

技术面试Do & Don't

Do

良好的算法题习惯

- 自己刷算法题的语言标准库函数（输入/输出/常用数据结构/字符串操作）一定要熟练。不一定每个面试官都会要求你把code跑起来，但是在这些地方卡住往往被认为是基本功不扎实。
- 面试官会实时看到你的whiteboard，因此写算法最好先用注释标清楚每一段打算干什么再慢慢填充。即使在某个环节卡住了，面试官也能知道你是哪个点没有想明白从而给予提示。切忌让面试官跟你一起对着空白发呆，否则面试官可能需要提示很多次才能命中你卡住的点，会非常影响评价。

体系化的知识

- 大部分面试官都很喜欢问follow-up，因此八股准备时最好以知识导图的形式进行延伸。
- 合理的知识框架有助于保持叙述的条理性，直接背答案的候选人在实际面试中比较容易出现颠三倒四的情况（说着说着又想起前面的点），观感不好的同时面评也容易记漏。

回顾自己的经历

- 梳理一遍自己过往的科研/实习/项目经历，尤其是简历上列出的那些。其中关键的技术细节、决策过程要烂熟于心。
- 过往经历中使用的技术栈要着重准备。
- 有意识地区分“我”（个人贡献）和“我们”（团队工作），并在叙述时分配合适的详略。不要连篇累牍地描绘团队的整个工作有多么强，否则会被认为是蹭经历的。

面试结束及时整理面经/回顾

- 一般来说一面官认为有缺陷或较为弱势的方面会进行备注，二面时会着重考察。因此如果觉得自己哪一科被问懵了一定要及时补漏。
- 不要认为一面被问过的问题后续一定不会被问到，尤其是没有答满的问题。（我的实习生被连问了三次Java数组扩容==）

Don't

不要作弊

任何情况下都不要尝试作弊！

- 以字节为例，字节的面评会一直保留并对后续所有进入简历筛选流程的部门和面试官可见。
- 常见的面试软件（飞书/牛客）均有切出页面检测，包括次数、时长等详细数据。这些数据同样会进入面试记录。
- 面试官培训中都会包含对常见作弊现象（眼神左右扫/磕磕绊绊然后开始念答案）的识别，并且大部分情况下从面试官视角看作弊是非常明显的。
- 被认为有作弊嫌疑的候选人会被复核（如安排第二机位或现场面试），因此为了避免麻烦，建议瓜田不纳履。

不要背题/套路答案

- 算法题库每年会进行更新，一般都会对常见的LC题做变形。同时HR也会监控外网面经中算法题的频率，频率过高的题会被从题库中暂时移除，所以不要抱着押题心态进行准备。
- 常用的算法不要直接背代码实现。一方面面试官可能会要求现场变形（e.g. 递归改非递归），另一方面某些题目的边界条件需要单独处理。

不要“即兴创作”

特别针对STAR类问题。

- 对于自己做过的事情平铺直叙，没有做过的事情不要现场扩写。“将会”“一般”“通常”“我认为”等假设性与含糊不清的表意通常被认为是假STAR。
- 如果面试官提出的就是假设性问题“假如.....”，证明这个面试官不是一个好面试官：）或者他接受的面试培训并不是STAR模式，按自己的理解回答即可。

不要被问崩心态

- 有的面试官习惯每一个方面都问到候选人不会为止，而有的面试官认为你了解不深之后就会跳过这一块甚至于直接对你的技术能力“盖棺定论”，因此你感受到的面试压力与你实际的面评并没有强相关性。不论面试官是热情还是冷淡，有反馈或是无动于衷，始终保持稳定的心态和有条理的语言。