Atitit 判断程序员能力

目录

[1.1. 工作年限 非常重要 手感 1](#_Toc2594)

[1.2. 项目经历尽可能全面些（数据库 文本nlp爬虫 图形 音视频 ）有技术含量 1](#_Toc18017)

[1.3. 高质量的开源项目 1](#_Toc5627)

[1.4. 博客等文章记录 1](#_Toc25239)

[1.5. 抽象分析能力 提升总结 xx体系 2](#_Toc12598)

[1.6. 考量思路而不是考虑代码， 2](#_Toc13298)

[1.7. 把面试当做一场技术讨论 2](#_Toc31780)

[1.8. 使用-原理 --自我实现 金字塔 2](#_Toc7632)

[1.9. 技术水平才是根本 2](#_Toc18353)

[2. 其他 3](#_Toc25680)

[2.1. 少问记忆性问题和太理论性问题 3](#_Toc10976)

[2.2. 白板编程真的有必要吗 3](#_Toc90)

[2.3. 智力问题为什么会收到青睐 3](#_Toc18717)

## 工作年限 非常重要 手感

## 项目经历尽可能全面些（数据库 文本nlp爬虫 图形 音视频 ）有技术含量

## 高质量的开源项目

## 博客等文章记录

## 抽象分析能力 提升总结 xx体系

## 考量思路而不是考虑代码，

笔试仅仅对于三年以下程序员有用

高级程序员往往不记具体语法细节

## 把面试当做一场技术讨论

筛选候选人就是筛选将来与你共事的人，所以为了更准确的反应候选人在以后的工作中的表现，不妨把面试当做一场与未来同事的技术讨论，在讨论的过程中感受候选人的技术能力。

技术面试就好比打乒乓球，一来一往中感受彼此的技术实力，面试的过程切忌类似与笔试一样的一问一答单向沟通。特别是一些开放性问题，架构设计的问题，本身就没有标准答案，背景又过于复杂开放，如果只是丢给候选人回答，中间没有任何沟通交流和引导，候选人是很难抓住重点展现出面试官心里期望的表现。

比如我们面试过程中经常会让候选人介绍某个项目的架构设计，如果面试官能一语中的的提出设计中的缺陷，或者追问架构中的技术难点，深入的跟候选人讨论，这样一方面能给候选人充分发挥的机会，另一方面，也会赢来候选人对公司技术的认可

## 使用-原理 --自我实现 金字塔

## 技术水平才是根本

# 其他

## 少问记忆性问题和太理论性问题

## 白板编程真的有必要吗

教育背景不是很重要，因为精英教育，，很多人家庭经济不好无法上大学

大公司的工作经历不是很重要的能力背书

，因为大公司挑学历，，导致很多穷人无法去大公司打工

## 智力问题为什么会收到青睐