SOFTWARE ENGINEERING INTERVIEW PREP

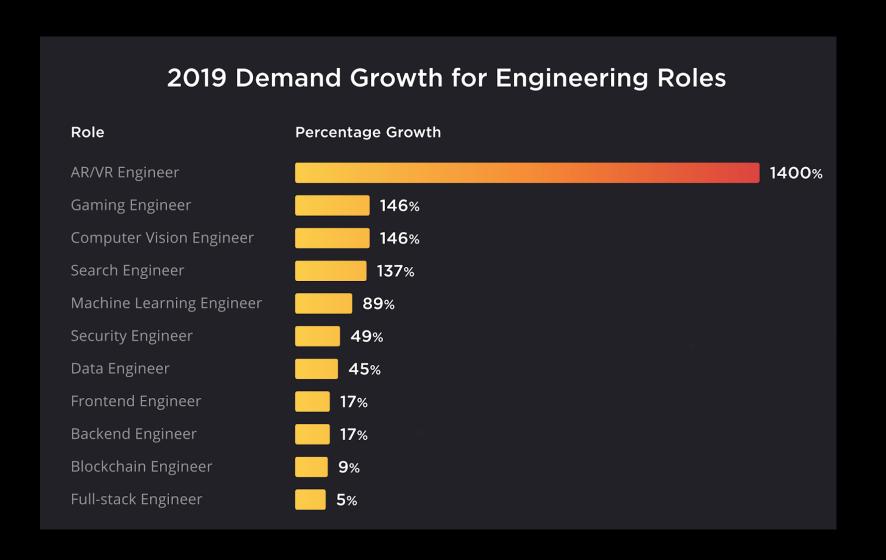


MEETING OVERVIEW

- Getting the Software Engineering Interview
- Overview of the Interview Process
 - Understanding the coding interview problems
- How to Prepare for the Interviews
- Lab: Understanding Time & Space Complexity



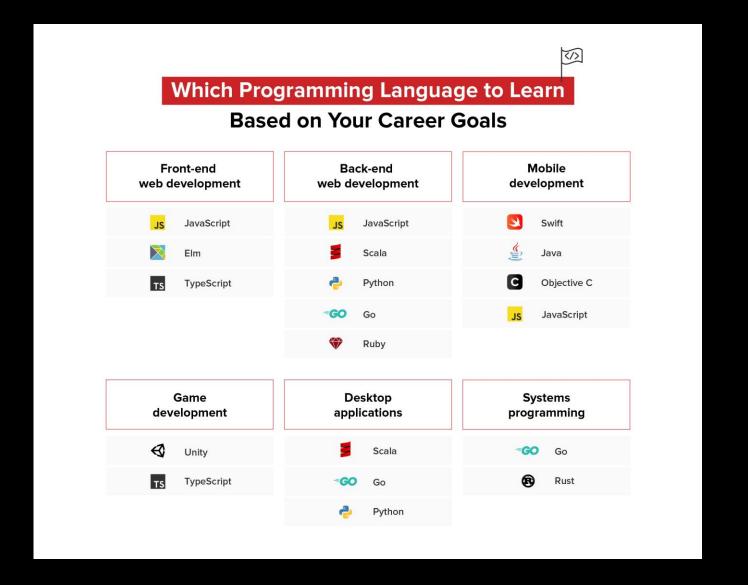
SOFTWARE ENGINEERING DEMAND







PROGRAMMING LANGUAGES TO LEARN



GETTING THE INTERVIEW

Reach out to people on LinkedIn! Referrals are the best way to get an interview

• Learn how to sell yourself – what value do you bring?

Have a polished resume – have clear and concise bullet points

FIRST LAST

New York, NY | P: +44 123456789 | first,last@resumeworded.com

RESUME WORDED UNIVERSITY

Boston, MA Expected May 2019

Bachelor of Engineering Major in Computer Science; Minors in Mathematics and Psychology Cumulative GPA: 3.93/4.0: Dean's List 2015-2016.

