作者：wait0804  
链接：<https://www.nowcoder.com/discuss/193598>  
来源：牛客网

Linux C/C 从零基础到入门、到熟悉的一些路线，对于想走这个方向的同学，可以认真看看，分享到这里来，是因为牛客网对我的帮助很大，当初找工作的时候，天天刷牛客，特此回来报答一波，真的感激牛客网，提供了这么一个好的平台。

我将大致按照以下的目录，分享自己一路走来的一些 C 经历。

**一、秋招自己的一些情况**  
**二、Linux C/C 方向的一些思考**  
**三、计算机基础知识的梳理**  
**四、C 方向的深入学习路线**  
**五、项目 亮点 面试的一些思考**  
**六、更多的个人分享**

在校期间，我一直走的都是 Linux C/C 学习路线，历经暑期实习、秋招决战、校招期间投的大部分岗位都是 Linux C/C 后台方向，自己对于这个方向的理解、以及学习路线有一些感受与心得，想把学习路线、以及学习编程的方法分享给各位，对于正在走 Linux C/C 路上的同学能有所帮助。

**一、秋招自己的一些情况**

就读于双非院校、非科班本科学生，专业是偏硬件的（学校课程只有一门 C 语言与编程相关，其余的课程跟编程一点点的关系都没有），在校期间也没拿过什么大的奖项，计算机基础为零，大二上学期接触 C 语言，大二一年学了 C 语言和数据结构，大三在学校的实验室开始学习计算机基础和学习 C 方向，一路自学编程，始终相信自己，也是拿到了大厂 offer。

**秋招 offer 情况：**  
**腾讯 后台开发工程师（实习转正）**  
**百度 手百 feeds 流，软件研发工程师**  
**美团 后台开发 C 方向面试的**  
**作业帮 后台开发**  
**好未来 后台开发**  
**迅雷 后台开发**  
**。。。**

在暑期实习过腾讯，大四上学期实习百度，校招拿了十几个 Linux C/C 方向的 offer，最终的选择也很玄学，也去了自己想去的公司，从事自己喜欢的岗位。

**二、Linux C/C 方向的一些思考**

**1、**  
对于方向的抉择，很多人都选择了 Java 或者 Python 方向，Java 或者 Python 方向确实学的人多，岗位需求很多，入门能比 Linux C/C 方向容易一些，但是往深了学，各有各的难度，先敲定了方向，在深入学习。

Linux C/C 方向，国内真正掌握的人，其实是极少数，这个方向的学习人数也是相对比较少的，入门一般首选 C 语言或者 Python，C、C 在编程排行榜也是经久不衰的，排名靠前，如果喜欢，有兴趣，走这个方向是可以的。

**Linux C/C 方向，国内很多大公司都有招聘这个方向的岗位，BAT 都有，尤其腾讯，底层架构都是 C 写的，百度核心搜索很多模块也是 C 写的，阿里云也在招聘 C 方向的工程师，这个方向，学的不错的，根本不用担心找工作的问题，学的一般的，其实就比较难找了。**

**2、**  
Linux C/C 的效率是比 Java、Python 快一些的，更偏向于底层，能直接操作内存，对于编程思维、逻辑能力是有帮助的；其实学习这个方向，是有一点枯燥的，不像 Python 爬虫、Java、前端能做出一些有趣的东西，小程序开发出一些产品，都能很快用于实战，获得编程的乐趣，进一步逼着自己学习，C 方向没有那么多有趣的东西可以做，在很长的一段时间内，要靠自己的兴趣去坚持学习。

我在大三的时候，用 C 写过五子棋、内存监控工具、压缩工具、以及实现一些比较复杂的数据结构，网络编程的并发模型，这些项目，当你学习 Linux C/C 方向到一定程度的时候，你会找到这个方向的乐趣，跟其他方向完全不一样的乐趣，不过，前期真的很难，有一段时间极其痛苦，是要经过很长一段时间的学习，才能有一定的效果，如果对这个方向，真的没有啥兴趣，最好选择走前端、Java、Python 等是最好的选择。

Linux C/C 方向，入门稍微难一点，一旦选择了要走 C 方向的，在心态上面，要做好准备，一定不能有畏难心里，做好接触底层、复杂逻辑的分析、以及具备面向过程 面向对象的思想；就是退一步讲，我学习了 Linux C/C 方向，具备了编程思维和熟悉了 C 语言，以后去公司上班了，也能很快的上手 Java、Python 等方向（我说的上手指的是语言基础很快的学习，要想深入，还得花大把的时间）。

