PhD Software Engineer Intern

Initial Interview Guide

Meta

Welcome to your Meta interview prep guide. This document covers what to expect and how to prepare for your interview(s). Use the table of contents below to find key sections quickly. If you have specific questions, reach out to your recruiter, they're here to help!

What you'll find in this guide

Accommodations Process

Meta is committed to providing reasonable support (called accommodations) in our recruiting processes for qualified candidates with disabilities, physical conditions, mental health conditions, neurodivergence, sincerely held religious beliefs, pregnancy, childbirth, or related medical conditions, as required by law. If you need assistance or require an accommodation, reach out via the Accommodations Request Form.

Screen Stage Process Overview

This guide will explain what to expect during the screen stage of the recruiting process for this role. Your recruiter will be your guide throughout the process and aim to adequately prepare you for both the online assessments and the technical screen interview.

What will the screen stage process entail?

The screen stage at Meta will include **two online assessments and a technical screen interview.** You can complete the online assessments at the time and place of your choosing before the deadline you will receive via email. We recommend taking the online assessments in a quiet environment where you can focus, as once you begin the assessments you cannot pause them. **Your assessment links will be sent to you via email and are also accessible in your Career Profile.**

Please be sure to open the assessment links from a computer equipped with reliable internet connection, a webcam, and microphone. We recommend you use the most up to date version of one of the following browsers: Chrome, Firefox, Microsoft Edge (Windows only), Safari (Mac only).

You can expect the following steps in the Screen Stage:

Part 1: Online Assessments

- Online Coding Assessment
- Preferences @ Work Assessment

Part 2: Technical Screen Interview

- Virtual Technical Screen Interview with a Meta Engineer

Online Coding Assessment

What can you expect?

The 70-minute online coding assessment will focus on your programming skills, particularly your knowledge of data structures and algorithms. You will be given 4 coding questions of varying difficulty.

The coding assessment will be proctored and will undergo an integrity review after completion. While taking the assessment, you will be asked to share your screen, camera, and microphone with CodeSignal, the coding assessment platform. The assessment rules are as follows:

- You **cannot** consult or copy code from any external source, including but not limited to chatGPT, websites, programs, books, or other people — **except** to search the web for language documentation and syntax references.
- You **cannot** copy or distribute any information from the assessment, including but not limited to the question descriptions.
- You **cannot** write or execute code in the browser's developer console, an external editor, or an external integrated development (IDE).

How to prep

- We recommend you take advantage of CodeSignal's Practice Test, which you can access by creating a CodeSignal account. The practice test is designed to be a playground that lets you get acquainted with the testing environment and practice the platform's functionalities before taking the assessment.

- Here is additional information on preparing for your CodeSignal assessment.
- Before taking the assessment, you can select the coding language you are most comfortable with.

Coding assessment reminders

- Please take this assessment in one sitting. Once the timer starts, you cannot pause it.
- You will need to create a CodeSignal account to take the coding assessment.
- If you have any technical difficulties or any questions about the coding assessment platform itself, please use the Help Chat feature on CodeSignal or email support@codesignal.com.

Preferences @ Work Assessment

What can you expect?

The Preferences @ Work assessment is focused on how you work and the environment you feel most effective in when you work. You will have up to 40 minutes to complete the assessment, but most people complete it in approximately 20 minutes. The assessment consists of about 40 short questions.

You do not need to do any preparation for this assessment, and you should answer these questions honestly as they are about how you prefer to work.

Assessment reminders

- Please click “submit” after answering all questions.
- Keep an eye on the timer; if time expires before you click “submit,” you will not be able to continue with your assessment and it will not be submitted.
- If you have any technical difficulties or any questions with this assessment, please contact me.

Technical Screen Interview

In addition to the online assessments, you will also have a technical screen interview with a Meta engineer. We recommend taking both online assessments before your technical screen interview. However, you can take the technical screen interview first if it is scheduled before your assessment deadline. For more information and resources to assist in preparation for the technical screen interview, please see below.

What will your screen interview entail?

You will have a 45-minute screen interview, where you can expect the following:

- **Introductions (5 minutes)**
  We want to know more about you and how your experiences and interests align with our mission!
  Prepare

a concise, interesting description of who you are, where you've trained and worked, and what your areas of expertise are.