Relevant Coursework: Software Engineering; Operating Systems; Algorithms; Artificial Intelligence; System Design

LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

New York, NY

Study Abroad Program in International Business and Globalization Jul 2016 - Jul 2017

WORK EXPERIENCE

RESUME WORDED & CO. (8 employee venture-backed recruitment startup)

New York, NY

Software Development Intern Jun 2017 - Sep 2017

- . Developed web scraping program in Python to help the firm download public data, including over 10,000 company descriptions and stock quotes, enriching internal data and increasing research efficiency by over 50%
- . Implemented crash reporter and used findings to fix three biggest causes of crashes; reduced customer support calls by 30%
- Designed and implemented 50+ E2E tests using Selenium & Protractor, simulating each user group's actions

Java Developer Intern

Jun 2016 - Sep 2016

- Led the transition to a paperless practice by implementing an electronic booking system and a faster, safer and more accurate business system; reduced cost of labor by 30% and office overhead by 10%
- · Translated business questions into use cases and data model requirements with emphasis on anticipating future ad-hoc needs
- . Created use cases and developed UML diagrams activity, sequence and class using Visio in line with product goals

- Designed and implemented movie recommendation application in 4-person team using Python in 3-day hackathon
- Enabled users to be recommended movies based on 50+ data points; awarded most innovative project by Google engineer

PINTOS - MODEL OPERATING SYSTEM

- Designed and implemented Unix-based operating system in 4-person team using C++
- Awarded First Prize in Computing's Senior Design Projects (out of 100 teams)

RESUME WORDED FINANCE SOCIETY

Boston MA Sep 2017 - Present

· Founded the first ever Business Series to organize finance training for 500 students

· Organized and advertised 10+ quarterly networking events with 300+ participants in 3 universities in Boston

RWILTENNIS SOCIETY

Boston MA Jan 2017 - Present

Committee Membe

· Managed the launch of new booking system to improve organization of events; system now used across university

Technical Skills: Advanced in SQL, PHP, Javascript, HTML/CSS; Proficient in MATLAB, Python [enter keywords from job

description, where relevantl Languages: Fluent in French, English: Conversational Proficiency in Italian, German

Certifications & Training: Online Course in Management (Coursera), Passed Resume Worded examinations

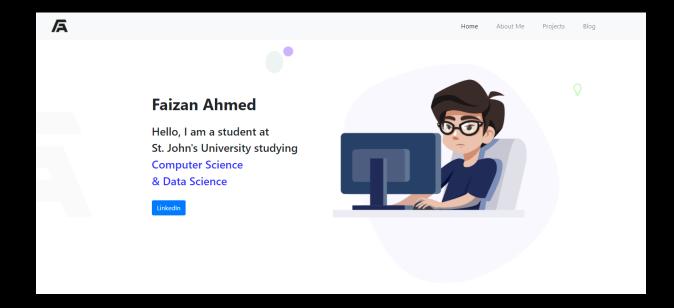
Awards: RW's Top 30 Under 30 (2011); Won RW's nationwide case competition out of 500+ participants (2013)

TIPS FOR GETTING THE INTERVIEW

- Work on personal projects
- Create your own personal website
 - Build your personal brand

Contribute to open-source projects!

• Get used to rejection!



