软件工程研究院

# 高质量程序设计指南
## ——C++/C语言

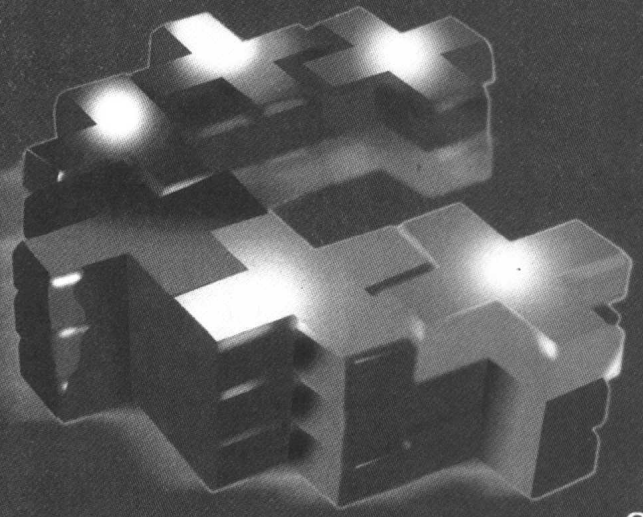
第3版

林 锐 韩永泉 编著
飞思科技产品研发中心 监制

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

软件工程研究院

# 高质量程序设计指南
## ——C++/C语言
## （第3版）



林　锐　韩永泉　编著
飞思科技产品研发中心　监制

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

## 内容简介

高质量程序设计是软件行业的薄弱环节，大部分企业只能依靠大量的测试和改错来提高软件产品的质量，为此付出了高昂的代价。因此，如何让程序员熟练地掌握编程技术和编程规范，在开发过程中内建高质量代码，是IT企业面临的主要挑战之一。

本书以轻松幽默的笔调向读者论述了高质量软件开发方法与C++/C编程规范。它是作者多年从事软件开发工作的经验总结。本书共17章，第1章到第4章重点介绍软件质量和基本的程序设计方法；第5章到第16章重点阐述C++/C编程风格、面向对象程序设计方法和一些技术专题；第17章阐述STL的原理和使用方法。

本书第1版和第2版部分章节曾经在Internet上广泛流传，被国内IT企业的不少软件开发人员采用。本书的附录C《大学十年》是作者在网上发表的一个短篇传记，文中所描述的充满激情的学习和生活态度，感染了大批莘莘学子。

本书的主要读者对象是IT企业的程序员和项目经理，以及大专院校的本科生和研究生。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

**图书在版编目（CIP）数据**

高质量程序设计指南：C++/C语言 / 林锐，韩永泉编著.3版.—北京：电子工业出版社，2007.5
（软件工程研究院）
ISBN 978-7-121-04114-3

Ⅰ.高… Ⅱ.①林…②韩… Ⅲ.C语言－程序设计 Ⅳ.TP312

中国版本图书馆CIP数据核字（2007）第038798号

责任编辑：赵红梅
印　　刷：北京智力达印刷有限公司
装　　订：北京中新伟业印刷有限公司
出版发行：电子工业出版社
　　北京海淀区万寿路173信箱　邮编：100036
开　　本：787×1092　1/16　印张：25.75　字数：659.2千字
印　　次：2007年5月第1次印刷
印　　数：6 000册　定价：39.80元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。
质量投诉请发邮件至zlts@phei.com.cn，盗版侵权举报请发邮件至dbqq@phei.com.cn。
服务热线：（010）88258888。

## 作者简介

~林 锐~

1973年生。1990年至1996年，就读于西安电子科技大学，获硕士学位。1997年至2000年，就读于浙江大学计算机系，获博士学位。大学期间两度被评为中国百名跨世纪优秀大学生，1996年获电子工业部科技进步二等奖，1997年获首届中国大学生电脑大赛软件展示一等奖。2000年7月加入上海贝尔有限公司，从事软件工程和CMM的研究推广工作，2003年7月当选为 Alcatel 集团软件工程专家。2004年初创建上海漫索计算机科技有限公司（http://www.chinaspis.com），致力于创作适合国内企业需求的软件研发管理解决方案，包括方法论和软件产品。工作期间出版著作7部。

