

我的秋招2022

【本文共计14372字，预计阅读时间为20分钟】

前序

在本文开始以前，首先需要声明的是这篇超详细面经**并不是****笔者本人的求职笔记**，它是笔者学校隔壁实验室22届毕业学弟的面试心路历程和经验心得。由于笔者和这位学弟经常讨论校招求职和职业发展的问题，并且在秋招以前就和他打好了招呼，于是就有了这篇学弟应邀输出的万字ssp面经：

2021年9月2日 下午 6:10

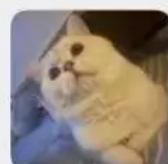
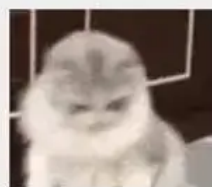
有没有可能



到时邀请你出一篇秋招心得博客啥的



你撤回了一条消息



🙋我吗

我感觉我挺菜的呀

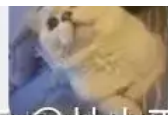
我现在劝退算法岗🙋

不不不



这位学弟可以说是非常谦虚了

都这么多 offer 了



简单而言，接下来你要看到的这篇**2022届秋招算法面经**，具备以下这几个特点：@林小平

- **全生命周期**：学弟的经验心得覆盖“论文/项目->实习->岗位投递->面试”的全部流程，非常适合什么都还没开始准备的研一/研二同学借鉴参考；
- **像素级别的经验分享**：可能也是事先和学弟打好了招呼，加上他十分细心认真的复盘总结，因此下面的面经很多都是像素级还原了面试题（不止一家公司的面试），因此也同样适合春招准备重新冲刺一把的应届同学；
- **offer大满贯**：拿到了**8家以上**的大厂offer（不少是核心部门），薪资方面就（高压线警告）了；
- **满满的心路历程**：笔者在拿到稿件后事先通读了一遍，发现学弟的面经风格并不像牛客网上的文章只有题目和回答，而是穿插着自己的感悟，像聊天一样的把心路历程和需要注意的地方分享给读者。这种面经看完不仅不容易焦虑，而且代入感更强，适合收藏反复阅读~~

下面就是学弟面经的正文部分啦~~



各位新时代的“农民工”，大家好！我是又卷又摸鱼的卷陌雨同学，受林小平学长的邀请，很开心可以和面友们分享我在NLP和搜广推算法岗位方面的秋招面试经验。****面试经验这个玩意非常主观，不同部门给不同人带来的感受千差万别。*本文会尽可能不带个人偏见地描述秋招的所见所闻，大家可酌情参考，希望能帮助到将来想要从事算法行业的每一位。**

先给面友们罗列下陌雨的背景信息以供参考。

****学历:** **211本 + 985硕

****论文:** **—A一作 + —A二作

****实习:** **一段鹅厂TEG的NLP算法岗暑期实习

****项目:** **无重大项目

****比赛:** **无权威比赛

****代码:** **leetcode 500+, 面试手撕代码环节没出过大问题; 笔试比较拉稀, 只乱杀过网易和半乱杀了华为, pdd压线, 其他厂几乎都是跳过笔试。

****基础:** **DL和NLP的相关知识都没大问题; ML相对薄弱一些; 搜广推没系统学习过。

****嘴炮:** **这是很容易忽视的一项, “你很优秀”和“你能让面试官觉得很优秀”是有gap的。陌雨自我感觉这次秋招嘴炮属于中等偏上、偶尔拉满。临场反应和freestyle发挥都挺好, 反问环节的面试反馈通常也比较正面, 反而是我的airpods pro被diss了好多次收音垃圾。

还有秋招的面试结果, 陌雨这次共计划投递12家公司, 非常有幸几乎都一次“混”过了大多数厂的面试环节, 最终也拿到了不少心仪公司的意向, 对秋招发挥整体还算比较满意。比较可惜的是不少外企和银行我没来得及投, **本文的参考性暂时只覆盖国内这部分大厂。**