- **Coding (35 minutes)**
  You'll solve two coding problems focused on CS fundamentals like algorithms, data structures, recursions, and binary trees. If your tech screen is by phone, the engineer will send you a collaborative
  editor (such as coderpad.io). If your tech screen is in person, you'll use a whiteboard.
- **Answering Your Questions (5 minutes)**
  Take this brief opportunity to learn more about working at Meta from an engineer's point of view. Think about what you find interesting and challenging about the work you'd be doing here or what challenges you're most interested in solving.

Video Conference interview best practices

- Make sure you're in a quiet environment.
- Double check that you have a reliable internet/phone connection.
- It's okay to ask the person you're speaking with to speak slowly if you can't catch what they're saying.
- You'll need a laptop with a webcam, speaker, and mic. We recommend using a headset headphones with a mic for better quality audio, but this is optional.
- While we embrace the usage of advanced tooling in our day-to-day work to build the future of social connection, we ask that you not use any unauthorized outside assistance during interviews. Throughout the interview process you may be asked to share your screen via Zoom. Here you can find guidance on how this works. Additionally, we ask that you disable and not use filters during your interview. This will help us get an understanding of how you approach problem-solving and coding independently. Outside assistance includes, but is not limited to, consulting external resources, seeing help from others, or using AI tools, like ChatGPT.
  - For your interview, ensure Zoom screen sharing is enabled in advance, as setup may require a few minutes. Here's how: Enable Zoom screen share.
  - A filter is considered any tool/feature that applies visual effects to your video feed, including virtual/blurred background.

Interview Dress Code

As you're probably aware, we promote a casual environment at Meta so that everyone can be their authentic selves. Formal dress is not required (jeans are definitely ok!). Dress comfortably. We care about what you can do, not what you wear.

What do we look for?

Your interviewer will be thinking about how your skills and experience might help Meta as well as how you tackle problems you're not as familiar with. In your screen interview, your interviewer will assess your performance on 4 focus areas:

- **Focus Area 1: Communication.** Are you asking for requirements and clarity when necessary, or are you just diving into the code? Your initial tech screen should be a conversation, so don't forget to ask questions.
- **Focus Area 2: Problem solving.** We're evaluating how you comprehend and explain complex ideas. Are you providing the reasoning behind a particular solution? Developing and comparing multiple solutions? Using appropriate data structures? Speaking about space and time complexity? Optimizing your solution?
- **Focus Area 3: Coding.** Can you convert solutions to executable code? Is the code organized and does it capture the right logical structure?
- **Focus Area 4: Verification.** Are you considering a reasonable number of test cases or coming up with a good argument for why your code is correct? If your solution has bugs, are you able to walk through your own logic to find them and explain what the code is doing?

How to prep

Interviewers can only assess your skills and abilities based on what you show them during your interview, so it's important to plan and prepare to best showcase your strengths. As you begin preparing, please reference your Career Profile for additional role-specific prep materials, if available. In addition to the preparation guidance below, this video will give you an example of what to expect during your technical screen.

- Tip 1: **Before you practice, plan!** Be honest with yourself—only you know how much prep time you'll need. Make the most of your prep time by following these steps to plan your approach before you start practicing.
    - **Schedule time to study and practice**. Block out time every day to write code. Be sure to practice
      with a friend or peer and create a Mock Interview for yourself.
    - **Prioritize breadth over depth.** It's much better to practice solving fewer example problems of many problem types than to become very familiar with one type at the expense of the others.
    - **Set aside time to review what you've practiced.** As you solve problems, make cheat sheets or flash cards to review later. Revision and repetition will strengthen your understanding of core concepts.
    - **Remember your goal.** Aim for confidently solving one to two questions— while thinking aloud—in about 35 minutes.

- Tip 2: **Use key practice strategies to practice effectively** Reading through sample questions, recognizing concepts, and having a vague understanding of these concepts won't be enough to help you shine. You need to practice! Make sure you're setting your practice sessions up for success by following these tips from engineers who've been through the process.

    - **Practice coding the way you'll code during your interview.** Use CoderPad.io if your interview is via phone or video call, or use a whiteboard or pen and paper if your interview will be in person. Check with your recruiter if you're not sure which format you'll use.
    - **Set a time constraint when you practice problems.** In your technical interview, you'll be asked to solve one to two problems in under 35 minutes. Practice coding solutions to medium and hard problems in less than 15 minutes each to help you be ready for the constraints during the interview. There are resources available in the Preparation Hub within your Career Profile such as coding puzzles and practice exercises.
    - **Code in your strongest language.** Provide the most efficient solution and find and fix the bugs yourself.
    - **Practice talking through the problem space and possible solutions before you dive in and talk**
      through your decisions out loud as you code. Interviewers will be evaluating your thought process as well as your coding abilities. Explaining your decisions as you code is crucial to helping them understand your choices. The more you practice this, the more natural it will feel during the interview.