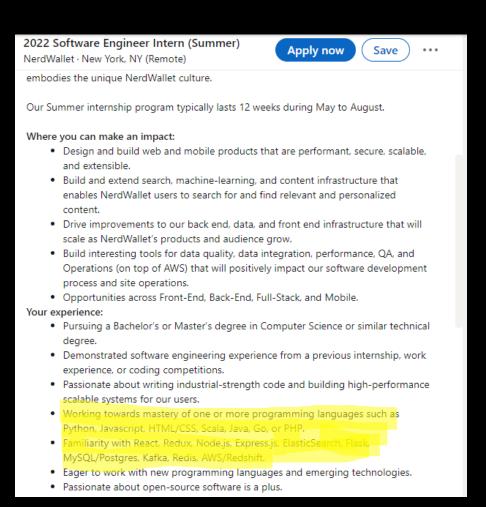
https://faizanahmed.me/



TIPS FOR GETTING THE INTERVIEW

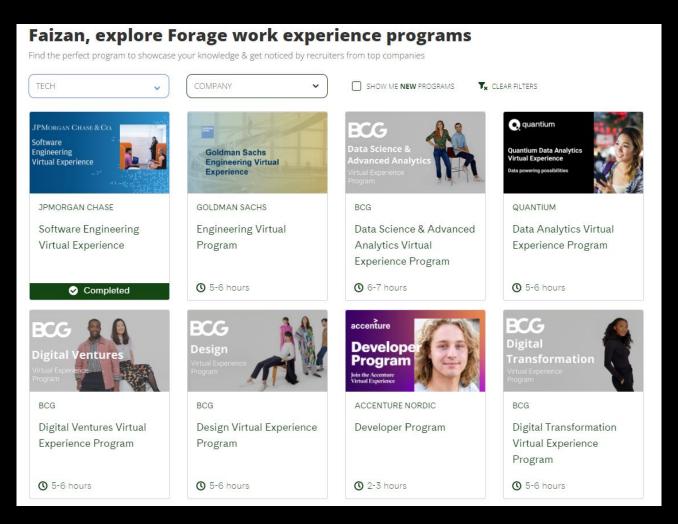
 Get familiar with the technologies that companies use

Check job descriptions



TIPS FOR GETTING THE INTERVIEW

- Complete virtual
 experience programs
 through <u>theforage.com</u>
- Experience the day-to-day tasks of a company and increase your chances of being hired there!





THE INTERVIEW PROCESS

- 1. Online Coding Assessment
 - Typically on HackerRank
- 2. Technical Phone Interview
 - Typically around 45 mins to 1 hour
- 3. On-site Interviews (now done virtually)
 - 4 6 additional interviews (behavioral and technical)



 Python is the recommended language for solving coding interview problems

 Just getting the solution is not enough! You need to get a solution that's fast and does not take up too much memory!



- Common interview question:
- There are many ways to solve this problem, but it needs to be done efficiently!





- Requires a very good understanding of data structures and algorithms
- Data Structures to know:
 - Lists
 - Linked-Lists
 - Dictionaries
 - Stacks
 - Sets
 - Trees
 - Graphs



Goo	gle coding interview questions types	iGotAnOffer
#1	Graphs / Trees E.g. Range sum of a binary search tree	39%
#2	Arrays / Strings E.g. Product of an array except self	26%
#3	Dynamic Programming E.g. Continuous subarray sum	12%
#4	Recursion E.g. Given a 2D board and a list of words, find all words in the board.	12%
#5	Geometry / Math E.g. Given a set of point on an xy-plane, find the rectangle with minimum of	11% area
Source G	lassdoor.com data for Google software engineer interview questions. Analysis by (GotAnOffer,	



THE BEHAVIORAL INTERVIEW

- 30-45 minutes long meant for the interviewer to gauge your personality and if it aligns with the company
- If your interviewer doesn't see
 themselves going to grab coffee with
 you, you will not pass the interview!
- Work on your communication skills!



COMPENSATION

View verified salaries on https://www.levels.fyi/



Software Engineer								
Average Compensa	+ Add Your Compensation							
Level Name	Total	Base	Stock (/yr)	Bonus				
T3 Software Engineer (Entry Level)	\$238K	\$136K	\$74K	\$31K				

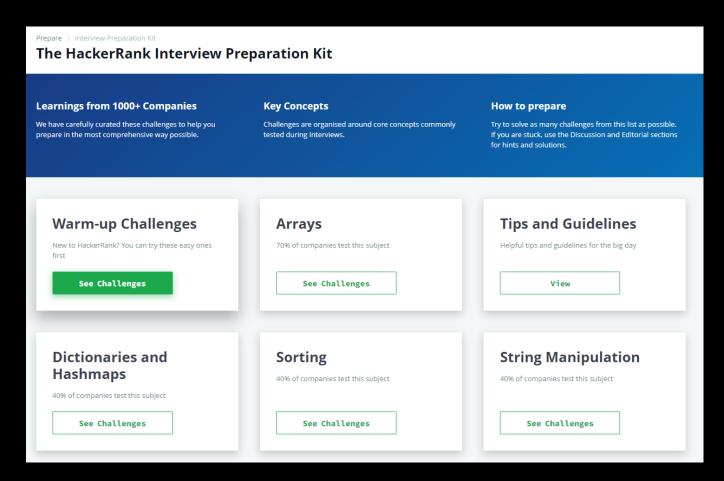


Software Engineer								
Average Compensation by Level + Add Your Compensation								
Level Name	Total	Base	Stock (/yr)	Bonus				
Software Engineer (Entry Level)	\$218K	\$149K	\$52K	\$17K				



