

# 码工入门

04-11-2014

## 1. 前言(a.k.a.废话)

我大概总结了一下适合化学、生物、材料三个专业或者别的实验学科的朋友转行CS的学习和找工需要的信息。希望能够帮助一些人。我觉得转行的时候最重要的是先对整个计算机学科有一个框架概念，然后掌握一些常用的基本的工具，接下来就容易很多了。以前从事实验学科的人的思维方式跟数理学科的人不太一样，所以同样是转行，接受跟码工有关的新事物的过程是不太一样的。

## 1.5. 工具准备

你要有一台能够上网的装有linux的电脑。推荐ubuntu。或者你可以买一个mac。

## 2. CS的知识框架

CS的知识不严格的讲可以分成软的和硬的两方面。

软的包括数据结构和算法(data structure and algorithm)、机器学习(machine learning)等，这些跟“计算机”本身其实没有什么太大关系，属于真正意义上的science。很多数学、物理、理论化学、生物信息学和搞各种科学计算的人适合转行的时候学这些，因为有相似性。

硬的包括计算机体系结构(architecture)、网络(network)等，别的像编译原理(compiler)、操作系统(operating system)虽然听上去是软件，但是跟硬件是紧密相关的，所以也算是这一部分的。很多转行的人不学这些东西，但是其实这些东西才是科班出身的CS本科生和半路出家的人的区别。这些是真正的“计算机”本身的知识，属于engineering。我觉得最重要的两门课，一个是computer architecture(计算机的硬件，顺便学习compiler和OS)，一个是software engineering(软件系统，虽然随便写的code也能用来做数据分析得到想要的结论，这个时候code只是作为一个工具，但是如何设计一个复杂的软件系统本身才是CS要研究的)。

### 2.1 从硬件到软件

对于做实验的人来说，最重要的一个概念是抽象。EE和CS里叫abstraction layer。下图大概展示了从底层的器件(做纳米的同学应该很熟悉)到电路，再到驱动和操作系统，再到软件应用的层层抽象。

```
#-----abstraction layers-----#
applications(excel, facebook 大部分人转行都是做这个的)
algorithms(这个理工科的大学都学过)
programming languages/compiler(C++, Java)
operating systems/virtual machines(linux)
instruction set architecture(X86什么的你肯定听说过)
microarchitecture(CPU的设计一定要学)
register-transfer level(EE里做CAD的intel nvidia qualcomm里很多)
gates(与 或 非之类的逻辑)
circuits(电路)
devices(晶体管做纳米的都想替代它)
physics(坑)
```

### 2.2 学习

#### 2.2.1 在线学习的网站

coursera  
udacity  
edx

### 2.2.2 我查到的上面不错的课

linux

<https://www.edx.org/course/linuxfoundationx/linuxfoundationx-lfs101x-introduction-1621#.U0fC9eZdWS8>

embedded

<https://www.edx.org/course/utaustinx/utaustinx-ut-6-01x-embedded-systems-1172#.U0fD0-ZdWS8>

startup

<https://www.coursera.org/course/startup>

machine learning

<https://www.coursera.org/course/ml>

android

<https://class.coursera.org/android-001>

<https://class.coursera.org/posa-002>

<https://www.coursera.org/course/mobilecloud>

### 2.2.3 有必要学习的课 学完对CS了解的就差不多了

data structure and algorithms

software engineering

computer architecture

operating system

computer network

security

programming languages

compiler

database

computer vision

computer graphics

embedded

machine learning

### 2.2.4 别的需要学习的東西

linux commands

<http://ss64.com/bash/>

你在ubuntu下打开terminal 然后在里面输入这些命令就可以了

你需要安装emacs或者vim 这两个是文本编辑器 可以在里面直接写程序和做各种事情 就没有必要用IDE(就是visual studio这种)了 如果喜欢IDE的话 可以用eclipse

去github上注册一个账号 平时可以在上面做开源的项目 自己的code也可以放在上面

CS里有很多黑话, 这个行业的人喜欢用些缩写、奇怪的名字、合成的单词, 给人制造神秘感和高大上的感觉, 平时多留意下。

### 2.2.5 找个师傅

其实课上学的东西和具体做项目写code都简单, 最难的是最初上手, 就跟学做实验一样, 最好有个大师兄带一带。这行geek特别多, 有很多很拽的人, 不要找这样的, 要找个脾气好并且肯一步步做给你看的。有很多小的细节, 比如用什么小工具, 碰到问题的时候去哪找答案, 这些才是需要跟他学的。具体的核心的东西反而课上都有。