**3、**  
**按照目前身边人找工作的情况，普遍来说，Java、Python、前端、数据分析、测试方向都要比 C 好找一些，更容易找到工作，虽然我是走的 Linux C/C 方向，但是我是真的不建议对这个方向，没有极其浓厚兴趣的也走这个方向，我一路走过来，真的很痛苦，在校期间都想过放弃这个方向，学个 Java 可能会好一些（身边学习 Java 的人还是挺多的），每当放弃的时候，想想都学了这么长时间了，再坚持坚持，就这样一路走过来了，我觉得自己真的挺幸运的，有时候，真的不一定非要头铁 C 方向，要看整个市场行情，顺势而为，选择好方向，在深入！**

**选择走 C 方向的，一定要走 Linux C/C ，面向网络编程，学会 Linux 至关重要！**

**大厂特别喜欢考察计算机基础知识，对于基础掌握的如何，在很大程度上将决定你是否能进入大厂，一定要重视基础知识的学习；秋招准备：基础 算法 项目，加上自己的方向（C 、Java、Python、前端等），其中每个环节都得准备到位，才能最大程度上拿到大厂 offer，意识很重要，有了意识，在看具体的学习路线。**

**三、计算机基础知识的梳理**

**计算机基础：数据结构 算法、操作系统 计算机网络、数据库 Linux、C 语言**

**工作必备工具：Linux Git Sql**

**不管你是学习什么方向的，Java、Python、前端、测试、运维等等，这些基础都不能忽视，有时间的话，一定要打扎实了，有了扎实的功底，在这行才能走的更远更快！**

**数据结构：**对于常用的数据结构，链表、栈、队列、矩阵、树(BinTree、BST、AVL、RBtree、B 、B-)、跳表、图等，学习从定义--->原理--->实现--->应用，都要有不同程度的掌握，前面的数据结构一定要非常的熟练，对于树、图复杂的数据结构，最起码的掌握定义、原理、部分实现，我就记得：我在面试的时候，手写过红黑树的旋转算法，对于图的一些最短路径算法也是当面手写过，头条的面试基本上就是全程算法，难度不低（我个人博客上面有所有的数据结构的整理）。

书籍：  
<大话数据结构>，适合入门学习  
<数据结构>，清华大学严蔚敏写的，作为进阶和深入  
视频：<https://pan.baidu.com/s/1qQrHTdkvxMLSGv7G4nHWBw> 密码:4o9p

**算法：**一定要保证掌握基础算法 常见算法，十大排序算法（冒泡、插入、选择、快排、希尔、堆排、归并、桶排、基数、计数）是最简单的，也是一定一定要熟练掌握的，另外还有字符串常见算法、数组常见算法、递归算法等等，对于算法，从易到难，先把我列的一些算法掌握了，在去刷一些题，会有一些成就感，才能有自信，也要早早开始，一直刷一些算法题，保持住手感，及时的做好归纳整理、总结反思，视频面试就是编译器上面直接敲代码，现场面试，大部分情况下都是手写代码，写一些核心的算法逻辑，要特别注意判断边界情况，与面试官边讨论边写，《剑指 offer》至少刷 2-3 遍，在刷刷 leetcode 上面 easy、medium 类型的题目就好了（我 github 上面有自己整理的常见算法题目）。

书籍：  
<啊哈算法>，适合入门学习  
<编程之美>、<算法导论>，进阶提升必看  
对于算法导论，虽然比较难，数学公式很多，但是有视频的，有些学到了，真的不亏，可以学一部分的  
<剑指 offer>、leetcode 找实习/工作前，刷题就好。  
视频：左神的算法视频推荐看看，讲的很清楚。  
算法导论的视频：<https://pan.baidu.com/s/1Bm34-92TwN5TbMOXfPkY6g>  
密码:1i38

**操作系统：**很常问的一些技术点，堆和栈、内存分区、虚拟内存 物理内存、进程 线程 协程、死锁、分片机制、五大组件、中断和系统调用 、同步和异步等等问题，对于操作系统的理解，对于这些基础的计算机知识的掌握是必须深入学习，要花很大的功夫去理解清楚这些，工作中，对于真实线上系统的稳定性、对于底层技术的理解是有帮助的，操作系统是面试中常见问题之一。