****腾讯:** **NLP算法 (1面 -> 2面 -> hr面 -> gm面 -> **意向**)

****阿里:** **广告算法 (笔试 -> 1面 -> 2面 -> 3面 -> hr面 -> **意向**)

****字节:** **广告算法 (1面 -> 2面 -> 3面 -> hr面 -> 4面 -> **意向**)

****百度:** **搜索算法 (1面 -> 2面 -> 3面 -> **意向**)

****美团:** **NLP算法 (1面 -> 2面 -> hr面 -> **意向**)

****拼多多:** **搜索算法 (笔试 -> 1面 -> 2面 -> 3面 -> hr面 -> **意向**)

****京东:** **NLP算法 (1面 -> 2面 -> hr面 -> 3面 -> **意向** -> 4面 -> 5面 -> **等待ing**)

****快手:** **推荐算法 (1面 -> 2面 -> 3面 -> hr面 -> **意向**)

****华为:** **NLP算法 (笔试 -> 1面 -> 2面 -> 3面 -> **意向**)

****网易:** **NLP算法 (笔试 -> 1面 -> **2面挂**)

****虾皮:** **推荐算法 (**笔试挂**)

****米哈游:** **NLP算法 (**简历挂**)

陌雨将分别从“**简历、自我介绍、论文、实习、基础、手撕代码和反问**”这几个方面给面友们分享一下个人体会, 后六个刚好也是每场面试依次遇到的环节。每个环节占的时长大概是“自我介绍3min, 论文&实习30-40min (这两个不定, 有人只问论文, 有人只问实习, 有人两个都问, 合起来大概这么长时间), 基础5-15min, 手撕代码10-20min, 反问5-30min”。

1 简历

1.1 简历内容

遇到的面试官几乎只会focus下面两点，其他的一些东西得过什么奖学金啊，绩点多少，有无社交牛逼症，乐观开朗，精通windows啥的，感觉面试官都不太关注，可以选择性写一些丰富简历。本人面试的时候是几乎所有的火力都集中在第二点，我当时还专门有个other章节写了一些别的东西，想强调没有相应的项目，但是这些我也会，但从来没有面试官问过。所以简历可以做的有的放矢和稍微简洁一些，把**必要的个人信息 + 论文、比赛、项目和实习给陈述好**，再适当挑选一些其他比较有意思的丰富下简历。

- 家住哪/学校哪，愿不愿意到公司城市发展，意向和本岗位match吗，什么时候能毕业
- 有哪些论文、哪些比赛、哪些项目和哪些实习

如果n很大，1/n桶水的东西就别往简历上写了，会留大坑，容易偷鸡不成折把米。身边有同学刚入门某些东西就往简历上写，想添彩，面试的时候刚好就被面试官逮着了，非常减分，自然而然都是都被挂了。简历上的东西尽量吃到比较透，至少确保能和面试官来上几个回合。

另外，强迫症患者要注意别花过多时间在简历修改上，我秋招就三天两头改，怎么看怎么不顺眼，浪费了超级多时间。我简历当时在非上面两点的other信息上有及其离谱的错误，但投了一圈都没沉（除了米哈游）。后面反思这东西其实就是通过性考核，只要达标（80分），能帮你走到面试环节就行，没必要苛求细节完美（100分）。重点在面试表现，你精心在简历搞各种小细节，面试官可能压根就不care。另外，最好找有经验的师兄/姐帮忙check。

1.2 简历制作

同学几乎都用这网站，非常好用。不需要整得花里胡哨、五颜六色，黑白、突出重点即可。

1.3 简历投递

Tips-1 投的要够早，最好提前批开始的第一周就把简历甩出去。

多早算早。老话都说金9银10，但是这几年互联网招聘已经变成金8银9了，甚至一年比一年早，基本都在7月底和8月初就开始。所以一定要好好珍惜8月和9月，尽量不要拖拉，每天都安排利用起来，也随时保持好面试状态。

复习是永远复不完的。** **无论复习多久，一定会有知识盲区，要学会理性评估自己当前的准备情况，觉得差不多就尽早开干。面试官通常会从你身上出发，都是在聊你会什么，而不是你不会什么，所以不要老想着自己还有多少东西是不会的，畏手畏脚。只要你没菜到什么都不会，简历上有东西和面试官聊够半个多小时就可以甩简历了，后面就边实战边复习。

坑位有限，先到先得。 我的教训是shopee投晚了，我当时以为外企要全英文面试，想认真准备，拖了好几周才投。最终shopee早投的那波人表现比较common都可拿意向，后面投的不少top2或笔试全a或论文怪或bbater在笔试阶段都给挂了，甚至正式批国内直接“取消”算法岗。然后，我字节投的非常早，提前批第一周就开面。当时加面有好些个问题回答的很拉，自我感觉表现一般，却顺利通过了。所以一定要趁早，早期坑位多而且bar也会低一点。