## 3. 找工

### 3.1 刷题 下面是题库的名字

cc150  
leetcode  
epi

### 3.2 业界 平时可以关注下业界新闻 这样才能跟人扯淡

<http://online.wsj.com/news/technology>  
<http://www.bloomberg.com/technology/>

google news  
wired  
csdn  
pulse (an app on iphone)

### 3.3 码工的分类

从工作title上分大概这么几种：

- SDE Software Development Engineer 这个就是正经做开发的
- SDET Software Development Engineer in Test 这个是做测试的
- Infrastructure/Performance/Operation Engineer 这个很多是负责大公司内部的data center什么的 升级版的网管(很多都是正经CS的PhD在做) 小公司一般没这种职位
- DBA database administrator 这个不算码工 工资比较低 以前很多f2大妈做这个 主要就是会用sql
- data analyst 这个最近很多big data 很多原来做统计或者别的专业的人转过来做 也有以前做 business intelligence和传统数据库的人在做 反正什么人都可以搞

从平台上来讲可以这么分：

- web 各种网站搭建。包括backend和front end。backend就是服务器那边的。front end就是你浏览器这边的。现在很多要求fullstack就是要求你两边都会。这个是技术更新最快的，各种稀奇古怪的技术。
- cloud 做云的平台。分布式系统。不错的方向。
- mobile 移动平台嵌入式。也不错。
- data analysis 数据分析。要么是学计算机的人开始学统计。要么学统计的人开始学编程。
- 桌面软件。传统的。

### 3.3 简历上热门词汇

简历上多搞些热门的词汇比如hadoop什么的。平时也多关注各种经常看见的单词是什么意思。到时候就算不会也可以稍微扯扯。如果你看见一个单词不认识，无非这么几种：

- 一种奇葩的语言 跟c++ java一回事
- 一种奇葩的library, SDK, API, package, open-source software 等等 就是别人已经写好了一些code来帮助你开发的工具
- 一种奇葩的平台 比如AWS EC2 就是amazon的云计算平台 如果你想自学cloud的一些东西但是显然自己没实力买个机房只有一个破电脑的话 你可以上去租一些虚拟机 很便宜的 非常推荐 而且写在简历上也很好

### 3.4 找工作网站

dice linkedin indeed glassdoor(可以查工资) monster myvisajobs(查h1b 绿卡)

## 4. 创业和进IPO公司捞一笔

除了著名的facebook linkedin amazon/apple google(所谓的flag)这样的大公司和微软这样的老公司，很多人希望进pre-IPO的小公司拿干股。

### 4.1 startup的公司成长

seed ~ \$1 million  
round a ~ \$10 million  
round b ~ \$10 million  
round c ~ \$100 million  
IPO

#### 4.2 VC 风投

seed: ycombinator  
round a: Sequoia, A16Z, Benchmark, Accel, Greylock, Battery, CRV, Matrix, KPCB  
round b: IVP, GVVC, Meritech, DAG  
round c: Goldman Sachs, Morgan Stanley, DST, Tiger

#### 4.3 查小公司的网站

<http://www.crunchbase.com/>  
<https://angel.co/>  
[kickstarter](#) 可以多看看别人的idea  
<http://techorange.com/>  
<http://techcrunch.com/>  
<http://www.36Kr.com/>

#### 4.4 我列了写2014年很可能要IPO的公司和一些别的也不错的公司

[https://docs.google.com/spreadsheets/d/1HYQQELxDSKkRwlvNZzzlYnGhyu\\_GTv41xqiTbFi3XPI/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1HYQQELxDSKkRwlvNZzzlYnGhyu_GTv41xqiTbFi3XPI/edit?usp=sharing)

#### 5. 后记(a.k.a. 废话)

祝大家找到好工作早日发财！

如果大家有任何问题和建议 欢迎写邮件给我 [ninjacoder.california@gmail.com](mailto:ninjacoder.california@gmail.com) 本人水平有限 但是希望可以帮助到国人