书籍：  
<现代操作系统>、<深入理解计算机系统> 是比较全面，写的比较好的书，极力推荐。  
视频：  
<https://pan.baidu.com/s/1XiKe5e0UoI2Fp3Amn7aCCA>  
密码:y230

**计算机网络：**OSI 七层模型和 TCP/IP 四层体系结构，TCP 三次握手与四次挥手、常见的网络协议（网桥、ARP、IP、ICMP、TCP、UDP、DNS、DHCP）、TCP 粘包、流量控制 拥塞控制、数字签名原理、http https，http 的状态码，https 的安全机制，网络安全、输入 [www.baidu.com](http://www.baidu.com) 背后发生了什么等等问题，其实工作中，有时就会出现，网络抖动、网络延迟，网络拥塞的情况，此时就需要具备一定的网络知识，及时的解决问题，计算机网络是面试中常见问题之一。

书籍：  
<计算机网络>（谢希仁第 7 版）；  
<TCP/IP>（卷一）  
视频：  
<https://pan.baidu.com/s/1xRvWUlWzQ9c67XTo0Lkg-g>  
密码:c85g

**数据库：**数据库作为存储数据的地方，其不同的数据结构、与应用的场景不一样，采取不同的数据库，对于 mysql 基本操作、写 sql 的优化、表的设计、索引优化、如何更快的查询、底层数据结构的设计原理等等问题，常见的设计模式、主要是学习设计模式的思想、单例模式是必须写代码实现的，其他设计模式理解思想，作为了解内容，后台开发工程师与 mysql 打交道挺多的，也是面试常见问题之一。

书籍：  
<MySQL 必知必会>  
<高性能 Mysql>  
<大话设计模式>  
视频：  
数据库视频：  
<https://pan.baidu.com/s/1yIT0nVwFazu0f7mJA0pBiA>  
密码:33n1

**Linux：**Linux 的目录结构、文件系统、启动过程、用户环境，Linux 下常用命令（find、grep、awk、xargs）、正则表达式、软/硬链接、重定向、日志信息、网络配置（top、ps、ifconfig、ping 等）、简单的 shell 脚本会写（常见的脚本写一些自动化工具或者定时任务）、熟练掌握 vim，用 C 写代码的话，还需要掌握 gcc、g 、gdb 调试，makefile 的编写，工作基本上离不开 Linux，也是面试常见问题之一。

书籍：<鸟哥私房菜> 入门学习 Linux 足够了  
视频：Linux 这块的视频，目前最好的是：马哥的和老男孩的视频，建议大家去网上搜着找一下。  
<https://pan.baidu.com/s/1y9Nw2sL0tcFzej2q6DzUwg>  
密码:3o32

**C 语言：**C 语言是我学习编程的第一门语言，是面向过程的语言，对于 C 语言中的数组、函数、指针、内存对其模式、大小端问题、野指针、内存泄露、static、register、define、typedef、struct、union 等一些关键字的考察，我把 C 语言当做基础，不是方向，在 C 语言这里学到了很多编程的思想，对于 Linux C/C 方向，C 语言是基本功，也是必定考察的地方，当初我也是因为特别喜欢 C，走上了 C 的道路。

书籍：  
<C 程序设计语言> 适合入门学习；  
<C 缺陷与陷阱>、<C 专家编程>、<C 和指针> 适合进阶提升  
视频：B 站郝斌的 C 语言强烈推荐

C 语言可以学，C 语言是基础，不是方向，但是选择走 C 方向要慎重！

对于上面的每一个模块，其实都能单拿出来，写的更为具体一些，后面有时间的话，我会写的尽量详细。

**四、C加加方向的深入学习路线**

**C 基础：**C 是面向对象的语言，一定要理解清楚面向对象的思想，先把 C 的基础知识点打牢，刚从面向过程中转变过来，一定一定要适应面向对象的写法，在学习面向对象的时候，也要考虑如何用面向过程去实现面向对象（其实也就是结构体中有一个成员是函数指针），C 的基础知识，我简单罗列如下：

