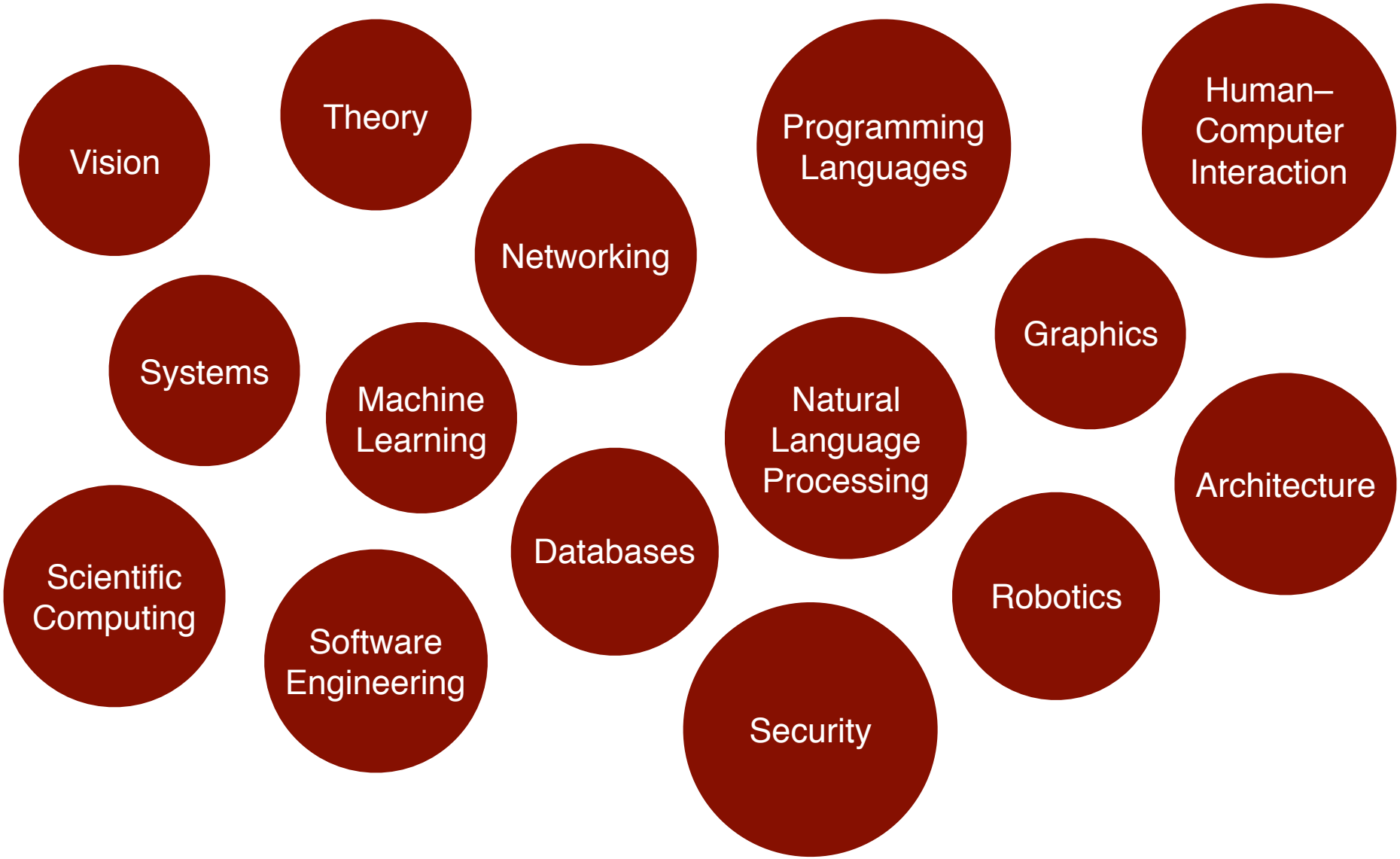
Lecture 0: Introduction to Computer Science

