Running the Gauntlet - LeetCode Discuss

Summary

This post details interview experiences from earlier in 2020. Recruiter calls took place mostly during February, phone screens during March, and (virtual) onsites during April. Due to hiring freezes at some companies, some of the interview processes were cancelled midway through. I applied only via company websites (no referrals). A couple of company processes started from recruiter messages on Linkedin.

I hope that this post can provide some insight into the various processes at different companies. I interviewed at a few companies which I don't see mentioned often on this site so perhaps this can be useful to anyone preparing for interviews at these places.

The overall results:

Reject: Dropbox, Nextdoor, Facebook
Cancelled (hiring freeze): Akuna, Two Sigma
Cancelled (by me): Spotify, Google
Offer: Amazon, Microsoft, Snapchat, Robinhood

Background

- YOE: 3 (as a SWE at a well-known investment bank)
- Bachelor's in CS
- US Citizen
- Current TC: 125k
- Leetcode: 350

Timeline

- Early December 2019: Started preparing for interviews
- Mid-February 2020: Started applying to companies online
- Mid-May 2020: Signed an offer

Dropbox [reject]

Applied on company website and was invited for a 1-hr phone screen. The interviewer was some kid who graduated like 4 months earlier. That's when I knew how this was about to go. In my experience, getting one of these hotshots means the interview won't go well, and unfortunately, I was right. The question was called space panorama or something (pretty sure it can be found online) which is a bit more of a OOD design question rather than a typical leetcode algorithm question. The interviewer had negative communication skills and I couldn't make sense of what was being asked, so I couldn't make too much progress on this. Got the rejection I expected a few days later. It was one of the first technical interviews I did during this process and I wasn't quite "in the zone" yet, so I just took it as a learning experience.

Nextdoor [reject]

Applied via Linkedin and was invited by a recruiter for two phone screens (both had video as well). These are actually part of the same round (back-to-back) but I did them on two different days.

Phone 1: Leetcode 150 with a follow-up to convert a postfix expression to infix.
Phone 2: Leetcode 56

After that I was invited to a virtual onsite, which consisted of a system

design, 3 coding rounds, and a discussion with the hiring manager.

1. Coding: Not a leetcode question, but a fairly simple question using hash tables.
2. Coding: Leetcode 165
3. Coding: Not a leetcode question, something about writing a connection manager for HTTP requests. Interviewer also asked about what happens when a URL is typed into the browser.
4. System design: Design Zillow. They called it a "problem solving" interview, but it's really just system design.
5. HM round: Behavioral. Work experience, interests, what other companies I am considering, etc.

Overall I thought it went well. The recruiter said I would hear back at the very latest by the following Tuesday. Then I got ghosted and got a standardized rejection email the next Thursday. I felt it was a pretty poor candidate experience to get rejected in that way, although all of the interviewers were nice enough to talk to. I think they were for some reason trying to slot me into an infra role, whereas I'm more of a product engineer, so they may have determined it wasn't a good fit.

Facebook [reject]

Applied via website and was invited for a 45-min phone screen. I haven't seen this question on leetcode but it was about concatenating a given list of strings into a single string based on some rules that involve modifying the strings. (Such as enclosing a string with double quotes if it contains a semicolon, and replacing any double quotes with double-double quotes.) The follow-up was decoding the single string back into the original list. I didn't actually complete the follow-up part but I was somehow invited for an onsite anyways. I spent some time on my own after the interview finishing

my implementation, and I realized I was actually pretty close to solving it during the interview itself. I got feedback from the recruiter that I got points for writing and running through my own additional test cases for the problem.

The onsite consisted of 3 coding rounds, a behavioral round which also included a coding question, and a "product design" round.

