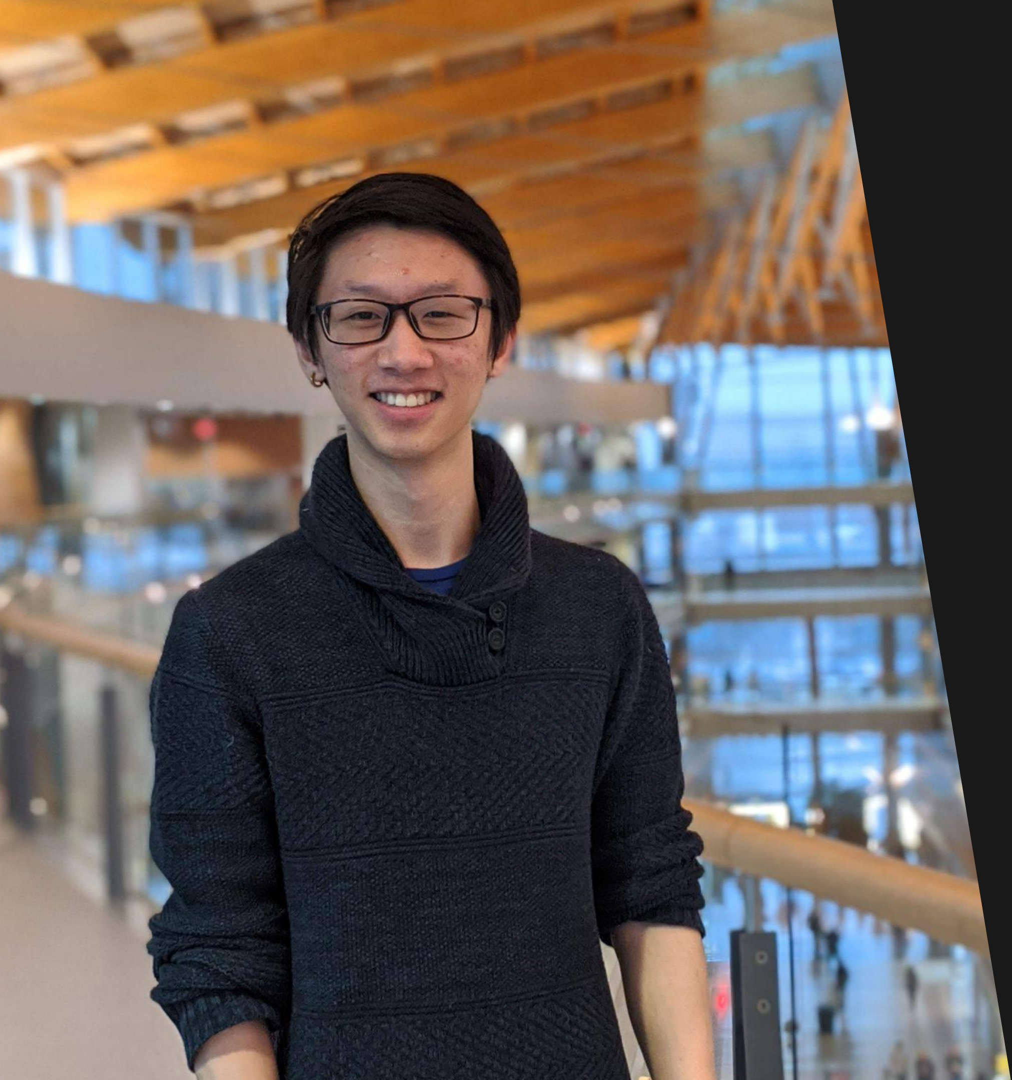


be not afraid of the technical interview

Jeffrey Leung
Amazon Software Developer (October 2021 - Present)
SAP Tech. Project Coordinator Intern (2020)
Samsung Back-end Developer Intern (2017)
SFU Software Systems (2014 - 2021)

TABLE OF CONTENTS

01

ABOUT ME

02

INTRO TO TECH INTERVIEWS

03

STRUCTURE YOUR ANSWERS

04

EXAMPLE PROBLEM

05

PRACTICE STRATEGICALLY

06

QUESTIONS

01

ABOUT ME

Fill out Survey at sfusurge.com/project hub
Collab 2
Whiteboard
A. Website
① Gift
② Degree Navigator

02.1

INTERVIEW STRUCTURE

Behavioural Questions (20 minutes)

Technical Questions (20 minutes)

02.2

WHAT ARE TECHNICAL INTERVIEW QUESTIONS?