CS 51P

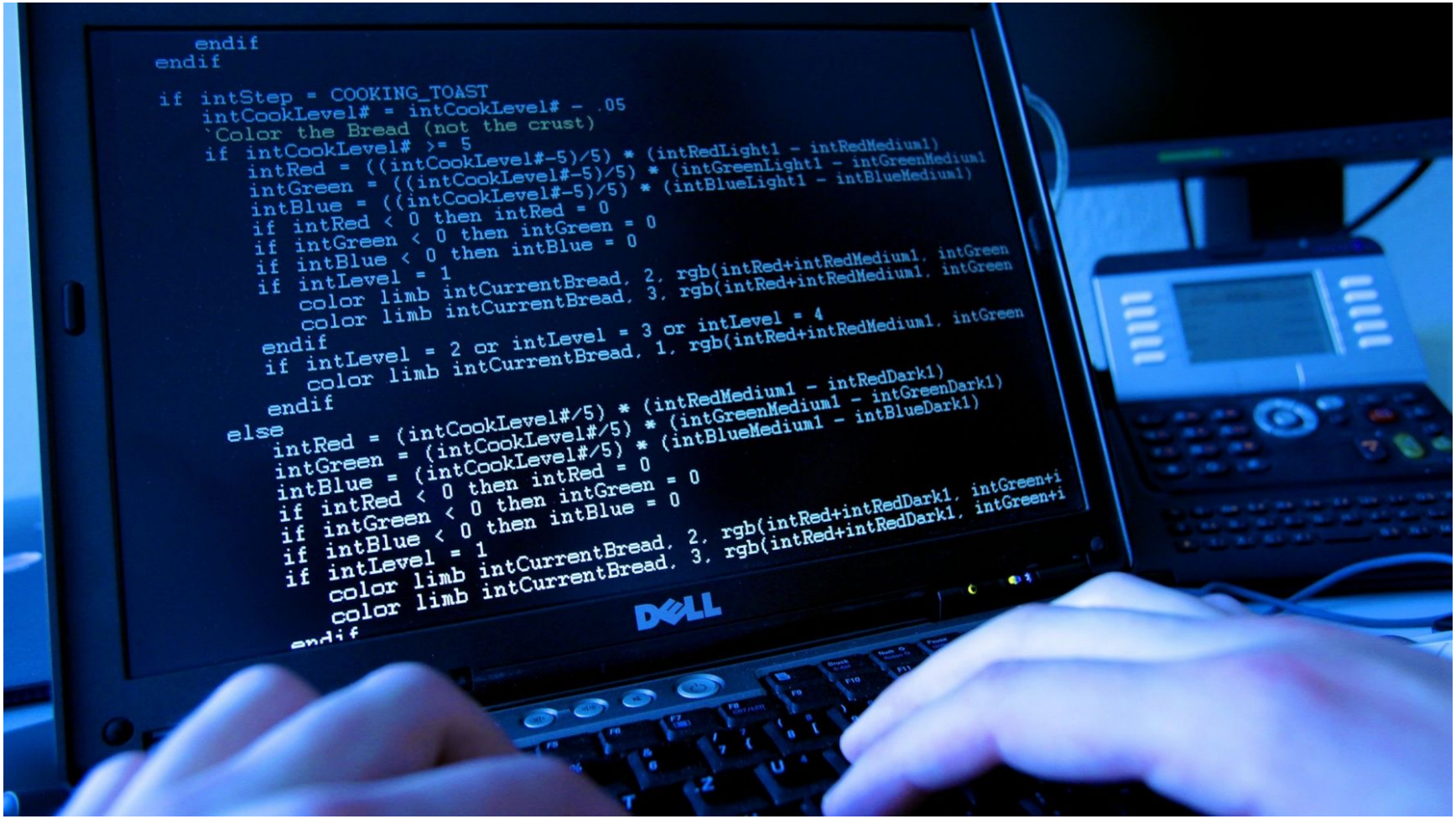
August 28, 2023

Computer Science



Computational Thinking

Programming

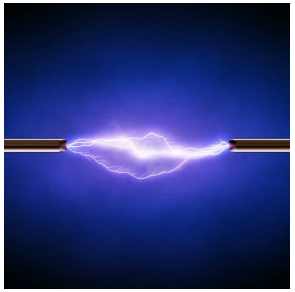


Bits

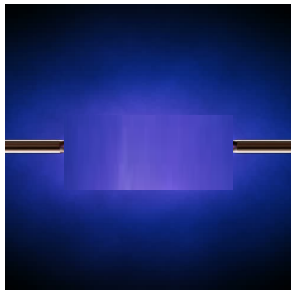
- a **bit** is piece of data that can have two possible values
- can be physically represented with a two state device