1、const的用法  
2、引用的用法  
3、#define用法  
4、构造函数  
5、析构函数  
6、拷贝构造  
7、new、delete 和 malloc、free 的区别  
8、访问限定符 public、private、protected  
9、深拷贝和浅拷贝  
10、友元函数  
11、static  
12、内联函数  
13、继承、虚继承  
14、钻石继承问题  
15、同名覆盖问题  
16、虚函数表  
17、虚指针  
18、虚函数、纯虚函数  
19、接口  
20、多态  
21、重写  
22、重载  
23、函数重载  
24、运算符重载  
25、流类库和文件

书籍： <C Primer> 第 5 版先全面的了解清楚 C 的基础。

**C 进阶：**函数模板、类模板，C 中对于异常的处理，对于继承和多态底层的理解，对于 virtual 底层的理解等，以及对于 C 中 boost 库八大智能指针的掌握与理解，其核心是理解并且剖析过这几个核心的智能指针，**理解清楚智能指针的本质是，内存的申请与释放全部交给了对象管理，以避免人为疏忽，造成内存泄露。**

书籍：  
<Effective C >  
<more Effectice C >  
<深度探索 C 对象模型>  
<C 沉思录>

**STL 源码：**对于 STL 中的容器剖析，常见容器有 list、vector、stack、queue、map 等，考察自动扩容的原理、map 的底层实现（RBtree），源码是必须手动至少剖析一遍的。

对于迭代器、空间配置器的理解，比如：一级空间配置器、二级空间配置器的运用场合分别是什么？一二级空间配置器的本质是什么，如何用内存池去管理？所存在的问题又有哪些，源码又是如何实现的等等，这些问题都需要去思考。

书籍：  
<STL 源码剖析>  
<泛型编程与STL>

**Linux 网络编程：**  
1、Linux 进程环境：僵尸进程、孤儿进程、守护进程、进程组、会话、前台进程组、后台进程组  
2、Linux 进程七大通信方式：signal、file、pipe、shm、sem、msg、socket  
3、Linux 线程：互斥量、锁机制、条件变量、信号量、读写锁  
4、Linux 下并发模型：多进程、多线程、线程池  
5、Linux 下 I/O 复用：select、poll、epoll 高并发  
6、Linux 网络编程  
7、静态库和动态库

书籍：  
<Linux高性能服务器编程>  
<UNIX网络编程卷一>  
<UNIX网络编程卷二>  
<UNIX 环境高级编程>

视频推荐看：  
陈硕的网络编程：  
链接:<https://pan.baidu.com/s/1vSKPRpNEPSj59iASaZD38A> 提取码:e4q3

**内核源码剖析：**对于 Linux 内核源码，可以先看 Linux 内核的设计与实现，了解清楚每部分的构造与原理，前期多看书、多看相关视频，对一些源码的解读，到一定程度，最好拿到 Linux 2.6 版本内核源码，我是用 Source Insight 工具辅助分析源码的，这个工具对于源码的分析特别友好，很快定位变量、追踪函数，其实重点应该放在内核文件系统与内核数据结构的实现上面，多看看源码是如何实现的，比如：内核链表的源码实现，真的是一种非常独特的思想，没有看的可以去看看（没有源码的可以找我要）。

书籍：  
<Linux 内核设计与实现>  
<深入理解 Linux 内核>  
内核视频：  
链接:<https://pan.baidu.com/s/1jvLYQFJa5ZTZ0_E3kZ-pcQ> 提取码:yn06  
链接:<https://pan.baidu.com/s/1ZsngBAllXGEkThSVearOuQ> 提取码:h9qs

**C 相关视频：推荐看黑马的 C 视频，是全套的，讲的比较好，对于推进的书籍，先看看目录，就大概了解其内容了。**

**开源网络库：**对于 Linux C/C 方向，还需要关注一些业内开源的网络库，比如：MemCached、libevent 等，在 github 上面可以直接搜，对于分布式、高并发、集群和负载均衡等知识，这部分作为了解，要是有兴趣，也可以深入看看，在深度的前提下，作为技术广度的扩充。

