
九章算法 帮助更多中国人找到好工作



扫描二维码，获取“简历”“冷冻期”“薪资”等求职必备干货

九章算法，专业的IT 求职面试培训。团队成员均为硅谷和国内顶尖IT企业工程师。目前开设课程有《九章算法班》《系统设计班》《Java入门与基础算法班》《算法强化班》《Android 项目实战班》《Big Data 项目实战班》。

Google 面试流程、考核能力要点及准备建议

Google的招聘流程涉及两种面试。在每一轮面试中，Google都会从面试官那里收集反馈，然后确定后续步骤。

（一）面试流程

第一阶段：电面（Phone Screen）

电面的面试官很有可能是你以后的同事或经理。

如果你申请的是软件工程相关职位，电面将持续 30 到 60 分钟。回答编码相关问题时，你需要一边在与面试官共享的 Google Doc 上编写代码，一边清晰地阐述你的思路。建议你使用耳机或扬声器，以便你能够自由输入相关内容。

电面会问到数据结构和算法。使用你最擅长的编程语言写 20-30 行代码。根据如下步骤做编码练习（注意你的代码风格，参考文章：[《Google 官方代码风格指南》](#)）：

1. 你会被问到一個开放式的问题。面试官会据此问一些问题，然后制定一些要求。
2. 用一个算法解释这个问题。
3. 将其转换为可行的代码。（提示：因为有时间限制，不要想着一下子写出完美的代码，写下你所想到的，稍后再进行修改，还要确保考虑了边界情况和运行问题。）
4. 优化代码，测试用例并发现错误。

对于所有其他职位，电面将持续 30 到 45 分钟。请准备好回答涵盖职位相关知识的行为问题、假设问题或案例式问题。

第二阶段：现场面试 (Onsite)

一般情况下，会有 4 位 Google 员工（一些可能成为你以后的同事，还有一些是跨职能部门的员工）面试你，每个人的面试时间大约为 30-45 分钟。

对于软件工程职位的应聘者，面试官希望了解你的编码技能和技术领域的专业知识（其中包括工具或编程语言，以及有关数据结构和算法等主题的一般知识）。面试官通常会在这种讨论中反复提及与此相关的问题，就像讨论实际工作中遇到的问题一样，因为他们喜欢激发彼此的思想碰撞火花，进而了解不同的方法。因此，请准备好深入探讨你的解决方案。超越自我界限，找到最佳答案，这很可能就代表了您的工作方式。

工程职位以外的应聘者将有机会从 4 个不同的方面突出展示自己的优势：

- **一般认知能力：**面试官会提出一些开放式问题，以便了解你如何着手处理问题和解决问题。这些问题没有标准答案，因此，他们最看重的是，你是否具备一定的能力，可以清晰地阐明你的思路以及你如何利用数据做出明智决策。
- **领导能力：**准备好谈一谈你曾如何利用自己的沟通和决策能力调动他人积极性。这种案例可以是你在工作期间或单位的领导岗位晋升经历，也可以是你带领团队取得成功（即便在你并未正式成为领导者的情况下）的经历。
- **职位相关知识：**面试官希望可以了解你如何将个人的优势与经验相结合以带来积极的影响。面试官不仅想知道你目前能做出的贡献，还要了解你打算如何提升自身能力以便挑战不同的职位，甚至包括目前尚不存在的职位。
- **Google 精神：**分享你如何独立工作以及如何与团队协作，如何帮助他人，如何在迷茫中找到方向，以及如何激励自己居安思危、不断进步。

在整个面试过程中，你可以放心大胆地请面试官明确说明面试问题，以确保你完全理解所问的问题。你也可以畅所欲言，面试面试官们，询问有关工作、团队和文化的问题，这样做可以帮助你确定相应职位是否适合你。

（二）如何准备

1. 适用于所有职位的面试

- **预测可能遇到的面试问题：**你可以预测到面试官会向您提出的 90% 的面试问题。例如“为什么想要这份工作？”、“你曾解决过什么棘手的问题？”

