Logic

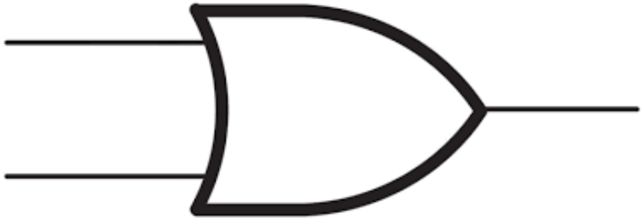
And Gate Truth Table

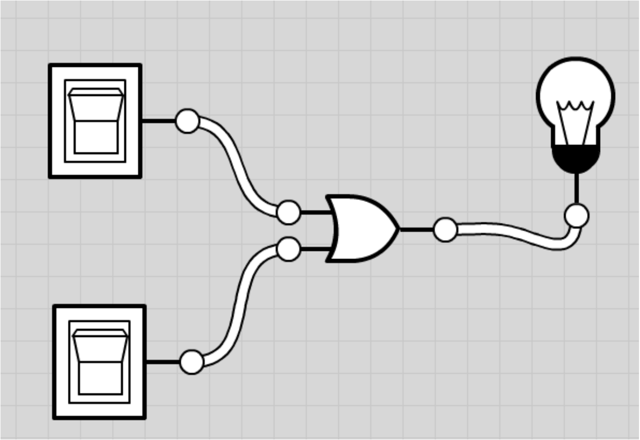


input 1	input 2	output
0	0	0
0	1	0
1	0	0
1	1	1

Digital Logic

Or Gate

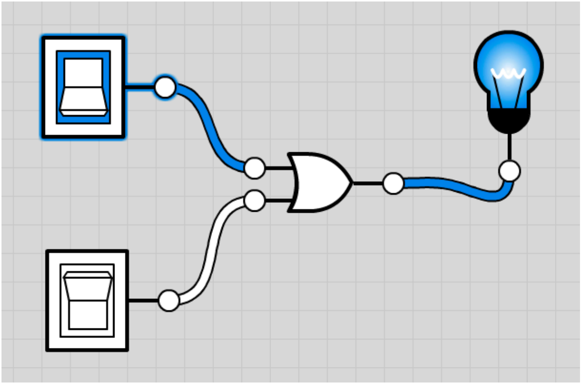




- The light bulb turns on only when **either or both** of the input bits are on (are 1).
- The light bulb here is itself a bit that is either 0 (off) or 1 (on).
- Both inputs are off - the lightbulb is off.

Digital Logic

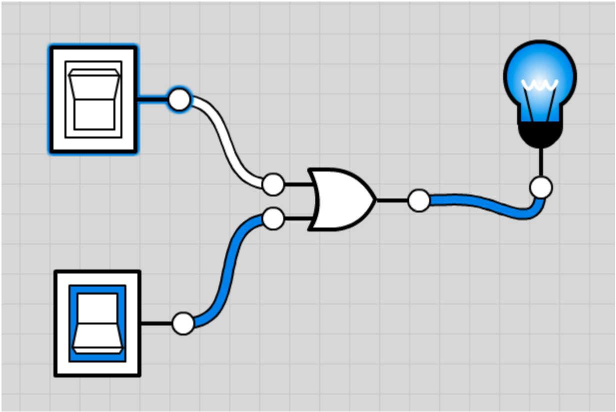
Or Gate



- One input is on - the lightbulb is on!

Digital Logic

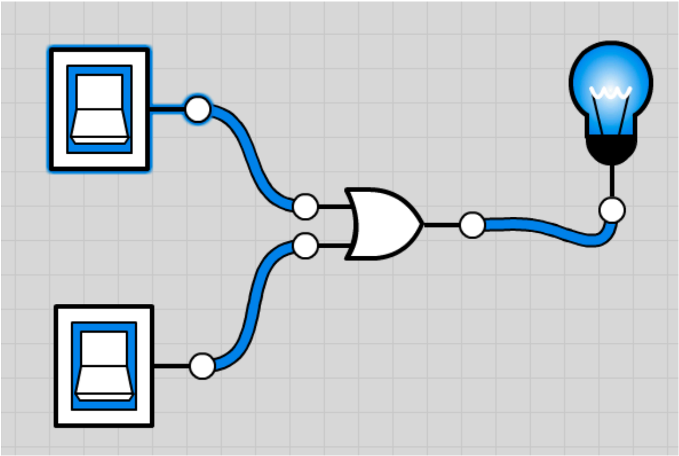
Or Gate



- The other input is on - the lightbulb is on!

