

Summary of <<The Ph.D. Grind>>

书籍原作: Philip J. Guo

Summarized by JerryLead@ISCAS

csxulijie@gmail.com

2012-11-25

【前言】

好久没认真读一本书了,这次实验室组织读书推介活动,加上最近被实验数据分析整的头大,想读本书陶冶下情操。想来想去选择了这本电子书《[The Ph.D. Grind](#)》,之前在微博上被师兄推荐过,当时收藏了下,也没多看(微博都喜欢 mark, mark 以后就船到桥头自然沉了)。现在重新选这本书,一方面出于对 Stanford 的膜拜,一方面是跟自己目前的生活关联紧密。本书最近也被国内的热心人士翻译成了中文版,不过对这本书来说,读中文版犹如去买了个 Mac,回去卸掉 Mac OS X,装了个 XP。

【简单评论】

由于刚刚读完,忍不住在介绍主要内容前先评论几句。本书讲述一位华裔、懂事、独立性强、刻苦努力、执着、略带叛逆、有追求、有能力的斯坦福大学计算机系博士生 Philip J. Guo (以下简称 Guo) 的六年读博经历。全书 122 页,以时间顺序介绍了自己六年(06 到 12 年)求学生涯中的重要事件和心路历程。

词法方面: native speaker, 挺多 GRE 词汇,用词精准有力,双关语多

修辞方面: 多比喻,暗喻,排比

写作风格: 逻辑性强,考虑面宽,略带嘲讽和搞笑,重视心理活动,《明朝那些事儿》风格。

先八卦一下,Guo 就是之前风靡人人网、微博上的 Python 执行可视化 [visualize](#) 的作者。

【主要内容】

【以后想干嘛】

Guo 出生在一个美籍华人家庭,父亲是一个高技术人才,母亲是 UCLA 的社会学教授。受社会背景和父母影响,长大想找一份稳定又有点地位的工作,首选教授。Guo 本科去 MIT 计算机系读本科,在低年级时候暑期去外面公司实习,在公司里干了一些对他来说是没什么挑战,不需要太多思考的活。有点厌倦以后,在大三决定加入 MIT 的 5 年制的本硕连读项目,决心以后读 PhD 做研究。由于硕士期间是导师(下面也称老板)发钱,需要做老板的项目。项目是对计算机编程语言 C 和 C++ 进行运行时分析。由于做的不错,读硕期间发了两篇二档的文章,其中一篇是一作,加上老板的推荐,顺利拿到心仪的 Stanford 的 offer,还有两个奖学金,可以支持他五年的学费,并且衣食无忧。

【很糟糕的第一年】

在去 Stanford 前的暑假里,Guo 考虑以后研究啥方向呢?思索来思索去,决定研究能够提高编程人员效率的工具。这个编程人员不仅指程序员(俗称码农),也包括数据分析师、需要写程序的科研人员等。之所以要做这个方向,是他感觉自己本科实习和硕士期间编程繁琐,处理数据麻烦,工具不够友好。

这个领域算是软件工程领域,可惜 Stanford 搞这个的比较少,Guo 选来选取选了一个做

系统程序分析的副教授 [Dawson](#)。这里我忍不住八卦下，这位教授是操作系统领域泰山北斗 MIT 教授 Frans Kaashoek 的学生，Dawson 在其主页上写到“I now I have my own students, who sometimes listen to me”，在其有限的几个学生中，有一个本科清华的 Junfeng Yang。Junfeng Yang 教授现在在 Columbia，在去年的 SOSP11 上连灌 3 篇。作为操作系统顶级会议，该会议每年也就 28 篇左右。Yang 的姐姐 Xiaowei Yang 也是清华毕业，目前是 Duke 的副教授，我师兄的导师。

回到正题，Guo 进入 Dawson 的组后，被分配的第一个活是测试 Klee 是否能找到 Linux（跟 Windows，Mac 一个级别的操作系统）驱动程序的 bug（错误）。Klee 是 Dawson 最关心的工具，主要用来找程序 bug。

