Technical Interview Prep

Interview Overview

Our interview style may differ from what you've experienced elsewhere and **tends to be highly technical**. The interviewer will be interested in the specifics of your past projects, implementations and how you arrived at your conclusions.

Technical Assessment

Our interviews will typically cover the following topics: CS Skills/Theory, Algorithms & Data Structures (i.e. Big-O Notation), Debugging & Performance, Object-Oriented Development, Multithreading, Operating System Fundamentals / Wildcard, System Architecture, Coding, Software Testing, System Design. It may also be worth refreshing on hash tables, heaps, binary trees, linked lists, depth-first search, recursion (basically CS 101 concepts that everyone knows but sometimes doesn't come right off the tip of one's tongue at a moment's notice).

Coding: Did you write readable code? Checked errors at boundary conditions? Took advantage of language constructs and data structures, and corrected errors? Used a coding style that was self-consistent? Were you able to produce a desired or intended result? Think about ways to improve the solutions you present. In many cases, the first answer that springs to mind may need some refining (especially concerning edge cases you may have initially overlooked). It is worthwhile to talk about your initial thoughts to a question. A brute force explanation will be received less well than taking time to compose a more efficient solution. **Ask clarifying questions** if you do not understand the problem or need more information. Some questions may be deliberately underspecified because our engineers are looking to see how you engage the problem. In particular, we are looking to see which areas leap to your mind as the most important piece of the technological puzzle you've been presented. **Think aloud & about ways to improve the solutions you present, give interviewers a window into how you think.**

Data Structures and Algorithms: Did you identify data structures to design a solution with optimal efficiency? Did you propose multiple algorithmic approaches to single and compound problems? Did you identify an optimal solution? Did you propose creative solutions and/or come up with alternative optimal solutions for different constraints?

Comprehension and Communication: Did you show comprehension of the problem(s) and identify supplemented contingencies/challenges? Were you able to provide examples and/or alternative explanations to clarify rationale? How effective were you at communicating code, algorithms, etc.

Machine Learning example questions: predicting the likelihood of user engagement for ads, classifying spam tweets, predicting user's interests.

Common Evaluation Areas

Problem solving

The ability to create solutions to interesting & challenging problems

Iteration

The ability to break large challenges into smaller batches

Ownership

The ability to take responsibility for work, solve roadblocks

Craftsmanship

Are you really, really good at what you do and the tools of your trade

Collaboration

Team-oriented; enjoys and favors solving problems with others

Communication

The ability to clearly and concisely explain technical topics; ability to listen and take feedback

Impact

Focus on having a large impact on users and the company you work for

Learning

Continuously learns to strive for both depth and breadth of knowledge including why things work the way they do

Adaptability

Passionate about solving user problems but open to whatever it takes to solve that problem

Technical Interview Prep

Helpful Resource Links

Project Fuller

TopCoder

Cracking the Coding Interview

Programming Pearls

Intro to Algorithms

Programming Interviews Exposed

Twitter Engineering Blog

About Twitter

Additional Preparation Tips

Past Projects & Resume
Brush up on your past projects and your resume in general.

Your Interviewer is your collaborator
Consider your interviewer a collaborator on the problem. Share what you're thinking with them and ask them for help when needed.

Why Twitter?
At Twitter, our team is passionate about the product and our engineering culture. Before your interview, ask yourself "Why Twitter?" Take some time to think about what you like about Twitter's product, what you could do to improve it, and what skills you could bring to the team and organization as a whole.

Behavioral Questions
Be prepared to answer questions around: Building Trust, Conflict Resolution / Getting It Right, "Simplify"

Questions to ask your Interviewer
What do we like most about working here?
What do we struggle with?
How do we balance our professional and personal lives?
And/or anything else you're interested in knowing!

And Most Importantly...
Have fun! And, good luck!