**针对上面，从基础到 C ，我列出来的都是非常重要的知识点，当然了，也不可能全部列到，有很多技术问题，很多细节，我可能没有考虑到，没有写到，这个还需要在学习的时候，认真看看视频、看看书，把相应的知识都学到位，基础打扎实一些。**

**五、项目 亮点 面试的一些思考**

**项目**  
对于暑期找实习，或者秋招找工作，除了基础跟方向，接下来就是项目的准备了，对于 C 方向，很多人不知道要准备什么项目？其实项目不在于有多大、有多难，在于项目是否是自己真正做的，自己是否完全消化了，项目中是否有什么亮点？

**C 方面的实战项目，以数据结构 网络编程 实用工具为主，数据结构方面：可以实现红黑树、B 树，又或者用图解决什么实际问题，这些复杂的数据结构实现可以作为项目的；**

**7 大进程间通信，6 大高并发模型，可以写线程池，select、poll、epoll等局域并发聊天项目；实现工具，以自己写个简单的压缩工具、编译器、内存泄露检测工具等，这些都是项目的来源，我的项目就有：网络编程实现在线群聊、压缩工具，对文件或者图片进行压缩。**

对于项目这块，面试的常见问题有：  
a、做这个项目的背景是什么？  
b、项目中具体有哪些模块，都是怎么实现的？  
c、项目中你的技术亮点有哪些？  
d、项目中有什么难点，你是怎么解决的？  
e、要是重新做这个项目，你这个项目有什么不足，你会进行哪些改进？

**亮点**  
找实习或者找工作，必须的有自己的亮点，我觉得在那么多基础知识中，数据结构 C 语言就是我的亮点之一，只要问到关于这 2 个方面的问题，尽量回答全面清楚一些，还有就是自己对于 Linux 内核数据结构的剖析也是亮点之一，总之，一定要找到自己的亮点，与别人不一样的地方，这个就是面试的加分项。

**搜索引擎的底层原理、通讯的底层原理，对于这些底层的实现，有兴趣的可以了解看看，其实也没那么难，要是在面试的时候，能回答出对这些技术的了解，以及自己的一些思考，那就是加分项了。**

**面试**  
春招实习、秋招决战，我不知道投了多少家公司，也不知道面试了多少场，基本上都是海投的，有了面试机会，就要抓住每一次，面试完多做总结、归纳、反思；其实面试很玄学，有自身实力的原因，也有运气的成分，相信自己，好好表现就好。

**对于面试，才开始我也是比较紧张的，但是当你面的多了，慢慢的就会调整过来，就会好很多，我现在把面试当做是，这是一次近距离和优秀的人交流的机会，是一次学习进步的机会，有了这种心态，其实就好的多了。**

**六、更多的个人分享**

推荐一些好的视频网站：b 站、牛客网、极客时间、51cto、七月算法、网易云课程，基本上你所需要的视频资料都可以去找，实在不行上淘宝看看，有些视频文中没有给到，需要自己去找找，我的网盘满了，清空了一些，这些都很好找到的。

对于实在自己找不到视频、书籍资源的同学，你私聊我，我有时间给你找找，视频其实核心就是：抓住黑马（找一个全套的，跟着看、跟着学、一直走下去）。

我走的是 Linux C/C 路线，秋招投递这个岗位也很多，对于没有我简历的、没有 Google C 编程规范的，你私聊我，把这些该有的资料都送给你。

对于这方面的分享，鉴于一篇文章根本写不完，自己也是从去年开始分享自己一路走来的心得、体会，学习的编程经验，目前分享了大约有 20 余万字，设计考研/就业、方向的抉择、以及春招暑期实习、秋招备战的一些思路。

对于很多细节，上面的分享可能远远不够，因为我自己也在不断的学习路上，接触编程的 3 年左右的时间，对于我的感受就是，一定要多实践，多去敲，在多做一些好的项目，对于自己的提升是非常的大的。

**想进大厂，多提升自己的技术能力，基础扎实、算法在强一些，再加上玄学秋招，有些运气，秋招的过程，真的很考验一个人的心态，经历过春招暑期实习、秋招的，多少会学到一些东西的，对于自己哪方面不足，也会有更明确的认识，总之，坚持下去 相信自己。**

**没有谁会一夜之间突然长大，都是日积月累的厚积薄发！**

**下图是我在实验室的座位，以及 20 余万字的个人分享，都是自己一路走过来的真实经历。对于需要我简历、需要查看我 20 余万字的真实经历分享，又或者是需要 google C 编程规范，我的 C 项目等等，都可以加我，和我交流。**

**个人微信：18519336960**  
**公众号：【编程剑谱】，有兴趣的可以关注一波**

  
**我在实验室的座位（当年买的计算机方面的书籍挺多的）**

**下面是我公众号分享的精选文章，值得一读！**  