HOW TO PREPARE

- Complete as many problems as you can on <u>LeetCode.com</u>
- Complete the <u>HackerRank</u>
 <u>Interview Preparation Kit</u>
- Complete company-specific problems on <u>GeeksforGeeks</u>





HOW TO PREPARE

- Have a strong understanding of time and space complexity
- Do mock interviews with friends, explain your thought process while solving the problem
- Use https://www.pramp.com/ to practice interviewing with a peer

How Pramp Practicing Works



Get Paired

Tell us when and what you want to practice and we'll pair you with an optimal peer. We provide interview questions and a collaborative environment for you to conduct the interview.



Practic

Interviews are conducted using a collaborative environment over video. You and your peer will interview each other for 30-45 minutes. After the interview, you'll each provide feedback on to help each other improve.



Ace

Learn from peers' feedback, gain confidence and master the art of interviewing. Keep practicing until you interview like a rock star. Impress recruiters and land awesome job offers.

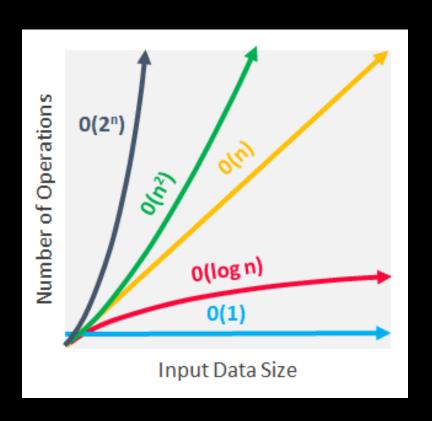


WHAT IS TIME COMPLEXITY?

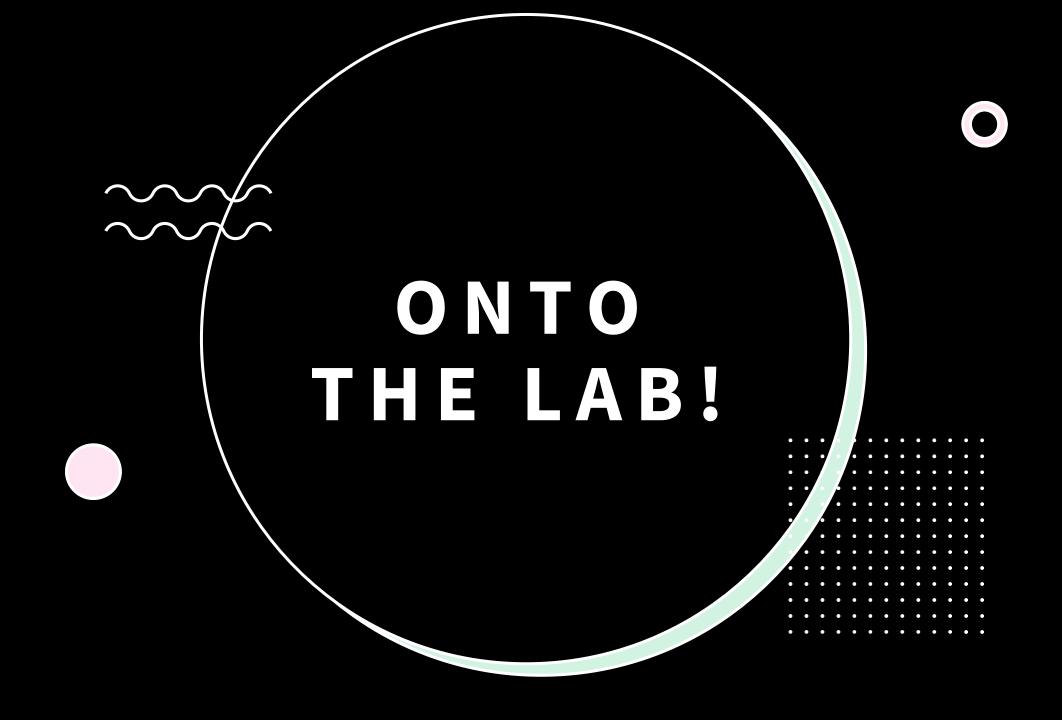
 Allows us to compare the performance of algorithms