Guo 想，这跟我想做的“用来提高编程人员效率的工具”目的差的很远啊，有些不爽，便找师兄们诉苦。师兄拍拍他肩膀说，新来的一般都要干一些实际的，不需要太多想法的活。Guo 说，好吧，认真干活吧，随着工作的进行，发现这活不是人干的，每天要花很长时间配置 Klee 需要的环境，而且有些时候要检测的软件的 bug 没检测出来，反而发现这个组开发的 Klee 的工具具有 bug。痛苦测试一段时间后，终于能发现驱动程序的 bug 了，但讽刺的是，配置 Klee，并搭建好测试程序的环境需要 n 个小时，而且对于依赖操作系统的长程序需要手工切分才能测试。

这个时候如果是我这种功利型的，直接就写 paper 了，写我们的工具能够发现 Linux 驱动的 bug，多牛。但 Guo 是一个近完美主义者，既然 Klee 有大漏洞，就一定要补上。他直接找老板说，这不成啊，要改进。老板碰到这种学生也无奈，开了个小组会，叫一个高年级的去改进，同时对 Guo 说，你继续做测试。Guo 无语，继续痛苦测试，直到 paper deadline 的前三天，Klee 终于被初步改进，并测试出了一些结果。但只有三天写 paper，一般对 Stanford 这样牛校来说，老板加上小工需要 1 个月时间组织文章、写、贴实验数据、融合各自的部分、润色。但现在有且仅有三天，Dawson 还是冒风险将小组人员集结在一期通宵达旦写文章，东拼西凑一篇文章后提交到顶级会议。最后得到的审稿意见是“哥，你是来搞笑的吧，这样读不通的文章你也来碰瓷？”

Guo 很无奈，不能一开始做自己想做的研究方向，而且迫于规则苦干了三个月不需要太多思考的工作，自己提的 idea 也没被实现好，最后还被这么无情拒掉。

由于 Guo 有奖学金支持，不像其他依靠老板项目资助的 PhD，一直需要为老板的项目干活。Guo 果断退出 Klee 项目，开始独自一人漫步于校园，思考人生方向，读大量 paper，想一些自己想干的项目，并在暑假时找了一个与自己研究方向不相关的 Google 实习，仅仅是为了换个环境，想起来 Klee 头就大。

由于毕业论文需要 3 个教授来担任指导委员会，而换导师需要再次接触，干活，Guo 就没换。当然除了休息以外，Guo 思考得到了自己新的研究方向应该是经验软件工程，确切说是软件质量测试，那么首先要做的事情就是在软件开发整个周期里观察程序员如何写程序，然后分析出一些规律规则，并根据规律规则来判定软件质量。

【无奈的第二年】

要观察程序员开发过程得找人啊，他跟 Google 的 mentor 说能找几个人让我观察下他们怎么写程序不能？mentor 说可以，结果程序员不同意，一方面出于代码保密的原因，另一方面谁写程序时都不想被一个站在旁边的人监视着。那找开源的的行不？找到 Mozilla（开发火狐浏览器，雷鸟邮件客户端等软件的开源组织），结果人家程序员也不答应。这就是软件工程实战派遇到的最大问题啊，程序好找，人难找啊。

怎么办？Guo 回到了 Stanford，找到一个搞 HCI（人机交互）方面的助理教授 Scott 攀谈，两人一拍即合。Scott 派他的博士生 Joel 跟 Guo 组队研究，他们找来了上 Scott 的 HCI

课程的本科生，观察他们如何写一个课程作业的程序，发现一般是遇到问题先搜索，然后贴代码等等行为。最后他们把各种有意思的发现写成 paper，经过 Scott 的润色，投到了人机交互的顶级会议 HCI 上，可喜的是 paper 中了，可悲的是 Guo 不是一作（第一作者），不会对最后的学位论文有贡献。不过 Guo 慢慢体会到了做研究的感觉，最起码比第一年痛苦干活几个月，最后毫无结果强。

到第二年快结束时，由于没有一作文章，Guo 开始有点着急，HCI 那边 Joel 可以完全 hold 住了，自己需要想新的研究项目。无奈之中，Guo 又回到了 Dawson 的研究组，回到了让他痛苦的 Klee 项目。