- Coding questions
- Requires a self-directed answer
- Often has both a brute-force algorithm and an efficient algorithm
- Designed to evaluate:
  - **Communication** skills
  - Understanding of time/space **complexity**
  - Data structures/algorithm abilities
  - **How much you have practiced**

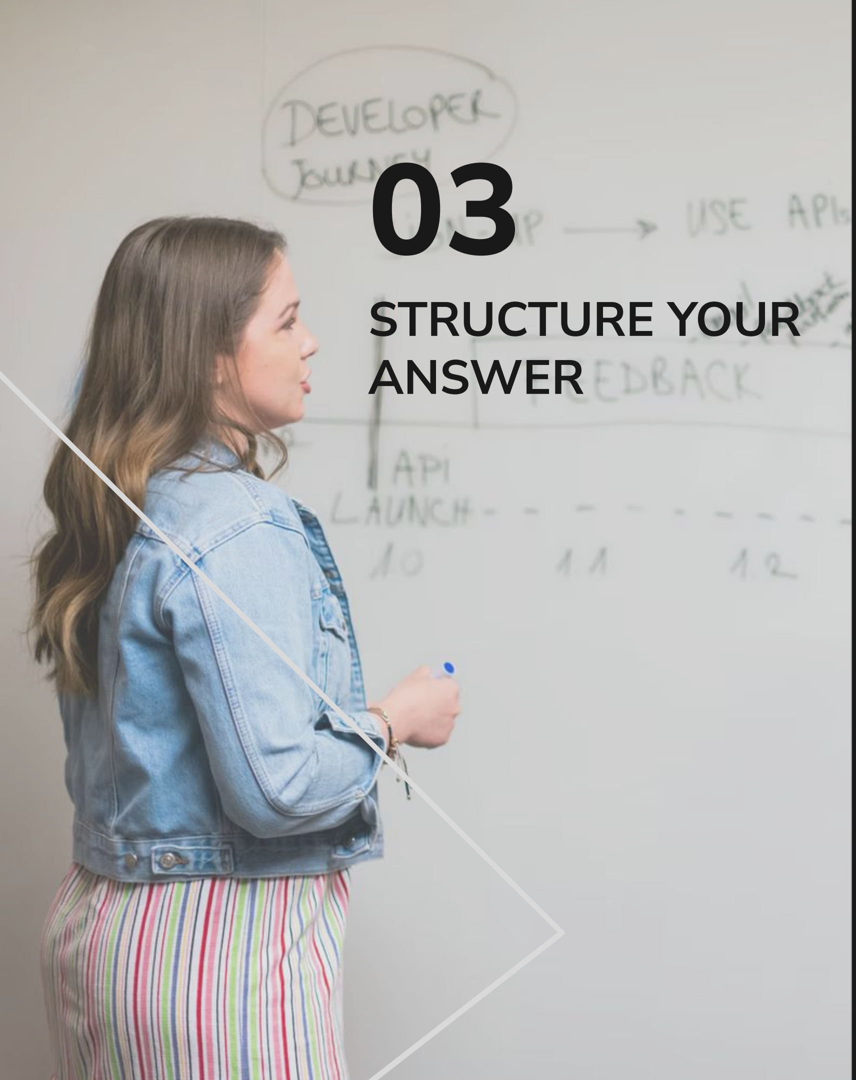
Example question: Given an array, return whether a value appears at least twice.

02.3

EXPECTATIONS OF YOU

- Technical knowledge (a little)
- Adapting to technical problems with new constraints
- Ability to solve problems independently & together

- Soft skills:
    - Ability to communicate thought process
    - **A team member and normal human being**



03

STRUCTURE YOUR ANSWER

1. **Investigate.**
   Ask clarifying questions. Define the input/outputs. Suggest constraints. Discuss obvious edge cases.