Digital Logic

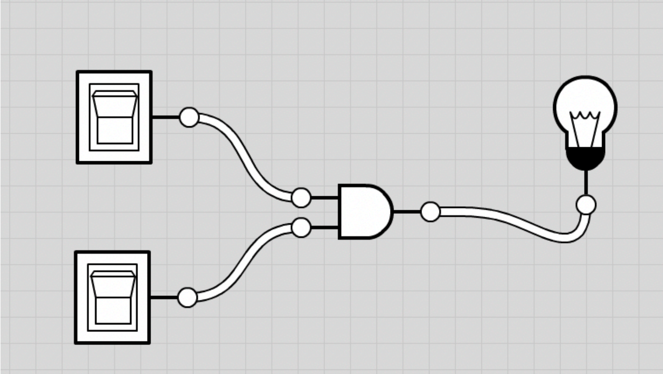
Or Gate



- Both inputs are on - the lightbulb is on!

Digital Logic

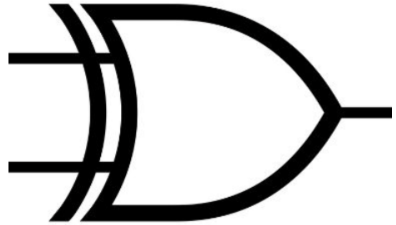
Or Gate Truth Table

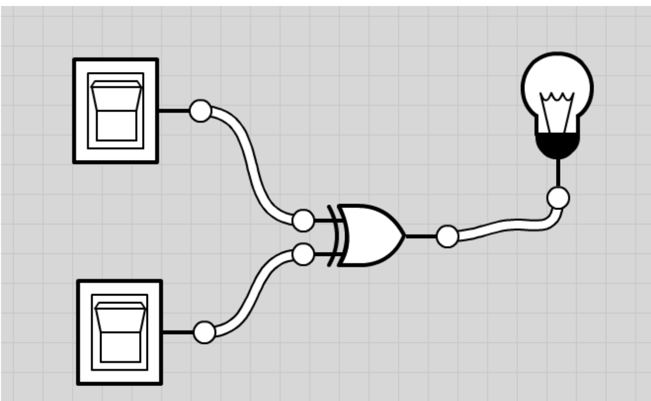


input 1	input 2	output
0	0	0
0	1	1
1	0	1
1	1	1

Digital Logic

XOR (exclusive or) Gate

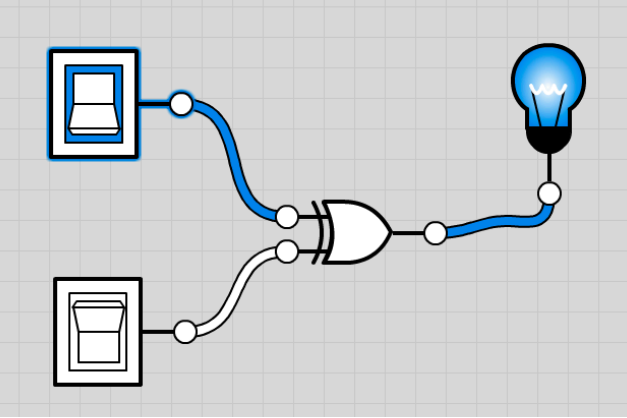




- The light bulb turns on only when **either, but not both,** of the input bits are on (are 1).
- The light bulb here is itself a bit that is either 0 (off) or 1 (on).
- Both inputs are off - the lightbulb is off.