提前批是复活甲，面试是无限火力。 提前批等于白送一次机会，结果不影响后面批次，不投白不投。多数大厂都不会严格限制次数，听到都是挂了隔几天又被别的部门捞起来，无限循环。其次，即使面试不通过也不要让负面情绪影响自己，不要因为一次不好的体验就aoe整家公司，个人觉得大厂更像是很多个用了同一个名字的小厂，不同部门和不同小组差距太大了（在我实习的同一个中心下，我们组leader人超好，氛围超级和谐，隔壁组的leader就天天排队训话，差别就是那么大），不可能次次都是奇葩，相信总会遇到好的。认识一位7篇paper的巨佬，一面被宇宙公司的奇葩面试官给弄了，他中间一度曾放弃这家公司，觉得很莫名其妙+面评被写死了，中间耽误好几周都没进展。最后想通了，第二趟轻松找回场子。

Tips-2 投哪些公司，面试难度、体验和效率如何，部门怎么选，有什么渠道

面试难度，个人的排序大概是字节 阿里 腾讯 百度 美团 快手 拼多多 京东 华为 网易。前三家的难点不太一样，字节是难到让你挂的明明白白，腾讯和阿里则是难的不太明白，通常是面的比较轻松但过不了。个人觉得华为和网易会相对简单一些，技术问题比较少，像聊天，有些轮次面试官讲的时间比我还多。剩下的难度没特别明显的差距，如果信心还不太足，一上来尽量不要刚太难的厂，容易受打击，尽可能先从中间这波厂开始。

面试体验，讲道理个人觉得大厂的面试体验整体都非常好，包括在脉脉上被喷成shit的字节和阿里（我字节一面和阿里一二面是我整个秋招最舒服的Top5）。我没遇到过传说中的kpi面，最坏只遇到过几个静态脸或者纯问一些不知道想考核啥的面试官，btw对面都让我通过了。我觉得这可能只是他们的面试style，和面评无关，所以面友们遇到这种情况也不要轻言放弃。我秋招制定了策略，如果三面往后遇到传说中让人忍受不了的业余面试官，就舔着苟过去，入职大概率见不着他，但一二面遇到，反问环节就主动说不够match，让释放流程。

面试效率，字节独一档，我一面到意向11天，结果反馈超级快，都是“私人专享”的HR实习生在半小时内通知。字节的每个部门差不多都这样，效率特别稳定，身边同学拿到的第一个意向十有八九都是字节。其次是百度，一面到OC一共4天（不包周末2天），百度的效率没字节稳定，

不同部门差距较大，但是百度允许多线程作战，不同部门、甚至一个部门的不同组可以同时发起面试流程，所以整体效率也比较高。剩下大厂没太大差别，都维持一周一到两面的频率，整体流程大概是20-30天不等，而且它们多数都还会设鱼塘，hr完要继续等一段时间才有结果。想快速拿保底offer的卷王们建议秋招初期从字节和百度下手，真的超级快。

业务部门**，**我的选择方式很粗暴，我印象中百度搜索很牛逼，就投大x；字节和阿里广告很牛逼，就投商xx广告和阿里xx；快手算是推荐业务支撑，就投社x推荐。其他也是类似的想法，都是尽量选印象中比较卷、核心和技术有一定沉淀的业务部门。不同人有不同需求，千人千选，核心和卷并不是唯一标准，面友们也可以参考一下下面这个 项目。

投递渠道，包括企业微信（腾讯）、招聘官网（快手、拼多多、网易、虾皮）、面试群JD（字节、京东、米哈游）、熟人递简历（华为、美团）、牛客网员工内推（阿里、百度）。招聘官网和面试群遇到的JD都太随机了，建议主动出击，尽可能让简历直达你的意向部门。

2. 自我介绍

我每次自我介绍都讲大概3分钟左右，甚至可以讲更长，不然1个小时的面试时间都不知道要怎么耗过去，只能这些有的没得多讲一些。这一部分感觉不是很重要，我每次都是一样的说法，还没被面试官challenge过。下面是我的模版，面友们也可以参考一下。

面试官老师，您好！我是xx学校xx专业的硕士研究生，大概xx年xx月毕业。本科就读于xx大学xx专业，在本科的时候学过xx，有过xx成果，最终在大四的时候顺利保送到xx大学xx专业，然后跟随xx老师从事NLP的研究快xx年了，我们实验室的研究方向是xx，研究生期间我发表了xx论文，做过xx项目，参加过xx比赛。此外，我平时还关注xx领域，对xx和xx也有一定的了解。我前段时间在xx公司的xx部门实习了xx久，部门主要负责xx公司的xx业务，我在实习的时候主要做了xx工作。我个人的情况大概是这样，老师，您有啥问题吗？

