USF UNIVERSITY OF SAN FRANCISCO
CHANGE THE WORLD FROM HERE

Course Recap

CS 272 Software Development

Professor Sophie Engle
Department of Computer Science

Evaluations

Course Evaluations

- BLUE Evaluations
  - Conducted by university
  - Used by instructors AND university
- Course Survey
  - Specific to course
  - Used by instructor to improve course

Where did we start?

What is your guess?

https://pollev.com/sjengle

What was the first code we covered?

Project Euler, Problem 1

If we list all the natural numbers below 10 that are multiples of 3 or 5, we get 3, 5, 6 and 9. The sum of these multiples is 23. Find the sum of all the multiples of 3 or 5 below 1000.

https://projecteuler.net/problem=1

What was the first code we covered?

```
int max = 1000;
int sum = 0;

for (int i = 0; i < max; i++) {
  if (i % 3 == 0 || i % 5 == 0) {
    sum += i;
  }
}
```

https://github.com/usf-cs272-spring2023/cs272-lectures/tree/main/ProjectEuler

What was the first homework?

ArgumentMap

For this homework, you will create a class to parse command-line arguments and store them in a map. For example, consider the following command-line arguments:

```
"-a", "ant", "-b", "bee", "-b", "bat", "cat", "-d", "-e", "elk", "-f"
```

In this case, `-a` `-b` `-d` `-e` and `-f` are all flags since they start with a `-` dash followed by at least 1 non-digit character. The values are `ant` `bee` `bat` `cat` and `elk` since they do not start with a `-` and non-digit character.

Note that `-42` is *not* a flag because the `-` dash is followed by a digit character. Instead it should be interpreted as a value representing a negative number.

Not all flags have values, not all values have associated flags, and values will be overwritten if there are repeated flags. For example, flag `-`

https://github.com/usf-cs272-spring2023/homework-ArgumentParser-template

Where do we end up?

back-end and dynamic **front-end** (**full stack**) of a

Multi-Threaded

in-memory **inverted index**, **web crawler**, and

SEARCH ENGINE

(**database-backed**?) (user tracking via **cookies** or **sessions**?)

What have we learned?

Java Keywords

abstract	continue	for	new	switch
assert	default	if	package	synchronized
boolean	do	goto	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp	volatile
const	float	native	super	while

https://docs.oracle.com/javase/specs/jls/se17/html/jls-3.html#jls-3.9

Java Keywords

abstract	continue	for	new	switch
assert	default	if	~~package~~	synchronized
boolean	do	~~goto~~	private	this
break	*double*	implements	~~protected~~	throw
byte	else	import	public	throws
case	enum	~~instanceof~~	return	~~transient~~
catch	extends	*int*	*short*	try
char	final	interface	static	*void*
class	finally	*long*	~~strictfp~~	volatile
~~const~~	*float*	~~native~~	super	while

https://docs.oracle.com/javase/specs/jls/se17/html/jls-3.html#jls-3.9

Java Keywords

abstract	continue	for	new	switch
assert	default	if	package	synchronized
boolean	do	goto	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp	volatile
const	float	native	super	while

https://docs.oracle.com/javase/specs/jls/se17/html/jls-3.html#jls-3.9

Java Packages

- java.math, java.text, java.time
- java.io (print writer, buffered reader)
- java.lang (thread, object)
- java.net (socket, uri, url)
- java.nio (path, directory stream)
- java.util (collections framework, concurrent, regex, function, stream)
- java.logging (logger, level)
- java.sql (statement, result set)

https://docs.oracle.com/en/java/javase/17/docs/api/index.html

Concepts

- Exception Handling, Resource Handling, File NIO
- Data Structures, Collections Framework, Iteration
- Mutability, Objects, Static, Final
- OOP Principles, Encapsulation, Generalization
- Inheritance, Casting, Nested Classes, Generics
- Functional Interfaces, Lambdas, Stream Pipelines

Concepts

- Logging, Assertions, *Software Testing & Unit Testing**
- Multithreading, Synchronization, Work Queues
- Web, URL/URI, HTML/CSS, HTTP/S, Sockets
- Regular Expressions
- Jetty, Servlets, HTTP Cookies, Sessions
- Relational Databases, SQL, JOINs, JDBC

** See Thursday videos*

Tools & Skills

- Git, Github
- Apache OpenNLP
- Apache Log4j2
- Apache Commons
- Eclipse IDE, Maven
- JUnit
- Jetty, Servlets
- URI, URL, Sockets
- HTTP, HTML, CSS
- SQL, JDBC

Piazza Statistics

...as of Sun May 7, 2023

Piazza At A Glance

Total Posts:	
Total Contributions:	
Instructors' Answered:	
Students' Answered:	
Average Response Time:	

Contributions include posts, notes, answers, replies, and other activity.

