

# A Guide to Tech Interviews



**You probably want to earn money...**



# Since we're CS kids, we have tech interviews!



- Focused on solving problems/testing your knowledge
- Not the same as behavioral interviews
- Kinda like your coding exam but in-person and 1-1 with your professor

# Technical Questions Usually Involve:

Binary Search

DFS

Linked lists

Trees

Queues

Heaps

BFS



Sorting

Graphs

Finding bugs

*\*cats aren't involved\**

# There can be many technical interviews...

- Different companies have different procedures
- Some companies have just 1, some can have up to 5



# Do your research!



- Usually the bigger the company, the more complicated the process
- Read them up on career sites(e.g. Glassdoor) to see their process and the questions they ask
- Or you can just, ask your recruiter...

# There are many different types



- There are many different ways for them to test your technical skills
  - **Take home assessments**
  - **Video call technical interviews**
  - **Onsite technical interviews**

# Take Home Assessments (Format)

- No interviewer involved, just a link to complete an assessment
- Similar to coding assignments you'd get in class
- There's usually a time constraint





# Take Home Assessments (Tips)



- Read the instructions carefully
- Don't start right away
- Don't cheat
- Get a good environment

# Video Call Technical Interviews (Format)



- You're meeting the interviewer
- Solving technical problems while being interviewed
- Usually sharing your screen and writing code
- 1-2 hours in length

# Video Call Technical Interviews (Tips)

- Say your every step
- Ask questions, especially if you are struggling
- Act human, remember this is an INTERVIEW
- Pls wear pants



# Onsite Technical Interviews (Format)

- Doesn't always happen
- You're brought in-person to the company
- Usually longer and harder questions
- You might have a white board to discuss your solutions
- 3-4 hours in length



# Onsite Technical Interviews (Tips)



- Communicate often
- You might not solve the problem, be honest and still try your best
- Make your mistakes in the beginning
- Yes you'll be super nervous
- Again, act human(they see if you're a good culture fit here too)



# Technical Interviews Are Difficult

- They require you to know difficult topics
- They're nerve-wracking
- Grinding leetcode is stressful
- It's not fun, like at all
- How do I get good....?



# Practice, practice, practice



- There's no way around it
- Don't just **grind** though, practice questions strategically
  - Research the company
  - Use sites like **NeetCode.io** to get a roadmap
- Make sure to practice **saying** your solutions



**Questions?**