Workshop 2
Proactive Communication

Above & Beyond Computer Science (ABCS)

Meta

RECAP: The Goal of a Coding Interview

…is to get signal on things that we do at Meta every day.

- How you think about and tackle hard problems and how you communicate about code
    - Evaluate your problem-solving skills to see if you can translate thought into reasonably correct, well-structured code
- How you consider engineering tradeoffs
- Limits of what you know

ABCS Tips and Tricks

Today, we’re covering:

The first 5....

- Work the clock
- **Communicate proactively**
- Design your algorithm

The next 10...

- Using collaborative editors
- Talk through your code/solution
- Handle your mistakes

The last 2-3...

- Test your code
- Increase your coding speed!
- Tackling imposter syndrome

01 Before you start coding...

1. Communicate Proactively
2. Design Your Algorithm

Proactive Communication

Huh? What does that mean?

Pro • ac • tiv

/pro'aktiv/

(adj) controlling a situation by causing something to happen rather than responding to it after it's happened.

- Proactive communication is any communication before you code
- Helps get what you need, get organized, and set the tone
- Don't let the first 5 minutes impede the next 30 minutes

*****Continue proactively communicating throughout your entire interview*

Do.

Not.

Start.

Coding.

Right.

Away!

Proactive Communication
Don’t start coding right away!

Don’t assume that the question is easy

- Repeat the question and rephrase it in your own words
- Assume all information given by the interviewer is necessary to solve the problem

Ask A LOT of questions

- Resolve any areas of ambiguity; clarify problem scope and intention
- Make sure that you fully understand Inputs & Outputs
- Write down edge cases → how they should be handled → come back to when testing
- Validate or state assumptions

For Example

You are asked to design an algorithm to sort a list. What do you ask?

For Example

You are asked to design an algorithm to sort a list. What do you ask?

- What sort of list? An array? A LinkedList?
- What does the array hold? Numbers? Characters? Strings?
- Are the numbers integers?
- Where did the numbers come from? Are they IDs? Values of something?
- How many numbers are there?
- Increasing or decreasing order? Or a custom comparator?
- Can it be empty?

Other Common Questions to Ask

These are just a few to get you started...

- How big is the size of the input?
- How big is the range of values?
- What kind of values are they? Are there negative numbers? Floating points? Will there be empty inputs?
- Are there duplicates within the input?
- What are some extreme cases of the input?
- How is the input stored? If you are given a dictionary of words, is it a list of strings or a trie?

Other Forms of Proactive Communication

Be Visual!

- Use diagrams
- Illustrate the problem and your high-level solution approach
- Write things down to make it easier to refer to when coding and testing your code

Workshop Summary

1. Do not start coding right away!
2. Proactive communication is any communication that happens before you start writing code
   - * Continue to communicate throughout your entire interview
3. Repeat and rephrase the question, ask A LOT of questions, and be visual!

Please scan the QR code and check-in to this workshop!

Additional Technical Resources

1. Sliding Window Technique:
    - Technique for getting O(N) runtime on many string/array questions
    - https://www.youtube.com/watch?v=jM2dhDPYMQM
    - https://medium.com/outco/how-to-solve-sliding-window-problems-28d67601a66

2. Additional String Practice Questions:
    - https://www.geeksforgeeks.org/category/data-structures/c-strings/

3. Additional Array Practice Questions:
    - https://www.geeksforgeeks.org/top-50-array-coding-problems-for-interviews/