- Tip 3: **Understand the types of problems you may encounter.** Practice a variety of different problems—and understand why we ask them—so you're prepared to solve them during your interview.
    - **Don't be surprised if the questions sound contrived.** Problems may be different than what you're probably tackling in a day-to-day job. We won't ask a "puzzle" question, but questions may be different than real-world questions because they need to be described and solved in 10-20 minutes.
    - **Problems may assess the depth of your knowledge and your versatility.** For example, your interviewer might ask you to solve a problem any way 5 you want. Then, they could add constraints on the running or space characteristics and ask you to solve it again.
    - **Problems may focus on edge cases.** You might be asked to parse some data format or mini language. Your answers demonstrate your ability to handle multiple states in your head.
    - **Problems may test how well you know how things work under the hood.** For example, you might be asked to implement well-known library functions.

- Tip 4: **Decide what resources you'll use to prepare** It's easy to be overwhelmed by the number of online resources or the detail in an entire theoretical algorithms book. Here are some sites that our engineers found helpful when preparing for their coding interviews.

Top sites for practice problems from Meta:

- Meta Sample Interview Problems and Solutions
- Interview Bit

Other websites:

- LeetCode
- Hacker Rank

Video prep guides for tech interviews:

- Cracking the Meta Coding Interview: The Approach
- Cracking The Meta Coding Interview: Problem Walk -through

The password for the two videos above is "FB_IPS". Portions of the videos that cover soft skills tips may be more relevant for preparing for your onsite interview than for preparing for your initial tech screen.

Example tech screen study list:

- See page 8 for an example list of exercises from Meta's engineering team you can use as a starting point to help you prepare. Feel free to tailor it to your specific practice needs.

  **It's not necessary to review these resources when preparing for your initial tech screen, but engineers recommend them to understand the entire technical interview process.*

How to Approach Coding Problems During Your Interview

Before you code

- **Ask clarifying questions.** Talk through the problem and ask follow-up questions to make sure you understand the exact problem you're trying to solve before you jump into building the solution.
- **Let us know if you've seen the problem previously.** That will help us understand your context.
- **Present multiple potential solutions, if possible.** Talk through which solution you're choosing and why.

While you code

- **Don't forget to talk!** While your technical interview will focus heavily on coding, the engineer you're interviewing with will also be evaluating your thought process. Explaining your decisions and actions as you go will help the interviewer understand your choices.
- **Be flexible.** Some problems have elegant solutions, and some must be brute forced. If you get stuck, just describe your best approach and ask the interviewer if you should go that route. It's much better to have nonoptimal but working code than just an idea with nothing written down.
- **Iterate rather than immediately trying to jump to the clever solution.** If you can't explain your concept clearly in five minutes, it's probably too complex.
- **Consider (and be prepared to talk about):**
    - Different algorithms and algorithmic techniques, such as sorting, Divide-and-conquer, recursion, etc.
    - Data structures, particularly those used most often (array, stack/queue,hashset/hashmap/hashtable/dictionary, tree/binary tree, heap, graph, etc.)
    - O memory constraints on the complexity of the algorithm you're writing and its running time as expressed by big-O notation.

- **Generally, avoid solutions with lots of edge cases or huge if/else if/else blocks,** in most cases. Deciding between iteration and recursion can be an important step.

After you code

- **Expect questions.** The interviewer may tweak the problem a bit to test your knowledge and see if you can come up with another answer and/or further optimize your solution.
- **Take the interviewer's hints to improve your code.** If the interviewer makes a suggestion or asks a question, listen fully so you can incorporate any hints they may provide. Your interviewer is there to collaborate with you, not trick you! If they are leading you in a direction, please feel empowered to follow.
- **Ask yourself if you would approve your solution as part of your codebase.** Explain your answer to your interviewer. Make sure your solution is correct and efficient, that you've taken into account edge 7 cases, and that it clearly reflects the ideas you're trying to express in your code.

