



# Above & Beyond CS (ABCS)

## Coding Interview Workshop Series

Workshop 4  
Writing Code at the Whiteboard



# Recap: The Goal of A Coding Interview

...is to get signal on things that we do at Facebook 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** (memory vs. time)
- **Limits** of what you know

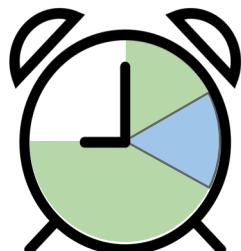
# What have we seen so far?



Before coding

...the first 5'

1. Communicate Proactively ✓
2. Design Your Algorithm ✓
3. Work the Clock ✓



During coding

...the next 10'

4. Writing code at the whiteboard
5. Talk through your code / solution
6. Handling mistakes



“After” coding

...the last 2-3'

7. Test your code
8. Ask questions!

# While you're coding...

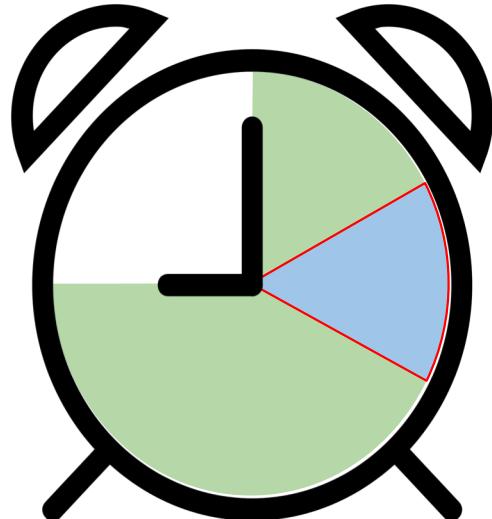
- 4. Write code at the whiteboard
- 5. Talk through your code/solution
- 6. Handling mistakes



4. Write code at the whiteboard

# Writing Code at the Whiteboard

Practice makes better. There's no substitute.



- Your ability to articulate your ideas clearly is necessary to judge how well you communicate
- Writing clearly will help you avoid confusion, catch and handle mistakes, increase complexity, and save time to present your best work
- Practicing a few simple tricks can go a long way

# Build Your Confidence Writing Code

Perfect practice makes perfect.

- You won't have the benefit of writing code at the computer

 Syntax highlighting

 Code completion

 Compiling

# Build Your Confidence Writing Code

Perfect practice makes perfect.

- You won't have the benefit of writing code at the computer
  - ~~✗ Syntax highlighting~~
  - ~~✗ Code completion~~
  - ~~✗ Compiling~~
- **That means, practice on pen/paper or at the whiteboard**

# Writing Code at the Whiteboard

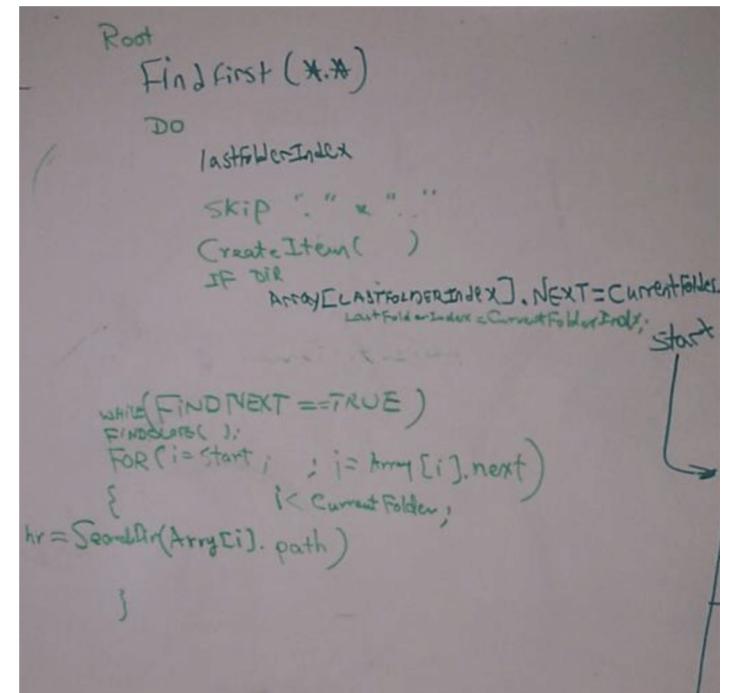
Begin with the end in mind.

- Outline the structure of the solution.
  - Include high-level functions (input, the purpose of the method, and its output)
  - Order of loops and conditions
- Start with pseudo-code.
  - Writing pseudo-code first may help you outline your thoughts clearly and reduce the number of mistakes that you commit
  - Tell your interviewer what you're doing and that you'll follow-up with "real" code next
  - If you do run out of time in the end, the interviewer will at least know how you'd planned to finish out the task, even if you didn't get to the details

# Writing Code at the Whiteboard

Don't crowd your code.

- Start in the upper-left hand corner, not in the middle
- Use a lot of white space
  - You might need this space to add extra lines of code, notes, or show test cases
- Use the right letter sizes and smaller names for variables and functions



# Writing Code at the Whiteboard

The end is just the beginning.

- Check your work in a compiler
  - When you're done, type the written code as-is into a computer to identify your mistakes. You'll probably make a lot at first – that's okay!
- Track changes
  - Start a list of all the mistakes that you've made so that you can keep these in mind in the real interview
- You'll be writing quality code on the whiteboard in no time!



# Thank you!



## Workshop 5

[INSERT REGION DATE/TIME]

### Complete Pre-Work

- ✓ Review screencast
- ✓ Solve HackerRank problems
- ✓ Be prepared to walkthrough your submitted code