咩啊🐑量变-质变-事情是干不完的但是有轻重缓急

## 21年的时间表，很多东西被拆分出去了

## 时间规划表

* 没事就看经典
* 我好像总有事，忙得要死
* 算法
* 日语-JLPT
* 英语
* 数据库
* golang
* aws
* 基础情报考试
* 人工智能
* 计网，操作系统，ai技术内参是需要看完的，还有数据库那个和牛客网的练习
* <https://matt.might.net/articles/what-cs-majors-should-know/>

## 算法🔞

* 🐮【看看英文版本题目】
* labuladong 2021年度文章总结 <https://mp.weixin.qq.com/s/ir1Hk06HcT8W_qz0MtyONA>
* ~~那个谁的一个仓库有分类高频题~~

1. ~~牛客SQL练习~~
2. 顺序看看到了399 到2000
3. 计数⭐搞定
4. 枚举⭐搞定
5. 数学⭐都实际写写代码 1588 到2119
6. 博弈⭐搞定 1561到最后看labuladong
7. 模拟题⭐搞定
8. 位运算⭐搞定810
9. 贪心⭐ 搞定1647到 2027
10. 二分查找⭐搞定 从中等开始852
11. 分治⭐搞定
12. 前缀和⭐搞定基本都得配合滑动窗口
13. 递归 (深度广度搜索) ⭐841搞定
14. 记忆化搜索⭐搞定
15. 回溯 ⭐搞定 638 到
16. 图(深度广度) ⭐ 搞定 都还没做
17. 动态规划⭐搞定 从中等 354.俄罗
18. 矩阵 ⭐还剩中等
19. 简单题到先搁了，去做SQL
20. 栈 ⭐搞定
21. 树 ⭐搞定
22. 队列 ⭐搞定
23. 滑动窗口 ⭐搞定
24. 链表 ⭐搞定
25. 双指针 ⭐搞定
26. 哈希表 ⭐搞定
27. 字符串 ⭐搞定
28. 并查集
29. 字典树，Trie前缀树⭐
30. 树状数组
31. 线段树

……

## 极客时间

* ennnnn搞了半天，有营养的还得自己筛选，多看看系统设计的视频吧，培养系统设计的思想

### 算法🌺

* 算法面试通关40讲⭐❤
* 在看
* 算法实战高手课⭐❤
* 数据结构与算法之美 挑着看⭐❤
* 在看
* 动态规划面试宝典❤

### Java

* Java核心36讲，视情况而定吧，要不要准备面试到时候再看
* 在看
* Java并发编程实战
* 【视频课】玩转Spring全家桶⭐❤
* 在看
* 【视频课】小马哥讲Spring核心编程思想
* SpringXXX⭐

### Go

* ~~Go语言项目开发实战⭐️Done~~
* ~~讲的是一个IAM应用，后面的云原生部分讲得不错~~
* ~~✅命令行工具上面的设计和权限控制啥的都有很大参考价值~~
* ~~Go语言从入门到实战Done~~
* ~~一般~~
* ~~❌23协程之前没看，后边都看完了到最后~~
* ~~Go语言核心36讲-先不看了❌~~
* ~~X~~

### Web

* ~~后端技术面试38讲⭐【听】~~
* ~~08在看~~
* ~~从08开始听，第二章略过~~
* ~~直接从21讲开始听，这一章不错~~
* ~~听完了done，最后一章一般~~
* 213-手把手带你写一个Web框架✨
* 在看19-应该是看到了最新版
* ennnnnnnn，还行吧
* 后端存储实战课
* Django快速开发实战
* 高并发系统40问⭐❤【听】
* ~~全栈工程师修练指南【听】~~
* 第一章的讲HTTP的历史的不错
* 20讲讲全栈团队的组成的不错
* 在听21，中间都还行，还算行
* 在听28，最后一章还行吧
* ~~第一遍done~~

### 音视频

* 攻克视频技术
* 搞定音频技术
* 从0打造音视频直播系统
* 在看

### 前端

* 【视频课】Vue开发实战⭐
* 【视频课】React实战进阶45讲⭐
* TypeScript开发实战
* 9小时搞定微信小程序开发
* Electron开发实战
* 图解谷歌V8⭐️❤️
* 待看
* 92-浏览器工作原理与实践⭐️❤️
* 待看

