Broadview
WWW.BROADVIEW.COM.CN

编程高手箴言

梁肇新 编著

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

编程高手箴言

梁肇新　编著

電子工業出版社
Publishing House of Electronics Industry
北京·BEIJING

内 容 简 介

本书是作者十余年编程生涯中的技术和经验的总结。内容涵盖了从认识 CPU、Windows 运行机理、编程语言的运行机理，到代码的规范和风格、分析方法、调试方法和内核优化，内有作者对许多问题的认知过程和透彻的分析，以及优秀和精彩的编程经验。

本书将为广大程序员打下牢固的知识基础，适用于立志成为编程高手的广大程序员，是一本不可多得的国内编程高手的鼎力之作。

图书在版编目（CIP）数据

编程高手箴言 / 梁肇新编著. —北京：电子工业出版社，2003.10
ISBN 7-5053-9141-0
Ⅰ.编... Ⅱ.梁... Ⅲ.程序设计 Ⅳ.TP311.1
中国版本图书馆 CIP 数据核字（2003）第 080528 号

责任编辑：朱沐红 zsh@phei.com.cn
印　　刷：北京民族印刷厂
出版发行：电子工业出版社 http://www.phei.com.cn
北京市海淀区万寿路 173 信箱 邮编：100036
经　　销：各地新华书店
开　　本：787×980 1/16 印张：27 字数：453 千字 附光盘 1 张
版　　次：2003 年 10 月第 1 版 2003 年 10 月第 1 次印刷
印　　数：8 000 册 . 定价：50.00 元（附光盘）

凡购买电子工业出版社的图书，如有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系。联系电话：(010)68279077。质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权请发邮件至 dbqq@phei.com.cn。

本书精彩导读

- 如何达到无我之化境的程序员高手思维境界。
- CPU 的工作机制可以让你无需操作系统就可验证机器的 4GB 内存空间，可用于改写成内存检测工具。
- 通过内核的驱动方式令 CPU 降温及实现的源代码。
- Windows 的线程真的会泄漏内存吗？还是人云亦云。
- PE 格式解析，如何不让别人看到你的 dll 的函数名字。
- 语言的调用约定是怎么回事？非得从 main 开始执行吗？stdcall 及 cdecl 的不同之处。编写不从 main 开始的程序。
- 函数挂钩及类接口挂钩方法，DirectX 挂钩技术。
- 如何才能写出像诗一样的代码？代码规范化的原则及成对编码方法。
- 软件开发项目的进行方法及分析方法。如何进行开放性思维思考问题，以及进行项目的分解？
- 游戏修改软件的开发技术，以及开发过程实例。
- 什么时候才需要编写 dll？什么时候才需要对象化？并非 C++就一定是最好的。
- 主干及分支分析方法。
- 编写软件 RAID 硬盘阵列。
- 超级解霸软件体系结构。
- 软件是调试出来的，不是编写出来的。调试的方法及调试分析。
- MMX 指令及优化方法。
- 图像淡出淡入效果的实现以及用 MMX 进行优化。

序

有人说，“编程是一种艺术”，这句话的意思一方面是说，编程技巧像艺术技巧一样，深不可测、奥妙无穷；另一方面是说，程序员像艺术家一样，也有发挥创造性的无限空间。

在传统产业中，比如，一个人操作一部机床，他所需掌握的技术是有限的、几乎不变的，而他的创造性的发挥也基本上受到机床能力的局限。但是搞艺术就不同了。比如，拉小提琴的技巧，从“会拉”到“大师”，其水平相差何止十万八千里！同一架小提琴，初学者拉起来会令人生厌，而大师演奏出来却能使听众陶醉、痴迷。

当然，将程序员与艺术家相比也不完全恰当，而一个软件企业也需要各种层次的软件人才的组合。比如所谓“软件蓝领”，就是指在软件项目中从事编写代码或测试这类工作的，他们的工作性质与其说像艺术家，不如说更像传统产业工人。所以，这里将程序员与艺术家相比，更多的是指所谓“软件白领”。不过，“软件蓝领”如果能提高自己的编程水平，也对做好工作大有帮助，何况，“软件蓝领”是可以向“软件白领”发展的。

现在都说人才重要，那么对软件来说，人才更是无比重要。如果说我们的某些传统产业落后于人可以归结为装备落后的话，那么，我们的软件产业落后于人只能归结为人才了。因为在今天，我们和发达国家的软件人员所使用的硬件和软件装备基本上是一样的。

不管怎样，一个软件企业需要有一些高水平的程序员来做架构设计、系统分析或进行重要部分的编码、调试等工作。他们的水平、他们的创新，对于软件企业至关重要。记得当比尔·盖茨被问及：如果只允许他在全部金钱和最有才华的20个雇员之间做出选择的话，他将选择哪个？比尔·盖茨毫不犹豫地选择后者。他说：“我不在乎钱”！这说明在盖茨心目中，这20个雇员（显然很大一部分是顶级程序员）是微软公司最宝贵的财富。

