

聊聊算法面试

自我介绍

无隅

- 毕业于北京邮电大学
- 现就职于亚马逊中国研发中心 SDE
- 擅长于Java/Python/数据结构与基础算法
- Interest: 分布式系统, 机器学习

提要

- 关于算法面试
- 面试中的常见误区
- 算法问题解答流程
- 算法面试考察的知识点
- 如何准备算法面试
- Q & A

影响算法面试是否成功的几个因素

- 公司招人多与少
- 应聘者的多与少
- 应届求职者 v.s 有相关经验求职者
 - (1) 对于应届生着重考察基础，算法面试相对多一点
 - (2) 对于有经验求职者，算法面试相对少一些，会增加相关项目经验与系统设计方面的考察

算法面试的常见误区

- 算法思路正确，想出来就能过
- 把实现代码写出来就能过
- 做过的题和简单的题一定没问题

面试官视角下的求职者

- 求职者是未来的同事，一起工作的人
 - 求职者的思维方式与工作模式是否符合公司价值观与工作流程
- 求职者的代码可读性
 - 是否在code review中需要浪费大量的时间
- 求职者的编程习惯
 - 是否遵循基本的编程原则
 - 是否有良好QA的意识

面试的关键点

- 代码风格
 - 变量命名，代码缩进
- 编码习惯
 - Bug free
 - 异常检测，边界处理
 - 抽象
- 沟通交流能力
 - 让面试官明白你的意图与想法
 - 展示自己的沟通交流能力
- 测试
 - 给出测试用例
 - 覆盖所有情况

算法面试问题解答流程

- 明确问题：与面试官交流明确要解决的问题
- 举例分析：用一些例子来尝试解决问题
- 描述大体思路：让面试官了解你的思维过程
 - 先给出暴力解法：不要担心效率，Done is better than perfect!
 - 尝试优化：BUD原则, 头脑风暴
 - Walk through
- 代码实现：写干净整齐的代码，注意抽象模块化
- 给出测试用例
- 分析算法复杂度，回答问题

怎么优化和解决问题

- BUD原则
 - Bottlenecks
 - Unnecessary Work
 - Duplicated Work
- 模拟
 - 尝试从基本例子入手
 - 简化和抽象问题，从特殊推广到一般
- 头脑风暴：利用数据结构特性与边界条件

算法面试考察的知识点

- 数据结构与算法
- 面向对象设计与系统设计
- 计算机基础知识

数据结构

- 数组与字符串
 - 数组的实现原理与基本操作
 - 字符串基本操作
- 链表
 - 链表的实现原理与基本操作
- 栈与队列
 - 栈与队列的实现与基本操作
- 散列表与堆
 - 散列表与堆的实现与基本操作
- 树与图
 - 二叉树的实现与基本操作
 - 二叉搜索树（BST）的实现与基本操作

算法

- 复杂度分析
- 排序算法
- 递归与搜索
- 动态规划
- 贪心算法

面试准备建议

- 夯实基础，别做难题
- 是面试而不是考试
- 着重总结，理解而不是背诵
- 时间要用在重要的地方

夯实基础， 别做难题

- 不要花时间攻关难题
- 把时间花在如何做到bug free和如何提高编程速度上
- 集中突破简单和中等难度的题目上
- 理解数据结构与算法背后的故事

是面试而不是考试

- 面试是为了招聘合适的人一起工作
- 面试是双方选择的过程
- 和面试官愉快交流，一块合作解决面试问题
- 证明自己的实力，但别去证明面试官的实力

着重总结，理解而不是单纯的背诵

- 主要学习的是思维方式和分析技巧
- 要理解数据结构与算法的内在原理，熟练地给出模板实现
- 善于总结一类题目的模版与内在原理
而不是某个题的解法

时间要用在重要的地方

- 不要把时间浪费在那些基本不会考你又很心虚的内容
- 比如KMP, 红黑树, 线段树, AVL以及很复杂的题目上

Q & A