## 作者简介

**韩永泉**

1975年生。1994年至2001年就读于西安电子科技大学计算机系，获硕士学位。2001年4月加入上海大唐移动通信技术有限公司，担任高级软件工程师，从事电信设备网管软件的设计和开发。2004年加入北京新岸线软件科技有限公司，从事数字电视相关软件产品的设计、开发和管理工作，曾负责所在公司与上海漫索计算机科技有限公司合作开展的软件过程改进和研发管理解决方案的实施工作，现为公司软件项目经理。他是面向对象和面向组件软件开发技术及编程技术的爱好者。

此为试读，需要完整PDF请访问：www.ertongbook.com

# 第3版

大约在2005年年初，本书第2版（和第1版）已经售完。至今本书仍然受到软件公司和C++程序员的关注，不断有读者询问我从何处可以买到本书、什么时候再版。

说来惭愧，我从2002年写完本书第1版后，再也没有接触过C++编程，现在对C++已经很陌生了。2004年1月我离开上海贝尔，创办了上海漫索计算机科技有限公司，专注于IT企业的研发管理整体解决方案（包括软件产品和咨询服务）。我自己已经从技术专家转型为企业管理者，关注商务多于软件技术。对于出版本书第3版，我的确心有余而力不足。幸好第2版的作者韩永泉仍然从事应用软件开发，宝刀未老，他全面操办了第3版，我只是挂名而已。

在撰写第3版的时候，为了更进一步突出本书一贯强调的“高质量程序设计”理念，对原书第2版的内容做了一些调整：

首先是对第2版进行了全面的修订，改正了所有已经发现的错误，并对原有部分章节的内容进行了补充；

其次，删除了第2版的第2章和第17章（名字空间和模板）。根据我们的观察，除非是开发类库等通用程序，第17章的内容在现阶段对应用软件开发人员一般不具有实际指导价值；

最后，增加了大约10个小节的内容，分散在各章中。这些增加的内容是实际应用软件开发过程中经常会用到的技术，可以显著地提高编程效率，增强软件的健壮性和可移植性。

不论本书第1版和第2版是好是差，它都被过度地使用了，产生了令作者始料不及的影响。本书的试题被国内软件公司大面积地用于C++程序员招聘考试，结果事先看过答案的应试者考了高分而被录取，还真有人向我致谢；也有不少人未看过答案而考了低分未被录取，在网上把作者骂一通。本书的试题和答案早在2002年就公开了，不知有多少人看过，我很奇怪怎么到现在还被煞有介事地用于考试。

本书第3版即将出版，我希望读者正确地使用本书：请您学习和应用您（或公司）认为好的东西，不要把本书当做标准来看待，不要全部照搬，也不必花费很多时间去争议本书是好还是坏。如果您发现书中的错误或不妥之处，请及时告知作者韩永泉，或发邮件至northwest_wolf@sina.com，或直接上他的Blog与他交流：http://blog.csdn.net/northwest_wolf/。

林锐

2007年1月

上海漫索计算机科技有限公司
http://www.mansuo.com
linrui@mansuo.com

# 第2版

《高质量程序设计指南——C++/C 语言》第 1 版上市后，一度成为畅销书。网上评论甚多，褒贬参半。我们分析了读者的批评和建议，总结了本书第 1 版的主要不足：

由于第 1 版原本是企业的培训教材，初衷是为了帮助程序员提高程序质量，假设读者已经熟悉 C++/C 语法，所以内容薄而精练、前后章节不连贯，看起来更像专题讲座。出版社的宣传工作做得很好，本书吸引了很多 C++初学者和高级程序员。由于书中不讲解入门知识，导致很多初学者看不下去。有一些大学生听了我的讲座后，为表敬意特地买书让我签名，翻阅之后就塞进书架当做纪念品了。对于那些高级程序员而言，本书的大部分内容他们早已经熟悉，好不容易看到几处精彩的章节，却翻了几页就没有了，真是不过瘾啊。