### 机组&操作系统⭐❤

* 深入浅出计算机组成原理⭐
* 待看6
* 操作系统实战45讲✨⭐【听】
* 在看18
* 很硬核，很多汇编
* 趣谈Linux操作系统✨⭐
* 03待看
* 【视频课】Linux实战技能100讲⭐❤
* Linux内核技术实战课
* Linux性能优化实战

### 编译原理⭐❤️

* ~~①编译原理之美✨⭐Done~~
* ~~Done，✅很好，值得再次品读~~
* ~~②编译原理实战课✨⭐满分✅~~
* ~~💯✅看完了~~
* ③205-手把手带你写一门编程语言✨✅
* 待看33之后，已经到最新
* 深入拆解Java虚拟机
* ①219-编程高手必学的内存知识✨
* 深入C语言和程序运行原理
* JavaScript核心原理解析
* WenAsemmbly入门课

### C++&Rust

* 【视频课】张汉东的Rust实战课⭐️✨
* 第1章over-集合容器和crate那没看
* 49难💔
* 206-陈天 · Rust 编程第一课✨⭐
* 064 现代C++实战30讲【完结】
* 罗剑锋的C++实战笔记

### 网络

* 网络编程实战
* 【视频课】Web协议详解与抓包实战
* 在看
* 透视HTTP协议
* 趣谈网络协议✨【听】
* 待看14

### 数据库⭐

* MySQL必知必会⭐❤
* 在看第6讲
* MySQL实战45讲✨❤
* 在看
* ~~SQL必知必会⭐️~~**~~Done~~**
* ~~TODO 21集、22集值得后面再仔细看⭐️~~
* ~~【视频课】MongoDB高手课⭐️Done~~
* ~~done还行吧~~
* Redis核心技术与实战
* 在看
* Redis原码剖析与实战

### 杂项

* Kafka核心技术与实战⭐
* etcd实战课⭐
* 容器实战高手课⭐
* TODO: 2待看
* makefile
* Vim实用技巧必知必会
* OAuth 2.0实战课
* 检索核心技术20讲
* 恋爱必修课
* 打造爆款短视频
* 手机摄影
* 互联网人的英语私教课⭐
* RPC实战与核心原理
* Web安全攻防实战
* 实用密码学

### 机器学习&AI【听】❤️⭐️

* 程序员的数学基础课⭐
* 211-零基础实战机器学习【听】
* 在听
* ~~人工智能基础课【听】~~
* ~~done，还行吧~~
* ~~AI技术内参⭐️【听】~~
* 45待看
* 论文解读部分没仔细看，看不太懂
* 擦，有点不清楚，图很少，但是涉猎广泛
* 从45开始听，听到47
* 推荐系统63开始到83，我擦是真的难
* 04-自然语言处理及文本处理核心技术 (19讲)
* 到90
* 05-广告系统核心技术 (19讲)
* Done
* 06-计算机视觉核心技术 (13讲)
* Done
* 07-数据科学家与数据科学团队养成 (25讲)
* 08-人工智能国际顶级会议 (31讲)
* 09-热点话题讨论 (7讲)
* Done
* 成为AI产品经理【听】⭐❤
* 03正在看
* 在听03
* ~~机器学习40讲【听】⭐❤~~
* ~~在听11~~
* ~~还是很好的，目前听不太懂……~~
* ~~一遍done~~
* 【视频课】NLP实战高手课⭐【听】
* 在看
* 推荐系统三十六式⭐【听】
* 在看
* 深度学习推荐系统实战⭐【听】
* 在看
* 【视频课】深度学习应用实践60讲⭐❤️【听】
* 在看21

### 容器&微服务&云&DevOps

* DDD实战课⭐️【听】
* 在看
* 容器实战高手课❤⭐
* 11待看，难💔
* 深入剖析Kubernetes⭐❤
* ~~DevOps实战笔记⭐【听】~~
* ~~待看08，先不看了，重要性不高，及时止损~~
* ~~❌这个……都是概念的~~
* ~~持续交付36讲【听】~~
* ~~待看02，同上，不太重要及时止损~~
* ~~❌这个……都是概念的~~