在原书的第二年讲述中，讲了很多论文撰写、投稿的一些门道，或者叫潜规则吧。分析了教授、学生、项目等等的利益关系和心理活动，很值得关注。

【悲剧的第三年】

Guo 又回到了 Klee 项目，仍然作为低年级学生对待，因为这时候 Klee 项目组人员走的差不多了，只剩老板和马上要毕业的博士生 Cristi 了。Guo 心情复杂，想要搞经验软件工程，帮助编程人员提高效率，老板却十分看好找软件 bug 的方向。无奈之中，Guo 开始做一个叫做交叉检测的项目，目的是为了发现同一软件不同版本之间的不一致性。但由于 Guo 是这个组里面最低年級的，测试任务又落到他身上，整天就是配置环境，测试，收集数据，分析，跟第一年没什么实质区别。（这里我要评论一下：我靠，这不是哥天天干的事么，忍了）。四个月过后，仍然没有可发表的结果。

这个时候 Dawson 的得意弟子 Cristi 毕业答辩了，Guo 参加了一下，感慨想超越这个师兄太难了，毕业前景渺茫。八卦一下，Cristi 最后拿到了伦敦帝国大学的教职，在国外，能拿到高水平大学教职对 PhD 毕业生而言是至高荣誉。Guo 心想这可怎么办，浪费了四个月，一无所获，跟老板攀谈后。老板说你要不改进下 Klee 的搜索算法吧，现在的算法虽然能工作，但不高效，这听起来是一个很 cool 的工作，经过 Guo 辛勤工作后，Klee 搜索算法召回率（学过 IR 的人，你们懂得）提高到了 94%，可惜的是 Klee 之前的搜索算法已经有 91% 的召回率，显然，效果肯定不会被顶级会议接受啊。Guo 心碎了，这个工作一开始就是有巨大风险的，改进也得看可改进空间大不大啊，这不害人么。

Guo 没办法，想起了第一年在测试时遇到的问题，提给老板的 idea，最后因为 paper 写作时间太短而被拒。Guo 再一次捡起了之前的 idea，这时候 Klee 组新加入了一个 PhD 叫 Peter，后来两个人一块开始重新实现这个 idea。在实现过程中，老板开始不耐烦，嫌他们工作进度太慢。不过也是，能够在 Stanford 这样档次学校找到教职的人，平均每年顶尖大学的 PhD 毕业生 75 个中才出一个。后来 Guo 和 Peter 感觉让老板失望了，终止了这个项目。Peter 更是失望地换了导师，没多久也博士也退了。

插曲：两年后，Dawson 新招的博士中有一个叫“David A. Ramos”的融合了 Guo 之前做过的两个 idea，终于发表在计算机辅助验证的顶级会议 CAV。但这代价太大了，前后经历了三次尝试，4 个 PhD，5 年才得到一篇文章。4 个 PhD 中，只有当前的一作幸存，Guo 离开了 Klee 项目，另外两个最后连博士也一起退了。

【暑期实习插曲】

经过三年的折磨，Guo 在 09 年暑假去西雅图的微软总部研究院（MSR）实习。其实 Guo 早在第二年暑假去投了 MSR 实习，结果被拒，原因是 paper 不够吸引人。现在 MSR 有个研究员看到 Guo 和 Scott 合作发的 HCI 论文后，发邮件问 Guo 要不要去实习。原来硕士导师也向这位研究员推荐过。Guo 对这位研究员说自己已经不再搞 HCI 方向了，这位研究员很好，帮他推荐到了另一个专门搞经验软件工程的学术新星 Tom 那。Guo 也把之前做的

Linux 驱动测试项目的 paper 发给 Tom。两人很合的来，开始研究关于 bug 提交、分发、修复中的人为因素。依靠几个月的努力，加上微软庞大又规整的 bug 库，Tom 高超的写作水平，他们最后投出一篇 ICSE（软件工程领域顶级会议）的论文，Guo 是一作，该论文在 10 年被录用。之后的几年中，Tom 将现有工作推进，又发了一篇 CSCW11 以及今年的 ICSE（而且还是 SEIP Track 的 best paper）。