Digital Logic

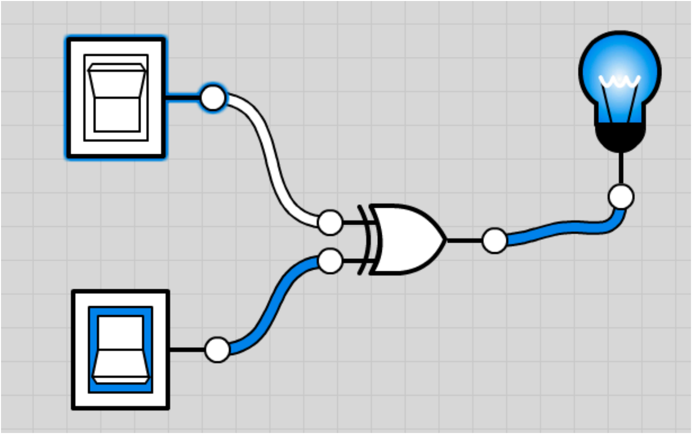
Or Gate



- One input is on - the lightbulb is on!

Digital Logic

Or Gate

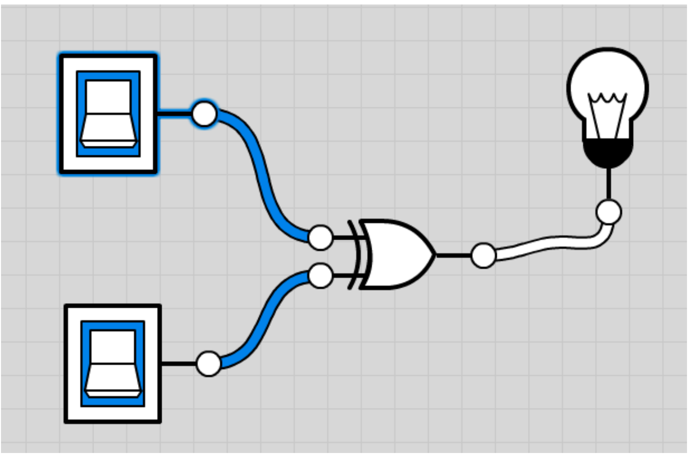


- The other input is on - the lightbulb is on!

Digital Logic

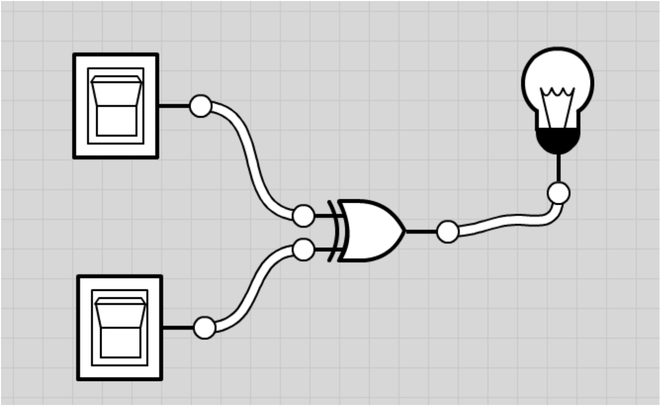
Or Gate

- Both inputs are on - the lightbulb is off!



Digital Logic

XOR Gate Truth Table



input 1	input 2	output
0	0	0
0	1	1
1	0	1
1	1	0

Digital Logic

One Digit Binary Math

$$\begin{array}{r} 0 \\ +\ 0 \\ \hline 0 \end{array} \qquad \begin{array}{r} 1 \\ +\ 0 \\ \hline 1 \end{array} \qquad \begin{array}{r} 0 \\ +\ 1 \\ \hline 1 \end{array} \qquad \begin{array}{r} 1 \\ +\ 1 \\ \hline 1\ 0 \end{array}$$

- Can we design a circuit, using an XOR and an AND gate to perform addition on two bits??
  - two inputs
  - and two outputs (one for each output digit)

Digital Logic

logic.ly Live Demo

Digital Logic

Lecture-1-AddingBits.logicly

Interactive Questions

Poll Everywhere

https://pollev.com/edsolovey024

questions 3 & 4

Digital Logic

Binary Numbers 0 - 15

0	0	0	0	0	1	0	0	0	8
0	0	0	1	1	1	0	0	1	9
0	0	1	0	2	1	0	1	0	10
0	0	1	1	3	1	0	1	1	11
0	1	0	0	4	1	1	0	0	12
0	1	0	1	5	1	1	0	1	13
0	1	1	0	6	1	1	1	0	14
0	1	1	1	7	1	1	1	1	15