1. Coding: The first question was to validate a string compression, given the original string and the run-length encoded version of the string. The second question was Leetcode 124. Solved both without much difficulty.
2. Coding: Leetcode 716 and Leetcode 609. Solved both of these as well.
3. Behavioral: Basic questions about work experience and past projects. Then a very quick coding question which was basically Leetcode 332.
4. Product Design: I was expecting something like a system design round but this guy instead started asking me about designing a client-side library for a mobile application. I was like I have literally no clue about either client libraries or about mobile development. I have no idea where this came from or what signal he was trying to establish by asking this. I discussed some of the APIs that would be used, but I imagine this round didn't go too well.
5. Coding: First question was a much easier version of Leetcode 621, in which the final task sequence had to be in the original order of the given tasks. It can be found in the discussion section on leetcode. The second question was Leetcode 983. I started talking about dynamic programming and the interviewer stopped me saying let's not go the DP route. So I said ok let's do backtracking. I wrote a solution (which I later verified the correctness of) but I think the interviewer wasn't convinced. This guy seemed pretty arrogant (when I asked why he joined FB, he said because it's prestigious and hard to get into??). I

assume he failed me.

Got a rejection email from the recruiter after 3-4 days. I was pretty surprised because I thought the many (literally 7) coding questions went well, but I was probably just very optimistic given some of the acceptances I had received earlier that week. I'm a bit envious reading on here that some people were given a choice between a "system design" round and a "product design" round. I would've happily gone the system design round, but I wasn't given a choice. I also seem to have had more coding rounds than is typical.

Akuna [hiring freeze]

I applied on the website and was invited by a recruiter to begin the interview process for a junior quant developer role. The first round was a phone screen by a senior quant researcher. The problem was to design a class that would be a real-time performance monitor. The goal was to implement max, mode, and mean of a set of numbers, and the follow-up was to implement sliding window max, mean. I also had to implement max stack and max frequency stack. After that we had a discussion around compiler optimizations and arrays vs linked lists. At the end, the interviewer said this went really well and that I would be invited for an onsite. Unfortunately I heard from the recruiter a few days later saying that hiring for this role had been paused.

Two Sigma [hiring freeze]

I applied on the website and was invited to complete a Hackerrank assessment. I don't really remember either of the questions but they were probably at Leetcode-Medium level. One of them was called jumble sort or something, you can probably find it online. After that I was invited for a

phone screen, in which I was asked to design a random number generator class. The goal was to create an API that would output random numbers from a given range with no repeats. I was able to solve this as well as a related follow-up question. After that I was invited to provide my availability for an onsite interview, but later was told that hiring for the role had been paused.

Spotify [cancelled]

Applied on the website and was invited by the recruiter to interview for a data engineering role. First round was a phone screen with video. There were some questions about CI/CD and MapReduce (which I don't know much about). This was followed by a coding question Leetcode 23, which I solved pretty easily.

The recruiter followed up saying they felt a backend software engineering role would be a better fit for me, rather than a data engineering role. (This is absolutely true). So I was invited to another phone screen for that. There were some basic questions about past experience, and then a coding question Leetcode 108.

I was invited to schedule an onsite, but I declined to do so after accepting another offer.

Google [cancelled]

A recruiter had reached out to me many months ago when I wasn't looking for a new job. I reached back out when I was ready and began the standard process.

The 45-min phone screen problem had to do with detecting whether RPC calls had timed out, given a list of start and end times for calls and a timeout

threshold. I don't think it's a leetcode question, but I've seen something like it in the discussion section. It took a long time for the interviewer to explain it, and we wasted a lot of time because he messed up the explanation initially. I coded in the last 10 minutes but it wasn't an optimal solution. The problem isn't hard so I assume this was a fail.

The recruiter came back and said they recommend a second phone screen. In that I was asked to implement a skip iterator (seen something like it in the leetcode discussion section). I did that quickly and was asked a second question about finding the max root-to-leaf path sum in a k-ary tree. There are probably similar leetcode questions.

I was invited to schedule an onsite, which I declined after accepting another offer. I'm sure many would regard this as a bad move, but I knew Google would take eons to finalize an offer and I didn't have that kind of time. They also have severely limited hiring across the board and I was afraid this would affect the outcome.