Piazza At A Glance

Total Posts:	709 posts
Total Contributions:	2,404 contributions
Instructors' Answered:	504 answered
Students' Answered:	32 answered
Average Response Time:	57 minutes

	Students				Teacher and Assistants			
Category	**Min**	**Median**	**Mean**	**Max**	**Min**	**Median**	**Mean**	**Max**
Days Online								
Views								
Contributions								
Posts								
Answers								
Endorsements								

	Students				Teacher and Assistants			
Category	**Min**	**Median**	**Mean**	**Max**	**Min**	**Median**	**Mean**	**Max**
Days Online	5	46	47	101	62	84	82	97
Views	23	204	202	364	141	634	527	698
Contributions	0	17	26	124	33	204	318	829
Posts	0	9	13	57	15	25	44	111
Answers	0	0	1	7	9	95	138	352
Endorsements	0	1	2	20	5	51	53	105

GitHub Statistics

...as of Sun May 7, 2023

Total Statistics

Description	Total
Repositories*	
Issues**	
Pull Requests***	
Action Minutes	

Description	Total
Java Files	
Java Comments	
Java SLOC	

*Student repositories only **Without error label ***Reviewed PRs only

Total Statistics

Description	Total
Repositories*	463
Issues**	629
Pull Requests***	403
Action Minutes	?????

Description	Total
Java Files	1,601
Java Comments	188,116
Java SLOC	274,133

*Student repositories only **Without error label ***Reviewed PRs only

Projects Only

Metric*	Files	Blank Lines	Comments	Code
Minimum				
Median				
Mean				
Maximum				

**Filtered out projects with only 1 Java file.*

Projects Only

Metric	Files	Blank Lines	Comments	Code
Minimum	3	64	381	404
Median	13	299	1,301	1,452
Mean	14	300	1,229	1,402
Maximum	25	517	1,914	2,276

Code Reviews by Type

Statistic	Total Count	Per Item	Per Term	Per Week
Code Reviews		20m		
Quick Reviews		10m		
All Reviews*		---		
Lectures**	30	105m	52.5h	3.5h

*Assumes 10 weeks of code reviews. **Assumes 15 weeks per term or semester.

Code Reviews by Type

Statistic	Total Count	Per Item	Per Term	Per Week
Code Reviews	340	20m	113.3h	9.4h
Quick Reviews	063	10m	10.5h	0.9h
All Reviews	403	---	123.8h	10.3h
Lectures	30	105m	52.5h	3.5h

**Assumes 10 weeks of code reviews.* ***Assumes 15 weeks per term or semester.*

Code Reviews by Project

	Grade Issues			Code Review (20)		Quick Review (10)	
Category	**Test v#.0**	**Test v#.1**	**Design**	**Total**	**Average***	**Total**	**Average***
Project 1							
Project 2							
Project 3*							

*Average per test issue. **Many students still working on project 3, results inaccurate.

Code Reviews by Project

	Grade Issues			Code Review (20)		Quick Review (10)	
Category	**Test v#.0**	**Test v#.1**	**Design**	**Total**	**Average***	**Total**	**Average***
Project 1	41	38	34	138	3.6	28	0.7
Project 2	34	31	33	104	3.4	21	0.7
Project 3*	33	32	24	98	3.0	14	0.4

**Average per v#.1 test issue.* ***Many students still working on project 3, results inaccurate.*

What's Next?