Digital Logic

Binary Math

	0	1	0	1	5
+	1	0	0	1	9

Digital Logic

Binary Math

			1		
	0	1	0	1	5
+	1	0	0	1	9
				0	

Digital Logic

Binary Math

			1		
	0	1	0	1	5
+	1	0	0	1	9
			1	0	

Digital Logic

Binary Math

			1		
	0	1	0	1	5
+	1	0	0	1	9
		1	1	0	

Digital Logic

Binary Math

			1		
	0	1	0	1	5
+	1	0	0	1	9
	1	1	1	0	14

Digital Logic

Binary & Negative Numbers

0	0	0	0	0	1	0	0	0	8
0	0	0	1	1	1	0	0	1	9
0	0	1	0	2	1	0	1	0	10
0	0	1	1	3	1	0	1	1	11
0	1	0	0	4	1	1	0	0	12
0	1	0	1	5	1	1	0	1	13
0	1	1	0	6	1	1	1	0	14
0	1	1	1	7	1	1	1	1	15

- 0-15 with 4 bits is great!
- Addition works!
- **But how would you represent a negative number with binary ??**

Digital Logic

One's Complement

- First widely used attempt to represent negative numbers.
- Rules:
  - when leading bit is 0, the rest of the number is treated as positive
  - when leading bit is 1, the number is positive and equal to
    - negative value of leading bit + positive value of other bits + 1

1	0	0	0	-8 + 0 + 1 = -7
1	0	0	1	-8 + 1 + 1 = -6
1	0	1	0	-8 + 2 + 1 = -5
1	0	1	1	-8 + 3 + 1 = -4

Digital Logic

One's Complement

- One's complement negative representation of a number can be arrived at by taking the positive number and flipping all of its bits

0 1 0 1 5

Digital Logic

One's Complement

- One's complement negative representation of a number can be arrived at by taking the positive number and flipping all of its bits

0	1	0	1	5

flipped bits:

1	0	1	0	-8 + 2 + 1 = -5

Digital Logic

One's Complement

- One's complement negative representation of a number can be arrived at by taking the positive number and flipping all of its bits
- Called one's complement because adding a positive number and its negative representation results in all 1's

0	1	0	1	5
1	0	1	0	-8 + 2 + 1 = -5
1	1	1	1	-8 + 7 + 1 = 0

Digital Logic

One's Complement

Bits	Unsigned value	Ones' complement value
000	0	0
001	1	1
010	2	2
011	3	3
100	4	–3
101	5	–2
110	6	–1
111	7	–0

- Note two different ways to represent zero…
- There is one more imperfection that led engineers to look for other ways to represent negative numbers in binary…

Digital Logic

One's Complement

- Let's add -3 and -4

0	0	1	1	3	start with 3

Digital Logic

One's Complement

- Let's add -3 and -4

0	0	1	1	3	
1	1	0	0	-3 = -8 + 4 + 1	flip bits to get -3

Digital Logic

One's Complement

- Let's add -3 and -4

0	0	1	1	3	
1	1	0	0	-3 = -8 + 4 + 1	
0	1	0	0	4	start with 4

Digital Logic

One's Complement

- Let's add -3 and -4

0	0	1	1	3	
1	1	0	0	-3 = -8 + 4 + 1	
0	1	0	0	4	
1	0	1	1	-4 = -8 + 3 + 1	flip bits to get -4

Digital Logic

One's Complement

- Let's add -3 and -4

	1	1	0	0	-3
+	1	0	1	1	-4

Digital Logic

One's Complement

- Let's add -3 and -4

	1	1	0	0	-3
+	1	0	1	1	-4
				1	

Digital Logic

One's Complement

- Let's add -3 and -4

	1	1	0	0	-3
+	1	0	1	1	-4
			1	1	

Digital Logic

One's Complement

- Let's add -3 and -4

	1	1	0	0	-3
+	1	0	1	1	-4
		1	1	1	

Digital Logic

One's Complement

- Let's add -3 and -4

	1	1	0	0	-3
+	1	0	1	1	-4
	0	1	1	1	

Digital Logic

One’s Complement

- Let’s add -3 and -4

	1	1	0	0	-3
+	1	0	1	1	-4
1	0	1	1	1	

- What do we do now? We want a 4 bit number not 5 …
- One’s complement rule is that a carried over fifth bit has to be carried around - added to the front of the result…