Amazon [offer]

Similar to Google, a recruiter reached out a while ago when I wasn't looking, and I followed up when I was ready to begin the interview process.

First round was the online assessment. There were two questions, both of which are in the list of Amazon OA questions. They were Leetcode 937 and the treasure island problem. I passed all test cases in both problems. There was also some kind of personality test.

After that I had a 30 min call with the recruiter in which I was asked a behavioral question regarding constructive feedback I had received from a manager. They also asked some simple theory questions about binary

search, hash tables, and recursion.

Next was the onsite, 4 rounds, including 1 system design.

Round 1

- LP: time when you did something out of scope of your role
- LP: time when you had a disagreement with your manager
- Coding: something about Quadtrees, I think there is a similar leetcode question

Round 2

- LP: time when you had to take a calculated risk
- LP: time when you innovated something
- Coding: a variation of Leetcode 347

Round 3 (bar raiser round)

- LP: what changes you would make to a system in hindsight
- LP: negative feedback received from manager
- Coding: something about designing an online poker website

Round 4

- LP: time when you solved a complex problem and how you got information needed to solve it
- LP: time when you delivered under a tight deadline and what was the outcome
- System Design: news aggregation service

After the onsite, the recruiter called the next business day to say they would be giving an offer.

However, the onsite was an SDE2 loop but I was being downlevelled to SDE1. Between that, the need to relocate, and the terrible things I've heard about the culture, I opted against taking this offer.

Microsoft [offer]

So Microsoft was interesting because I actually went through two loops with two different teams at the same time (I didn't tell them). I got the first one by applying online and the second from a recruiter messaging me on Linkedin. Both loops consisted of an online assessment (codility) followed by a virtual onsite. The first was for a specific team (got the offer for this) and the second was an Azure hiring event.

Azure - Codility

1. Fizz Buzz
2. Find the max possible value of a number achieved by inserting a '5' at any position

Azure - Onsite

1. I was asked Leetcode 126 and Leetcode 127 and solved both.
2. This guy asked me some random questions about security issues in designing systems. My background has literally nothing to do with this (and neither does the interviewer's) so I don't know what his deal was. He also didn't bother turning on the camera. Each round was supposed to be 1-hr, and he was done asking questions after 20 min. All the questions were about computer security and I just had to BS everything.
3. This was with a fairly senior person who asked some questions about my background and interests, and then asked a coding question Leetcode 340. Solved it without issues.

The recruiter called me to say they would not be proceeding with the offer. I got feedback that the team thought my technical skills were strong, but they were "hoping I dug deeper into the technical details of my past work rather than talking at surface level". I was thinking, how stupid can these people be? If they wanted deeper technical explanations then they should ask. Candidates can't read their minds.

Msft (Other Team) - Codility

1. Debugging a short program meant to reverse the digits of a number
2. Minimum cost to remove adjacent duplicate characters in a string (each index has a cost)
3. Fair index (can be found in the Microsoft OA problems list)

Msft (Other Team) - Onsite

1. Validate an arithmetic expression that has both numbers and variables, as well as parentheses. I basically made absolutely zero progress in this and the interviewer gave no hints. Also there were technical glitches with the codility. I entirely lost all motivation to continue this process after this.
2. Some basic questions about my past projects followed by Leetcode 387. Solved it easily.
3. Some more questions about my past projects followed by a coding question about finding the max width of a binary tree. There is a similar leetcode question, but the interviewer defined width of a level as the number of nodes at that level, which is different from the leetcode question. I solved it first with a BFS approach, but the interviewer was clearly looking for something else. Unfortunately he kept asking "what is wrong with this solution?" which is an entirely useless statement because the solution is correct. After some time I realized he was

referring to optimizing the space complexity. I explained to him that even a recursive DFS solution would require the same space complexity. I still don't know if there is a O(1) solution for it.