本来 Guo 打算一直实习下去，毕业论文也在 MSR 做。但之前 Stanford 还没有这个先例，如果回到 Stanford，将失去微软强大的 bug 库，没有数据，会为研究带来巨大困难。Guo 又联系了 Google 以前的 mentor，mentor 表示可以过来做研究，但该 mentor 不是搞学术的，不以发 paper 为第一目标，Guo 最后还是决定回 Stanford 做新的研究。

【重生的第四年】

一直以来，Guo 都使用 Python（不是蟒蛇了，一种计算机编程语言）来写程序进行数据清洗、分析和可视化。由于处理的数据多而且各种各样，Guo 不得不针对每种数据写相应的 Python 处理程序，管理数据依赖和包依赖等问题。而且每次有新的分析需求，需要加入新的代码，处理程序越来越复杂。Guo 想通过改 Python 的运行环境（也就是解释器）来减少上面提到的一些任务。Guo 将新工具命名为 IncPy（增量 Python）。上网搜，发现这个 idea 还没被实现过，Guo 就带着这个 idea 回到了 Stanford。

对于大多数研究生来说，一般是跟着自己老板做项目，项目一般来源于老板申请的课题（当然可以是老板新想的 idea，当然最好是自己的 idea 并受到老板认可），运气好的话能够发篇高水平文章。老板负责找另外两名教授组成指导委员会，因此研究生只负责做好自己被分配到的任务或者做经过老板认可的自己的研究项目即可。但对于 Guo 来说，这些都是奢侈，他基本脱离了 Dawson 的具体指导，项目要自己想，其他两名指导教授还要自己找，而且现在还没有对毕业论文有用的 paper，你说捉鸡不着急。

Guo 开始给其他教授发邮件阐述自己的想法，看有没有对他项目感兴趣的教授，不出所料，没人甩他。Guo 不管那么多了，开始冒险，继续实现自己的 idea。一边写代码实现，一边与人交流。某天中午吃饭，Guo 的 PhD 同学 Robert 说他正在考虑投篇论文到一个 workshop（一般计算机会议包含正会和针对某个专题的小型研讨会称为 workshop，正会的 paper 都是正式完整的 paper，workshop 一般用来发一些初步结果、idea 或者不完整的工作）。Guo 一开始没在意，因为 workshop 上论文一般对毕业论文没啥作用，只是可以去开会交流。其实这个 workshop 是 USENIX 办的，还不错，有很多主流学者参与。Guo 考虑地点也很近，在 San Jose，决定先投稿，去交流一下。当论文被录用后，Guo 在会议上做了报告，认识了来自 Harvard 的 Margo 教授，她赞赏了一下 Guo 并说她的一个学生 Elaine 也在做相关项目，让 Guo 去联系 Elaine。

Guo 与 Elaine 保持了联系，也发现她做的东西与自己并不是竞争关系，聊的挺好。Guo 开始继续 IncPy 项目，想投 7 个月后的顶级会议。要验证软件的效率提升效果得找人试用啊。Guo 求 n 多人试用其软件，结果仍然没人甩他，大家都不愿意再花时间去学一个 PhD 做的不成熟软件。发现 Stanford 没人愿意用，就找旁边的学校，Guo 给 UC Davis 的一个同学发信，同学也给了很好的反馈，但没有实质的进展。这时候 Guo 贸然给 UC Berkeley 的神经学家 Fernando 发信，Fernando 一直在用 Python 来进行计算。Fernando 让 Guo 来做个报告，并安装一下他开发的 IncPy。Guo 兴致勃勃地过去做了报告，安了软件，心想这次实验部分有的写了。几天后，Guo 收到自己软件不能处理 Python 依赖包问题的邮件。Guo 心想这次坏了，自己在测试时只测试了一些常用的包，没测试过他们用的包。Guo 没有放弃，接下来几周，努力改进，但当再次联系 UC Berkeley 的时候，没有了回复。是的，有时候，机会只有那么一次。