3. 论文

怎么发论文**。**论文比较吃自身科研sense和实验室科研氛围。最重要是实验室得有经验丰富的师兄带你。如果没有，建议多打比赛和多搞项目，尽可能让面试可以有些自己的东西讲。又或者多去企业Lab实习，企业Lab招实习生听说不一定需要paper，反而是更看重coding。Lab里面有大牛mentor+多数实验室都望尘莫及的计算资源，出paper概率要大很多。

论文的作用**。**首先只要论文数量达到阈值就可以走头部计划，据我了解某些头部计划面试难度和普通批次并不会有质变，无非就是有更高级别的leader来面你，区别更大的是论文数量

有没有达到阈值。其次，硕士生两篇CCF A一作 + 其他方面不太拉垮，今年绝对大概率乱杀除头部计划以外的所有招聘。最后，论文可以让你每场面试都拥有接近30min的“安全期”。

论文怎么讲。**除了pdd，其他公司都允许共享屏幕。为了方便演示和更稳定发挥，建议提前准备ppt。演示内容包括“任务定义(输入是，输出是，有无监督等) - 任务难点 - 相关工作 - motivation - 提出的方法(创新点，有啥实现难度) - 实验结果 - 后续改进”。**

4. 实习

实习选择。****我个人理解是有两种情况，第一种是选意向的核心部门，然后实习期间好好干，转正拿好offer；另一种是选一个还行的大厂养老部门镀金，每天快速干完mentor分配的任务，稍微没那么好好干和想办法多挤出一些时间准备秋招，再拿其他意向部门的好offer。

实习怎么讲。****和论文类似，建议做一个ppt展示，介绍一下你们组是干啥的，以及这几个月你做了些什么事，目的是什么，效果怎么样，后续会怎样改进。因为业务算法长时间都是和数据打交道，几个月的时间也很难做出很大的成果，如实写上去就ok。个人发现，做一些实际的数据处理带来的性能提升比包装一些乱七八糟的吹牛逼模型更能引发面试官的共鸣。

通过实习 = 秋招有优势。对于不少公司，你通过了它们的暑期实习面试，即使没接offer，秋招也是有优势的，可以少面的几轮的或者招聘还没正式开始就私下约你面试。我了解到的是阿里可以直通终面，美团可以少面1-2轮不等，腾讯和字节是招聘没开始就私约面试。建议卷王们暑期实习也多多折腾一下，既可以锻炼和积累经验，又对秋招有好处。

暑期实习 = 半个offer。****公司通常都更愿意把坑留给组内的实习生，所以暑期实习除了丰富简历和增加秋招竞争力以外，还等于你收获半个offer。个人的体会是只要不犯大错，不删库跑路，mentor交代的東西也有踏踏实实的在做，通常有hc都会先留给你，大厂对实习生的预期没有我们想象中那么高。身边小伙伴暂时没听到转正被刷的，然后开奖几乎也都给sp+。

导师不放实习，怎么办。**建议先尽力和导师沟通，让导师了解实习对于秋招的重要性。如果硕士期间你一直勤勤恳恳、踏踏实实和已经顺利达到毕业条件，导师多数情况也会理解你。实在不行就只能给你们一个不那么好的建议，可以在尽可能保证自己能顺利毕业的条件下**开溜。首先是实习不需要全勤，通常一周只需要三天，有些甚至可以远程。然后大厂的研究型或者养老部门的算法岗实习生任务比较轻，不用担心挤压掉实验室的“科研”时间导致没法交差。对于实验室不需要打卡的同学，时间可以卡极限一些，想办法一周挤出三天去搬砖，市内有急事来回顶多1小时。对于实验室要天天打卡的同学，就只能耐心点找远程实习了。

5. 基础

5.1 机器/深度学习

ML本人的基础相对薄弱一些，因为我开始学NLP的时候，已经是DL年代了，当时没啥人搞传统的机器学习方法，很多东西我都是匆匆茫茫快速翻了一遍，没深入的推导过。我这次面试的时候，也很少会被问到传统的机器学习方法，印象中只有字节和阿里的广告算法岗会问一些，每次都是问我LR相关的，搜+NLP岗位的面试几乎是没问过。我的感受是，只需要将李航老师的《统计学习方法》认认真真翻完一遍，再额外补充一些面经常见问题就可以应对秋招了。另外，我还看了推荐度挺高的葫芦书《百面机器学习》，一本QA形式的书，里面的问题都非常的经典，十分适合秋招复习。

DL的话，感觉没什么好讲的，只要你平时有认认真真搞过NLP相关的任务，这一部分肯定是难不倒的，都是问一些平时能够接触到的东西。我在复习的时候是看了邱锡朋老师的《神经网络与深度学习》，整个秋招体验下来，面试已经是非常够用了。