由此可见，中国软件产业能否赶上国际水平，关键在于中国是否能够拥有一大批具有国际先进水平的软件人才。

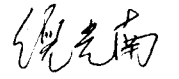
现在，随着我国软件产业的高速发展，每年吸引着数以十万计的相关专业的毕业生和众多爱好者加入到这片热土中来。但另一方面，很多企业还是感到饥渴，深感合格的软件人才还是不多，他们迫切地呼唤具有丰富工作经验，技术扎实精深的高层次

复合型人才。这种人才的成长需要一定的环境、一定的时间以及一定的实践磨炼。他们也很希望得到高手的指点，从高手丰富的实践经验中吸取营养、迅速成长。

本书作者梁肇新是国内为数不多具有十多年编程经验，而依然战斗在软件开发第一线，且依然对其热情执著的编程高手之一。梁肇新将自己厚积薄发的编程经验集结成书，相信对广大程序员大有裨益。本书通篇没有时髦的 IT 新名词或新思想，而是踏踏实实地对很多知识进行了深刻的剖析，这有助于为编程打下坚实的根基。只有这样，才能在飞速变化的软件领域免于雾里看花，才能更快更深地认识许多新问题、新知识，也才能更从容地应对未来之挑战。

千里之行，始于足下。希望本书的出版有助于每位程序员走好坚实的每一步。

中国工程院院士



2003 年 8 月 12 日

前　言

我的这一本书很难写，因为试图在这本书中把我所了解的东西和盘托出，但是，其中的每一个部分其实都可以单独成书。由于本书的目的是让那些彷徨于如何提高自己而不得其法者以最大的启迪，所以，我认为必须把握的内容都有必要提及。

本书是以先讲基本知识，然后用实例来开悟读者的，可能基本知识中的内容有些枯燥，这也是我的第一本书一开始不知如何下手的缘故。由于无经验，所以一开始方法也有一些不当，从而把精彩的内容略去了，在这本书即将完成之际已经无法再补上。如果会再版的话，将会更加精彩有趣。

此书的精要在于悟，即在真实的实验之中进行领悟。必须提到的是，在此书的创作中起关键作用的易虎刚从 2001 年开始一直努力协助，同时以亲身的进步来证明我所讲的内容和方法，并且在现实的工作中得到提高。

此书写作之时正值 IT 业狂躁之际，所以书中一再强调耐心，强调方法，所以很难写。难写之处在于思考成分多，实例讲解少，而计算机书籍中以实例讲解，不需要思考的成分多，即一本不需要动什么脑筋的书最容易写。像这本几乎是挖空心思的书最难写。

高手在我看来是可以达到对从 CPU 本身到操作系统内核、到系统平台、到应用软件体系，到软件的具体工作都有着深层次的掌握者。可以达到出神入化境界者谓之高手。

我在书中试图把十几年来的精髓写出来，但可能会存在书不达意之处，请众高手笑纳。

计算机技术的本质是什么？它是一个精心积累人的思维智慧的机器。现在技术越积越多，功能也就越来越强大，智能化也就越来越无所不在。千万不要以为软件无非如此，现在还没有到达它发展的登峰造极的时候。

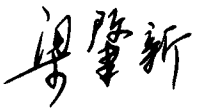
有人认为现在是 Java 及.Net 的时代，有谁还需要 C 及汇编呢？要知道，Java 及.Net 是建立在软件之上的，是为了垄断市场而建立的体系，犹如挖好一个金碧辉煌的坑让你往里跳，还自以为站在巨人的肩膀上，事实上成了坑底之蛙。所以，我认为要作为高手，就必须从机器本身出发，从 CPU 到操作系统、再到软件体系。当你做到这一点时，你会发现自己也可以建立一整套体系，如可以为 Ne(tja)va 之类。

在本书中，从 CPU 入手到 Windows 体系，到接口方式，再到代码规范化；从分

析方法，到优化的技巧，逐步深入，畅游计算机关键技术，用真实的例子来深入引领读者的思维方式。以期每一位认真的程序员可以提升自己的功力，渐入化境。

这可能是我惟一的一本以讲解思维方式的书，以后不会有这种方式，所以不管你认为它是好是坏，但这是我的真实体会，这种以“说”为主的书确实太难！

以后我还会出一些书，但将以某一个技术的深入为主，即以“代码”为主，以“说”为辅的方式。一方面这种书好写，另一方面技术的专业化也不太可能包揽计算机技术的内外一切。以后的书将全在此书的基础上进行，即在这里讲过的东西将不再重复。但不管是什么内容，基本的原则是不会变的：（1）成对编码原则；（2）代码规范化原则。



