作者：代兴　邮箱：503268771@qq.com　Vx 公众号：东方架构师
https://github.com/drink-cat/Book_Program-Principles

汇编语言和程序原理很抽象，学习过程枯燥，需要读者和 CPU、指令、寄存器、内存打交道。
同时，关联的技术很多，除了编程语言之外，还包括 CPU、内存、操作系统、编译原理、运行原理等。
这些原因导致汇编语言的学习成本很高。根据自己的学习过程，我总结了一些学习技巧。

技巧：坚持手写汇编
熟能生巧，是一个真理。
在学习的前期阶段，会感觉到手写汇编很累，这时就需要坚持。强迫自己手写大量的汇编代码，把每个知识点都手写很多遍，过几周或几个月再来手写一遍。
写好汇编代码，还需要编译成可执行文件，然后运行程序，查看执行结果是否符合预期。
使用 objdump、readelf 查看程序的内部细节，和手写的汇编代码做对比，加深理解。
经历高强度的练习，你会发现汇编指令不再陌生，汇编程序有套路，信手拈来。

技巧：看书，不要看视频
人的思维过程，很缓慢。
视频给出的信息太快，还有各种娱乐诱惑，导致注意力不集中，学习效率低下，所以不建议看视频。
准备安静的环境，慢慢看书，把每个知识点都手写几遍，慢慢思考。
看书时，多做练习，遇到疑问，及时查资料。
抽出整块时间，比如工作日的晚上、周末的下午。思考过程需要时间积累，下定决心付出时间。
推荐几本重要的书：《汇编语言程序设计(ATT 语法)》，《C 语言程序设计》，《linux 高级环境编程》，《程序员的自我修养(链接、装载和库)》。

技巧：持之以恒，长期目标
底层技术没有速成的方法。
越是难的技术，越需要作为长期目标，小步慢走，点滴积累。
急功近利使不得。如果规划的学习阶段比较短，遇到困难或者学的很累，看不到学习效果，就会感到迷茫，进而可能放弃。
长期目标，建议以 0.5 年为单位。先用 0.5 年熟悉汇编语言，熟练手写汇编代码。再用 0.5 年掌握关联的汇编知识，比如 linux、ELF 中的汇编。之后用 0.5 年融会贯通，把很难的知识点摸清楚，比如内存模型、动态库中的汇编。

技巧：举一反三，多查资料
一个知识点，和多个知识点有关联，顺藤摸瓜，学习一遍。
遇到关联的知识点，一定要查资料。有时查资料也找不到答案，可以先放放，可能过 2 个月又找到答案了。
知识点是网状结构，触类旁通，举一反三，积累自己的知识库。别人总结的知识，经过思考、验证、消化、总结，才能变成自己的知识。
比如，cmpxchg 指令是 CAS 操作的基础，CAS 操作又是 linux Mutex、Java AQS、Atomic 的基础。那么，Mutex、AQS、Atomic 怎么使用 CAS/cmpxchg 呢？
翻阅 linux 源码、Java 源码，再结合测试代码，终于知道，cmpxchg 指令实现内存整数值的原子操作、互斥操作，上层还需要自旋、队列、线程调度。
可见，知识有层次性、相关性。