Guo 带着有冤无处申的心情继续自己的工作，搭了个网站，录了像，介绍自己的 IncPy。几个月过后，只收到 3 个人的反馈，说这个东东在他们的研究计算工作中确实有用。Guo 心想聊胜于无，自己写了个 paper 老实地说他的软件经过 3 个人试用，提高了效率。顶级会议肯定对这个效果不感冒。

一直以来，相对于实际的软件开发，教授们更偏爱理论创新。Guo 更看重自己软件的实际效果，将大量精力放在了开发和维护上，做实用的东西，虽然最后的结果可能不好。

【高产的第五年】

前面的四年都是悲剧大于喜剧。各种项目失败，各种论文被拒。少有的成功也不能对最后的毕业论文做贡献。而这时，大多数同学至少发了一篇算数的一作文章，加上自己的奖学金是最后一年了，Guo 开始热锅上蚂蚁了。情急之中，Guo 想出了下一个项目 Sloppy Python (Sloppy Python)，帮助用户在 Python 程序崩溃时分析查看部分计算结果，仍然是要改 Python 的解释器。Guo 花了两个月写了个原型系统，做了一些初步试验，投了篇二档会议，因为他不认为这是个很强的 idea。

这个时候，Guo 有两篇正在被申，已经放弃做教授的梦想，因为这样的成果，在 Stanford 毕业都困难啊。一段时候，Guo 收到几乎崩溃的结果，两篇都被拒。Guo 仍然没有放弃，分析了理论和实践的关系，研究下各个会议的胃口，改了后将 IncPy 的那篇转投另一个二档会议，最后被接受。将 Sloppy Python 那篇转投 USENIX 的 Workshop，最后也被接受，并在参会时一块介绍了 IncPy。虽然有一定好的反馈，但这次工作要毕业远远不够。

Guo 又联系了 Berkeley 的 Fernando，Fernando 给他提了个需求，说现在计算神经学里面很多实验不能够重现，即是得到同样的实验数据和源代码，也会因执行环境的不同，不能得到同样的结果。不能重现，工作就得不到有效评价，也不容易继续跟进。Guo 心想，啊，好 idea，原来只考虑代码和数据，现在加上环境因素，有的做了。回到 Stanford，跟老板 Dawson 讨论，说要做一个针对 Python 的可移植的执行结果可重现的工具。姜还是老的辣，老板说，你为啥要限定是 Python，忽视广大的兄弟语言。

Guo 大彻大悟，干脆做个通用的可移植的代码执行工具，简称 [CDE](#) (代码，数据，环境)。这个项目的牛逼之处在于，帮助广大 Linux 苦逼青年摆脱了困扰十几年的不同发行版的软件安装、配置、执行问题。因此，在还是半成品的时候，Guo 建了网站，发了一些 demo，被同学推到了 Slashdot 上，被评价是“U ROCK”。随后 Guo 将工作发了篇短文到 USENIX ATC 上，还有篇长文到 USENIX 的 LISA 上，都是非常好的会议。CDE 也被 NASA，开源组织，欧洲网格项目，各类研究人员，Google 等采用。Guo 也被邀请去 Google 做技术 talk，Youtube 直播。以前这个 talk 只邀请著名教授或者杰出工程师。这个应该是 Guo 最成功的也是实用价值最大的项目。

此时，如果是一个普通的 PhD，就该转到下一个新项目了，因为学术界主要负责产生新 idea 和原型系统，如果实用价值很大，会有热心人士或者公司专业人士负责做成产品。Guo 个性显然与其他人不一样，一定要做的满意为止。

在 Google 完善 CDE 的同时，由于担心 paper 可能不中，他又联系了一名新的助理教授 Jeff，也是搞 HCI 的。Guo 给 Jeff 讲，说自己当前的想法就是做个新工具，发顶级，两人一拍即合，准备开始一个新项目 ProWrangler，是对之前的 Wrangler 工具的扩展和改进，具体干什么事，书中没有细讲。Guo 花了两个半月实现系统的原型加评价，并写了篇 paper，Jeff 写了论文中的 Introduction 和实验评价与结果解释部分。Jeff 的强大论述能力和文章发表经验，让这篇 paper 被顶会相中。Jeff 在自己读博时发了 19 篇高水平文章，而且大部分是顶会。