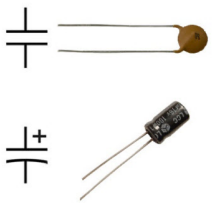




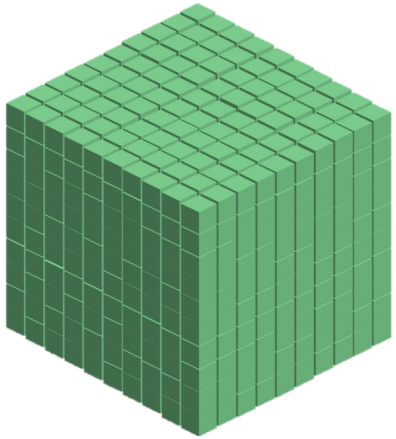
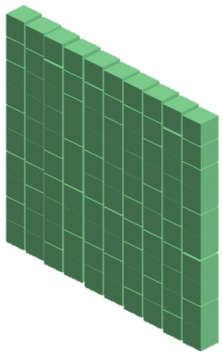



Storing bits

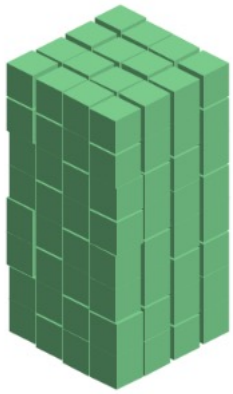
- Dynamic Memory (DRAM): stores each bit of data in a capacitor, which stores energy in an electric field (or not)
- Magnetic Disk: regions of the platter are magnetized with either N-S polarity or S-N polarity
- Optical Disk: stores bits as tiny indentations (pits) or not (lands) that reflect light differently
- Flash Disk: electrons are stored in one of two gates separated by oxide layers



"Normal" Integers (aka Base-10 Integers)

1000 (10^3)	100 (10^2)	10 (10^1)	1 (10^0)
			
0	0	0	5
0	0	4	7
2	0	2	3

Binary Numbers (aka Base-2 Integers)

128 (2^7)	64 (2^6)	32 (2^5)	16 (2^4)	8 (2^3)	4 (2^2)	2 (2^1)	1 (2^0)
							
0	0	0	0	0	1	0	1
0	0	1	0	1	1	1	1
1	1	1	1	1	1	1	1

Exercise 1: Binary Numbers

Consider the following four-bit binary values. What is the integer interpretation of these values?

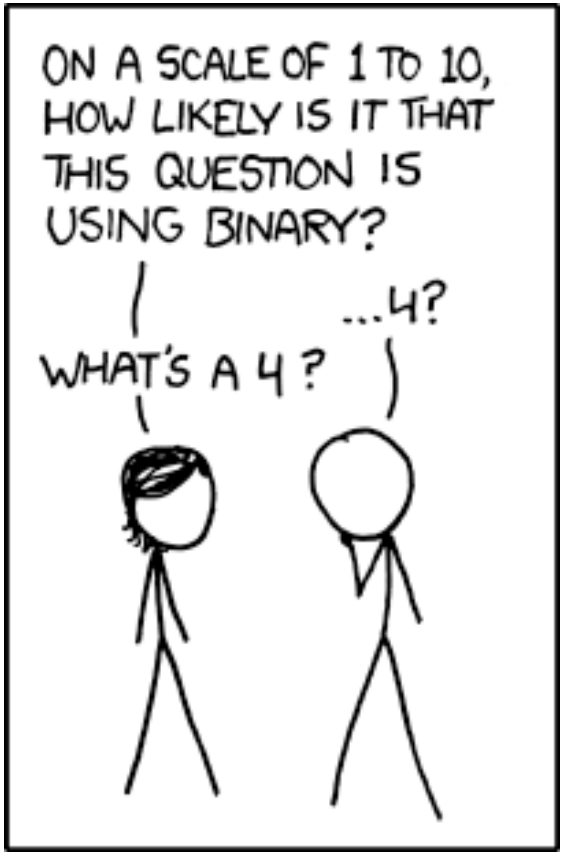
1. 0001
2. 1010
3. 0111
4. 1111

Exercise 2: Binary Numbers

Consider the following integer values. What is the 4-bit binary representation of each of these values?