我的研究专长是软件工程和企业研发管理，而非程序设计。在 C++/C 编程方面自己仅仅是一名老工匠而已，我的确没有时间没有能力写出让初学者和高级程序员都喜欢的 C++/C 书籍。这本书炒作得过火，让我骑虎难下了。从 2002 年 11 月起，我就开始物色真正的 C++/C 高手来写本书的第 2 版。

恰好上海大唐移动通信设备有限公司的韩永泉正在为公司写 C++/C 培训教材，他也是本书第 1 版的读者。韩永泉提出了很有价值的建议和批评，并把他写的培训教材发给我看，真是自投罗网啊！

韩永泉是在西安电子科技大学计算机系读的本科和硕士，内功扎实。我和他碰头交谈了 2 小时，就把第 2 版托付给他了。两个月后，他把第 2 版的书稿交给我审阅。第 2 版的内容比第 1 版多了一倍，其广度符合我的设想，其深度完全出乎我的意料。**为了阐述清楚 C++/C 程序之中的许多“为什么”，本书给出了大量的“提示、建议、规则”，并从编译器实现的角度论述原理。**这种深度非一般教科书能比，我用了一个月时间才审阅并且学习完毕，删除了几十页过分深奥的内容（免得让我自己看昏倒）。我相信第 2 版可以让大多数高级程序员看过瘾了。

网上有一些忌世愤俗者认为计算机领域的每个分支都已经有了世界名著，不具有世界顶尖水平的中国人再写类似书籍都是欺世盗名的行为。这种极度自负和极度自卑的心态导致他们专爱骂国内作者。如果中国作者的书籍中的技术错误被他们抓住，经过放大、推理、演绎之后基本上就能断定作者是卑鄙之徒，于是砖头就拍过来了（简称“拍砖”）。拍砖者们遥相呼应，很快就能拍出江湖豪情，被拍的作者就成了倒霉蛋。有位好心的读者怕我经受不起，特意发给我一本拍砖大法——《拍砖十二流》以增强内功。

网上自由漫骂既是网络价值的体现又是民主的体现，这是物质文明和精神文明发展到一定境界的产物。《高质量程序设计指南——C++/C 语言》第 2 版即将出版，作者忐忑不安地等待第二轮“拍砖”。

2003 年 2 月

上海贝尔阿尔卡特股份有限公司

# 第1版

软件质量是被许多程序员挂在嘴上而不是放在心上的东西！

除了完全外行和真正的编程高手外，初读本书，你最先的感受将是惊慌："哇！我以前捏造的C++/C程序怎么会有那么多的毛病？"

别难过，作者只不过比你早几年、多几次惊慌而已。

请花几小时认真阅读这本经书，你将会获益匪浅，这是前面$N$-1个读者的建议。

## 编程老手与高手的误区

自从计算机问世以来，程序设计就成了令人羡慕的职业，程序员在受人宠爱之后容易发展成为毛病特多却常能自我臭美的群体。

如今在Internet上流传的"真正"的程序员据说是这样的：

（1）真正的程序员没有进度表，只有讨好领导的马屁精才有进度表，真正的程序员会让领导提心吊胆。

（2）真正的程序员不写使用说明书，用户应当自己去猜想程序的功能。

（3）真正的程序员几乎不写代码的注释，如果注释很难写，它理所当然也很难读。

（4）真正的程序员不画流程图，原始人和文盲才会干这事。

（5）真正的程序员不看参考手册，新手和胆小鬼才会看。

（6）真正的程序员不写文档也不需要文档，只有看不懂程序的笨蛋才用文档。

（7）真正的程序员认为自己比用户更明白用户需要什么。

（8）真正的程序员不接受团队开发的理念，除非他自己是头头。

（9）真正的程序员的程序不会第一次就正确运行，但是他们愿意守着机器进行若干个30小时的调试改错。

（10）真正的程序员不会在上午9:00到下午5:00之间工作，如果你看到他在上午9:00工作，这表明他从昨晚一直干到现在。

……

具备上述特征越多，越显示程序员水平高，资格老。所以别奇怪，程序员的很多缺点竟然可以被当做优点来欣赏。就像在武侠小说中，那些独来独往、不受约束且带点邪气的高手最令人崇拜一样。我曾经也这样信奉，并且希望自己成为那样的"真正"的程序员，结果没有得到好下场。