目　录

读书笔记

第1章 程 序 点 滴

好的开始是成功的一半，本书首先会试图告诉你什么是程序员？为什么要做这样的程序？正确的入门方法是什么？

程序员只有在理解了以上内容的基础上，才能进一步更快地提高自身技能，这时候再开始程序的设计。其实，对一个软件的开发者来说，真正重要的不在于这行代码怎么写，那些代码应该怎么写，关键是思路的问题，而思路事实上是经验的积累。经验是使你从最初的封闭的思维方式，到最后形成开放式的思维的一个过程。将我在十几年程序生涯中获得的一些经验告诉读者，使大家少走弯路，这也是我想写这本书的主要目的。

1.1 程序≠软件

现在很多人以为程序就是软件，软件就是程序。事实上，软件和程序在20世纪80年代时，还可以说是等同的，或者说，在非PC领域里它们可能还会是等同的。比如说某个嵌入式软件领域，软件和程序可能是等同的。但是，在PC这个领域内，现在的程序已不等于软件了。这是什么意思呢？

1. 软件发展简述

在20世纪80年代的时候，PC刚诞生，那时国内还没有几个人会写程序。那么，如果你写个程序，别人就可以拿来用。那时候的程序就能产生价值，那个程序就直接等同于软件。

但软件行业发展到现在，这里以中国的情况为例（美国在20世纪80年代，程序已经不等同于软件了），程序也不等同于软件了。因为现在写程序很容易，但是你的这个程序很难产生什么样的商业意义，也不能产生什么价值，这就很难直接变成软件。要使一个程序直接变成软件，中间就面临着很高的门槛问题。这个门槛问题来自于整个行业的形成。

现在，你写了一个程序以后，要面临商业化的过程。你要宣传，你要让用户知道，你要建立经销渠道，可能你还要花很多的时间去说服别人用你的东西。这是程序到软件的一个过程。这门槛已比较高了。

我们在和国内的大经销商的销售渠道的人聊天时，他们的老板说，这

几年做软件的门槛挺高的，如果你没有五六百万元做软件，那是“玩”不起来的。我说：“你们就使门槛很高了。”他说：“那肯定是的。如果你写个“烂”程序，明天你倒闭了，你的东西还占了我的库房，我还不知道找谁退去呢。我的库房是要钱的呀！现在的软件又是那么多！”

所以，如果你没有一定的资产的话，经销商都不理你。实际情况也是这样的，如果你的公司比较小，且没什么名气，你的产品放到经销商库房，那么他最多给你暂收，产品销不动的话，一般两周绝对会退货。因为现在经销商可选择的余地已很多了，所谓的软件也已经很多了。而程序则更多，程序都想变成软件，谁都说自己的是“金子”。但只有经受住用户的检验，才能成为真正的“金子”。

这就是美国为什么在 20 世纪 90 年代几乎没有什么新的软件公司产生的原因。只是原来 80 年代的大的软件公司互相兼并，我吞你，你吃我。但是，写程序的人很多，美国的程序变软件的门槛可能比我们还高，所以很多人写了程序就丢在网上，就形成了共享软件。

2. 共享软件

共享软件是避开商业渠道的一种方法。它避开了商业的门槛，因为这个行业的门槛发展很高以后就轻易进不去了。我写个程序丢在网上，你下载就可以用，这时候程序又等于软件。共享软件是这样产生的，是因为没有办法中的办法。如果说程序直接等于软件的话，谁也不会轻易把程序丢到网上去。

开始做共享软件的人并不认为做它能赚钱，只是后来用的人多了，有人付钱给他了。共享软件使得程序和软件的距离缩短了，但是它与商业软件的距离会进一步拉大。商业软件的功能和所要达到的目标就不是一个人能“玩”得起来的了。这时的软件也已不是几个人、一个小组就能做出来的了。这就是在美国新的软件公司没法产生的原因。比如 Netscape 网景是在 1995～1996 年产生的新软件公司，但是，两三年后它就不见了。

1.1.1 商业软件门槛的形成

1. 商业软件门槛的形成

商业软件门槛的形成是整个行业发展的必然结果。任何一个行业初始阶段时的门槛都非常低，但是，只要发展到一定的阶段后，它的门槛就必然抬高。比如，现在国内生产小汽车很困难，但在 20 世纪 50 年代～60 年代的时候，你装 4 个轮子，再加上柴油机等就形成汽车。那时的莱特兄弟装个螺

读书笔记

旋桨，加两个机翼，就能做飞机。整个行业还没有形成的时候，绝对可以这样做，但是，到整个行业形成时，你就做不了了。所有的行业都是这样的。

为什么网站一出来时那么多人去挤着做？这也是因为一开始的时候，看起来门槛非常低，人人都可以做。只要有一个服务器，架根网线，就能做网站。这个行业处于初始阶段时，情况就是这样的。但这个行业形成后，你就轻易地“玩”不了了。