4. The last round was with the head of the group. I was asked Leetcode 273 which is a terrible interview question but I knew how to do it, so at least this round went well.

I was definitely expecting a rejection from this but somehow the recruiter said that feedback from the onsite was positive (???), but unfortunately the team paused hiring. After like 3 weeks they reached out again saying that they were able to get one more slot and they could extend an offer. They were offering at L60 which I thought was low for my YOE. I've had people with one less year of experience on my own team leave for Microsoft at L61. This combined with the very low compensation I was offered led me to decline this offer.

Snapchat [offer]

Applied on the website and was invited by a recruiter to interview for a particular team. The first step was a phone screen with video. The question was basically Leetcode 146 (implement LRU Cache). I tried to use the OrderedDict approach in Python but the interviewer said to do it the long way. This question should pretty much be memorized during any interview prep process so at this point it was just a matter of rolling my eyes, shutting off my brain, and banging it out.

I was then invited for a virtual onsite which consisted of 4 rounds.

1. Leetcode 53 with a follow-up to return the actual start and end indices of the max subarray.
2. Shortest path in a binary matrix (find shortest path from src to dest in a matrix of 0s and 1s, where the 1s are obstacles, and you can travel

up/down/left/right). I'm sure there are equivalent leetcode questions but I couldn't find an exact copy. The follow-up was to print the path.

3. Leetcode 253 with a follow-up to write a class that would assign meetings to room numbers
4. System design question about a cloud photo storage system, including a discussion about scalability.

All the rounds went really well. The recruiter messaged me to set up a call the next day and then ghosted me til the next week. During the call he said feedback was positive, but they were still interviewing other candidates and are waiting to make a decision until all of them are done. Why he told me this is beyond me. He said he would be in touch by the end of the week with the outcome. Then he ghosted me for like two more weeks and said they were ready to make an offer, and that he would talk to the compensation team to get initial numbers. Then he ghosted me again. No seriously, I actually never heard back from them after they said they would follow up with the compensation package.

Robinhood [offer]

Applied online and a recruiter reached out and sent me an online assessment. There were three questions, all of which were basically Leetcode-Medium level. One was validating whether a given string matches a given pattern. Another one was about renaming files in a given list of files to avoid duplicate filenames. The last one was to convert a grid of numbers to a grid where each value is the average of its neighboring values from the original grid.

Recruiter emailed me the next day saying I passed the online assessment and they want to schedule a 1-hr phone screen. The phone screen is 1 coding question and 1 architecture question. The coding question was

about computing the number of trades executed from buy/sell orders with prices and could be solved using two heaps. I had the correct approach but had some bugs in the code which the interviewer helped correct. The architecture question was about concurrency and reliability issues with database read/writes. I don't know too much about this topic but I just guessed the answers and got lucky.

I was invited to the onsite next, which was 4 rounds.

1. First was a coding question. I don't remember what it was but it was basically a non-leetcode style easy question. It was solved fairly simply using hash tables.
2. This round was an algorithms round. The interviewer asked what I imagine is some kind of variation of the knapsack problem. It was something about maximizing the profit of jobs where each job has some profit and a deadline by which it has to be completed, if at all. The goal was to discuss how to solve it, not write any code. We used a whiteboarding tool rather than a code editor.
3. This round was a deep dive into a past project. I simply drew the architecture of a system I had worked on in the past and explained it to the interviewer, and I answered his questions about it. I guess the aim is to assess communication skills. The interviewer wasn't even a backend engineer so he didn't really dig too deeply into the technical details, so I got lucky in that sense.
4. Generic system design round. Design a cron job scheduler.

I heard back 1 week after the onsite that feedback was positive and they are making an offer. It also turned out to be an exploding offer, which was annoying. That meant I unfortunately had to decline this offer as well, while waiting for some other decisions.

