**提升讲题能力**:

虽然你不可能总是找别人帮你mock interview，但在自己平时在做题之后，对着小黄鸭自言自语，复述解题思路是非常重要的。

除了说，学会在共享文档里画图举例子，走case，对于把问题讲清楚有着极其美妙的效果。

能不能把题讲明白和能不能把题做出来同等重要。

题库，一般是 Leetcode 为主，可以偶尔辅助 LintCode。**题目范围**主要是总表前两三百道高频，按公司分类的高频，和网上的各种面经。

**题目分为四个优先级**：

<1> **必背**，大约20~30道，都是各类型题目的典型模板题，基本需要刷十几遍，做到迷迷糊糊半昏迷状态也能熟练默写的肌肉记忆状态。

<2> **核心**，大约100~150道，主要是各种高频题和经典题，基本在5~8遍以上，需要做到最优解，medium难度10分钟以内，hard难度15分钟以内，无错一遍过，同时要能解释清楚思路。另外有多重解法的也要掌握，知道不同解法间的优缺点和trade off原因。

<3> **重点**，大约200~300道，核心题之外的高频题，基本在5遍左右。这样遇到原题或者类似题的时候，基本思路、逻辑不会错。能不能临场完全bug free要看基本功和运气。

<4> 普通，上述题之外所有你刷过的题。基本上做过一两遍，有个思路，临场遇见了不会慌。

**初刷：**

新手期遇到新题，如果短时间内（五分钟）没有思路没关系，不要死缠烂打，早点看答案。

找到最优的答案并尽力去理解，用自己的面试语言和规范的代码风格进行“抄写”，不完全懂也没关系，标注一下，以后再看。

每题时长控制在半小时之内。

**精刷：**

针对初刷过还未充分理解的题。

尽量全面理解答案的要义，并且至少在能自己“默写”出来，并调试通过。

将解题思路和形成的标准答案加入到笔记当中。

如果实在不能完全领悟，需要在笔记中用显著符号标注该题，写下困惑的地方，过一段时间再试试看。

每题时长控制在一小时之内。

**复刷：**

针对做过的题进行复习。

自言自语clarify题意，讲请楚自己的思路，写题，检查，跑case，争取bug free一遍过。

只有参考笔记和标准答案复盘自己的错误和不足。

每题时长控制在20分钟内。

**回顾：**

按照笔记的标签（比如按分类，或者按优先级）大量复习题目。

针对做过的题进行复习，如果对这一题还有映象，则在大脑中过一遍解题思路及实现中的细节要点。

如不记得，则进行主动思考寻求解法，之后对比之前的答案和笔记，看是否正确。

如果不正确则需要强化思考并记忆，不需要写题。

每题时长控制在5-10分钟内。

**讲题：**

在对此题解题思路正确的情况下，尝试在白板上画图并讲解自己的解题思路，看是否能够流畅完成此过程。

找到一个让自己满意的描述方式，将图表和讲题思路/要点加入到笔记当中。

**模拟：**

抽取一道新题，看能不能做到复刷中的效果目标。

把题目记入笔记。

每题时长控制在30分钟内。

分类知识学习+核心题目+反复训练+进阶题目+复习升华+融会贯通的系统性方法.

**分阶段使用不同的方法**，实现螺旋形上升的学习过程. 循序渐进的循环刷题效率可能更高.

刚开始刷题的时候初刷、精刷比较多，中期过题、复刷比较多，最后阶段过题和模拟比较多。

（1）基本知识学习

（2）初刷必背+核心

分类学习各算法分类的基本概念和思路，以初刷模式刷一遍必备+核心题目（30 + 100左右）。

其中概念和思路要做到基本理解，但是题目一时半会无法完全理解很正常，把题号+题目+参考答案 按照第二章说的方式记在笔记里即可，刷题记录可以标记为深红（尚未理解）。

目标：知道算法大概有哪些门类，题目大概长什么样，都是在问些什么，总结100+题的答案，并收录在笔记里。

（3）精刷必背题

按分类尝试深入理解每个分类下面的必背题。

对于核心题可以按照初刷的方式再过一遍，看不懂就跳过。

目标：基本理解了必背题的解题思路。

（4）精刷核心题

尝试默写出必背题，对核心题使用精刷操作。

目标：能够基本写出必背题的答案，基本理解核心题的解题思路。（第一层境界）

（5）初刷重点题

按照分类对重点题使用初刷模式。

每道题自己先思考几分钟，有思路就先自己试一试，没思路就看答案，不管是否成功写对，最后都要找到标准答案。

目标：把重点题过一遍，至少知道前400题长什么样。

（6）复刷必背+核心

继续熟练默写必背题，并尝试讲题，包括clarify，描述思路，过一些test case。

尝试自己写出核心题。

目标：能都随手写出必背题；碰到核心题，都能有思路，虽然可能有小问题，但基本都能写出来。（第二层境界）

（7）精刷重点题 + 回顾必背、核心题

按照高频顺序（不再是分类）精刷重点题，如果觉得比较费时间，就从前往后刷到哪是哪。

同时回顾（过题）必备题和核心题，做过时间的比较久的重点题也可以放在回顾计划中。回顾时主要关注思路和实现要点、易错点。

目标：对于前300 ~ 400题形成相对完整的知识体系、笔记体系。

（8）复刷必背、核心题 + 回顾重点题

做到必背、核心题基本都能一次写对，尝试讲题。

目标: 能秒杀核心题。

（9）复刷重点题 + 回顾所有做过的题

如果重点题能直接做出来（可能只是一小部分），就对照一下之前的答案对比一下小的细节差异，如果五分钟没有思路，或者三十分钟没写对，直接看之前笔记的答案修正问题。

目标：对于重点题，也基本看到就都有思路。

（10）回顾所有做过的题 + 模拟

按频率或者面经，以模拟的形式尝试还没做过的题。

包括clarify，解法描述，写题，跑case等等。

不管有没有做对，都要看答案形成笔记。

目标：对于必背、核心题都可以秒杀并详细讲解；对于重点题可以在没有太大逻辑错误的情况下写对；对于400道以外的题目也做过一些，并且看到大部分新题都至少有思路。（第三层境界）