2. **Brute-force it.**
   Suggest the obvious answer. Solve as if solving manually. Explain and code-comment your solution. Draw whiteboard diagrams if available.

3. **Calculate Complexity.**
   How much **time** does each line take? What’s the O(n)? How much **space** is needed?

4. **Optimize.**
   Pinpoint the heaviest operations. Eliminate duplicated effort. Use data structures to speed up saving/retrieval. Write down the new time and space complexity.

5. **Code.**
   Code alongside your comments. Write simple, readable, and maintainable code. Mention edge cases.

6. **Test and Verify.**
   List and check all possible edge cases. Walk through the code with a sample input to verify output.

04

EXAMPLE PROBLEM

1. **Investigate.**
   Ask clarifying questions. Define the input/outputs. Suggest constraints. Discuss obvious edge cases.

2. **Brute-force it.**
   Suggest the obvious answer. Solve as if solving manually. Explain and code-comment your solution. Draw whiteboard diagrams if available.

3. **Calculate Complexity.**
   How much **time** does each line take? What's the O(n)? How much **space** is needed?

4. **Optimize.**
   Pinpoint the heaviest operations. Eliminate duplicated effort. Use data structures to speed up saving/retrieval. Write down the new time and space complexity.

5. **Code.**
   Code alongside your comments. Write simple, readable, and maintainable code. Mention edge cases.

6. **Test and Verify.**
   List and check all possible edge cases. Walk through the code with a sample input to verify output.

05.1

PRACTICE STRATEGICALLY

Languages: Python, Java, JavaScript, C++

Practice by yourself & with friends

Start with LeetCode **EASY** questions

Struggling with LeetCode?

- Don’t read answers until you have been stuck for 10+ minutes
- Read sample answers from the Discussions tab
- Go back to easier questions
- Save questions you could not solve by yourself, and revisit them in future