### 分布式&微服务⭐❤

* 深入浅出分布式技术原理--还没买⭐️❤️
* ~~114-分布式协议与算法实战✨~~**~~Done~~**
* ~~Done~~
* 深入浅出区块链⭐️❤️【听】
* 在看03
* 从03开始听书，听到24
* 后续再听，有点听不太懂……
* 主要是讲各种币的原理的
* 说透区块链⭐️
* ~~3技术篇ok~~
* 待看04-万物皆可区块链+ (5讲)
* 147-分布式数据库30讲✨❤⭐
* 在看
* 【视频课】Service Mesh实战⭐
* 【视频课】Zookeeper实战与源码剖析
* 【视频课】分布式缓存高手课❤
* 【视频课】分布式系统案例课⭐️❤⚡
* 到19讲了，有点难，进行不下去了❌💔
* ~~分布式技术原理与算法解析~~
* **~~Done~~**
* ~~【视频课】微服务架构核心20讲~~ **~~DOne~~**
* ~~很短，几个小时看完了，前半段没啥有用的。很多是概念性的东西，但是都是核心概念。~~
* 从0开始学微服务【听】
* 待听14
* ❌咋感觉没啥营养呢
* 【视频课】微服务架构实战160讲
* 待看01
* ~~深入浅出云计算⭐Done~~
* ~~Done-都是一些概念性的东西，还凑合吧~~
* ~~【视频课】SpringBoot与Kubernetes云原生微服务实战Done~~
* ~~还可以，还不错Done~~
* Kafka核心技术与实战
* 126-Kafka核心源码解读
* 消息队列高手课❤⭐
* 即时消息技术剖析与实战

### 架构【听】❤️⭐️

* 从0开始学架构⭐️【听】
* 待看05
* xushiwei的架构课
* 系统性能调优必知必会
* 架构实战案例解析
* 郭东白的架构课
* 数据中台实战课
* 高并发系统设计40问⭐️
* 如何设计一个秒杀系统✨⭐
* 在看
* 214-手把手带你搭建秒杀系统✨⭐
* 在看

### 大数据

* 21-从0开始学大数据✨⭐
* 05-模块三 大数据开发实践 (8讲)待看
* 212-大数据经典论文解读
* 先不看了，放放
* 数据分析实战45讲⭐️
* 【视频课】Spark核心原理与实战

### 软技能&面试【听】❤️⭐️

* ~~左耳听风✨⭐【听】~~
* 在听11%12，机器学习
* 6月26
* ~~陈皓老师这个专栏介绍了太多的资料，有时间梳理一下，慢慢学！~~
* ~~听过一遍✓~~
* <https://coolshell.cn/>
* 职场求生攻略
* 技术管理案例课
* 技术面试官识人手册
* 大厂晋升指南
* 面试现场⭐❤
* 程序员的个人财富课
* 讲好故事

### 产品

* 硅谷产品实战36讲

### 训练营-【先过文档，再选择性看视频】

* Go进阶训练营第1期⭐
* 第二周-讲什么鸡儿Error
* 在看9-1
* 极客-算法训练营⭐挑着看看
* 在看
* 极客架构师训练营
* 在看
* 大数据训练营
* 云原生训练营
* 课程有点重，可以先看一些专栏课
* Java训练营✨-先看文档不要看视频去浪费时间，视频废话太多
* 07周开始看-从07周开始分布式
* 产品训练营-有空再看

## YouTube课

* AWS⭐❤
* 基础 AWS Basics for Beginners - Full Course <https://www.youtube.com/watch?v=ulprqHHWlng>
* full course
* edureka！
* <https://www.youtube.com/watch?v=k1RI5locZE4>
* Simpilearn
* <https://www.youtube.com/watch?v=uQdzcIf_KII>
* 2022版本 <https://www.youtube.com/watch?v=ZB5ONbD_SMY>
* SAA
* <https://www.youtube.com/watch?v=2tAN9FUzCmw&list=PLiH9_MU-6RjI9gdFqmvUfKRfw_zRxIb6o>
* <https://www.youtube.com/watch?v=Ia-UEYYR44s>
* SAP
* <https://www.youtube.com/watch?v=QxMZDzyWJj4&list=PLYD7D0d62WlABQJODWIq-Eks1B36OYzl8>
* 分布式-System Design Interview
* IBM科普⭐️
* 各种收藏夹视频

## 国外CS网课【做lab很关键⭐️✅】

* 计网⭐
* 视频
* lab
* 操作系统⭐
* 视频
* lab
* 数据库⭐
* 视频
* lab
* 编译原理⭐
* 视频
* lab
* 分布式系统⭐
* 视频 04待看
* lab
* 论文
* **还是得看论文，老师讲的听不懂~**
* sicp⭐
* 视频
* lab
* 配套图书
* csapp⭐
* B站视频
* lab
* 配套图书

