

用户故事-因为热爱，所以坚持

你好，我是宫文学。

很高兴能够看到你分享自己的学习故事。

通过你的留言和故事分享，我能深刻地感受到你对编译原理的热爱，感谢你能与我一起，坚持学习，努力进步，把编译原理这门硬骨头一点一点、一步一步地消化掉，学有所用。

你好，我是雲至，今年38岁，现在在电力公司做信息运维工作。

虽然我的工作与编译技术并不相关，学习编译原理似乎没有用武之地，但是，编译原理于我而言有着特殊的意义，它伴随了我整个大学时代，“啃”下它，攻破它，成了我多年后的目标。

大学时，我学的是信息与计算科学，那时，接触了很多计算机的数学原理，出于好奇心，我尝试去了解编译原理教材，却觉得像天书一样，整整看了50多遍就是看不懂。虽然不服输，但因为各种客观原因，只好放弃。

其实，我特别想知道计算机语言是怎么样变成能被计算机执行的语言的，步入中年后，我开始计划学习编程语言和计算机基础课，**可在学习编程语言时发现如果不懂编译原理的话，自己的认识水平根本无法提高**，而那时，我心里那股不服输的劲儿又燃了起来，所以当“极客时间”开设《编译原理之美》课程时，我马上就报了名。现在，学到18讲，我想把自己的感受分享给大家，可文笔不佳，还希望大家不要见怪。

感受一：宫老师讲解思路特别清晰，课程设计比较巧妙。

在原理上，老师讲了很多书本上看不到的编程思想，比如清晰化，简单化和好维护。并用清晰的AST还原了程序代码从代码变成可执行的代码的过程。

在学习方法上，宫老师提供了一个比较高效的学习方法，先帮助我建立了对编译器前端技术的直观理解，在“[01 | 理解代码：编译器的前端技术](#)”里，让我真正理解了词法分析、语法分析和语义分析到底是什么意思。然后宫老师由浅入深，展开解析，帮助我理清了编译原理的知识体系。

感受二：原理和实践并行，让我通过动手提升认知。

在学习完词法分析讲之后，我认识到有限状态机的编程思路可以大大简化编码的难度，老师通过计算器的例子，特别清楚地讲明白了这个方法。

而“[08 | 作用域和生存期：实现块作用域和函数](#)”则解决了我很多年的困惑，让我明白了变量的作用域的概念具体是怎么一回事。与此同时，课后老师及时提供了示例代码的链接，我通过动手演练，明白了一些没有搞懂的内容，比如函数、作用域等等。

随着课程不断深入，我的困惑也多了起来，当我在留言区提出自己的疑问时，宫老师总是能不厌其烦地讲解，十分认真负责！十分感谢宫老师带来这个课程，我也会继续努力的。

你好，我是沁园，是一名软件工程专业的研二学生。

研一时，我曾学过编译原理的课，但课上老师只讲了一些理论，没有结合实例，学的不明所以。而我自己一直对编译原理非常好奇，好奇编程语言底层到底是怎么实现的，也一直想要探知，本想啃下“龙书”和“虎书”，却因其厚重、难懂而搁置了。

后来，宫老师在极客时间上开设了《编译原理之美》的课程，三个月讲完编译的前端与后端技术，我毫不犹豫地入了坑，并从第一讲一直坚持，学到了现在。在这个过程中，我有一些学习的心得，所以想借此分享给大家，也向宫老师表达感谢之情。

心得一：在我看来，这门课不能只利用碎片的时间，而是需要课下动手和思考的。

因为编译原理本身就比较有难度，外加篇幅所限，只看文本的话，还是会产生困惑。我记得自己在“08 | 作用域和生存期：实现块作用域和函数”时，走入了死胡同，后面的连续几讲都看不明白，几乎快要放弃。

不过，宫老师贴心地在GitHub上提供了全部的源码，而且用到了我比较擅长的Java语言。我相信Java语言的程序利用IDEA的调试器就没有什么看不明白的，于是利用一个周末，把老师提供的示例脚本全部放到解释器中跑，并把解释器用IDEA的调试功能单步执行一遍，观察解释器都是怎么处理类、对象、函数以及闭包的。

调试的过程中，我**边调试，边思考，边在笔记上总结**，最后才恍然大悟，真正明白了老师讲的内容。真正搞懂之后，一直以来，编译器在我心中的神秘色彩也就消失了，**编译中的类型推导，引用消解等高大上的概念也不过是由判断，循环等简单逻辑组成，只不过需要考虑的东西相对多些**，如果几十年前让我来创立第一门编程语言，我肯定也这么搞。（目前只学了前端，学完后端以后可能观念还会有所改变）。

心得二：除此之外，这门课非常注重实战，先帮助我们建立直观认识，再去讲细节的算法。

一开始我还很奇怪，课程怎么这么“水”？编译原理不应该先把DFA、NFA、NFA转DFA、LL、LR这些经典算法作为开场吗？这个课程怎么用前三讲就把这些东西“水”过去了，然后开始讲Antlr以及语义分析了呢？

后来我发现，这些内容放在了“算法篇”中讲解。在将老师写的解释器源代码过了一遍之后，我越发地感觉到老师用心良苦。**对于实现一门编程语言，实现语义才是最重要的，之前学编译的时候完全陷进了NFA、DFA、LL、LR等算法的细节中，完全没有意识到语义才是一门编程语言的灵魂。**

宫老师一开始就教我们如何复用现有的成熟的Antlr规则，然后基于这些规则实现自己的语义，在学习算法篇之前，我就已经有了“如果哪天有需要，我可以徒手写一个编程语言解释器”的自信，之后虽然算法学起来也很吃力，但是不会因为陷进去而感到慌张，**因为已经对解释器有了全局的把握。**

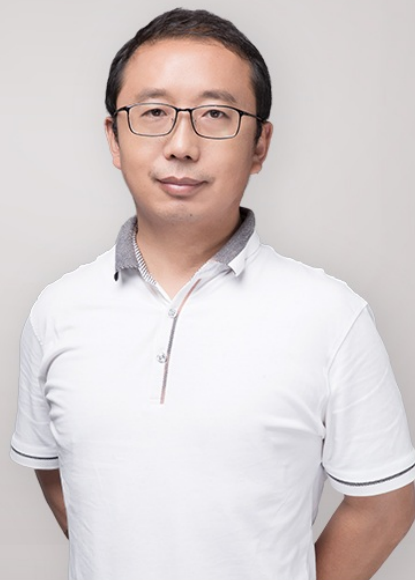
一年前，我也像很多人一样，觉得编译原理是没有用的屠龙技，后来，我越发感觉编译原理在工作和学习中无处不在，比如Java程序员都会深入学习的JVM，不懂编译很多概念就只是听听，完全不理解。当你需要深入研究某个DB的时候，SQL解析优化器也绕不过去的一个坎，只有懂编译才能搞明白。所以，我庆幸自己能够接触到宫老师的《编译原理之美》，也感谢老师的良苦用心，**我会一直坚持学下去，趁年轻，趁热爱，趁一起都还来得及。**

编译原理之美

手把手教你实现一个编译器

宫文学

北京物演科技CEO



新版升级：点击「 请朋友读」，20位好友免费读，邀请订阅更有**现金**奖励。