（10 +）回顾旧题 + 做新题

进入了一个开放性的成长阶段，面试能用到的理论体系已经完整建立，主流高频题都能搞定。

通过各种新题磨炼自己的临场技能，补充知识体系中的细节。

目标：前往第四层境界的路上。

第三层境界：

总题量 400~500。

熟练掌握十几种类型，300~400的核心/重点题，对每一类题的套路了然于心。

看到原题或类似的题基本能在规定时间内完成高质量，最优的代码。可能有一些小问题（typo，API记错），导致不能一遍过，但没有思路和逻辑上的问题，基本上不会提交两次以上。

能够清晰讲解解题思路，并让别人听懂。

对于没做过的题目，能迅速地进行分类；或者对于有多重解法的题目，能分析不同类解法的要点和优劣，理解取舍的原因。

基本上能根据自己想出的思路正确地实现最优解的代码，但可能会有一些细节考虑地不周到，造成提交超过两次。

在讲解这些没做过的题目的时候，可能不太流畅，但最终也能说得通。

第四层境界：

实现了对面试中所有常见类别的算法题套路与解法轻车熟路，完成了理论体系的融会贯通。

深刻理解题目的内涵，清楚的理解当一些题目条件发生变化时将对题目的解法与复杂度产生什么样的影响，能够大致猜到follow up的方向。

不管题目有没有做过，看到以后都很快形成清晰的思路，找到面试环境下的最优解，同时对一些需要注意的常见代码细节了然于胸。

能清晰流畅地表达讲解自己的思路。

在较短时间内流畅且无错地完成符合规范的代码，一气呵成。

整体而言，为了能较好的通过大部分面试，一般需要修炼到第三层境界。

当然如果经过努力最终到达第四层境界，至少在算法这个分项上，你大概已经咸鱼翻身成为大佬，具备手握包括顶级大厂在内的N个offer，在HR之间比价聊包的资本了。

**笔记内容**

每一题记录为一页笔记。

首先会抄下题号、难度、链接和截图题面。这样的好处是复习的时候不用再点开leetcode，省时间。

其次作者一般会添加这几个部分内容：

讨论澄清：

这个其实也是你需要和面试官clarify的内容。

因为实际面试不会像leetcode把一道题明明白白的摆在你面前，而是先给你一个模糊的描述，需要你问问题、讨论逐步细化明确需求。

所以在平时做题的时候，多想一想这些问题，对面试非常有帮助。

一个小tip，leetcode题目里面的notes，constraints 很多时候就是我们面试中需要先澄清的问题。

仔细想一想为什么要加这些条件，如果不加的话题目解法会出现什么样的变化，带着这样的思路去看题，也就知道面试的时候需要clarify什么了。

解法思路：

如果有多重解法，分开写。

对于每一种解法，描述一下核心思路和主要解法步骤。

其实这也是面试中，你向面试官简要描述你的解法的过程，对着屏幕自言自语练一练。

注意要点：

代码实现中需要注意的细节，这主要来自于你写这道题是遇到的具体问题，和 debug中领悟的要点，往往也是易错点或者比较tricky的地方。

复杂度：

常见的复杂度分析，每次做题想一想，分析一遍，自然就熟悉了。

同时也是不同算法之间trade off重要准则。

代码：

贴上你自己准备的"golden"答案，一方面方便复习，同时也促进了代码风格的形成。

下面是一个笔记的例子，因为优先级相对较高，所以写的比较细。

其实不用每题都写这么复杂，抓准每题的重点即可。

也不需要一次性写这么多，很多内容实在反复刷反复想的过程中逐步添加的。



**刷题记录表**

除了分类笔记，我们还需要一张总体的刷题表来记录自己当前的状态。

它可以帮助我们一目了然的看出哪些题我们做的还不够，下一步需要加强哪些方向。

这张表包含：

要求：必背，核心，重点，普通，分别用不同颜色区分，方便观察。

题目链接：指向leetcode的链接，并用颜色区分难度。

分类：分类标签。

当前刷题状态：

包括最近刷题、过题的一些信息。特别是最近一次的写题时长和重交次数，这个决定刷题状态的颜色条：

深红：连答案都没看懂。

浅红：最终提交通过，但是时间太长或者重交次数过多（简单/中等超过10min，hard超过15min, 重交达到2次且有逻辑漏洞）。

黄色：第一次能在规定时间内完成，顶多有一两个typo造成的错误。

绿色：连续两次按时无错通过。

备注：一些其他信息，灵活使用，比如某题是其他题的follow up之类的。



**刷题计划表**

也可以定制一个自己的刷题计划表（不同于刷题记录表），方便计划和打卡。

一般来说，刷题这样的高脑力活动，一天4~6个小时效率会比较高。

超出的时间做一些知识学习、回顾旧题这样相对较轻松的任务会比较合适。

如果经常无法按时完成，需要及时调整。

注意劳逸结合。大脑和肌肉一样，练多了效率会降低，特别是对于二十多岁的中老年人。

把每天有限的心力及时用在最困难的任务上。



**一些重复三遍的小建议：**

学不明白不是你的错，是题目确实难

不要和某一题死磕太久

劳逸结合

能讲和能写一样重要；但是在会讲题之前，得先会写；会写之前，得先理解

按照现在的市场价格，学会一题相当于赚 $500，而且是每年

过程是艰难的，结果是美好的