下面给大家罗列一些印象中我或小伙伴秋招会被问到的问题。

- SVM（原理）
- LR（为啥用sigmoid函数，交叉熵推导，MAE和MLP，反向传播，归一化，正则化）
- 降维算法（SVD和PCA）
- K-Means（手撕代码实现）
- 决策树（各种生成和剪枝方法）
- 集成学习（随机森林、XGBoost、AdaBoost、GBDT）
- EM算法（原理）
- 过拟合（正则化、增加训练数据、数据增强、标签平滑、BatchNorm、Early-Stop、交叉验证、Dropout、Pre-trained、引入先验知识）
- 方差偏差分解（解释什么是方差什么是偏差，公式推导）
- 正则化（L1和L2的会有啥现象、解释原因、分别代表什么先验，bias要不要正则）
- 初始化（不同网络初始化有啥区别，神经网络隐层可以全部初始化为0吗）

- 激活函数（优缺点，sigmoid、tanh、relu、gelu）
- 损失函数（用过哪些损失函数，为啥分类不用MSE）
- 信息论（信息熵、条件熵、联合熵、相对熵、互信息的概念，交叉熵和KL散度区别）
- 归一化（为什么要做归一化，各种归一化的区别和优缺点，NLP为啥不用BatchNorm）
- 梯度消失（残差、门控、sigmoid换relu、归一化）
- 梯度爆炸（截断）
- 优化器（原理和演进过程，SGD、AdaGrad、RMSprop、AdaDelta、Adam、AdamW）
- 显存爆炸（重计算、梯度累加、混合精度训练、Adam换成SGD、多用inplace）
- 学习率（衰减、warmup、自适应、平时自己使用的时候对lr有什么调整心得吗）
- 样本不均衡（降/过采样和带权重的loss）
- 数据预处理（离散特征和连续特征）
- 评价指标（Acc、Precision、Recall、F1、ROC、AUC、代码实现AUC）
- 神经网络（优缺点、演进和公式推导，lstm、cnn、transformer）
- OOV咋办

5.2 NLP

NLP包括两板块，一个是通用的基础（所有面试官都可能问）和你个人研究方向的基础。

个人研究方向基础没啥好说的，比如你做生成，面试官就很可能让你手写beam search；做序列标注的就可能让你推viterbi解码、HMM和CRF区别之类的；做文本匹配可能就问你双塔和concat模型、CLS塌缩和对比学习之类的。这个需要大家根据自身的情况选择性去复习。

通用NLP基础我和小伙伴暂时能想起来的主要包括下面这些了。

- **经典模型的原理和手撸代码**

- **文本关键词抽取** (textrank和tf-idf)
- **词向量模型** (NNLM、word2vec和glove)
- 相对于NNLM, word2vec的改进有哪些
- 哈夫曼树的构建方法, 在NLP有啥应用
- hierarchical softmax和负采样的原理和复杂度
- 负采样的具体实现方法
- word2vec和glove的区别
- 怎样评估词向量的质量
- 选出当前query和100万个key词向量相似度的TopK, 复杂度尽可能低 (faiss)
- **预训练发展**
- word2vec - glove - ELMo - GPT - BERT - others的演进, 每个模型分别解决什么问题
- **Transformer的细节**
- **为什么要用多头** (多个空间学习多种pattern、降低注意力学习的风险)
- **为什么Q和K的映射矩阵不相同** (关系对称、容易得到单位矩阵)
- **为什么注意力权重要除 $\sqrt{d_k}$** (防止梯度消失)
- **为什么用乘性注意力不用加性注意力** (乘性计算量更小?)
- **为什么要有FFN模块** (增加模型的非线性能力)
- **Transformer和BERT的位置编码有啥区别** (三角函数式和可学习向量)
- **残差结构及意义** (防止梯度消失和网络退化)
- **哪个block中更耗时, 哪个更占显存** (序列短的时候FFN耗时, 长的时候MHA耗时; FFN更占显存)