## bilibili

* 多线程
* JUC
* ~~尚硅谷mysql~~ https://www.bilibili.com/video/BV1iq4y1u7vj ~~不看啦，极客时间的更好~~
* https://www.bilibili.com/video/BV1xW411u7ax~~170+~~
* Rust rust-in-databend高阶 <https://space.bilibili.com/275673537>
* <https://github.com/wubx/rust-in-databend>
* Rust-杨旭
* Golang-杨旭

## 知乎收藏夹

* 分布式系统
* 收藏夹
* 关注的人+Functional Programming
* 经典论文
* 我在日本企业工作-live

## 电子书～仓库

* MySQL是怎样运行的 ，先把极客时间的看完吧
* 数据密集型系统设计 《》<https://github.com/Vonng/ddia>
* 《《七周七并发模型》
* 《《七周七数据库~》
* 《七周七语言》
* 北航程序设计语言原理教材（共18章）
* 到第5章 在看
* 《自己动手写Lua虚拟机》
* 《自己动手写Java虚拟机》
* 《自己动手写Git？》
* 《自己动手写操作系统》
* 《程序员的修养》
* 《》unix编程编程艺术
* 《Go底层原理分析》

## 日语

* 喜马拉雅50天听力文稿背诵

## 英语

* 外文IT论坛
* 之前收集的硬盘里的视频

## Github

* 各种有趣的库
* 编写 Rust 语言的操作系统 <https://github.com/rustcc/writing-an-os-in-rust>
* 分布式系统模式 <https://github.com/dreamhead/patterns-of-distributed-systems>
* 分布式系统 <http://book.mixu.net/distsys/index.html>

## 系统设计

[github系统设计](https://github.com/donnemartin/system-design-primer) [中文🇨🇳](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md) ⭐