Try other websites - NeetCode, HackerRank

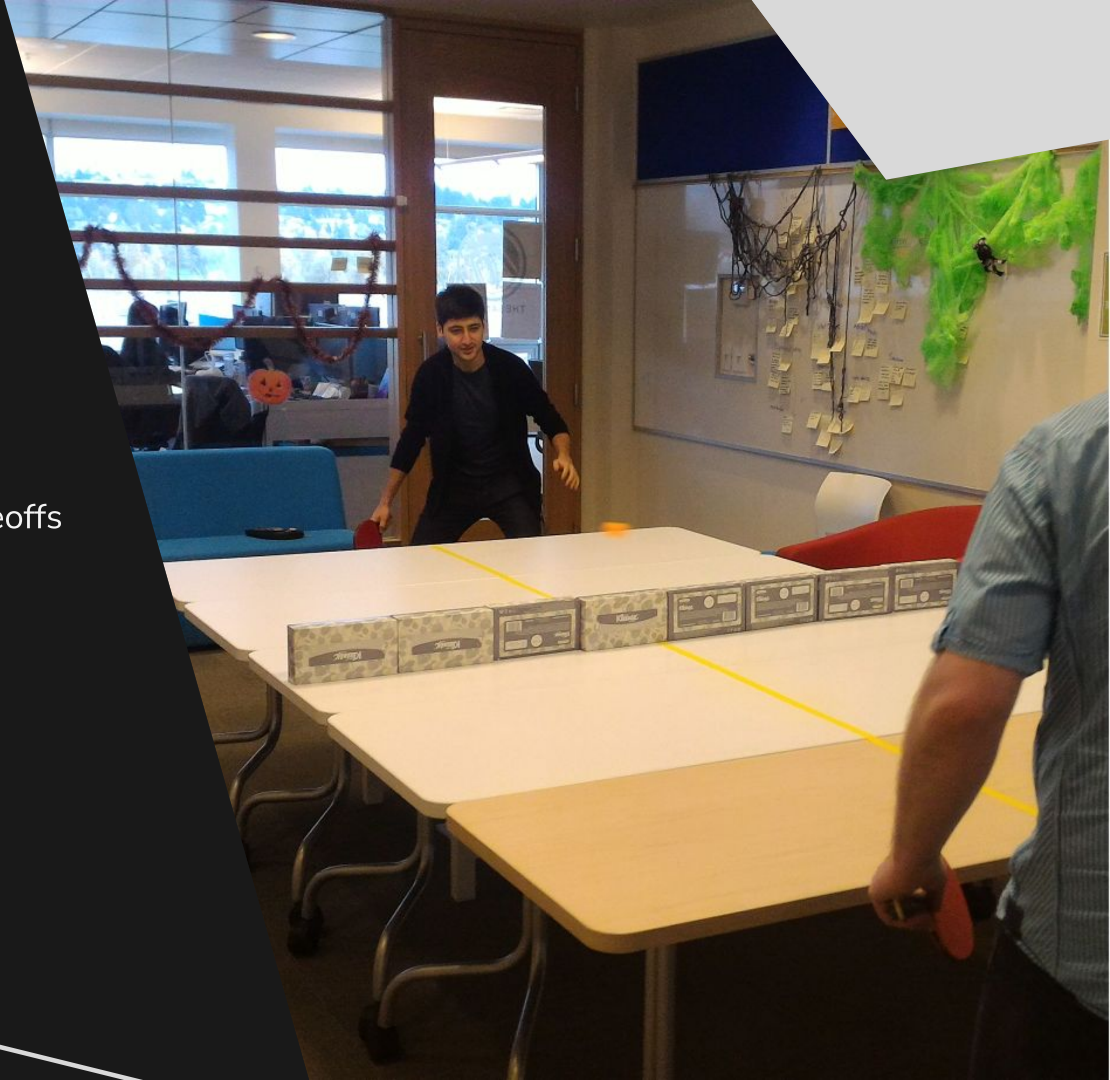
05.2

WHY PRACTICE?

Pause to think about a design and its costs

Vocalize your work and actions

Quickly analyze time/space complexity for tradeoffs



05.3

WHAT ABOUT FAANG?