我从读大学到博士毕业10年来一直勤奋好学，累计编写了数十万行C++/C代码。有这样的苦劳和疲劳，我应该称得上是编程老手了吧？

我开发的软件都与科研相关（集成电路CAD和3D图形学领域），动辄数万行程序，技术复杂，难度颇高。这些软件频频获奖，有一个软件获得首届中国大学生电脑大赛软件展示一等奖。在1995年开发的一套图形软件库到2000年还有人买。罗列出这些"业绩"，可以说明我算得上是编程高手了吧？

可惜这种个人感觉不等于事实。

读博期间我曾用一年时间开发了一个近 10 万行 C++代码的 3D 图形软件产品，我内心得意表面谦虚地向一位真正的软件高手请教。他虽然从未涉足过 3D 图形领域，却在几十分钟内指出该软件多处重大设计错误。让人感觉那套软件是用纸糊的华丽衣服，扯一下掉一块，戳一下破个洞。我目瞪口呆地意识到这套软件毫无实用价值，一年的心血白花了，并且害死了自己的软件公司。

人的顿悟通常发生在最心痛的时刻，在沮丧和心痛之后，我作了深刻反省，“面壁”半年，重新温习软件设计的基础知识。补修“内功”之后，又觉得腰板硬了起来。博士毕业前半年，我曾到微软中国研究院找工作，接受微软公司一位资深软件工程师的面试。他让我写函数 strcpy 的代码。

太容易了吧？

错！

这么一个小不点儿的函数，他从三个方面考查：

（1）编程风格；

（2）出错处理；

（3）算法复杂度分析（用于提高性能）。

在大学里从来没有人如此严格地考查过我的程序。我花了半小时，修改了数次，他还不尽满意，让我回家好好琢磨。我精神抖擞地进“考场”，大汗淋漓地出“考场”。这“高手”当得也太窝囊了。我又好好地反省了一次。

我把反省后的心得体会写成文章放在网上，引起了不少软件开发人员的共鸣。我因此有幸和国内大型 IT 企业如华为、上海贝尔、中兴等公司的同行们广泛交流。大家认为提高质量与生产率是软件工程要解决的核心问题。高质量程序设计是非常重要的环节，毕竟软件是靠编程来实现的。

我们心目中的老手们和高手们能否编写出高质量的程序来？

不见得都能！

就我的经历与阅历来看，国内大学的计算机教育根本就没有灌输高质量程序设计的观念，教师们和学生们也很少自觉关心软件的质量。勤奋好学的程序员长期在低质量的程序堆中滚爬，吃尽苦头之后才有一些心得体会，长进极慢，我就是一例。

现在国内 IT 企业拥有学士、硕士、博士文凭的软件开发人员比比皆是，但他们在接受大学教育时就“先天不足”，岂能一到企业就突然实现质的飞跃。试问有多少软件开发人员对正确性、健壮性、可靠性、性能、易用性、清晰性、可扩展性、安全性、兼容性、可移植性等质量属性了如指掌？并且能在实践中运用自如？“高质量”可不是干活小心点就能实现的！

我们有充分的理由疑虑：

（1）编程老手可能会长期用隐含错误的方式编程，习惯成自然后，被人指出发现毛病时都不愿相信那是真的！

（2）编程高手可以在某一领域写出极有水平的代码，但未必能从全局把握软件质量的方方面面。

事实证明如此。我到上海贝尔工作后，陆续面试或测试过近百名“新”、“老”程序员的编程技能，合格率低于 50%。很少有人能够写出完全符合质量要求的 if 语句，很多程

序员对指针、内存管理一知半解……

领导们不敢相信这是真的。我做过现场试验：有一次部门新进 14 名硕士生，在开欢迎会之前对他们进行“C++/C 编程技能”摸底考试。我问大家试题难不难？所有的人都回答不难。结果没有一个人及格，有半数人得零分。