* YouTube视频
* 看英文版

系统设计主题的索引

各种系统设计主题的摘要，包括优点和缺点。每一个主题都面临着取舍和权衡。

* [系统设计主题：从这里开始](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E7%B3%BB%E7%BB%9F%E8%AE%BE%E8%AE%A1%E4%B8%BB%E9%A2%98%E4%BB%8E%E8%BF%99%E9%87%8C%E5%BC%80%E5%A7%8B)[第一步：回顾可扩展性的视频讲座](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E7%AC%AC%E4%B8%80%E6%AD%A5%E5%9B%9E%E9%A1%BE%E5%8F%AF%E6%89%A9%E5%B1%95%E6%80%A7scalability%E7%9A%84%E8%A7%86%E9%A2%91%E8%AE%B2%E5%BA%A7)
  + [第二步：回顾可扩展性的文章](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E7%AC%AC%E4%BA%8C%E6%AD%A5%E5%9B%9E%E9%A1%BE%E5%8F%AF%E6%89%A9%E5%B1%95%E6%80%A7%E6%96%87%E7%AB%A0)
  + [接下来的步骤](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%8E%A5%E4%B8%8B%E6%9D%A5%E7%9A%84%E6%AD%A5%E9%AA%A4)
* [性能与拓展性](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%80%A7%E8%83%BD%E4%B8%8E%E5%8F%AF%E6%89%A9%E5%B1%95%E6%80%A7)
* [延迟与吞吐量](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%BB%B6%E8%BF%9F%E4%B8%8E%E5%90%9E%E5%90%90%E9%87%8F)
* [可用性与一致性](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%8F%AF%E7%94%A8%E6%80%A7%E4%B8%8E%E4%B8%80%E8%87%B4%E6%80%A7)[CAP 理论](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#cap-%E7%90%86%E8%AE%BA)[CP - 一致性和分区容错性](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#cp--%E4%B8%80%E8%87%B4%E6%80%A7%E5%92%8C%E5%88%86%E5%8C%BA%E5%AE%B9%E9%94%99%E6%80%A7)
  + - [AP - 可用性和分区容错性](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#ap--%E5%8F%AF%E7%94%A8%E6%80%A7%E4%B8%8E%E5%88%86%E5%8C%BA%E5%AE%B9%E9%94%99%E6%80%A7)
* [一致模式](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E4%B8%80%E8%87%B4%E6%80%A7%E6%A8%A1%E5%BC%8F)[弱一致性](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%BC%B1%E4%B8%80%E8%87%B4%E6%80%A7)
  + [最终一致性](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%9C%80%E7%BB%88%E4%B8%80%E8%87%B4%E6%80%A7)
  + [强一致性](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%BC%BA%E4%B8%80%E8%87%B4%E6%80%A7)
* [可用模式](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%8F%AF%E7%94%A8%E6%80%A7%E6%A8%A1%E5%BC%8F)[故障切换](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%95%85%E9%9A%9C%E5%88%87%E6%8D%A2)
  + [复制](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%A4%8D%E5%88%B6)
* [域名系统](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%9F%9F%E5%90%8D%E7%B3%BB%E7%BB%9F)
* [CDN](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%86%85%E5%AE%B9%E5%88%86%E5%8F%91%E7%BD%91%E7%BB%9Ccdn)[CDN 推送](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#cdn-%E6%8E%A8%E9%80%81push)
  + [CDN 拉取](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#cdn-%E6%8B%89%E5%8F%96pull)
* [负载均衡器](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E8%B4%9F%E8%BD%BD%E5%9D%87%E8%A1%A1%E5%99%A8)[工作到备用切换（Active-passive）](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%B7%A5%E4%BD%9C%E5%88%B0%E5%A4%87%E7%94%A8%E5%88%87%E6%8D%A2active-passive)
  + [双工作切换（Active-active）](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%8F%8C%E5%B7%A5%E4%BD%9C%E5%88%87%E6%8D%A2active-active)
  + [四层负载均衡](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%9B%9B%E5%B1%82%E8%B4%9F%E8%BD%BD%E5%9D%87%E8%A1%A1)
  + [七层负载均衡](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E4%B8%83%E5%B1%82%E8%B4%9F%E8%BD%BD%E5%9D%87%E8%A1%A1%E5%99%A8)
  + [水平扩展](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%B0%B4%E5%B9%B3%E6%89%A9%E5%B1%95)
* [反向代理（web 服务器）](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%8F%8D%E5%90%91%E4%BB%A3%E7%90%86web-%E6%9C%8D%E5%8A%A1%E5%99%A8)[负载均衡与反向代理](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E8%B4%9F%E8%BD%BD%E5%9D%87%E8%A1%A1%E5%99%A8%E4%B8%8E%E5%8F%8D%E5%90%91%E4%BB%A3%E7%90%86)
* [应用层](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%BA%94%E7%94%A8%E5%B1%82)[微服务](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%BE%AE%E6%9C%8D%E5%8A%A1)
  + [服务发现](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%9C%8D%E5%8A%A1%E5%8F%91%E7%8E%B0)
* [数据库](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%95%B0%E6%8D%AE%E5%BA%93)[关系型数据库管理系统（RDBMS）](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%85%B3%E7%B3%BB%E5%9E%8B%E6%95%B0%E6%8D%AE%E5%BA%93%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9Frdbms)[Master-slave 复制集](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E4%B8%BB%E4%BB%8E%E5%A4%8D%E5%88%B6)
  + - [Master-master 复制集](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E4%B8%BB%E4%B8%BB%E5%A4%8D%E5%88%B6)
    - [联合](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E8%81%94%E5%90%88)