- **transformer的LayerNorm有哪些** (post-norm和pre-norm)
- **如何对pad进行mask** (将pad的attention_score加上-np.inf, 过softmax后会变0)
- **怎样解决曝光偏差** (训练时以一定的概率用上一时刻的输出、NAR、占位符生成、基于负梯度构建对抗样本)
- **transformer加速** (NAR, 知识蒸馏、剪枝、动态退出、稀疏注意力、线性注意力等)
- **attention瓶颈** (low rank, talking-head等)
- **BERT的细节**
- **BERT怎么做分词**(Basic Tokenizer和WordPiece Tokenizer)
- **WordPiece和BPE有啥区别**
- **BERT的两个预训练任务分别有啥优缺点, 后续有啥改进工作**
- **为什么BERT要用自己学习的位置编码** (在训练充分的情况下, 可学习的比三角函数式的表示能力要更强)
- **BERT的位置编码有啥缺点, 还有哪些位置编码** (绝对位置并不能很好的表示距离和方向, 后面有相对位置编码、复数位置编码以及加入树形的位置编码等, 参考t5, transformer-xl, t5, deberta, tupe和roformer等等)
- **BERT的FFN为啥要用GeLU激活函数** (非饱和区大同时非线性也更强?)
- **BERT-Related-Works**
- **Prompt**
- ****HuggingFace** (**有些面试官会直接让你看源代码解释某一些模型核心代码块的具体作用, 建议把BertModel给过一遍)

5.3 搜广推

本人没有搜广推经验，但是经过一些师兄和脉脉了解到搜广推是当前非常好的算法赛道。因此，秋招刻意选几家大厂的看家业务部门尝试了一下，我投递的仍然是NLP算法工程师，但面了后发现这些部门里面多数岗位几乎都不是纯纯的NLP，会用到但不局限于NLP，更多的是排序和预估相关的工作。秋招刚开始的时候非常担心没优势，抱着无限火力+复活甲的心态在面，没想到最后都很顺利挺过去，也挺舒服。

大家应该比较关心“校招生没有搜广推相关经验，面试这些岗位会特别吃亏吗”，这个问题我问过很多面试官，得到的回复大概都是“拥有搜广推经验的校招生是非常少的，他们也很想找，但是很难有人在学校就专门研究，不强求，NLP和这些岗位要求的能力有一定的重叠性和可迁移性，所以也非常欢迎厉害的NLPer投递这些岗位”。

关于搜广推的面经，我大概只能一段话总结：**NLPer面试这些岗位几乎所有的时间都在聊你的论文、实习、项目和一些通用的DL/ML知识+一丢丢业务场景题，你做的东西被challenge的问题相对会少很多，面试翻车的概率感觉比NLP岗更低，面友们可以放心投。**另外**，**如果你平时还学习过推荐和广告系统，那么优势就非常明显了，顺便附上一位og写的文章。

5.4 通用CS

计算机网络、操作系统、计算机组成原理、数据库和设计模式，这些我从来没被问过。反而经常被问数据结构和一些常用的搬砖工具，但是非常浅，这部分适当瞄一眼就ok。

- 数组、链表、栈、队列、堆
- list, array, vector, unordered_set, set, dict等数据结构的底层怎么实现的
- 二叉树、平衡树、搜索树和红黑树，各种树的演变
- python装饰器、深拷贝浅拷贝、垃圾回收机制
- shell、git、hadoop、spark

还有些面试官喜欢问一些偏概率和数学的东西，下面是一些常见的概率题，最好准备一下。

- 用rand7构造rand10
- 山羊汽车问题
- 木棍截成三段，形成三角形的概率

- 抛一个6面的骰子，连续抛直到6为止，问期望的抛的次数是多少
- 给定概率不均匀的0和1随机生成器，怎样等概率随机生成0和1
- 三角形里面随机选3个点，构成锐角三角形的概率
- 两个人轮流抛硬币，规定第一个抛出正面的人可以吃苹果，求先抛的人吃苹果的概率
- 一副扑克牌，分成三堆，大小王出现在同一份的概率
- 在半径为1的圆内随机等概率采样一个点

6. 手撕代码

6.1 刷题网站

- **LeetBook**: leetcode-cn.com/leetbook/
- **剑指Offer**: [leetcode-cn.com/problem-lis...](https://leetcode-cn.com/problem-list/)
- **Hot-100**: [leetcode-cn.com/problem-lis...](https://leetcode-cn.com/problem-list/)
- **Tag**: [leetcode-cn.com/problemset/...](https://leetcode-cn.com/problemset/)
- **Labuladong**: [labuladong 的算法小抄](#) :: labuladong的算法小抄
- **LeetCode 101**: github.com/changgyhub/...
- **CodeTop**: codetop.cc/home

6.2 刷题经历

面试得分公式大概是“总分 = Sigmoid(手撕代码) * 面试表现分”，手撕代码更像面试表现分前面的门控系数，撕不出来大概率凉，所以要重视一下。**我没有参加过权威的coding比赛，也没有系统刷过oj，是从基础比较薄弱的菜 状态下复习的，平时刷题会比较反胃，但是面试多数时候都觉得在牛刀杀鸡。如果你学习能力和自制力都比较强，准备2个月差不多可以搞定。另外，coding这东西不是一朝一夕可以锻炼出来的，除了秋招要准备外，平时也应该多注意练习，最好每周参加一下周赛，每天也抽空写一下每日一题，下面是我的刷题经历。**

