链滴

顶尖程序员 5 个思维模式，你可能想到却没有做到

作者：gentoo666
原文链接：https://ld246.com/article/1513579589132
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

《The Effective Engineer》的作者在写书的过程中，为了了解那些顶级程序员和普通程序员的区别采访了很多硅谷顶级科技公司的顶尖软件工程师。他发现这些给世界带来巨大影响的的工程师们至少以下5个共同的思维模式。

1.勇于去研究你不懂的代码

一般人都不愿意去研究自己不曾接触过的代码，很多人都没有尝试就放弃了。如果你经常去研究你没接触过的代码，你就会越来越熟悉不同的代码结构和设计模式。现在人们很容易就接触到优秀的开源码资源，你可以很方便的就下载下来做一些改动或者调试，去研究为什么代码可以这么写。

除了代码之外，很多人对于陌生的工作内容也会感到恐惧。每次换工作的时候，你可能都会遇到新公的工作内容和以前工作的内容不一样的情况，以至于刚开始的时候工作效率没有以前那么高。很多人至觉得，他们是不是骗了面试官。

其实，大家都是在学习的过程中。在一个陌生的领域，没有人从一开始就是大神。如果你想变得越来好，无论是写代码，与人沟通或者其它的技能，都是需要投入时间去学习的。

2.精通代码调试(debug)

很多人在写代码的过程中，经常会有的一个问题就是：为什么我写出来的代码不能运行？为什么运行结果不是我想要的？

几乎所有的程序员写代码都不是一遍就能写好的。但是顶尖的程序员非常快的就明白自己代码的问题能是什么。这是一个很重要的能力，但是偏偏学校里不教，面试的时候考官也不经常提及。

那么怎么去调试代码呢？其实核心就是以下几个方法：

不妨先猜测一下到底发生了什么。假设你的猜测是对的，想想你的猜测会导致程序有什么结果。试着察这些结果有没有异常的地方。如果你没有发现异样，那么说明你的猜测就是对的。如果你发现了异，那么说明你的猜测是错的，接下来换一个猜测试试。

对于顶尖程序员来说，这个过程在脑海中就是电光火石的一瞬间。只要你解决的问题足够多，你做出的猜测就会越准确。

至于如何发现异样？你就需要有一套自己的工具或者方法论了。最简单的就是在代码里输出日志来判。但是这是比较笨的办法，你需要去接触一些高级的工具或者直接带有Debug功能的编辑器。

3.重视能够节约时间的工具

最近打败人类的AlphaGo每天可以进行上百万局的下棋训练，我们人类一万个小时的训练却需要10年久。也就是说，电脑运行几分钟，可能就等于人类工作好几年。

曾经在Facebook担任技术总监的Bobby Johnson描述过，高效率的程序员都把时间花在制作工具上。

很多人也认为工具是很重要的，但是他们并没有花时间去制作、整合自己的工具。但是，Jonson团队出色的员工耗费了他们1/3的时间在工具制作上，这些工具可以用来发布代码，监控系统，以及能让们花更少的时间去做更多事情。

总之，不要花时间去做机器可以代替你去做的事情。

4.优化你的迭代速度

假设你要花12秒钟去搜索某个函数是在哪里定义的。再假设你每天做这个动作60次，那么你每天就要12分钟去搜索函数定义。

如果你用一个好一点的编辑器，每次找到函数定义只要2秒钟，那么你每天就会节约10分钟。每年你可以节约40个小时。

如果你能找到3个这样的场景去优化一下，那么你每年可以节约一个月的时间。想想这一个月你可以多少有意义的事情。

再假如你在调试一个App的bug的时候，改完一次代码都需要重启一下App，然后点击4、5次才能看bug有没有改好。那么你是不是可以先花几分钟设置以下，让App一启动就转到显示Bug的页面呢？

千万不要小看这些琐碎的细节，改善它们的回报是巨大。

5.系统性的思考方式

当你在写代码的时候，你很容易就认为只要你按照需求实现了指定的功能，你的代码就写完了。但是其实只是冰山一角。任何没有发布到生产环境的代码都不会产生任何价值。

如果想写出真正有影响力的代码，你需要从整个系统去理解你的工作：

你的代码和其他人写的代码在功能上是什么关系？你有没有好好测试你的代码？或者其他人是否很容测试你的代码？为了部署你的代码，线上生产环境的代码是不是需要改动？新的代码会不会影响到已运行的代码？在新的功能下，你的目标用户的行为是不是你期望的？你的代码有没有产生商业上的影？

这些问题都不是很容易就能回答的，但是在写代码的时候，你需要明白你的代码最后会不会得到最好结果。