    - [分片](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%88%86%E7%89%87)
    - [非规范化](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E9%9D%9E%E8%A7%84%E8%8C%83%E5%8C%96)
    - [SQL 调优](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#sql-%E8%B0%83%E4%BC%98)
  + [NoSQL](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#nosql)[Key-value 存储](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E9%94%AE-%E5%80%BC%E5%AD%98%E5%82%A8)
    - [文档存储](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%96%87%E6%A1%A3%E7%B1%BB%E5%9E%8B%E5%AD%98%E5%82%A8)
    - [宽列存储](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%88%97%E5%9E%8B%E5%AD%98%E5%82%A8)
    - [图数据库](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%9B%BE%E6%95%B0%E6%8D%AE%E5%BA%93)
  + [SQL 还是 NoSQL](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#sql-%E8%BF%98%E6%98%AF-nosql)
* [缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E7%BC%93%E5%AD%98)[客户端缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%AE%A2%E6%88%B7%E7%AB%AF%E7%BC%93%E5%AD%98)
  + [CDN 缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#cdn-%E7%BC%93%E5%AD%98)
  + [Web 服务器缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#web-%E6%9C%8D%E5%8A%A1%E5%99%A8%E7%BC%93%E5%AD%98)
  + [数据库缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%95%B0%E6%8D%AE%E5%BA%93%E7%BC%93%E5%AD%98)
  + [应用缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%BA%94%E7%94%A8%E7%BC%93%E5%AD%98)
  + [数据库查询级别的缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%95%B0%E6%8D%AE%E5%BA%93%E6%9F%A5%E8%AF%A2%E7%BA%A7%E5%88%AB%E7%9A%84%E7%BC%93%E5%AD%98)
  + [对象级别的缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%AF%B9%E8%B1%A1%E7%BA%A7%E5%88%AB%E7%9A%84%E7%BC%93%E5%AD%98)
  + [何时更新缓存](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E4%BD%95%E6%97%B6%E6%9B%B4%E6%96%B0%E7%BC%93%E5%AD%98)[缓存模式](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E7%BC%93%E5%AD%98%E6%A8%A1%E5%BC%8F)
    - [直写模式](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E7%9B%B4%E5%86%99%E6%A8%A1%E5%BC%8F)
    - [回写模式](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%9B%9E%E5%86%99%E6%A8%A1%E5%BC%8F)
    - [刷新](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%88%B7%E6%96%B0)
* [异步](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%BC%82%E6%AD%A5)[消息队列](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%B6%88%E6%81%AF%E9%98%9F%E5%88%97)
  + [任务队列](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E4%BB%BB%E5%8A%A1%E9%98%9F%E5%88%97)
  + [背压机制](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E8%83%8C%E5%8E%8B)
* [通讯](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E9%80%9A%E8%AE%AF)[传输控制协议（TCP）](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E4%BC%A0%E8%BE%93%E6%8E%A7%E5%88%B6%E5%8D%8F%E8%AE%AEtcp)
  + [用户数据报协议（UDP）](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E7%94%A8%E6%88%B7%E6%95%B0%E6%8D%AE%E6%8A%A5%E5%8D%8F%E8%AE%AEudp)
  + [远程控制调用协议（RPC）](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E8%BF%9C%E7%A8%8B%E8%BF%87%E7%A8%8B%E8%B0%83%E7%94%A8%E5%8D%8F%E8%AE%AErpc)
  + [表述性状态转移（REST）](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E8%A1%A8%E8%BF%B0%E6%80%A7%E7%8A%B6%E6%80%81%E8%BD%AC%E7%A7%BBrest)
* [安全](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%AE%89%E5%85%A8)
* [附录](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E9%99%84%E5%BD%95)[2 的次方表](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#2-%E7%9A%84%E6%AC%A1%E6%96%B9%E8%A1%A8)
  + [每个程序员都应该知道的延迟数](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E6%AF%8F%E4%B8%AA%E7%A8%8B%E5%BA%8F%E5%91%98%E9%83%BD%E5%BA%94%E8%AF%A5%E7%9F%A5%E9%81%93%E7%9A%84%E5%BB%B6%E8%BF%9F%E6%95%B0)
  + [其它的系统设计面试题](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%85%B6%E5%AE%83%E7%9A%84%E7%B3%BB%E7%BB%9F%E8%AE%BE%E8%AE%A1%E9%9D%A2%E8%AF%95%E9%A2%98)
  + [真实架构](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E7%9C%9F%E5%AE%9E%E6%9E%B6%E6%9E%84)
  + [公司的系统架构](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%85%AC%E5%8F%B8%E7%9A%84%E7%B3%BB%E7%BB%9F%E6%9E%B6%E6%9E%84)
  + [公司工程博客](https://github.com/donnemartin/system-design-primer/blob/master/README-zh-Hans.md#%E5%85%AC%E5%8F%B8%E5%B7%A5%E7%A8%8B%E5%8D%9A%E5%AE%A2)

## behavior question

* 面接100问中日英三语版本

## 得到

* 专栏
* 傅佩荣的西方哲学
* 薛兆丰经济学
* 香帅的北大金融学课
* 硅谷来信-1
* 硅谷来信-2
* 硅谷来信-3
* 李笑来财富自由
* 听书
* 心理学
* 经济学
* 自然科学
* 哲学
* 大师课
* 熊毅：佛学50讲
* 刘伟：存在主义哲学20讲
* 政治学通识30讲
* 吴军：信息论40讲
* 吴军：数学通识50讲

## 不知道