


Welcome to facebook Boston

Initial Interview Preparation Guide

Interview Process Overview

Initial Coding Interviews

These are 45-minute onsite interviews here at Facebook Boston. You will meet with an engineer one-on-one and answer technical questions by writing code on a whiteboard. Discussion is encouraged throughout the interview and you'll have an opportunity to ask questions.

Full Onsite Interviews

These interviews will include coding, architecture and career-focused questions in our Menlo Park, CA headquarters. This is a great opportunity to see the Facebook headquarters and learn more about the company.

Review

Our engineering leadership team will review all information about your interviews to determine how we will proceed.

Additional information:
https://www.facebook.com/notes/facebook-boston/the-interview-process-and-tips-for-success/640970546046514

facebook

Initial Coding Interviews

For the initial 45-minute coding interviews, expect 1-2 coding questions using a whiteboard (if on-site) or laptop (if remote). You will be tested on core CS fundamentals, including algorithms, data structures, recursion and your problem solving and logic skills. When coming up with approaches for the questions, think of an efficient, optimized solution in the language you code best in. Prepare questions for your interviewer for the end.

See Page 7 for Interview Tips

**Facebook is committed to providing reasonable accommodations for qualified individuals with disabilities and disabled veterans in our job application procedures. If you need assistance or an accommodation due to a disability, you may contact us at accommodations-ext@fb.com or you may call us at 1+650-308-7837.*

Initial Coding Interviews: Key Focus Areas

	Data Structures	Algorithms / CS Concepts
Critical	• Arrays and lists • Binary trees • Hash tables • Stacks and queues • Graphs	• Search – iterator, binary, hash • Sort – merge, quick, bucket • Graph traversals – BFS, DFS • Complexity, big O notation • Recursion
Nice to Have	• Trie • Heap • Set • Red-black trees	• Randomized quicksort • Dynamic programing • Heap sort, Radix sort • Spanning tree, minimum cut

Initial Coding Interviews: Practice & Preparation

Video Tutorials:

How to Crush Your Coding Interview:
https://www.facebook.com/Engineering/videos/vl.922576297773331/10153034510412200
What To Expect During Your Interview: www.facebook.com/Engineering/videos/10153034561822200/?theater

Coding & Interview Practice:

Facebook Code Lab: https://codelab.interviewbit.com/
Example Interview Questions: http://www.careercup.com/page?pid=facebook-interview-questions
LeetCode: https://leetcode.com/problemset/algorithms/

Other Resources:

Get that Job at Facebook: https://www.facebook.com/notes/facebook-engineering/get-that-job-at-facebook/10150964382448920
Careers at Facebook: https://www.facebook.com/careers
Engineering at Facebook Blog: https://code.facebook.com/
Algorithms class with Coursera: https://www.coursera.org/course/algs4partI

facebook

Interview Preparation Courses

These videos were created to give an even more in-depth look into our interview questions.

Cracking the Facebook Coding Interview - The Approach: This class will focus on how to do well in software engineering interviews. This is is deeply technical and will discuss specific algorithm and data structure topics. It will cover how to walk through an algorithm problem, how to brainstorm solutions, how to discuss the problem with your interviewer, and how to write code (and test code) on a whiteboard:
Video: https://vimeo.com/interviewprepsession/theapproach
Password: FB_IPS

Cracking the Facebook Coding Interview- Problem Walk-Through: This video will focus on how to walk through a technical coding problem on a whiteboard.
Video: https://vimeo.com/interviewprepsession/problemwalkthrough
Password: FB_IPS

Interview Tips

Prepare for technical questions involving coding or algorithms (in your preferred programming language), design patterns, and more specific questions to your background. It may also help to review core computer science concepts (data structures, binary trees, linked lists, object-oriented analysis/design, etc.) as well as subjects pertaining to the scale of our technology. People generally study 1-2 weeks and practice on a whiteboard or coding competition websites. Think about data structures, particularly the ones used most often (Array, Stack/Queue, Hashtable.) You may be asked about O memory constraints on the complexity of the algorithm you are writing and its running time - O(N^2) to O(N) etc.

Practice writing code by hand on paper or a white board. During the interview, explain your thought process out loud. Create a working solution (even if it feels inefficient) and then iterate rather than immediately trying to jump to the clever solution. Tip: If you cannot explain your solution clearly in 5 minutes, it's probably too complex for the interview question at hand. Generally avoid solutions with lots of edge cases or huge if/else if/else blocks. Deciding between iteration and recursion can be an important step.

Discuss initial ideas and solutions with your interviewer to clarify any ambiguity. If the interviewer gives you hints, be ready to discuss them against other options and implement them if appropriate. It is good to adjust and work through the problems with the interviewer to show your thought process and problem solving ability. Take hints from your interviewer to showcase your thought process and problem-solving ability.

Explain your motivation for pursuing an opportunity at Facebook. Be prepared to answer the question, "Why Facebook?"

Complete your solution. Done is better than perfect, but be able to identify any bugs and assumptions. The interviewer will be looking for an intersection between speed and accuracy.

Pace Yourself. The duration of the interview will be ~45 minutes. It helps to time yourself when practicing by hand at home. If your answers are taking longer, aim for simpler solutions.

Qualities We Look For In Engineers

A drive to make the world more open and connected

Facebook has played a critical part in changing how people around the world communicate with one another. Facebook Boston engineers are not only very talented, but also passionate about solving the challenges associated with connecting the next few billion users.

Ownership of projects from start to finish

Facebook engineers are always building new things, and have a demonstrated ability of owning projects from start to finish. Our engineers are able to take on every aspect of an project – from design and development to implementation and support.