- ****刷LeetBook入门。****从简单的数组章节开始，逐步加大难度，往难的动态规划之类恶心人的算法上刷。LeetBook把题目都给归类好了，刷题前也会介绍一些基本概念和常用的套路模版，方便没系统学习过的面友们入门和巩固。刷完LeetBook你的LeetCode榜大概就有100多道了，这时对各类算法问题都会有一个清晰的概念。有经验的LeetCodeer可以直接跳过这一步，或者选择性刷某些章节。
- ****重点刷剑指Offer和Hot-100。****这上面的题面试出现的概率非常高，而且难度也适中，不恶心人。最好重复刷，确保这上面的题目出现你是能秒的。刷完这部分就可以开始甩简历，难度一般都不会超出这个范围。
- ****选择性刷Tag拓展知识面。****例如蓄水池抽样、拒绝采样、数组差分、贪心、拓扑排序和并查集等等。做到心里有数就行，一般考察这类非剑指Offer和Hot-100的题目会非常baseline。
- **CodeTop刷意向部门岗位的面试原题。**通常面试都是约下午或晚上，当天你会特别紧张，啥也不想干，这时候可以刷一下这个爆料网站的题目缓解一下，安慰自己手撕代码准备到万无一失了，也可以保持下手感（虽然我每次面试遇到的都不是爆料的，可能投的太早了，好些题目是我爆料上去的，[裂开]）
- **Labuladong和LeetCode 101**是我秋招看的比较多的算法刷题总结，面友们可以瞅一下。

6.3 刷题心得

面试遇到没见过的题尽可能先往暴力的方向想，先确保你自己能做出来，不要嫌弃想法过于傻逼和繁琐就不说出来，没准就是这样做的，写出来一种方案总比啥都没写好。而且你想出了暴力写法，面试官大概率会提醒你，让你继续优化。甚至很多原创题目，你能写出暴力的解法，面试官就通过了。遇到不会的题也不用太紧张，一般手撕代码环节的题在代码实现上都不会特别复杂，就是思路可能会绕，尽可能静下心来思考分析，我有好几次上来一看到题目就觉得要翻车，但最后都成功做出来了，要慢慢学会享受现场解题的刺激。

面试遇到见过的题目，不要上来直接无脑套最优解。这样不会加分，只会让面试官觉得你刚做过原题，然后反手让你证明算法的正确性或者再给你整一道难难的。最好走完一套流程，尽可能让面试官觉得你现场是在思考的，先假装在纸上写写画画，然后有思路了和面试官分析这类问题（不要说题目）应该怎样解决，通常用啥算法或模版优化，最后再开始写代码。

语言可以直接用Python，没有任何的限制，但是如果你本身非常擅长c++，建议还是用c++刷，c++会有一定的印象分。没时间复习c++也没关系，对面试结果没影响，好些次面试官都问我能

不能写c++，我都反向画饼地说“本科肯定是用过的，但是最近两年一直在用python，上一次用c++还是保研机试，现在写不了了，不过以后工作有需要捡起来也不会很难”。

如果仅仅只是想上岸秋招（无所谓乱杀和收不收割机），我觉得有剑指Offer+Hot100就已经非常足够。剑指Offer+Hot100以外的很多题目都非常长尾，你整个秋招都不一定能遇到一次，不停地刷下去，性价比超低。如果搞完剑指Offer+Hot100后还有充足时间，建议你再复习一遍，因为真的超级容易忘记和面试超级大概率是这上面的原题。

手撕代码环节，我个人90%以上都只用写核心代码 + 面试官肉眼debug和人脑运行。但是笔试环节是都需要自己构建控制台输入，平常要注意练习一下，不要小看这小小的输入环节，可以去牛客浏览下有多少人笔试是死在输入边界条件处理不对。

笔试是一个容易翻车的环节，笔试和面试的题目差异非常大，并且笔试会相对偏门和琐碎一些，很容易出边界错误导致卡case，面友们可以到牛客网找一些企业真题体验一下。应对笔试环节，像我这等菜的建议是跳过它。首先是多数公司（字节，百度，京东和快手等）的提前批是无笔试的，大家只要抓紧时间，简历尽早投就ok；其次还有一些公司（腾讯，阿里，美团）是可以不看笔试的，直接找熟人递简历，看上你就会立马发起面试，整个流程也会顺畅很多。最后还有些公司的实习笔试可以顺延（华为），只要你实习通过笔试，秋招不用重新弄，建议趁实习有空的时候也多折腾一下，既能锻炼又有好处。