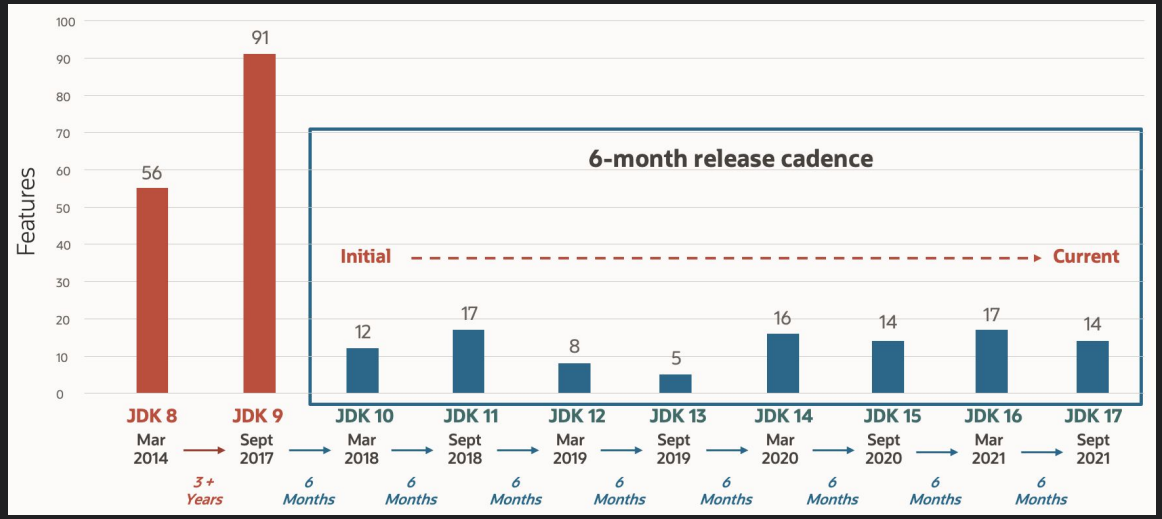
Missing Concepts

- Sockets+
- Generics+
- Testing+, Benchmarking+
- Multithreading+
- Servlets+
- Databases+
- Security+
- Deployment, JARs
- Packages, Modules
- AWT, Swing
- Graphics, Sound
- Serialization
- Internationalization
- *And More...*

Followup Courses?

- CS 256 **Career Prep Course** (2 Units, Optional)
- CS 490 **Senior Team Project** (4 Units, Required)
- CS 345 **Programming Languages** (4 Units, Theory)
- CS 333 Intro to **Database Systems** (4 Units, Application)
- CS 336 **Computer Networks** (4 Units, Application)
- Topics: **Computer Security**, **Distributed Systems**...

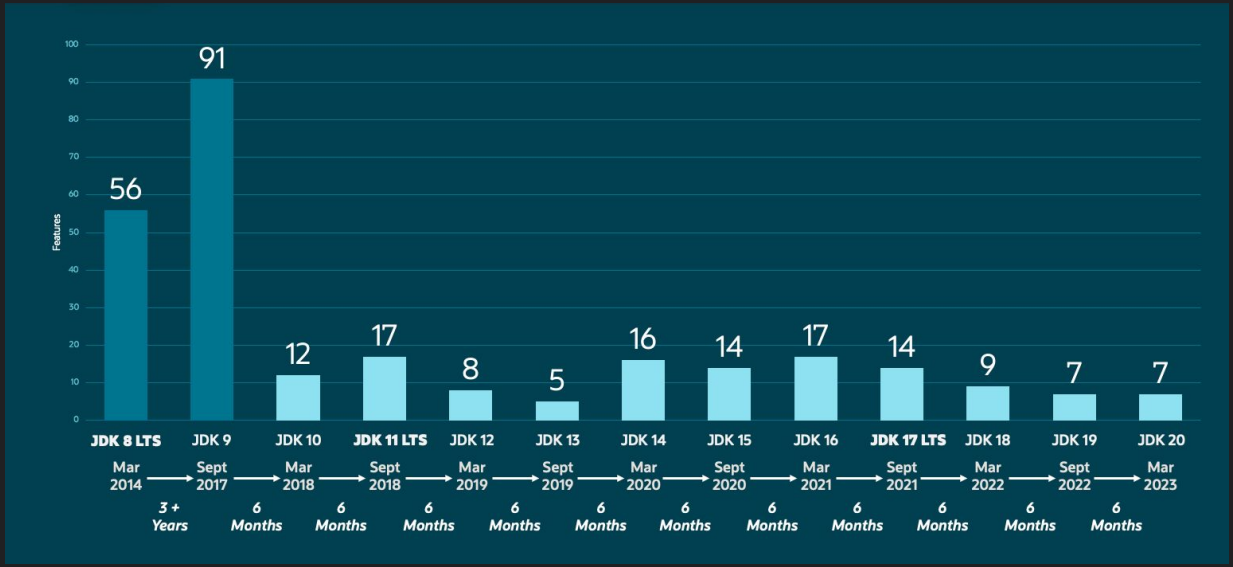
Java 17 (LTS)