1. 2
2. 12
3. 13
4. 14

Binary Numbers



There are
10 types
of people
in the world:

Those who
understand binary,
and those
who don't.

ASCII characters

Ch	Dec	Binary	Ch	Dec	Binary	Ch	Dec	Binary	Ch	Dec	Binary	Ch	Dec	Binary
!	33	00100001	1	49	00110001	A	65	01000001	Q	81	01010001	a	97	01100001
"	34	00100010	2	50	00110010	B	66	01000010	R	82	01010010	b	98	01100010
#	35	00100011	3	51	00110011	C	67	01000011	S	83	01010011	c	99	01100011
$	36	00100100	4	52	00110100	D	68	01000100	T	84	01010100	d	100	01100100
%	37	00100101	5	53	00110101	E	69	01000101	U	85	01010101	e	101	01100101
&	38	00100110	6	54	00110110	F	70	01000110	V	86	01010110	f	102	01100110
'	39	00100111	7	55	00110111	G	71	01000111	W	87	01010111	g	103	01100111
(	40	00101000	8	56	00111000	H	72	01001000	X	88	01011000	h	104	01101000
)	41	00101001	9	57	00111001	I	73	01001001	Y	89	01011001	i	105	01101001
*	42	00101010	:	58	00111010	J	74	01001010	Z	90	01011010	j	106	01101010
+	43	00101011	;	59	00111011	K	75	01001011	[	91	01011011	k	107	01101011
,	44	00101100	<	60	00111100	L	76	01001100	\	92	01011100	l	108	01101100
-	45	00101101	=	61	00111101	M	77	01001101	]	93	01011101	m	109	01101101
.	46	00101110	>	62	00111110	N	78	01001110	^	94	01011110	n	110	01101110
/	47	00101111	?	63	00111111	O	79	01001111	_	95	01011111	o	111	01101111
0	48	00110000	@	64	01000000	P	80	01010000	`	96	01100000	p	112	01110000

Exercise 3: ASCII Characters

Consider the following words. How would we represent each of these words in binary?

1. `a`
2. `I`
3. `Hello`

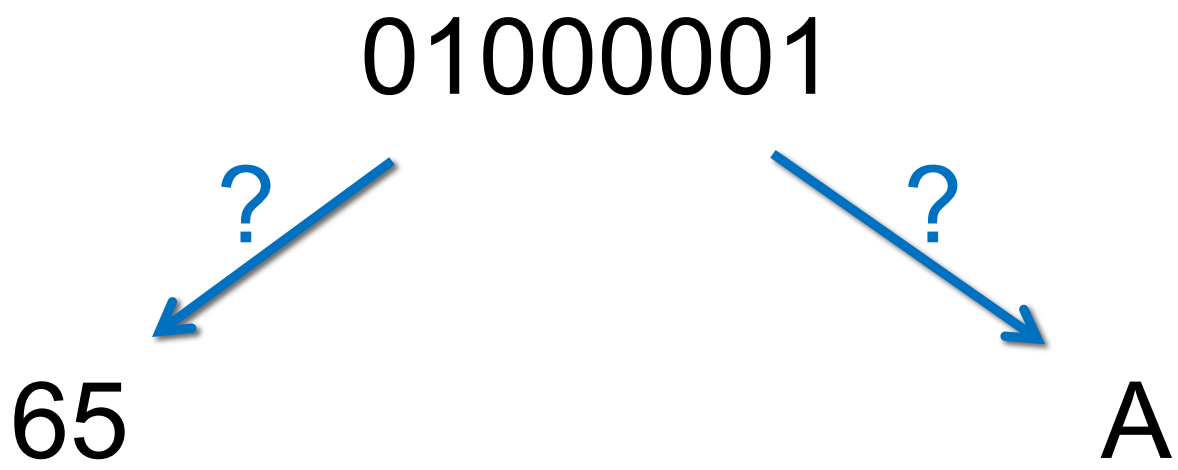
Interpreting Binary

What is the following value?