Ability to thrive in a flexible, flat environment

Facebook has a very flat organizational structure, and all of our engineers are flexible and agile enough to stop what they're doing at a moment's notice and code. Everyone is a software engineer, and everyone is hands-on.

Daringness to be Bold

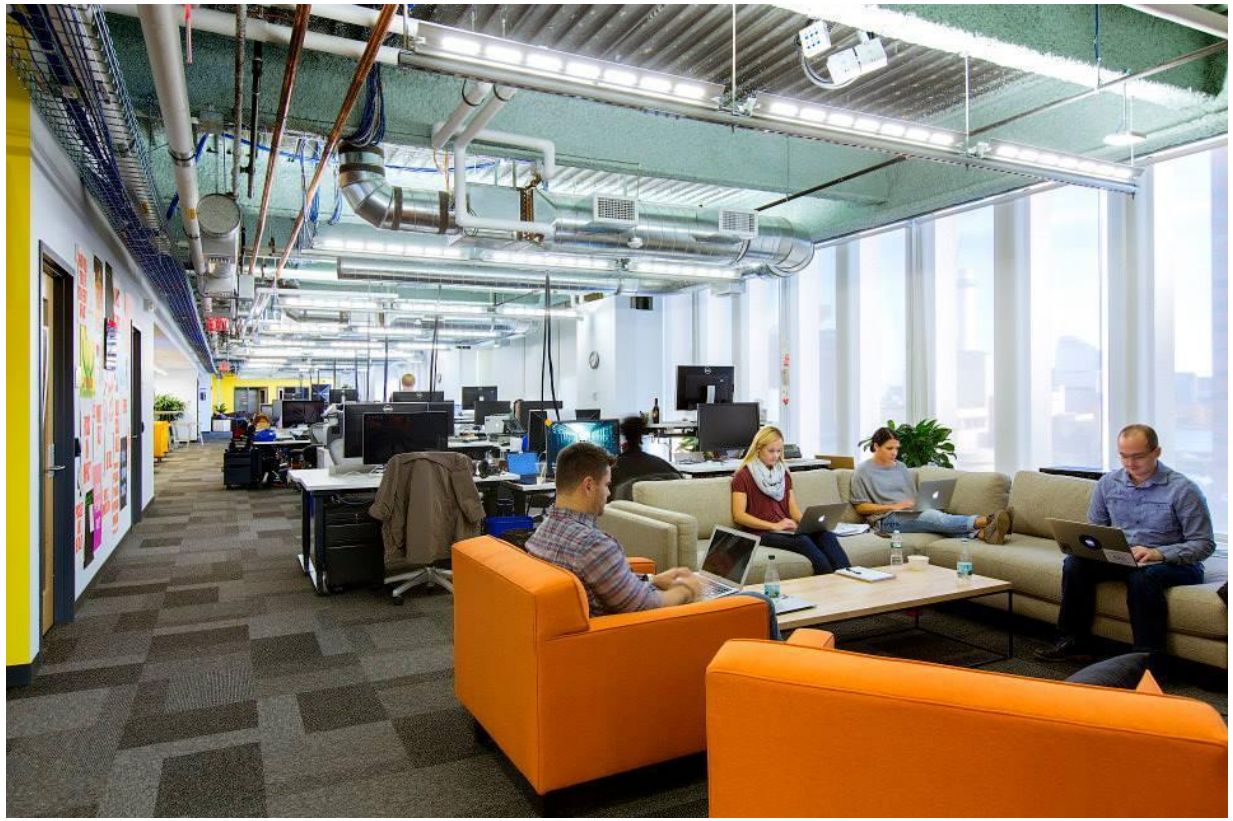
Facebook engineers have consistently taken on challenging projects throughout their education and careers. When you speak with engineers on our Facebook Boston team, you'll hear stories about times when they weren't quite sure about how they were going to accomplish a project, but dared to be bold, iterated quickly and found a solution. Our engineers embrace uncertainty, take risks and learn from failure.

facebook

Facebook Boston Office

Facebook Boston opened in the beginning of November 2013. We're located in the heart of Kendall Square, just off of the Kendall Square T stop. Most of our teams focus on large-scale, distributed back-end systems and infrastructure.

You can find full job descriptions on Facebook Boston's Careers Page:
https://www.facebook.com/careers/locations/boston/



facebook

Facebook Boston Engineering Teams

Core Data builds the majority of our back-end database and storage engines - everything from our low level file systems to cache infrastructure. There are currently two Core Data projects in Boston: the first is a back-end storage solution that handles all of Facebook's geo-tagged content. The second project is newer: a caching system.

Core Systems builds foundations for large-scale distributed systems and mobile apps at Facebook. The team in Boston is building a next-generation platform that supports the entire lifecycle of back-end systems.

Data Infrastructure deals with all of our real-time and stream processing and batch analytics. The team in Boston is building a scheduler which allows for internal partners to perform batch analytics, real-time and stream processing, production of models, apply machine learning applications, etc.

Security Infrastructure protects against abuse of our codebase and builds services, frameworks, and platforms that make it harder for developers to push insecure code into production. The team in Boston is currently developing new discovery and labeling systems to provide better safety for people's data on Facebook.

Product Infrastructure is responsible for the programming model used by Facebook product developers. The team in Boston works on GraphQL, a data query language that product developers use to describe their data needs. To learn more, visit http://graphql.org/.

Developer Infrastructure: HHVM (aka the "HipHop Virtual Machine") is an execution engine for PHP and Hack. Everything we open source is well-maintained and currently in production. We encourage you to collaborate with us on https://github.com/facebook/hhvm.

Android works closely with our product and design teams to optimize the Facebook experience across the Android platform.

facebook