 Lets us understand how algorithms would perform at scale

• Expressed in Big-O notation







LAB INSTRUCTIONS

Go to https://t.ly/SJUACM

Follow along on Google Colab

(Check out the full code with explanations <u>here</u>)



TIME COMPLEXITY OF THE RECURSIVE FUNCTION

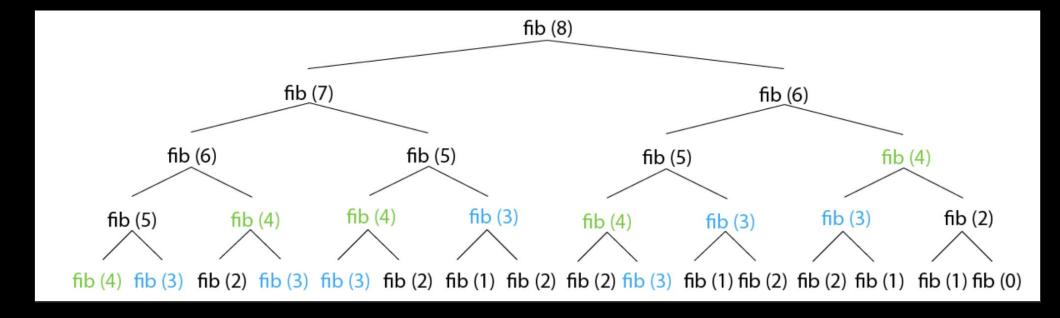
of Recursive

 2^1

 2^3

 2^4

Calls:



The time complexity of this algorithm is $O(2^n)$ (Exponential)

Each Fibonacci number takes twice as long to compute as the previous one!



RUNTIME ANALYSIS

Fibonacci Number (n)	Time to Calculate n
40	~28 Seconds
50	~119 Days
100	~2,700,000 Years
200	~2,500,000,000,000,000,000,000,000 Years



TIME COMPLEXITY OF THE TABULATION FUNCTION

tabulationFibonacci(7):

tabulationFibonacci(8):

The time complexity of this algorithm is O(n) (Linear),

since the algorithm takes n steps to calculate Fib(n)

Each Fibonacci number takes only one more step to calculate than the previous one!



CLASSES TO TAKE

 Understand the mathematical foundations of Computer Science, this will help you become a better programmer!

• CUS 1186 Theory of Computation

MTH 1017 Mathematical Logic



PATHWAYS TO TECH PANEL

Today at 7:30 PM

https://info.codepath.org/pat hways-to-tech-instructorpanel



HIRING WITHOUT WHITEBOARDS

- Companies that don't have traditional interview processes
- https://github.com/potet
 o/hiring-without whiteboards

Hiring Without Whiteboards

A list of companies (or teams) that don't do "whiteboard" interviews. "Whiteboards" is used as a metaphor, and is a *symbol* for the kinds of CS trivia questions that are associated with bad interview practices. Whiteboards are not bad – CS trivia questions are. Using sites like HackerRank/LeetCode *probably* fall into a similar category.

The companies and teams listed here use interview techniques and questions that resemble day-to-day work. For example, pairing on a real world problem or a paid/unpaid take home exercise. Read (and contribute to) our recommendations for ways to conduct better interviews.

tl;dr

- . Discussing a real world problem (with or without whiteboard) is
- Solving CS trivia, technical puzzles, riddles, brainteasers (with or without whiteboard) is \(\bigcirc\)



