1. 复习前的准备
2. 了解自己的职业方向

每个职位都有其对应的知识体系，所以强烈建议提前确定未来的就业方向，包括且不限于

研发（前端、后端、客户端）

算法（算法、机器学习）

数据（数据挖掘、大数据）

运维

测试

硬件

产品经理

1. 编程语言的选择

不同的语言同样方向不同，要根据自己的强项、兴趣、就业薪资选择几种（不要太多）进行重点复习

编程语言有很多的地方思想是通用的，有些高级特性是触及旁通的，希望不要过于纠结用什么语言

当敲定职业方向，主修语言之后，就可以开始复习了。

附1：如果哪个语言都不会的话，此处推荐学习Java

附2：

Java：服务端开发、数据挖掘、Android开发（互联网公司就没有不用的）

C++：服务端开发、客户端开发（Qt）、游戏开发（虚幻4）、编译器开发（腾讯很多C++）

C：操作系统、嵌入式（职位少，搞这个的更少）

Python：机器学习、服务端开发（数量少）（AI应该是今年最火的词了）

PHP：后端开发（百度最近的偏爱，坑少人也少）

C#：服务端开发、游戏开发（Unity）、客户端开发（WPF、UWP、ASP.NET）（虽然我是个软粉，且看了不少C#的书，但互联网公司基本没有用的，不建议学习）

Go：服务端开发（新兴的后端语言，在高并发、多线程开发下有很不错的开发效率，今日头条和Bilibili后台几乎都是用Go写的）

JavaScript等一系列前端语言（我不了解）：今年前端需求很猛，各大公司就没有不要的

最后，需要了解的是，任何编程语言学好了之后，找工作的机会都是很大的。

前言：我的复习方向基本以C++后端开发为主，所以复习的内容会比较偏后端

1. 需要复习的内容

编程语言：任何语言其实都是在实现基础的语言特性（变量定义、函数、递归、面向对象、垃圾回收等等），掌握语言特性。

以C++为例

至少需要了解：变量所占的字节数量、内存对齐、STL容器相关（vector、list、map）、指针和引用的区别、智能指针……

进阶：STL源码的实现、模板编程（模板元编程）、一些C++开源库、C++ 11，C++ 14特性

推荐书：

《C++ Primer》

C++入门不二之选。

《Effective C++》

面试的时候经常会问到里面的内容

进阶：

《More Effective C++》

《STL 源码剖析》

《Effective Modern C++》

操作系统：这里特指一下Linux操作系统，通常情况下，各大服务器使用的都是Linux操作系统，对于Linux操作系统的系统调用、进程线程、死锁、信号量等等就是需要了解的目标

进阶：银行家算法的实现、生产者消费者的实现、多路IO复用（select、poll、epoll的不同，实现）、进程间通信（IPC）

推荐书：

《UNIX高级编程》

数据库：大多数时候，不会问 太多数据库的内容，通常以MySQL数据库居多，了解到范式规约，主键外键唯一键等等定义，多线程下的锁的机制、索引

推荐书：

数据库是我目前相对薄弱的地方……

计算机网络：万年常考的TCP/IP三次握手四次挥手的知识、OSI 7层、Linux下Socket编程所需要用到的系统调用

进阶：TCP/IP握手挥手时候的状态，HTTP/HTTPS，加密算法（非对称加密、对称加密）

推荐书：

《计算机网络 自定向下方法》

《TCP/IP详解》

《UNIX 网络编程 卷1:》

算法与数据结构：队列、栈、快排、归并排序、堆、堆排序、树、二叉树、二叉排序树、红黑树、哈希，字符串（KMP）、动态规划：跳楼梯（或者变种的跳河挑砖头），背包问题

进阶：图（很少问）

推荐书

《算法》

附3：

书单总结

《C++ Primer》

《Effective C++》

《More Effective C++》

《STL 源码剖析》

《Effective Modern C++》

《UNIX高级编程》

《计算机网络 自定向下方法》

《TCP/IP详解》

《UNIX 网络编程 卷1:》

《算法》

《Head First设计模式》

《剑指Offer》

可以说非常针对面试的一本书，更有意思的是很多面试官也看这本书……就面试的经验来看，确实很多这里的原题

<https://leetcode.com/>

刷题刷题刷题

进阶：

《深入理解计算机系统》

多看看组成原理，码农迈向程序员

《算法导论》

和红皮《算法》可以互补

《离散数学及其应用》

编码重要，逻辑同样很重要

1. 关于校招

（1）我对于校招的心态

秋招下来，我的第一反应就是：秋招=985 211的招聘会。

当然，这并不准确。不过，要了解的就是，重点学校的学生确实有优势，这并不仅仅是学历上的优势，也有知识上的优势，所谓站得高看得远，平台不同，其能力确实通常会更强一些。

当然，落到自己，还是要好好学习，互联网公司，可以说还是相当不怎么看学历的了。

尽量提前复习，有些同学从3月就开始实习+复习了，我相信这样的人最后一定会拿到一个不错的心仪offer

（2）流程

以我参加的秋招为例

7-8月份陆陆续续就该有内推了，有些公司的内推是可以免简历筛选的，内推的机会要好好把握。

8月份开始，校招网申也陆陆续续开始了，校招的流程通常是

网申->笔试（宣讲会后现场笔试、网上在线笔试）->面试

在这里推荐牛客网，刷题和讨论都非常不错。

（3）面试

面试的时候，回答出面试官的问题。自己也要传递给面试官你的优势，包括不限于：对某一项技术的出众了解、对未来学习的充足规划、与公司文化的完美相配。

面试的时候不要背书，多阐述在学习（使用）这项技术的时候的思考。这不仅能够得到一个好的印象，还可以引导面试官往你擅长的地方提问。比如万年常问的死锁问题，背出4大必要条件是不够好的，可以结合实际在编程中遇到过的死锁场景（或者是可能出现死锁的场景）和面试官谈谈你是怎么预防的（解决的）。当询问map 和 unorder\_map的区别的时候，结合项目（或者你的理解）谈谈什么情况下用map什么情况下用unorder\_map，如果了解这两的内部实现（比如map的红黑树），可以说更好了。

遇到不会的问题，直接说不会然后跳入下一问会显得很生硬，可以和面试官阐述你的思路，你想要用什么方法解决这个问题（或者想的这个方法有什么问题），你想到的哪一点卡住了，因为什么卡住了，与面试官进行讨论，即使最后没过，也能够得到一些知识。

（4）心态

心态很重要，千万不要崩……

（5）我的秋招感受

Java需求量很大，待遇也挺不错的

算法岗今年新宠，薪资很好，本科难

研究生很多

自己还需要不断地学习

Android/iOS开发退热，岗位需求少了不少，但投的人少的更多

.NET开发很少，工控领域还有Unity开发还是有一席之地

1. 简历
2. 诚实，干过什么就写什么
3. 突出亮点（项目、实习经历、GPA）
4. 尽量1张纸
5. 排版……

最后

这几个月复习总结的东西

<https://github.com/sunkaiiii/study_computer_science>

附：甩出我复习时爱听的音乐

http://music.163.com/#/album?id=1921574