"According to an IDC report over ten million developers, representing 75% of full-time developers worldwide, use Java, more than any other language."

https://blogs.oracle.com/java/post/announcing-java17

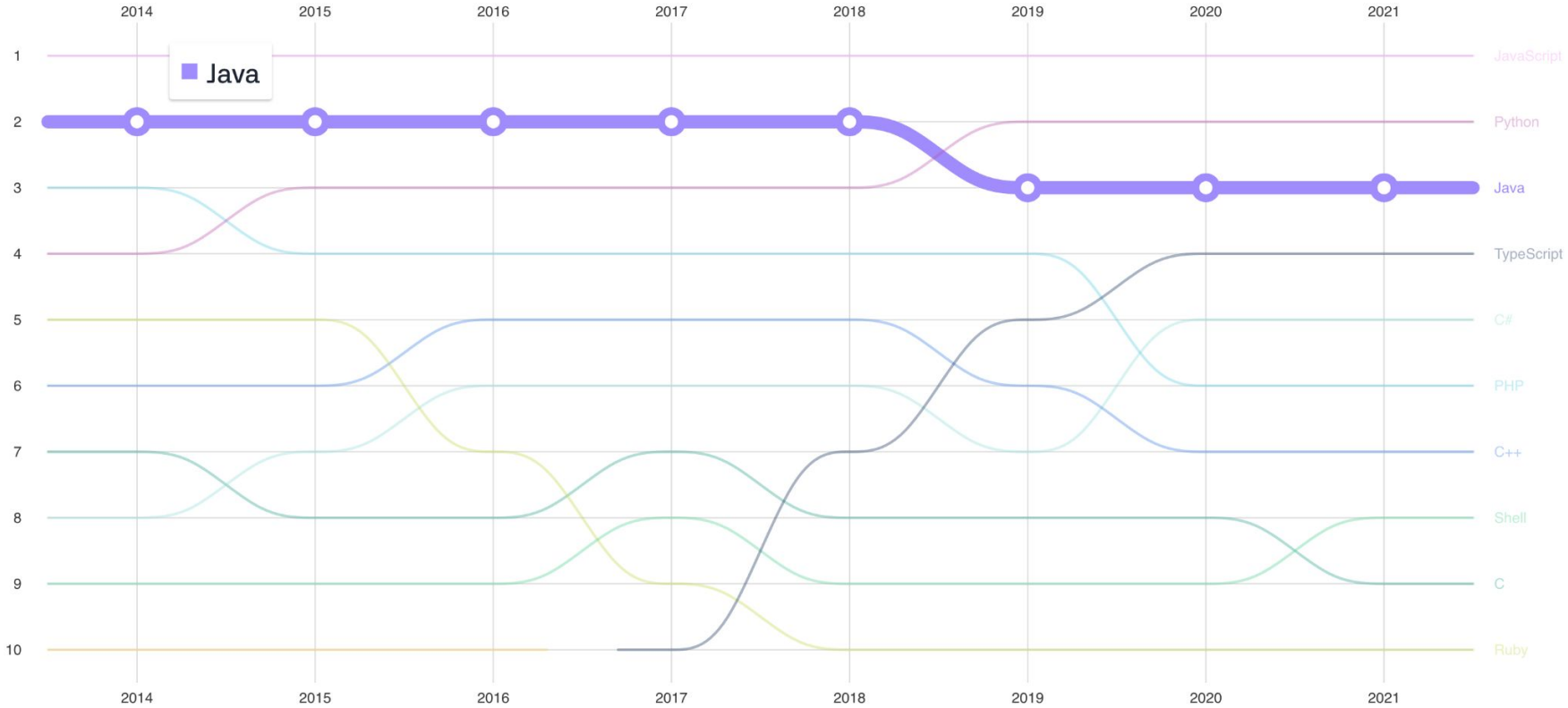
Java 20



https://blogs.oracle.com/java/post/the-arrival-of-java-20

Oracle Java SE Support Roadmap*†				
Release	**GA Date**	**Premier Support Until**	**Extended Support Until**	**Sustaining Support**
7 (LTS)	July 2011	July 2019	July 2022*****	Indefinite
8 (LTS)**	March 2014	March 2022	December 2030*****	Indefinite
9 (non-LTS)	September 2017	March 2018	Not Available	Indefinite
10 (non-LTS)	March 2018	September 2018	Not Available	Indefinite
11 (LTS)	September 2018	September 2023	September 2026	Indefinite
12 (non-LTS)	March 2019	September 2019	Not Available	Indefinite
13 (non-LTS)	September 2019	March 2020	Not Available	Indefinite
14 (non-LTS)	March 2020	September 2020	Not Available	Indefinite
15 (non-LTS)	September 2020	March 2021	Not Available	Indefinite
16 (non-LTS)	March 2021	September 2021	Not Available	Indefinite
17 (LTS)	September 2021	September 2026****	September 2029****	Indefinite
18 (non-LTS)	March 2022	September 2022	Not Available	Indefinite
19 (non-LTS)***	September 2022	March 2023	Not Available	Indefinite
20 (non-LTS)***	March 2023	September 2023	Not Available	Indefinite
21 (LTS)***	September 2023	September 2028	September 2031	Indefinite

https://www.oracle.com/java/technologies/java-se-support-roadmap.html