I also talked to a few other financial companies (Addepar, Virtu, Tudor, Optiver) but mostly was not interested in continuing the interview processes with them. I was contacted by these primarily via headhunters on Linkedin. Addepar had an extremely cumbersome take-home assignment about merging files. Virtu gave possibly the easiest online assessment I've seen, which was followed by a call with a recruiter who asked some of the strangest questions I've ever heard. Tudor asked simple coding questions like max stack and fizz buzz, but they were obsessing over programming language theory, such as tail recursion in Scala, mutable arguments in Python, and memory leaks in Java. Optiver gave a hackerrank assessment which had one question about error handling while parsing a binary tree string input and one question about finding the number of days between two given dates.

Resources

- Leetcode (obviously)
- Elements of Programming Interviews (EPI) - Python
- G_r_o_k_k_i_n_g - System Design
- Gaurav Sen (Youtube) - System Design
- System Design Primer
- The Amazon articles at Interview Genie were super helpful for behavioral round prep

I spent the majority of the time doing Leetcode problems closely following the EPI book front to back. Then I spent around 1-2 weeks studying system design. I had never done a system design interview before and had zero experience with the topics. The System Design Primer and Gaurav Sen videos were extremely helpful. G_r_o_k_k_i_n_g is good, but since I did it last, I didn't find any material there that was new to me. I spent 2-3 days

preparing stories for behavioral rounds.

Takeaways

It is widely known that Amazon really cares about the LP questions, and my experience was the same. I felt my performance on the coding questions was ok, but preparing for the LP questions and having solid answers definitely pushed me over the edge. I had at least 2 example stories for each LP. Amazon was the first onsite I had, so preparing for their behavioral rounds also helped tremendously in the behavioral rounds of other companies.

Microsoft interviewers are somehow looking for a particular solution that they have in mind. There can be multiple optimal solutions to a given problem, but they will always try to push you to the one they are thinking of. Other than that, there is a huge amount of variability in Microsoft interviews. This is likely because the interviews are team-dependent rather than company-wide. Whereas interviews at Google or Facebook are highly structured and one can expect the same kind of interviews each time, interviews at Microsoft depend entirely on whatever the interviewer wants to do. Personally, I strongly prefer the company-wide interview processes that are very structured.

Facebook interviews are ripped straight off of leetcode, and then they have the audacity to ask if you have seen the problem before. I'm curious if anyone out there is admitting to having seen the problem. The trick is to ask them to repeat the problem a few times as if you have no idea what they're talking about, and then ask them to run through a quick example for you. Then just go ahead and type out the code that you already have memorized.

Max/Min Stack are absurdly popular questions. I think I was asked this at

least 3-4 times. It seems to be especially popular with companies in the finance industry.

It is extremely hard, as a candidate, to judge how an interview went from the interviewer's perspective. In some cases I thought I had done well and did not receive an offer (Nextdoor, Facebook). In other cases I thought there was room for improvement yet I got an offer (Amazon). And there were even some in which I felt I had miserably failed and yet got an offer (Microsoft). It seems to me that which rounds go well is very important (rounds with senior interviewers carry more weight than rounds with junior engineers).

Perhaps I got lucky but it is super easy to game the system design round. It could be that I have only a few years of experience so the bar is not very high, but I was able to use the same basic template and got good feedback on every system design interview round. In reality I know nothing about designing systems and have no experience with it.

Things I would probably do differently next time: (1) use LinkedIn better by directly messaging recruiters, and (2) get referrals. I had a surprising amount of success just applying online.

Conclusion

First of all, I hope the walls of text aren't too cumbersome to read, so apologies if it gets too annoying. The same goes for not linking to the actual Leetcode questions, but I think it's not too tiring to just search them up by number.

Finally, if you read through the offers section, you probably figured out that I didn't actually accept any offers that I've listed here. In fact, I ended up accepting a role at another company that I haven't discussed here. Perhaps I will make a post about that some time. Until then, good luck to everyone

out there.