

<InfPALS/>

Internship Prep: Technical Interviews

InfPALS Week 9

Agenda

- > Introduction to technical interviews
- > Top tips
- > Best preparation resources
- > Practise, practise, practise

<InfPALS/>

Introduction to technical interviews



What is a technical interview?

- > **Part of the application process for most CS internships**
 - Along with a CV check and behavioural interviews
 - Can be the most intensive and important part of the application process
- > **30-90 minutes with a software engineer at the company**
- > **Normally can use any programming language, but can differ by company**
- > **Typically covers problem solving, algorithms, and data structures**
 - More specialised roles can ask for proficiency in other technologies: for example Data Science positions may test your SQL and front-end development positions may ask you to use their API

What could come up?

> Problem Solving

- Tests your ability to take a complex problem and break it down step-by-step

> Data Structures

- Frequent questions ask the candidate to manipulate strings, lists, arrays
- Should also be familiar with more complex data structures such as sets, hashmaps, dictionaries to decrease time complexity

> Algorithms

- You may be expected to know the operations and runtime of common Sorting (bubble sort, merge sort, quicksort, etc.) and Searching (linear, binary, etc.) algorithms and compare them

What could come up ... continued

> Time Complexity

- Covered in the second year course (Introduction to Algorithms and Data Structures), knowing the estimated runtime of the code you have written is a really useful skill
- Fairly simple concept to teach yourself (but gets more complex with harder algorithms)

> How the code could be more efficient

- A common question asked at the end of the interview
- Requires the candidate to be familiar with more complex algorithms and data structures (not necessarily by implementing them)
- Useful to also know the concept of caching

Top tips for a successful interview



Behaviour tips

- > **Stay calm and composed!**
- > **Ask for clarification in the problem**
 - Shows you have read and understood the task, and are thinking of edge cases
- > **Ask for a hint when needed**
 - The interviewer is here to for WITH you, not against you
 - Take the opportunity of having a professional software engineer at your disposal to ask for a suggestion if stuck and they will be happy to point you in the right direction
- > **Talking through your solution**
 - Showing your thought process is very helpful so make sure to talk through what you are coding as you do it!

Programming tips

> Good code organisation

- Break your code down into functions when needed
- Make sure to use accurate and sensible names for all functions and variables
- A neat code layout is easier for the interviewer to read and understand, so it goes a long way
- Use comments to explain anything important or unclear

Programming tips 2

> Don't overcomplicate it

- Start simple, then develop your idea
- A simple solution that is well-explained and works is generally better than a half-done complex solution that has not been explained to the interviewer
- That being said, the interviewer will generally let you know at the start whether they are looking for a simple, easy solution or an elegant one

Programming tips 3

> Testing

- After you have finished writing your solution, you are expected to step through the program and test it
 - This is usually done by adding comments to each line of the code depicting what the state of each variable is during that line
- Test any obvious or provided examples, but also think of edge cases to test if the code works fully!

<InfPALS/>

Best preparation resources



<InfPALS/>



LeetCode



GeeksforGeeks



HackerRank

<InfPALS/>

Time to practise!



Find the list of Leetcode Top Interview Questions

Choose 3 Easy level questions

Use the Hints, Solution panel, and Discussion panel to help you when needed!

Any questions?

Feel free to ask the leaders questions!

<InfPALS/>

<InfPALS/> Leader Applications

Gain **teaching and leadership experience**,
improve employability, refresh your technical
skills, and have fun social events with your peers!