https://octoverse.github.com/#top-languages

Don’t call it a comeback: Why Java is still champ

"After nearly 30 years of Java, you might expect the language to be showing some signs of wear and tear, but nothing could be further from the truth. Java in 2022 is not a language in decline, but rather a language preparing for the effervescent future of software development."

https://github.com/readme/featured/java-programming-language · August 2022

Historical Perspective

- Know someone **180** or older?
  - The first computer program or algorithm was published in 1843.
- Know someone **87** or older?
  - Formal models of algorithms and modern computers were invented in 1936.

https://en.wikipedia.org/wiki/Computer_science · https://en.wikipedia.org/wiki/History_of_computer_science

Historical Perspective

- Know someone **64** or older?
  - The term "computer science" first appears in literature in 1959.
- Know someone **61** or older?
  - The first CS degree in the US was offered at Purdue University in 1962.

https://en.wikipedia.org/wiki/Computer_science · https://en.wikipedia.org/wiki/History_of_computer_science

Historical Perspective

- Know someone **54** or older?
  - The first message was sent over ARPANET, the precursor of the Internet, in 1969.
- Know someone **52** or older?
  - The first personal computer was released in 1971.

https://en.wikipedia.org/wiki/Computer_science · https://en.wikipedia.org/wiki/History_of_computer_science

