**Facebook 面试流程**

[](http://www.jiuzhang.com/)

本资料由九章算法独家倾情整理

[www.jiuzhang.com](http://www.jiuzhang.com)

**目录**

[一、电面(Phone Screen) 2](#_Toc23025688)

[二、现场面试(Onsite) 4](#_Toc23025689)

拿脸书 offer 必备课程

《[面试软技能指导 - BQ/Resume/Project](https://www.jiuzhang.com/course/21/?utm_source=sc-libao-facebook)》《[九章算法班](https://www.jiuzhang.com/course/1/?utm_source=https:/www.jiuzhang.com/course/21/?utm_source=sc-libao-facebook)》《[⾯向对象设计OOD](https://www.jiuzhang.com/course/40/?utm_source=https:/www.jiuzhang.com/course/21/?utm_source=sc-libao-facebook)》

《[系统设计System Design](https://www.jiuzhang.com/course/28/?utm_source=https:/www.jiuzhang.com/course/21/?utm_source=sc-libao-facebook)》《[人工智能集训营](https://www.jiuzhang.com/course/20/?utm_source=https:/www.jiuzhang.com/course/21/?utm_source=sc-libao-facebook)》《[硅谷求职算法集训营](https://www.jiuzhang.com/course/29/?utm_source=https:/www.jiuzhang.com/course/21/?utm_source=sc-libao-facebook)》

Facebook的面试过程分为电面(Phone Screen)和现场面试(Onsite)两个阶段，下面将从面试流程、面试内容、面试技巧等方面来教你如何准备Facebook软件工程师面试。

# 一、电面(Phone Screen)

电面(Phone Screen)将会有一位Facebook工程师参与，主要面试内容为coding面试。

**面试流程:**

**· 自我介绍:** 首先面试官会要求你进行自我介绍，并说明想应聘的职位。

**· 职业期望:** 下面的5-10分钟， 面试官会对你的经历和职业期望进行提问。

**· coding:** 接下来的30-35分钟就要coding了。

1. coding面试会在一个你和面试官共享的在线编辑器.上完成(如果是现场面试就写在白板上)。

2.你会得到一个或几个coding question 问题都是长短合适的，可以在几分钟内解释好，10-30分钟解答完。

3.面试官会在这一环节 了解你解决问题的方式。

4.面试官一般不会问技巧或估算问题，他们并不关心海里能塞下多少个乒乓球。

5.你可以自己选择解题方式，但面试官可能会添加进一步的限制或要求。

**· 向面试官提问:** 最后5分钟是提问时间。这是一个从内部角度了解Facebook工程师的好机会。

**如何准备:**

**· 投入时间:** 温习面试技巧、编码技术和算法，对任何工程师来说都是很重要 的，即使资深工程师也是如此。面试和你的日常工作是完全不同的。尤其是第一次技术面试，一定要多做准备，对接下来的面试也是有益的。

**· 刷题:** 不使用编译器，用一 种最有效率bug free的方式练习解答算法问题，你可以在Lintcode.上刷题，同时你可以在uintcode标准答案查询上，到对应的最简洁、优质的答案。

**· 用简单的编辑器写代码:** 面试中你会用没有语法加亮和自动填充的工具写代码(比如CoderPad) 。

**· 练习手写编码:** coding面试将在白板 上完成，准备时可以用白板或纸和笔练习解题。

**· 计时练习:** coding面试的解题时间有限，按时完成至关重要。可以和朋友一起做一次模拟面试。

**·** **复习数据结构、算法和复杂度:** 要对时间和空间的复杂度进行分析。别忘了复习数据结构，比如列表、阵列、哈希表( lists, arrays, hashtables, hash maps, stacks, queues, graphs, trees, heaps)等等，还有排序、宽度优先搜索、深度优先搜素、和遍历(sorts, BFS, DFS, andtraversals)。还要记得复习递归和迭代方法(recursion and iterative)。

**· 想想你2-5年内的职业期望:** 面试官会问及你的兴趣和作为工程师的优势，准备1-2个问面试官的问题: 面试最后五分钟是向面试官提问，所以你要准备好1-2个能帮你增加面试官好感度的问题。

**coding面试技巧:**

**· 说出想法:** 面试官更多关注的是你解题的方法，这和答案的正确一样重要。把想法说出来可以让面试官了解你的思考过程，跟随你的解题思路。另外还可以让他们在必要的时候指点一:二。

**· 找一个好的面试地点:** 选择一个安静的地方,确保你的网络连接和手机信号良好。使用耳机，确保可以双手写代码。

**· 说话清晰:** 确保你说话清晰,如果你听不清面试官说话，立刻告诉对方,这样他们可以调整。谁也不想把时间浪费在努力听清对话上。

**·** **使用最擅长的编程语言:** 按时、准确地答题是很重要的，请使用你最熟悉的语言解题。

**· 高效管理时间:** 花时间找出问题最理想的解答方案,不要脑海里第一个蹦出什么想法就急着写什么。如果在一定时间内你还没能找到更好的方案，先动笔写一个可行方案，在解题过程中再迭代改进。有些面试者最终没有给出答案就结束面试了，就因为他们找不到理想方案。有一个可行但不是最佳的方案比一个随意而不加思考的方案强很多。有可行方案之后，你就可以在此基础上再进行优化。

