

Problem Solving Workshop #16

Tech Interviews and Competitive Programming Meetup

November 12, 2016

<https://www.meetup.com/tech-interviews-and-competitive-programming/>

Instructor: Eugene Yarovoi (can be [contacted](#) through the group Meetup page above under Organizers)

More practice questions: [glassdoor.com](#), [careercup.com](#), [geeksforgeeks.org](#)

Books: Elements of Programming Interviews, Cracking the Coding Interview

Have questions you want answered? Contact the instructor, or ask people on [Quora](#). You can post questions and [follow the instructor](#) and other people who write about algorithms.

Generally speaking, in design questions, there's always lots of ambiguity, so you would start by asking clarifying questions to the interviewer. Since you don't have an interviewer here, assume what you think are reasonable answers to the questions you would ask.

Requirements Analysis & High-Level System Design

How would you design "Amazon Lockers?"

Imagine Amazon Lockers is a new service that is going to be offered by Amazon.com. When a customer checks out on Amazon.com, instead of a shipping address, they can choose for their purchase to be delivered to a locker at an Amazon Locker facility near their residence, so that they can pick it up there whenever they're free. Suppose you're the technical lead on this project. What software components might need to be built to enable this service? How would you design them and how would everything come together?

Data Pipelines & Distributed Systems

Imagine you are Twitter. How would you design the "top 100 trending hashtags" feature, if it were not already designed? The goal of this feature is to inform Twitter users about the currently trending hashtags (which are understood to represent current "buzz", news, and events of cultural significance). Consider that there are approximately 500 million tweets/day on Twitter.

Some points you should be sure to address:

- How do you propose to define "top trending hashtags" so that it makes sense relative to the stated use case?
- How will you know which hashtags are the top trending 100 at any given time?
- When you design this feature, you should assume Twitter is already a large-scale distributed system. How will you add the logic necessary for your "trending hashtags" service to the existing Twitter infrastructure? To answer this, you will need to state some reasonable assumptions for what the Twitter infrastructure looks like, and how you would add your service on top of that.
- How will users access the top trending hashtags?

Distributed Systems & Algorithms

The interviewer states the problem like this: Consider an existing large website where there are lots of images submitted by users. For example, Wikipedia, Pinterest, DeviantArt. How would you design a web crawler to crawl all pages belonging to that domain, and download all the images?

At this point, you would need to ask clarifying questions before you can attempt the problem. Imagine the conversation goes as follows:

You: How large is the website and how many images do you estimate there are?

Interviewer: The site may have on the order of 1 billion images.

You: Each image may be 100KB-1MB, so we're talking hundreds of terabytes of data. I assume we want to download this data into some kind *distributed* database, since 1 machine can't hold this.

Interviewer: Yes. And no requirement to organize the images in any way, just download them all.

You: Can I assume I have the storage set up and an API call that adds 1 image to the storage?

Interviewer: Yes. Just design the crawler.

You: How do we know what pages are available in the domain? Is the website organized so that we will find all the pages by starting from the homepage for the domain and following links?

Interviewer: Yes. Also assume for the time being you don't have to deal with robots.txt or the like.

You: Can I assume a static page structure too, the link structure of the site isn't changing much?

Interviewer: Start with that assumption, you can think how to weaken this assumption later.

You: Can I assume images are referenced by img tags, and they're not embedded in some other medium like Flash?

Interviewer: Yes. The images will be easily accessible like that.

You: Because there's so much data, this process may take a long time.

Interviewer: Yes. Think about your design in the context of this huge scale.

You: And probably the websites will detect you're crawling them and block you.

Interviewer: True. Assume that doesn't happen, or that you can circumvent it.

At this point, there's still plenty of ambiguity, but assume reasonable answers of your choice for your remaining questions. Design the system.