What to Practice

Everyone could use a refresher in at least one core area! Before your initial tech screen, brush up on CS fundamentals— especially algorithms, data structures, object-oriented design, and design patterns in general. Review foundational techniques—recursion, graph theory, tree traversal, combinatorial problems, and so on.

Looking for more detailed guidance on what to review for your tech screen? The exercises below have been helpful for many engineers preparing for a Meta tech screen and can assist you in solidifying your

understanding of data structures and algorithms. Feel free to use this list as a starting point and tailor it to suit your areas of need.

Overview

- Each exercise could take you up to 1 hour
- These solutions are written in Java, but you will be able to use your language of preference in the assessment
- Remember how to analyze how "good" your solution is: how long does it take for your solution to complete? Watch this video to get familiar with Big O Notation.

Exercises:

Note: These exercises assume you have knowledge in coding but not necessarily knowledge of binary trees, sorting algorithms, or related concepts.

Topic 1 | Arrays & Strings

- A Very Big Sum (Warm-up, learning how to use HackerRank)
- Designer PDF Viewer
- Left Rotation

Topic 2 | Lists

- Pre-work: If you need to familiarize yourself with how lists work, watch this video
- Exercises:
  - Insert a Node at a Position Given in a List
  - Cycle Detection

Topic 3 | Stacks & Queues

- Pre-work: If you need a refresher, look at this video
- Exercises
  - Queue Using Two Stacks
  - Balanced Brackets

Topic 4 | Hash & Maps

- Pre-work: If you need a refresher, look at this video
- Exercises:
  - Ice Cream Parlor
  - Colorful Number (This one might be challenging. Remember, if you get stuck, refer to our proposed solution)

Topic 5 | Sorting Algorithms

- Pre-work: If you need a refresher, look at this video: Merge Sort
- Exercises:
  - Insertion Sort part 2
  - Quick Sort part 2

Topic 6 | Trees

- Theory: If you need a refresher, look at this video
- Exercises:
  - Binary Tree Insertion
  - Height of a Binary Tree
  - QHeap1

Topic 7 | Graphs (BFS & DFS)

- Theory: Watch this video to understand what a graph is and how to traverse it
- Exercises:
  - Breadth First Search
  - Snakes and Ladders

Topic 8 | Hash & Maps

- Theory: Watch this video to review concepts on recursion
- Exercises:
  - Fibonacci Numbers

Solutions: All solutions are available in this public repository.

Final tips for your interview

- **Be yourself.** This means being open and honest about your successes and ways you've improved
  throughout your career. Also, be sure to call out how you have specifically added value to your team or projects you've contributed to. We value teamwork and what each individual member brings to the table.
- **Carefully review and familiarize yourself with the job description and perform research on Meta and the role.** Be prepared to answer why you are interested in this specific role and in working at Meta.
- **Please take the time to review our mission statement and core values.** These values influence how we work together to fulfill our mission to build the future of human connection and the

technology that makes it possible. We also encourage you to take time using our technologies such as Facebook, Instagram, Messenger, and WhatsApp.

- **One important piece of advice for your interviews: it's ok if you don't know!** No one who works at Meta is an expert in all things, and we don't look for perfection in the people we interview. If you aren't sure of something during your interview, you're encouraged to ask clarifying questions and be upfront if there are topics that you have less experience with.
- **Prepare thoughtful questions for the interviewer(s).** Your interviewer may challenge your ideas, and you should be ready to speak not only to what you recommend or have experienced but the why as well. It is important to think outside the box and to approach problems from creative and different perspectives.

Post interview – What to expect

You can expect your recruiter to provide a specific timeline or updates along the way. Your recruiter will inform you of next steps after your interview and assessments as soon as they are available. Feel free to follow up with them if you have not heard back within a week of your interviews.

Appendix / Resources

Below is a curated list of resources to get started and help you prepare.

Role-specific prep

- Engineering at Meta - FB page
- Meta Code videos
- Meta Opensource website
- Engineering Leadership at Meta blog

Meta resources

- About Meta website
- Meta Newsroom website
- Meta Careers website and Meta Careers for Interns & New Graduates website
- Culture at Meta website
- Meta employee benefits website
- Interviewing at Meta: The keys to success blog

Update personal information, track interview progress, and send thank you notes.

At any time during the interview process, you can track your progress, send thank-you notes and update your
personal information all via the Career Profile. If you do not already have a Career Profile set up, you can create one here.

Thank you for taking the time to review this guide!