**·** **分享你的推理:** 确保被问到时你能解释你的方案。软件工程就是权衡的过程，你要时刻做好讨论的准备。

**· 自己找到并修复bug:** 不要等着面试官帮你找bug.

**· 留意面试官给出的提示:** 面试官通常都对问题有充分的了解，如果你卡住了，他们给出的提示可以帮助你。

# 二、现场面试(Onsite)

作为Facebook软件工程师职位的应聘者，你将会一天内经历4到5轮的面试， 这些面试会有3种类型:

**1.代码面试Coding interview** -你需要解决一些基本的代码问题。

**2.设计面试Design interview** -你会被要求展示设计能力。设计问题将根据你的背景决定集中在系统或产品上。

**3.行为面试Behavioral interview** -你会被问到你的工作经验和求职动机，还有其他一些行为问题。除了很早或很晚去参加面试的人，其他人都会由招聘或工程部的人带着吃午饭。这是向面试官以外的职员提问的好机会。

2020年初因受疫情影响，onsite面试改为virtual onsite，即通过线上视频或语音进行面试。

根据反馈，Facebook的VO面难度较大，但考的原题较多且更加注重系统设计；偶尔会出现加面情况。

1. **编程面试Coding interview**

**面试流程:**这次coding面试会比电面要难:面试官的提问会更难，评估标准也会更严格。面试时间约为45分钟，以下是比较经典的流程: .

● **自我介绍:** 开始的5分钟是介绍时间，可能还会简单问一下你的背景。

**● coding:** 接下来的30分钟会有一个或多个编程问题(coding question) ,大部分是算法面试题，可以参考Lintcode上的算法面试题。

**● 向面试官提问:** 面试官会留最后5分钟作为你对面试官提问的时间。这是从内部工程师身上了解更多Facebook和你感兴趣的事的好机会。

**如何准备:** 同电面(phone screen)

**2.设计面试Design interview**

**面试流程:**设计面试有45分钟。整个面试你只用说与写。正如所有面试一样，面试官会留最后五分钟让你提问。面试的目的是评估应聘者解决重要工程设计问题的能力，最后面试官会问你一个非常广泛的设计问题并评价你的方案。设计面试有两种类型:系统设计和产品设计。

**如何准备:**

**(1) 系统设计**

**·** 模拟系统设计面试。邀请-个工程师帮你模拟面试。让他提出一个系统设计问题，如果正好是他正在做的项目那就再好不过了。不要把它当成是一个面试，而是放轻松地去思考问题，并提出你能想到的最佳解决方案。

**·** 在实际的系统中去实践。你可以在既有的OSS中去练习,也可以与朋友合作搭建一个系统。对于课堂中的系统设计作业，不再把它仅仅当成一个学术训练，而是把它当成实际问题，思考系统设计过程中的架构和博弈。正如我们生活中遇到的大多事情一样，只有做了才知道其中会遇到什么问题，从而真正学到东西。深挖开源系统的运行特点。例如，你可以看看levelDB。这是一个干净、小、且编写良好的系统。你可以读读执行命令，了解它是如何在硬盘中存储数据的，如何将数据压缩成不同的层?你也可以多多反思一下的博弈问题:哪种数据和大小是最优的?什么情况下会降低读写速度? (提示:比较一下随机写和顺序写)

**·** 多了解一下系统中数据库和操作系统是如何运行的。这些技术并不只是你口袋中的工具，它们往往会在你设计系统的时候给你带来启发。如果经常像DB或OS一样思考它们如何处理各自的问题，你也会把这些思考方式应用到其它的系统设计中去。

**(2)产品设计**

· 反思自己的项目:思考已完成的项目，哪些是容易的部分，哪些是困难的部分?

面试官可能会问:“你会如何设计一个电子 邮件服务器?“你可以先考虑这些问题:

**·** 如何储存邮件?尤其是当系统扩“大到一台机器装不下时?

**·** 如何处理有大量收件人的邮件列表?

**·** 如何处理人们滥用垃圾邮件系统?

**·** 面对潜在的系统故障，如何保证系统是可靠的?

另一个面试官可能会问:“你会如何设计用来构建丰富文档编辑器的客户端服务器API?”你可以先考虑这些问题:

**·** 客户端如何从服务器请求文档数据?尤其是文档大到我们不想在单一请求中下载它的时候?

**·** 如何在API响应中显示文档中粗体和斜体这样的元素?

**·** 在不破坏原有客户端的情况下，想要给服务器添加新特性,应该如何设计系统?

**设计面试技巧:**

**· 提炼你的方法**: 当你采取你的方法时，要先概述它，然后再考虑如何把它进行拆分。

**· 找到关注点:** 你没有时间去没有讨论每个设计的细节,所以你需要找出有趣的关注点和难以解决的问题。

**· 注重整体和细节:** 你既要重视整体的解决方案，也要注意其中的一些细节。一个好的解决方案需要在方方面面都能做到完美的

**· 探索内在的权衡:** 在工程问题中,你需要对权衡做出明智的决定。你需要把你的方案与其他方案进行比较，探讨其中存在的问题，并且选择出最佳的解决方案

**· 主导讨论:** 面试官希望看到的是你通过艰难的经历学会了构建大型系统。你的预测能力和解决典型问题能力都能让面试官为你亮绿灯。

**· 积极对话:** 问清晰明确的问题，但确保话题要往好方案方向发展。