01000001

Interpreting Binary

What is the following value?



Types

A **type** is a set of values and plan for representing/interpreting those values in binary

integer (`int`)

- Values: 0, 1, -10, 34022, …

string (`str`)

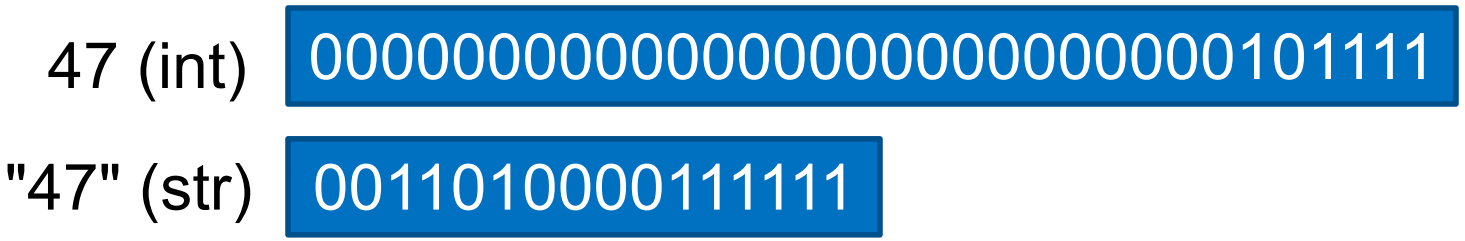
- Values: "Hi!", "", "2.0",…

All values have types.
Integers and strings are two common types

You can determine the type of a value using the command `type(<value>)`

Types

- Values that look similar to a person can have very different representations if they are different types



You can switch types (aka cast) using commands like `int(<value>)` and `str(<value>)`

ValueErrors

- You can't always do things for all possible values

Exercise 4: Casting and Representations

- How would we represent each of following?
  1. `13`
  2. `str(13)`
  3. `"Hi!"`
  4. `int("Hi!")`

Operations

Each type has a set of things you can do with values of that type. These are called **operations**.

int

- Values: 0, 1, -10, 34022, …
- Operations: +, -, /, *
  - ** (exponent),
  - % (remainder)
  - // (truncated division)

string

- Values: "Hi!", "", "2.0",…
- Operations: + (concatenation)
  - * (duplication),

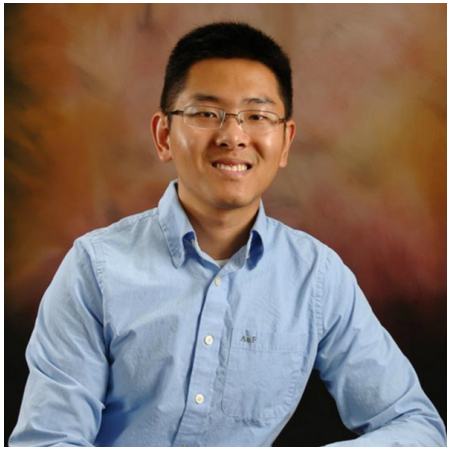
TypeErrors

- You can't always do things for all possible types

LOGISTICS

Course Logistics

Prof. Eleanor Birrell



Prof. Zilong Ye

- **Class Meetings:**
  - Monday and Wednesday, 1:15-2:30pm or 2:45-4pm in Edmunds 114
- **Lab:**
  - Monday or Tuesday, 7-9:50pm in Edmunds 219/229

Course Work

- Homework Assignments (20%)
  - Approximately 10 assignments
  - Mostly individual, some may be done in pairs
  - Assignments will be started in Lab
- Final Project (20%)
  - Use what you've learned!
- Checkpoints (60%)
  - Three checkpoints (Sept 27, Oct 23, and Dec 6, in class)
- No final exam

- All assignments will be due Mondays at 5pm PT.

Course Website

All information is on the Course website:

https://cs.pomona.edu/classes/cs51p/

51p communications, community, q&a, etc. on slack:

#cs51p-2023fa

CS Community

- Slack (Everyone!) pomonacs.slack.com
- BBICS
- WACM