Sample Tech Questions

Logical/Maintainable

* Implement a function to return the N highest rated products that are considered similar to a given product. Each product has an id, a rating, and a list of products it is similar to. Similarities are transitive
  + Ex of clarifying questions
    - Would all products in the example be similar
    - Can I assume all products are valid
    - Can I return an array
    - Can there be cycles
    - Is product ID unique
* Write a program that will give the fewest number of bills necessary to provide the correct change for a given dollar amount. In other words; given a sum of money, return the minimal bills that would make up that sum

Problem Solving

* Given two arrays, identify if they represent the same BST or not without constructing the BST (binary search tree)
* Amazon Locker system: if I give you a package and you tell me which locker it would need to go into?
  + Edge cases, etc.

Data/Algorithm

* Given a list of words the contain compounds. All words are in the list. Identify all combinations where one word is a composite of two or more words from the same list and print or return them
  + Ex. Air, plane, airplane

System Design

* Design a tinyURL system
* Design Uber or Netflix from scratch
* Design an air traffic control system

Sample Leadership Principle Questions/Answers

**Tell me a time when you took on something significant outside your area of responsibility**

During my time at “company” we had a customer that reported a bug every 2 weeks that surfaced when they tried to publish their company schedule for guard shifts. The bug would duplicate shifts for this particular customer, causing them to have to manually delete the duplicated schedules. The bug was passed around for a few months to various senior developers, but no one was able to reproduce it based on the customer's parameters. As my company was about to give up and mark it as a "won't fix", I asked to take the challenge on the side. I approached it differently; rather than trying to reproduce it like everyone else had done. I did a deep dive into the code base and identified the possible areas for the bug. Eventually, I was able to find a faulty logic that used ">=" instead of ">". In the end, I was able to resolve the issue for the customer (despite only 1 customer experiencing the bug). Then I wrote a new functional test to prevent the bug from happening again. I ended up receiving recognition from my manager, who called out determination as one of my biggest assets.

**Describe a time where you felt really strongly about something but ultimately lost the argument. How hard did you press the issue? What was your approach after you lost the argument?**

I was working on revamping a client's application as a team lead. During this time a bug was reported by stakeholders 1 week before the deadline. After investigation, the bug was in a dependency framework that the team had used. This bug was also fixed in the framework with a major version bump. A developer on my team wanted to upgrade the dependency, but I wanted to do it as a post-launch item since there could be compatibility issues with major version upgrades. While I believed that doing it later to de-risk the project would be the correct choice, I wanted to involve my team to promote healthy team culture. I gathered the team and proposed both solutions. The team wanted to do the major version upgrade, so I disagreed and committed to the decision by giving the developer 2 days, the maximum allotted time they have to avoid slipping, calculated from their existing sprint metrics. After 2 days, the developers couldn't do it without resolving new bugs, so we reverted the plan back to fixing the bug post launch. I had difficult, yet informed discussion with the stakeholders about the bug. In the end, the stakeholders agreed to the post-launch fix, and the team was able to deliver the fix 2 weeks after. The bug was not caught due to inconsistent behavior across various android devices, so I added new android devices for their QA testing process