等等。如果你实在想不出来，可以阅读[《硅谷程序员面试宝典》](#)，将你认为最有可能被问到的前 20 个问题写下来。

- **计划：**针对你列表上的每个问题写下答案。这有助于你牢记相关问题及答案，以便在需要时能够不假思索地说出来，这一点至关重要。
- **制定备用方案：**实际上，对于每个问题，建议你写下 3 个答案。为什么要写 3 个呢？对于每个问题，你都需要另外准备一个同样可行的不同答案，因为第一个面试官有可能不喜欢你讲述的案例，这时，你就得寄希望于下一位面试官听听你的另一个案例，或许他/她能因此成为你的支持者。
- **说出想法：**面试官想了解你的想法，在面试中说出你的思考过程和决策。请记住，面试过程中，不仅评估你的技术能力，更是要评估你如何处理问题并尝试解决问题。和面试官明确地说明和检查他给出的假设，确保它们是合理。
- **用数据说话：**在回答每个问题时，你都需要通过具体案例来说明自己具备问题中所要求的能力。例如，如果你被问到“你是如何履行领导职责的？”这样的问题，你应该回答：“我是一个具有协作能力/决策能力/诸如此类任何能力的领导，我来给你讲一下当我在...”
- **遇到问题询问：**面试题可能是有问题的，面试官故意不提，来深入了解你在技术难题中所看重的范畴和信息。他们是想看看你是如何参与解决问题的。一定要谈谈你的思考过程，如果遇到疑问，随时提问。
- **优化：**想想如何改进你提供的解决方案。在许多情况下，你一开始的解决方案可能需要一些细化和进一步的解释。如果有必要，从暴力解决方案开始，改进它 - 要让面试官知道你在做什么以及为什么。
- **练习：**每个人都可以不断练习获得进步。大声复述你的面试答案，直到你可以简明扼要地讲述每个案例。

2.适用于软件工程技术职位的面试

- **编码实践：**你可以在 LintCode、CodeLab、Quora 和 Stack Overflow 等网站上找到一些编码问题示例。[美国大公司面试真题训练](#)和 [Google Mock Interview](#) 也值得一做。面试会要求手写，所以请在纸张或白板上练习编写代码。一定要测试你的代码，并确保它易读而且没有错误。不要纠结于小的语法错误，比如对于给定的方法用哪个子串，直接选择一个，让面试官知道就可以了。
- **编码：**至少擅长一种编程语言，最好是 C++，Java，Python，Go 或 C。如果你擅长 API、面向对象设计和编程，知道如何测试代码，以及知道如何处理边界情况更好。请注意：面试注重的是概念理解而不是记忆。
- **算法：**掌握用自下而上和自顶向下的算法解决问题。知道算法的复杂性，以及如何优化/改变算法。熟悉解决 Google 问题的算法：排序，分治法，动态规划，贪心，递归或有关数据结构的算法。知道 Big-O 符号（例

- 如运行时间)，并准备好像Dijkstra和A*这样的复杂算法。在编写代码之前，建议你讨论或阐述你所想到的算法。
- **排序：**熟悉常见的排序类型以及它们有效的输入数据类型。考虑运行时间和空间。例如，在特殊情况下，插入排序或基数排序比通用的快速排序、归并排序、堆排序答案要好得多。
 - **数据结构：**研究尽可能多的数据结构。最常使用的数据结构是数组，链表，堆栈，队列，哈希集合，哈希图，哈希表，字典树，树和二叉树，堆和图。了解内部的数据结构，以及哪些算法倾向于和哪个数据结构一起使用。
 - **数学：**一些面试官会提一些基本的离散数学问题。尤其在Google，因为Google的工作经常会接触计数问题、概率问题和其他离散Math 101情况。在面试之前花上一些时间，记住基本概率理论和组合的基本要素。熟悉n选择问题和类似的问题。
 - **图：**考虑一个问题是否可以应用于距离、搜索、连接、循环检测等图形算法中。有三种图形表示方法：对象和指针、矩阵、邻接列表，你需要熟悉每个方法及其利弊。掌握基本图遍历算法、广度优先搜索和深度优先搜索。了解它们的复杂度，了解什么情况该用哪个，以及如何在实际代码中实现它们。
 - **递归：**许多编码问题涉及递归的思想和用递归编码给出解决方案。对于可以用迭代的方法解决的问题，用递归找到更优的解决方案。

(三) 其他问题

- **着装：**大部分面试对着装都没有特定的要求，但招聘人员会告知你如何着装最得体。如果你不确定哪种风格适合自己，那就做你自己，穿你认为舒适的衣服就好。
- **如何组织面试答案：**回答问题时，请务必说明你是如何得出解决方案的，因此，你可以想到什么就说什么。
- **做准备工作时要考虑的有帮助的问题：**独立工作以及参与团队协作时，你如何让自己的能力得到充分发挥？你在求学时和职场中遇到过哪些挑战？你是如何战胜这些挑战的？你有哪些技能和经验可作为你申请本职位的优势，为什么？
- 如果你没有理解面试问题，可以请面试官明确说明，且回答问题时可视需要从容作答。

免责声明：面试流程和面试内容因人而异，本文仅供参考，不承担相关的责任。