至此，Guo 已经做了 4 个项目，而且相应的文章都成功发表了。

【要结束的第六年】

虽然 Guo 做了 4 个项目，有相应的文章发表，不过现在只有自己的老板 Dawson 和新合作的 Jeff 愿意做他的论文指导委员会的成员，还需要一个。Guo 联系了曾经在 MSR 实习时的导师 Tom，问他愿不愿意客串一下，这样可以将之前在微软发的文章算上，可惜 Tom 表示兴趣不大。

插曲：经过几年的磨难与纠结，Guo 做出了人生的重大决定：毕业后告别学术界。原因挺多：想做的东西一直被认为是非主流，自己 paper 也不够牛，对以后自己撑起一片天没有足够信心 and so on。

还差一个人，怎么办？在 San Jose 的 USENIX 学术会议上，他再次看到了 Harvard 的 Margo 教授。他想上去再问个好，但担心人家是否还记得起自己。Guo 心想，哎，不管那么多了，反正准备撤离学术界了，上前搭讪并介绍自己的工作。还好 Margo 记起了他，之后发邮件邀请他毕业后到 Harvard 做博后，与自己的学生 Elaine 一块工作。Guo 想了想，说要不这样吧，我到你那完成最后一个项目，发最后的文章，Margo 答应了。

Guo 又回到了熟悉的波士顿，与 Margo 合作做一个帮助研究人员组织记录实验笔记的辅助工具，称作 BurritoBook。显然，这个工具与自己这几年一直关注的提升效率，更好的人机交互有关。一边实现，一边与自己以前的 MIT 同学、教授讨论。接下来开始了最漫长的 72 天写代码阶段，每天工作 10 个小时以上，中间休息了 5 天，里面有感恩节、圣诞节、新年等等。最后完成工具和一篇 paper，回到 Stanford。

回到 Stanford，跟老板 Dawson 商量毕业的事情，毕业的理由很简单：Guo 创造了 5 个软件工具 IncPy, SlopPy, CDE, ProWrangler 和 BurritoBook 来帮助需要编程计算的人员提高生产效率，这 5 个项目发了 1 篇顶会，3 篇二档会议，3 个 workshop，最后关于 Burrito 那篇被拒，转投中了一个 workshop。并且凑齐了 3 个教授：Dawson, Jeff 和 Margo，并且神奇地跟着后面俩人都有项目合作和文章发表。导师 Dawson 竖起大拇指，表示同意答辩。

虽然 Guo 的 5 个工具大目标都相同，但他不想写一篇只是介绍 5 个工具的毕业论文，所以仍然花了大量时间来组织和构思毕业论文（大概 230 页）。

最后，Guo 选择 2012-04-23 答辩，因为这时候 Harvard 的 Margo 教授在加州开会，比较方便。除了 3 个指导委员会的，还有两个教授参加答辩。答辩包含 1 个小时的公开报告和 1 个小时的不公开问答。不公开问答后会决定该博士生能够毕业。Guo 很紧张，以为会被问的很惨，还好，教授们提了些问题和建议，顺利通过。让 Guo 感动的是，之前有过接触的 Scott, Joel, Peter, Robert, Greg 以及 Berkeley 的 Fernando 都来参加答辩，给他鼓劲。

撒花，Guo 拿到了 Stanford 的 PhD 学位。在所有的祝福中，他收到了一个资深教授的评价“I just wanted to say that I really enjoyed it, partly because of the creativity of the work, partly because of the well-prepared talk, and partly because I had spent the previous year doing research programming.”这是对一个 PhD 莫大的肯定。

那一年，哦，也就是今年，Stanford 计算机系毕业了 26 个博士，Guo 自我感觉是中等水平，虽然 paper 没有想象的好，但是他横跨了编程语言、人机交互、操作系统三个计算机科学的子领域。而且解决部分编程计算人员一直以来遇到的种种困难。

最重要的是 Guo 已经选择好未来要走的方向：做一个专业技术人员。

【一些话】

先摘抄一些 Guo 的箴言

I feel extremely lucky to have been able to take charge of my Ph.D. career in creative ways.

