



Technical Interviews



GPI Extratation | 9/14
By: David Hashe & Amy Liu



A meme featuring a close-up of Steve Moss from the TV show 'The Office'. He has a worried and questioning expression on his face, looking directly at the camera. He is wearing his signature glasses and a light blue button-down shirt. The background is a blurred office setting with another person visible in the distance.

**AM I READY FOR THIS
INTERVIEW?**

**FALSE. THE QUESTION IS, IS THIS INTERVIEW
READY FOR ME?**

Outline

- What is a technical interview?
- Different types: phone screens vs on-campus vs onsite vs coding challenge
- How to even get a technical interview?
- Techniques
- Examples
- Questions to ask the interviewer
- After the interview
- How to practice, followed by some Leetcode example

What is a technical interview?

- You will be expected to write some code, usually on a whiteboard or a notebook
- The problem will usually have several reasonable solutions of varying time and space complexity, and your goal is to eventually arrive at the optimal one
- They usually test purely algorithmic knowledge, but sometimes they include domain knowledge like details of C++ syntax (if you're applying to a C++ role)
- They will also sometimes require writing a lot of code very quickly (Citadel and Jane Street love to do this)

Sub-Types of Technical Interview

- The do-you-know-this-language interview:
 - “Please explain what the Virtual keyword does in C++”
- The write-a-lot-of-code-quickly interview:
 - “Here is a (somewhat open-ended?) problem with low-algorithmic content, please write a reasonable solution and explain your design”
 - “Implement Tetris”
- The algorithms interview:
 - “Here is a problem that has many algorithms with different complexities. Find the best one and implement it”
- The domain-knowledge interview:
 - “Here is a problem from our industry, explain how you would solve it under these constraints”
 - May or may not involve code

General Interview Process

1. Hackerrank / Online Coding Challenge
2. Technical (and Recruiter) phone screens OR on-campus
3. Onsite interview
 - Also known as the “free stressful vacation”
4. Offer
5. Profit

How to even get a technical interview opportunity?

- For Hackerranks / coding challenges, some companies give them out almost automatically upon applying. If you want practice, APPLY APPLY APPLY!
- Go to career fairs / company events / stands / chats / etc and talk to recruiters or engineers.
- Get referrals from your friends (especially important for the Big-5)
 - Google, Apple, Facebook, Microsoft, Amazon
 - Upperclassmen are your friends here

Techniques

- Get a good night of sleep & eat properly

Sleeps for one hour

Fails interview



Techniques

- The flow
 - Clarifying questions
 - Brainstorm data structures before coding
 - Think of the brute force solution first
 - State that you know this is too long / inefficient / blah, state why
 - Now try to improve upon it
- You can ask for hints if you're absolutely stuck!

Techniques

- Think out loud
 - Interviewer is more concerned with your thought process in achieving the answer than correctness of the code
 - More than 10 seconds of silence **without progress** is bad
- Incremental difficulty
- Utilize constraints

Example: Intersection of 2 arrays

- You have two sets, represented as array with no duplicates. The arrays are small enough to fit in memory. Find the set-intersection of the two arrays (the elements present in both).
 - Store the elements of one array in a hash table.
 - Loop over the elements of the second array, outputting anything that was already in the hash table.
- Okay, now what if neither array can fit in memory, but the arrays are sorted?
 - Keep an index into both arrays. While neither index is at the end of the array, compare the elements at those indices. If they are the same, output that value and increment both indices. Otherwise, increment the index whose array element was smaller.

Example: The Virtual Keyword in C++

- In my experience this is the most asked interview question for C++ roles
 - I have answered it at least three times for three different companies
 - People ask it because it is a good separator between people who actually know C++ and people who only know C.
- You have two classes, A and B, where A inherits from B, and you construct an object obj of type A but cast it to type B. Then you call method obj.foo().
 - If foo() was virtual, then A::foo() would be called.
 - If foo() was non-virtual, then B::foo() would be called.
- For non-virtual functions, C++ bakes in the implementation from the “static type” of the object, whereas virtual functions call into a lookup table that retrieves the implementation for the object’s “dynamic type”.
 - This behavior is nicer, but has a runtime cost and so should be avoided for performant code.

Questions to ask the interviewer

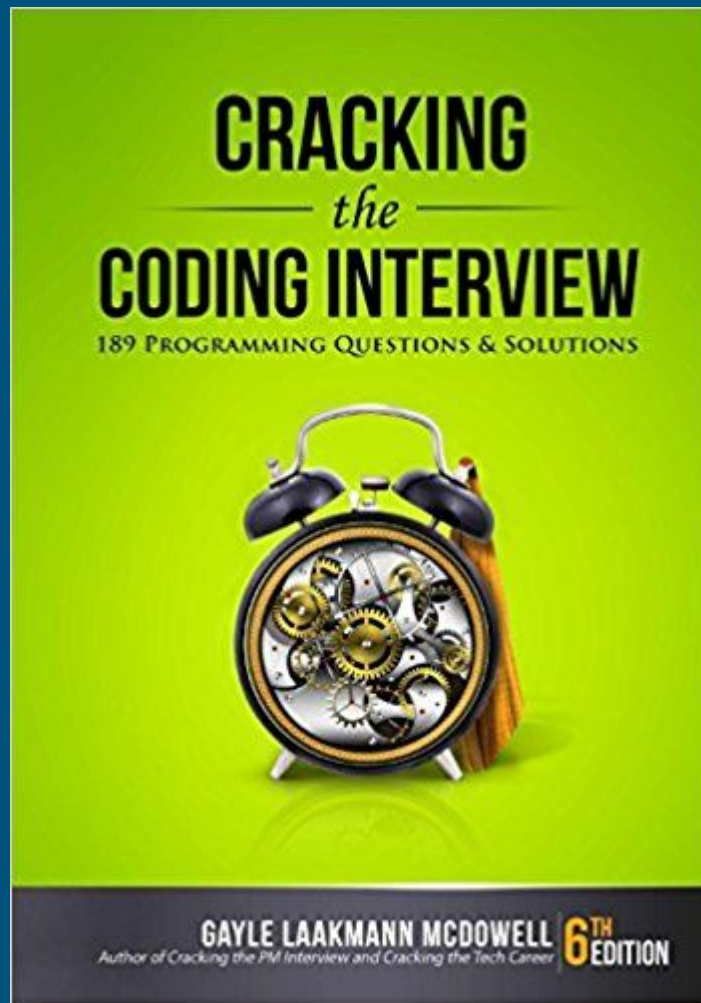
- Name, job title, maybe email (for reference later on)
- What made you choose company X?
- What's the most satisfying project you've worked on?
- What's a typical day for an intern like?
- Any example projects interns have worked on?
- What's your favorite thing about working for your company?
- How does this company compare to other places you've worked before?

After the interview

- Thank them for their time!!
- (If you didn't bomb it) and didn't hear back in 1 weeks, consider following up with the recruiter / interviewer!

How to practice

- Leetcode.com
- Cracking the Coding Interview
 - This book is amazing, seriously
- Being a TA
 - This is literally your job
- Rubber ducky method
 - Vocalize, vocalize, vocalize!
- Visit Kevin Collins or Career Center for tips or mock interviews!



For tomorrow:

Let's Talk happening on Gates 6 from
3:10-6:00pm.

Don't bring backpacks!

Do bring resumes!

Remember to take a selfie and submit it to
google form for another extratation credit!

<https://forms.gle/fydeeBJXUTBaQh1M6>

Leetcode time!
Q.538 & 983