Historical Perspective

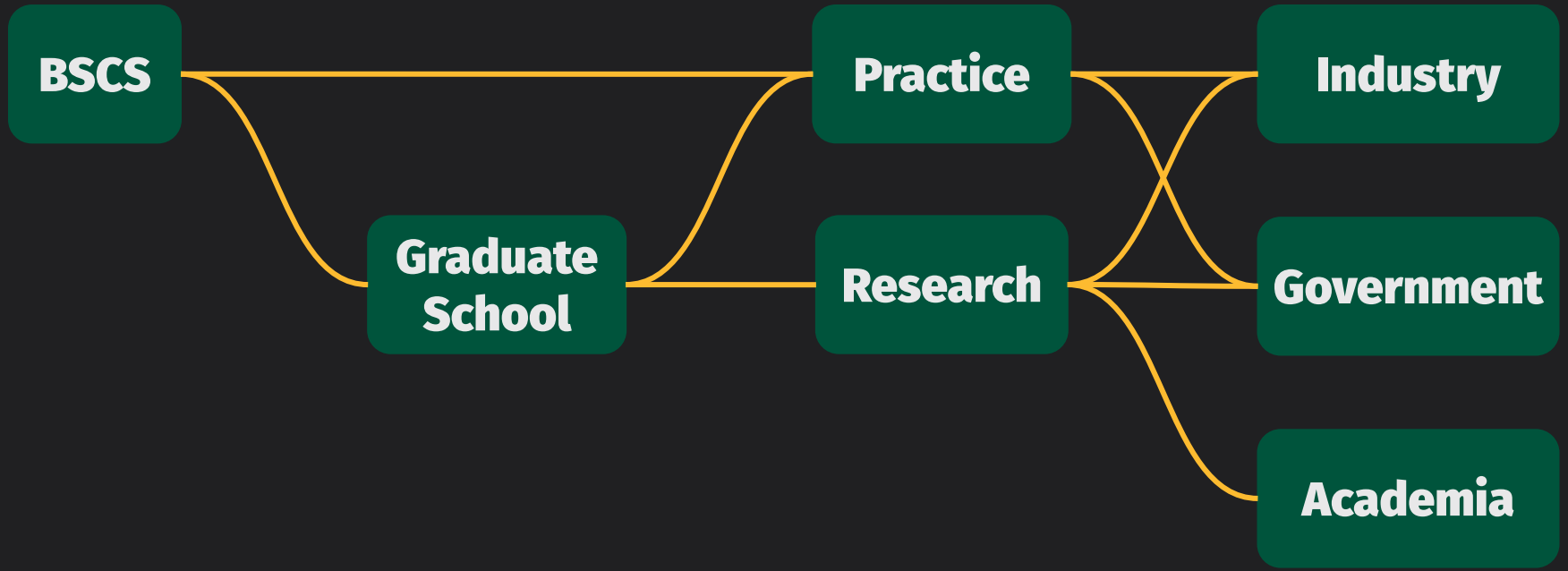
- Know someone **34** or older?
  - The World Wide Web was invented in 1989.
  - The Python programming language was first implemented in 1989.
- Know someone **16** or older?
  - Apple released the first iPhone in 2007.

https://en.wikipedia.org/wiki/Computer_science · https://en.wikipedia.org/wiki/History_of_computer_science

Computer Science...

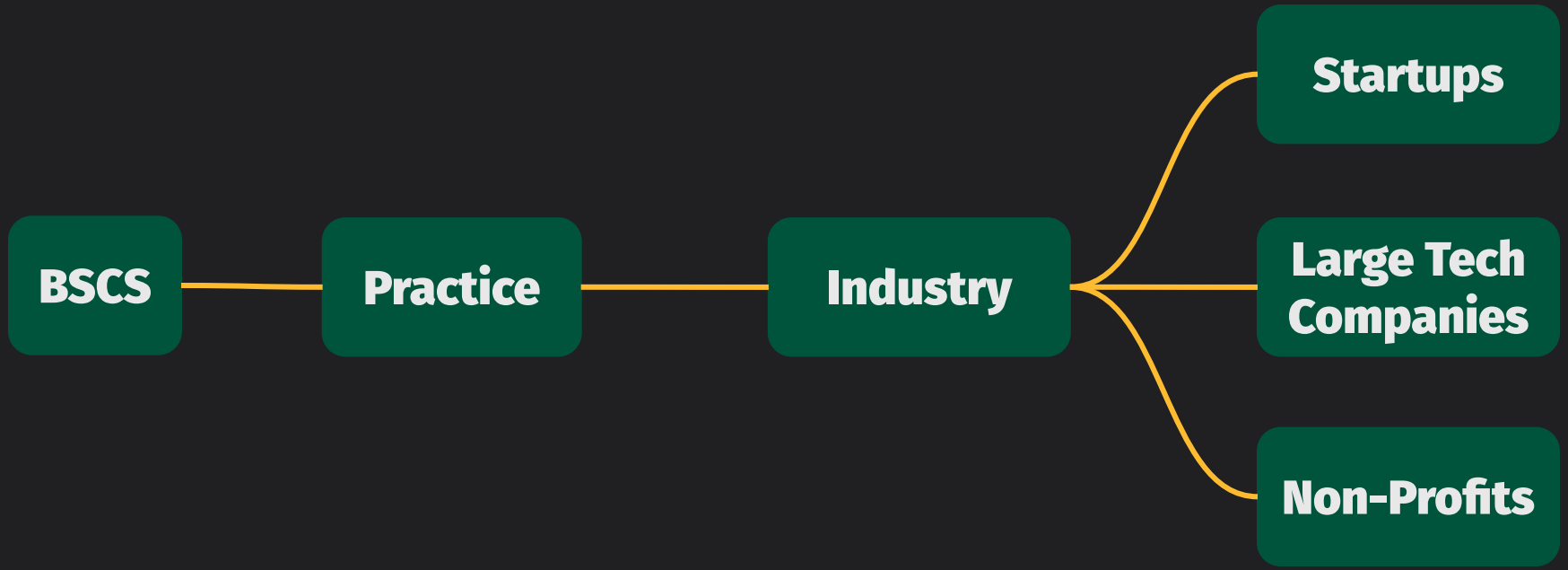
- Constantly changing
  - Must be constantly learning
- Increasing in ubiquity, impact, and importance
  - Must consider ethics, diversity, accessibility
- Multiple career paths and subfields
  - Sometimes tricky to find what is right for you