Phase 1: Application

Phase 2: 1-1 screening or online assessment

Phase 3: 4 consecutive rounds of behavioral & technical interviews

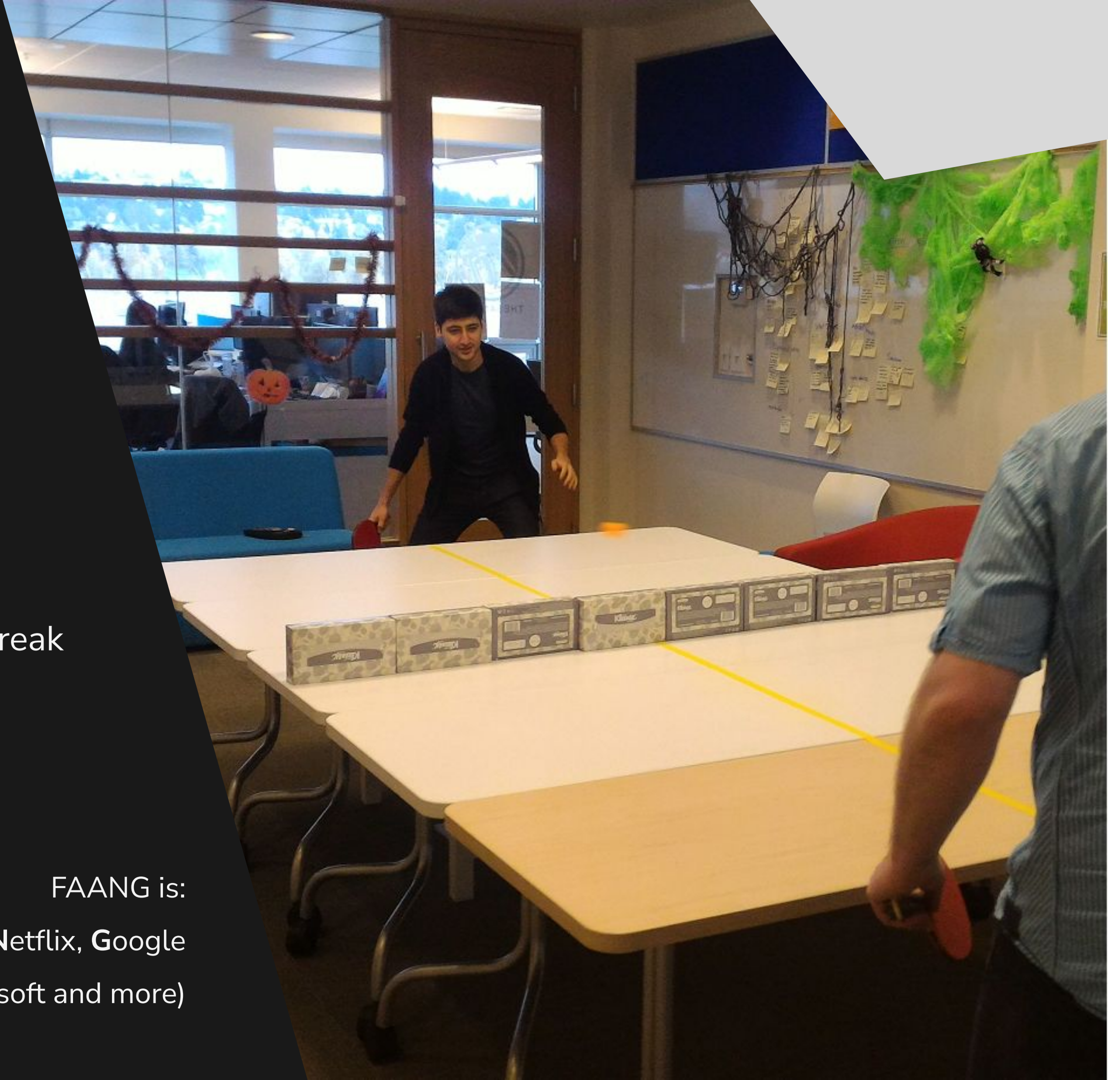
Each interview is 45 mins with 15 mins break

1-2 behaviorals, 1-2 technical questions

FAANG is:

Facebook, **A**mazon, **A**pple, **N**etflix, **G**oogle

(+ Microsoft and more)



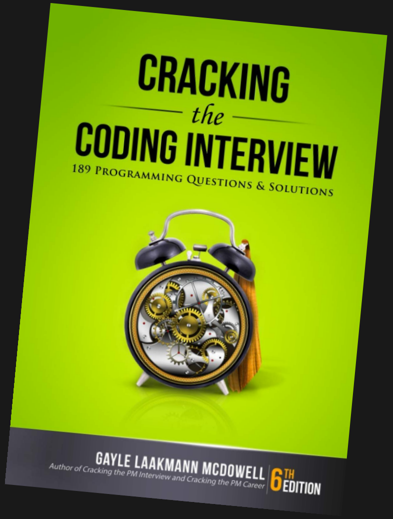
How do you conquer your fear of technical interviews?

Get Cracking The Coding Interview.

Start practicing now.

Practice consistently.

Fail until you start succeeding.


CRACKING
the
CODING INTERVIEW
189 Programming Questions & Solutions
GAYLE LAAKMANN MCDOWELL
Author of Cracking the PM Interview and Cracking the PM Career
6TH EDITION

QUESTIONS?

RESOURCES & CONTACT

RESOURCES

Cracking the Coding Interview: Learn to do coding interviews

LeetCode: Coding practice questions

Steven Zhang - Landed 18+ Software Engineer Offers

Work at Google — Example Coding/Engineering Interview

LeetCode Solution (217 - Contains Duplicate)

INTERVIEWING AT FAANG

How to Get a Job at the Big 4 by SJ Tech

How To Get Hired into a FAANG Company

ONLINE COMMUNITIES

Reddit:

- Computing Science
- CS Career Questions
- Programming

Hacker News

Nova - Software Engineering

Jeffrey Leung

Reach out to me through:
jleung513@gmail.com
LinkedIn
Facebook