国内的软件发展也是如此。国内的软件自从软件经销商形成以后，这个行业才真正地形成。有没有一个渠道是判断一个行业是否形成的很重要的环节。任何一个行业都会有一个经销渠道，如果渠道形成了，那么这个行业也就形成了。第一名的经销商是 1994 年～1995 年成立的，也就是说，中国软件行业大概也就是在 1995 年形成的，至今才经历 8 年时间的发展。

有一种浮躁的思想认为，中国软件产业应该很快就能赶上美国。美国软件行业是 20 世纪 80 年代形成的，到现在已经发展了 20 多年了。中国软件行业才 8 年，8 岁才是一个懵懂的小孩，20 多岁是一个强壮的青年，那么他们的力量是不对等的。但也要看到，当 8 岁变成 15 岁的时候，它真正的能量才会反映出来。

2. 软件门槛对程序员的影响

现在中国软件行业正在形成。所以，现在做一个程序员一定要有耐心，因为现在已经不等于以前了。你一定要把所有的问题搞清楚，然后再去做程序。

对于程序员来说，最好的工作环境是在现有的或者初始要成立的公司里面，这是最容易成功的。个人单枪匹马闯天下已经很困难了。即使现在偶尔做两个共享软件放在网上能成名，但是也已经比较困难了。因为现在做软件的人已经很多了。这也说明软件已经不等于程序了，程序也不等于软件。

程序要变成软件，这中间是一个商业化的过程。没有门槛以前，它没有这个商业过程，现在有这个行业了，它中间就有商业化的过程。这个商业化的过程就不是一个人能“玩”的。

如果你开始做某一类软件的时候，别人已经做成了，这时你再决定花力气去做，那么你就要花双倍的力气去赶上别人。

现在的商业软件往往是由很多模块组成的，模块是整个系统的一部分。个人要完整地写一个商业系统几乎是不可能的。软件进入 Windows 平台后，它已经很复杂了，不像在 DOS 的时候，你写两行程序就能卖，做个 ZIP 也能卖。事实上，美国的商业编译器也不是一个人能“玩”的。现在

你可能觉得它是很简单的，甚至 Linux 还带了一个 GCC，且源程序还在。你可以把它改一改，做个 VC 试一试，看它会有人用吗？它能变成软件吗？即使你再做个界面，它也还是一个 GCC，绝对不会成为 Visual C++那样能商业化的软件。

可见，国外软件行业的门槛要比中国的高很多了。我觉得我们中国即使再去做这样的东西，也没有多大的意义了。这个门槛你是追不过来的。不仅要花双倍的力气，而且在这么短的时间内，你还要完成别人已经完成过的工作，包括别人所做的测试工作。只有这样，才能做到你的软件与别人有竞争力，能与它做比较。

1.1.2 认清自己的发展

如果连以上认识都不清楚，很可能就以为去书店买一本 MFC 高手速成之类的书，编两个程序就能成为软件高手。就好像这些书是“黄金”，我学两下，学会了 VC、MFC，就能做一个软件拿出去卖了。这种想法也不是不行，最后一定能行，但要有耐心，还要有机遇。机遇是从耐心中产生的，越有耐心，就越有机遇。你得非常努力，要花很多的精力，可能还要走很多的弯路。

如果你是从 MFC 入手的，或是从 VB 入手的，则如要做出一个真正的能应用个人领域的通用软件，就会走非常多的弯路。直接的捷径绝对不是走这两条路。这两条路看起来很快，而且在很多公司里面确实需要这样的东西，比如说我这家公司就是为另一个家公司做系统集成的，那我就需要这样的东西，我不管你具体怎么实现，我只需要达到这个目标就行了。

任何软件的实现都会有 *n* 种方法，即使你是用最差的那种方法实现的，也没有问题，最后它还是能运行。即使有问题，再改一改就是。但是，做通用软件就不行了，通用是一对多，你做出来的软件以后要面向全国，如果将来自由贸易通到香港也好，通到国外也好，整个产品能销到全世界的话，这时候，通用软件所有做的工作就不是这么简单了。所以说，正确的入门方法就很关键。

如果你仅仅只是想混口饭吃，找个工作，可能教你成为 MFC 的高手之类的书对你就足够了。但是，如果你想做一个很好的软件，不仅能满足你谋一碗饭吃，还能使你扬名，最后你的软件还能成为很多人用，甚至你还想把它作为一个事业去经营，那么这第一步就非常关键。这时就绝对不能找一本 MFC 或找一本 VB 的书学两下就行，而是要从最低层开始做起，从最基本做起。