Career Paths in CS



*General picture, exceptions exist!

Career Paths in CS



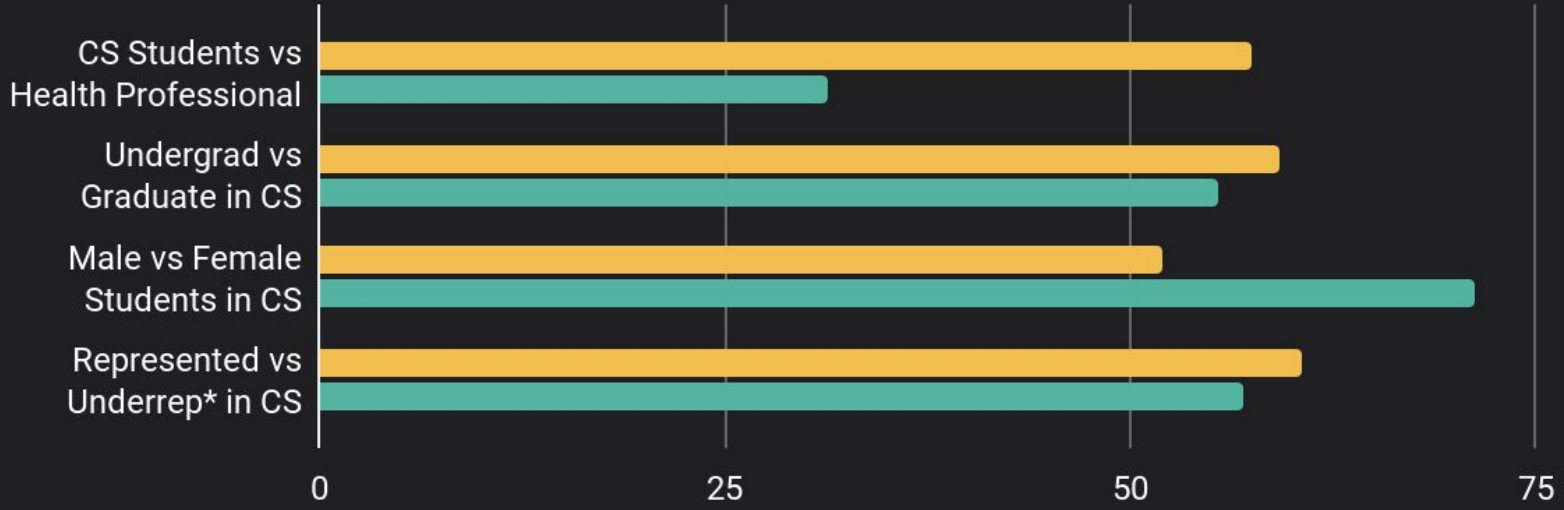
*General picture, exceptions exist!

Many Possible Job Titles...

- First, choose a broad specialization...
  - Software, Security, Network, Systems, Database, Data, Cloud, UX, etc.
- Next, choose a job title...
  - Developer, Engineer, Analyst, Administrator, Architect, Designer, Manager, Researcher, Scientist, Teacher, etc.
- Or, choose a more focused specialization...
  - Finance, Computational Biology, Bioinformatics, Medicine, Games, Environment, Education, etc.

Closing

Impostor Phenomenon in CS



https://dl.acm.org/doi/10.1145/3328778.3366815 Mar 2020 • **Low number of responses*

BE PROUD!

You are a computer scientist.

You have a powerful skill set.

Ask me (almost) **anything.**

https://pollev.com/sjengle

Thank You!