手撕代码不一定每轮都有，有些轮次是不用写题的，没有也没啥事，不代表面试官放弃你，我好多轮都没有。其次，一般手撕代码是在你面试流程的最后一环，如果你前面的项目和基础part聊的特别开心，题目相应也会简单一大截。为了手撕代码更顺利，前面要好好使劲。

6.4 面试题目

- **腾讯**

- 无代码考察

- **阿里**

- 实现堆

- 蓄水池抽样（数组抽1个、数据流抽1个、数据流抽k个、分布式数据流抽1个）

- **字节**

- 数组找Top-K（实现堆 + 快速选择）

- 二叉树的最大路径
- 代码实现AUC
- 给定两种沙漏，计时分别是3分钟和5分钟的，问两种沙漏能表示7分钟吗；能表示任意分钟吗；求表示任意分钟需要的最少沙漏数量
- 识别Python脚本的每行缩进是否正确
- **美团**
- 旋转数组找最小值（数字有重复）
- 用最少数量的箭引爆气球
- 旋转数组找target（数字有重复）
- **百度**
- 给定序列的入进栈顺序，求出栈有几种可能
- 括号生成（只有一种括号和有三种括号）
- 链表的归并排序
- **京东**
- 数组找Top-K（实现堆 + 快速选择）
- 旋转数组找target（无重复数字）
- 前缀树
- 链表相交节点的个数（2个链表、N个链表和当链表有环）
- 二叉树的层序遍历
- **华为**
- 最长有效括号

- 三数之和
- **拼多多**
- 一个讲不出来的特殊排序
- 中文分词的前向算法
- 快速幂
- **快手**
- 二叉树的右视图（非层序遍历写法）
- 给定一个升序数组，返回所有可能的二叉搜索树
- **网易：**
- 无代码考察

7. 反问

反问环节通常都会有15-20分钟左右，我基本都问下面几个问题，面试官也都很乐于解答。大伙有啥特别想知道的都可以大大方方的问，但是尽可能要注意下用词，礼貌一些。有些交叉面和加面没有反问环节，因为面试官不是你面试小组的人，写完题就结束面试。

- 小组大概是做什么的，业务方向、具体用到的技术栈和目前遇到的挑战和瓶颈
- 小组成立多久、规模多少人、有哪些base、在公司定位以及资源倾斜、未来几年发展规划
- 怎样培养新人，新人过去大概的上手流程是怎样的，会有多长的适应期
- 小组内的工作强度怎样，每天大概几点上班和几点下班，周末加班吗
- x老师，针对刚刚的面试表现，您觉得我还有哪些方面要再加强一下
- x老师，我最近比较有空，如果您觉得我还ok的话，可以尽快帮我排一下后面的流程

8. 其他

-
- 知识面宽更有利于面试，很少面试官会只抓住一个点狂问，除非你主动说你精通某个东西，一般都是问的相对比较浅和比较宽，而且知识面宽也有助于业务场景题和临场发挥。
 - 大厂实习比顶会论文要更靠谱。两者其实都可以帮你上岸，但明显大厂实习要更稳妥。顶会这个东西太玄学了，一年也投不了几次，投出去也要等好几个月，然后审稿又巨吃人品。相比之下，大厂实习的机会要多多了。如果你两个都没，建议优先考虑大厂实习。
 - NLP算法还没到算法岗劝退帖子那么邪乎，动不动就bar要高到会手写xlnet、n篇顶会和n段实习，又或者算法岗已经不招人啥的。我个人更倾向另一种观点“**不少部门算法岗其实都缺人，但是又招不到合适的人，有些连hc都发不出去**”。至于NLP算法岗的bar，我感觉我在众多NLPer里应该还勉强可以算得上中等水平，能代表大多数人，可以参考一下。
 - 避免尴尬，要预先准备好台词去“舔”投的每家公司。因为hr一定会问你为什么选择xx公司。每家公司也确实都有让人选择的理由，可以从企业文化、平台规模、业务方向和业务增长空间、技术口碑、公司所在城市以及薪资待遇等等角度去分析。
 - 本篇文章是陌雨本次秋招的所见所闻和心得体会，个人能力有限，肯定有局限和理解不到位的地方。如果有表述不好或错误的观点，求各位大佬轻喷，我们可以在评论区或者私信多多交流。最后，今年还在坚持算法岗的面友们是真的挺有勇气和挺不容易的，也该到收获的季节了，陌雨在这里祝愿大伙都可以顺利收割各大意向公司的offer~