

Technical Interviews

How to destroy the notorious technical interview

Overview: The Technical Interview Process

1. Application/ Resume Drop
2. Recruiter Phone Call*
3. Initial Screen
4. Phone Interviews
5. Onsite*
6. Final Decision

The First Technical Screen

Either:

- Online Screen
 - HackerRank coding challenge, etc
 - 1-2 hours
 - Solve 1-3 problems in online IDE and pass test cases
- Phone/Skype Interview...

The Phone/Skype Interview

Typically 45min - 1hr

- 5 min introduction
 - Tell me about yourself, tell me about a project you've worked on
- 35-45 min technical question(s)
- 5 min questions

Technical Interview - What questions to expect?

Most internship interviews will focus on general programming concepts, data structures, and algorithms.

In later stages and for full time recruitment, some companies will ask more specialized questions (systems design questions, domain-specific knowledge)

General Data Structures and Algorithms

Know your **general programming concepts + data structures and algorithms**

- Basic object oriented concepts, recursion
- Linked Lists, Arrays, Matrices
- Stacks and Queues
- Hash Maps
- Trees and Tries
- Graphs
- Heaps
- Tree Traversal
- Graph Traversal
- Sorting
- Dynamic Programming
- ... *and much much more!*

... and the Big O analysis (space and time complexity) for all of them

Behavioral Qs

- Why do you want to work for Company X?
- Tell me about a project you've worked on

Northwestern Classes

- **EECS 211 (Object Oriented Programming in C++)**
- **EECS 214 (Data Structures)**
- EECS 212 (Discrete Math), EECS 213 (Systems), EECS 336 (Algorithm Design and Analysis)

Interview Prep Resources

Print

- [Cracking the Coding Interview](#)
- [Programming Pearls](#)

Online

- [HackerRank.com](#)
- [Leetcode.com](#)
- [CareerCup.com](#)
- [Glassdoor.com](#)

Internship Resources

- Career Cat/McCormick Connect
- Career Fairs (SWE Industry Day next week, Tech Expo in Winter)
- Company Websites
- Piazza, Glassdoor, etc. Job Listings
- AngelList
- <http://www.intern.supply>

General Tips

- Pick a language you are comfortable in and have familiarity using
- Always aim to write idiomatic, executable code
- Think aloud
- Be friendly and personable
- Demonstrate passion!

How to approach a technical interview question

1. Make sure you understand the prompt. Ask clarifying questions, read carefully
2. Write out an example [test case] if one wasn't given
3. Say your initial thoughts & overview of your initial solution out loud
4. Write the method signature using the expected input and output
5. Start by implementing your initial solution
6. Optimize and discuss the time & space complexity
7. Check your code! Run through one of your example test cases
8. If you're stuck, ask for help!

Example

Write a function that converts a string to an integer

Ex: "123" -> 123