It's been a long, circuitous, and unpredictable journey, but I'm incredibly grateful that I was able to

turn this broad topic-one out of dozens that caught my interest over the years-into my Ph.D. dissertation.

This accomplishment wouldn't have been possible without a rare combination of great luck, personal initiative, insightful nudges from generous people, and nearly ten thousand hours of grinding.

在本书的后记部分，Guo 总结了几条经验：

- 1) Results trump intentions
- 2) Outputs trump inputs
- 3) Find relevant information
- 4) Create lucky opportunities
- 5) Play the game
- 6) Lead from below
- 7) Professors are human
- 8) Be well-liked
- 9) Pay some dues
- 10) Reject bad defaults
- 11) Know when to quit
- 12) Recover from failures
- 13) Ally with insiders
- 14) Give many talks
- 15) Sell, sell, sell
- 16) Generously provide help
- 17) Ask for help
- 18) Express true gratitude
- 19) Ideas beget ideas
- 20) Grind hard and smart

感慨：不经意的小事，不经意的联系，遇到的每个人也许都会成为人生中的重要引路人。

【感悟】

对这本书，只有作者自己才能评价，这里姑且叫做感悟吧。

人物方面：

Guo 不是一个喜欢按套路出牌的人，这点从大学实习和研一进实验室干活的心理活动可以看出。与大多数有想法、喜欢独立思考的人一样，有着“我的未来我做主”的气质。Guo 也有完美主义倾向，认准的事一定要做，一定要做好，不管是主流还是非主流。Guo 内心强大，具有执着的精神，纵有再多失败，仍然可以调整自己，适应环境，最后努力获得成功。Guo 善于学习，从各种事情中吸取教训，也能分析出和平衡各种利害关系。

教授也是人，Guo 说的。不够从这本书来看，至少这些名校教授在 PhD 时期相当于黄金圣斗士中处女座神一样的沙加了。

教育意义方面：

最早的叛逆性的文章是王垠写的，写过一篇从清华退学的文章，也写过一篇从博士退学的文章。跟这篇相比，王垠写的实在没什么可圈可点的地方，虽然他写的 Latex 教程挺好的。这本书从前到后没有炫耀作者怎么怎么牛，而是平铺直叙地讲出让人信服的道理。

另一篇最近比较火热的是程老师的《逃离科研》，感叹自己的 PhD 发了 n 多顶级，最后选择去人大附中当中学老师。这个讨论对很多人都有意义，但对于在读 PhD 的人来说，没啥太大意义，因为其实这是在讨论一个高富帅是买房还是买车的问题。

本书受到了各个年级：本科生、硕士生、博士生、毕业生、退学生、教授的关注，大家都从中学到了一些东西，这就够了。

学术方面：

研究系统的人真心伤不起，USENIX ATC 这样级别会议在 Stanford 只算二档，那一档的系统会议每年加起来也就不足百篇吧，具体可参考陈海波教授的[《一名系统研究者的攀登之路》](#)。

Guo 一开始就当起了自己的导师，过强的能力，过早的锻炼让他最后身心疲惫，我喜欢他写的这个词 “emotionally and mentally all-consuming”。

【By the way】

本书没有谈及 Guo 的感情经历啊，十分好奇，不过他的毕业论文中感谢了 wife，也许一开始就有稳定的感情吧。书还提到自己经常回家，看了他在 USENIX 的报告视频，很帅的。

原谅我没用 Latex 排版，中英混杂的 Latex 排版真心伤不起。

陶喆的《爱很简单》、《Melody》、《就是爱你》、《Susan 说》、《爱我还是他》等真心好听，信乐团的《天高地厚》、《海阔天空》、《离歌》、《One Night In 北京》等也好听啊。