Digital Logic

One's Complement

- Let's carry that bit around

	1	1	0	0	-3
+	1	0	1	1	-4
1	0	1	1	1	

	0	1	1	1
+				1

Digital Logic

One's Complement

- Let's carry that bit around

	1	1	0	0	-3
+	1	0	1	1	-4
1	0	1	1	1	

			1	
	0	1	1	1
+				1
				0

Digital Logic

One's Complement

- Let's carry that bit around

	1	1	0	0	-3
+	1	0	1	1	-4
1	0	1	1	1	

		1		
	0	1	1	1
+				1
			0	0

Digital Logic

One's Complement

- Let's carry that bit around

	1	1	0	0	-3
+	1	0	1	1	-4
1	0	1	1	1	

	1			
	0	1	1	1
+				1
		0	0	0

Digital Logic

One's Complement

- Let's carry that bit around

	1	1	0	0	-3
+	1	0	1	1	-4
1	0	1	1	1	

	0	1	1	1	
+				1	
	1	0	0	0	-8 + 1 = -7

- Correct! But a little tricky to remember to carry that bit around to the front…

Digital Logic

One's Complement

- First widely used attempt to represent negative numbers.
- Rules:
  - when leading bit is 0, the rest of the number is treated as positive
  - when leading bit is 1, the number is negative and equal to:
    - negative value of leading bit + positive value of other bits + 1

1	0	0	0	-8 + 0 + 1 = -7
1	0	0	1	-8 + 1 + 1 = -6
1	0	1	0	-8 + 2 + 1 = -5
1	0	1	1	-8 + 3 + 1 = -4

Digital Logic

Two's Complement

- Improvement on one's complement
- Most widely used binary representation of negative numbers
- Rules:
  - when leading bit is 0, the rest of the number is treated as positive
  - when leading bit is 1, the number is negative and equal to
    - negative value of leading bit + positive value of other bits

1	0	0	0	-8 = -8
1	0	0	1	-8 + 1 = -7
1	0	1	0	-8 + 2 = -6
1	0	1	1	-8 + 3 = -5

Digital Logic

Two's Complement

- Two's complement negative representation of a number can be arrived at by taking the positive number and then:
  1. flipping all of its bits
  2. adding 1

0	1	0	1	5

Digital Logic

Two's Complement

- Two's complement negative representation of a number can be arrived at by taking the positive number, flipping all of its bits, and adding 1

0	1	0	1	5
flipped bits:				
1	0	1	0	-8 + 2 + 1 = -5

Digital Logic

Two's Complement

- Two's complement negative representation of a number can be arrived at by taking the positive number, flipping all of its bits, and adding 1

0	1	0	1	5
1	0	1	0	
added 1 :				
1	0	1	1	-8 + 3 = -5

Digital Logic

Two's Complement

Bits	Unsigned value	Signed value (Two's complement)
000	0	0
001	1	1
010	2	2
011	3	3
100	4	–4
101	5	–3
110	6	–2
111	7	–1

- Only one way to represent zero
- Addition works without special rules…

Digital Logic

Two's Complement

- Let's add -3 and -4

	1	1	0	1	-3
+	1	1	0	0	-4

Digital Logic

Two's Complement

- Let's add -3 and -4

	1	1	0	1	-3
+	1	1	0	0	-4
				1	

Digital Logic

Two's Complement

- Let's add -3 and -4

	1	1	0	1	-3
+	1	1	0	0	-4
			0	1	

Digital Logic

Two's Complement

- Let's add -3 and -4

	1				
	1	1	0	1	-3
+	1	1	0	0	-4
		0	0	1	

Digital Logic

Two's Complement

- Let's add -3 and -4

		1	1	0	1	-3
+		1	1	0	0	-4
	1	1	0	0	1	

but working with 4 bit numbers so ignore fifth bit to get:

1	0	0	1	-8 + 1 = -7

Digital Logic

Interactive Questions

Poll Everywhere

https://pollev.com/edsolovey024

questions 5 & 6

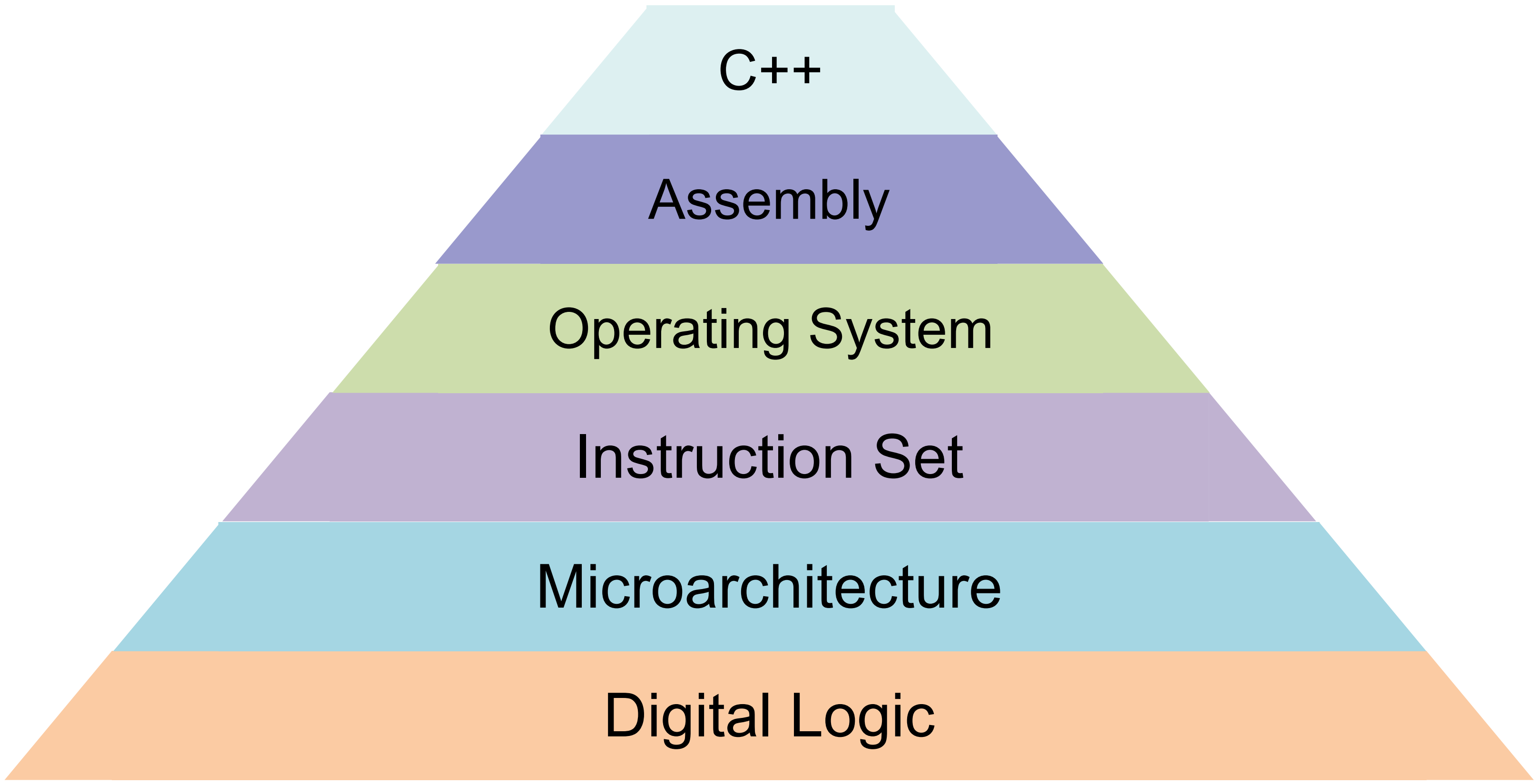
Digital Logic

logic.ly Live Demo

Digital Logic

Lecture-1-Memory.logicly

Six Level Computer Architecture



Reminders

- You'll complete lab-1 during your lab sessions this week
- Please read chapter 1 of Horstmann by Wednesday, 1/29