竞争对手如华为、中兴、大唐等公司的朋友们也做过试验，也是类似的结果。真的不是我“心狠手辣”或者要求过高（甚至变态），而是很多软件开发人员对自己的要求不够高。要知道这些大公司的员工素质在国内 IT 企业中是比较位列前茅前列的，倘若他们的编程质量都如此差的话，我们怎么敢期望中小公司拿出高质量的软件呢？连程序都编不好，还谈什么振兴民族软件产业。

多年来，我在软件开发过程中的苦头吃得实在太多了，现在总算被折磨清醒了。我打算定义编程老手和编程高手，请您别见笑。

定义 1：能长期稳定地编写出高质量程序的程序员称为编程老手。

定义 2：能长期稳定地编写出高难度、高质量程序的程序员称为编程高手。

根据上述定义，马上得到第一推论：我既不是高手也算不上是老手。

在写此书前，我阅读了不少程序设计方面的英文著作，越看越羞惭。因为发现自己在编程基本技能方面都未能全面掌握，顶多算是二流水平，还好意思谈什么老手和高手。希望和我一样在国内土生土长的程序员朋友们能够做到：

（1）知错就改；

（2）经常温故而知新；

（3）坚持学习，天天向上。

2002 年 4 月

上海贝尔阿尔卡特股份有限公司

# 致 谢

《高质量程序设计指南——C++/C 语言》第 3 版终于完成了！第 2 版在 2003 年出版后，接着第 1 版的名声，仍然受到不少读者的青睐。但是，出版后发现了一些错误，不管这些错误是什么原因造成的，林锐博士都替我挨了不少“砖块”，这令我深感愧疚，深怕影响了林锐博士的好名声，更怕误人子弟，对热心读者产生误导。我觉得有责任将这些错误改正后予以发表，于是 2004 年年末，在诸多热心网友的指正下，整理出了一份勘误表并粘贴在了 CSDN 网站上，但是只有少数人知道并下载，现在偶尔还能收到热心网友的 E-mail 要求给他们发送一份勘误表。

这次应电子工业出版社之邀撰写本书第 3 版，说明本书还有很强的生命力，更重要的是：给了我一个全面纠正错误的机会，同时，这几年在工作中也有很多新的感受，我很想总结出来与大家分享。

在第 3 版完稿之际，特别要感谢为我指正错误的热心网友，他们是：方如坤、smart_sonny、wfeng、fecita 等。感谢电子工业出版社飞思公司的编辑和策划，给我匡正失误、续写新篇的机会，没有他们的辛勤工作，本书现在不会出现在你的手上；这次帮我审稿的还有我的朋友陈亚明，他看到了许多我没有看到的问题，同时他也是 C++编程技术的爱好者，有他帮我审定稿件，相信有助于降低本版的错误率，感谢他的辛勤劳动；当然还有林锐博士，虽然我已穷思竭力避免错误产生，但毕竟金无足赤，新的错误仍然在所难免，林锐博士不免又要挡些“板砖”了。哈哈！还是欢迎大家多提批评意见。当然，有谁想夸我们两句，本人与林锐博士也是很愿意听到的☺。第 2 版也有不少网友对我们提出表扬，在此一并表示深深的感谢！

还有一个重要的感谢对象，那就是正怀着我孩子的老婆。虽然我不太希望我的孩子将来也从事 IT 行业，但我还是想用这本书的出版来庆祝我们的宝贝——兜兜——来到这个世界！

韩永泉

2007 年 1 月

北京新岸线软件科技有限公司

Yongquan.Han@nufrontsoft.com



此为试读，需要完整PDF请访问：www.ertongbook.com

# Contents

# 第1章　高质量软件开发之道

本章讲述高质量软件开发的道理。

为了深入理解软件质量的概念，本章阐述了10个重要的软件质量因素，即正确性、健壮性、可靠性、性能、易用性、清晰性、安全性、可扩展性、兼容性和可移植性，并介绍了消除软件缺陷的基本方法。

人们开发软件产品的目的是赚钱。为了获得更多的利润，人们希望软件开发工作“做得好、做得快，并且少花钱”，所以软件质量并不是人们唯一关心的东西。本章论述了“质量、生产率和成本”之间的关系，并给出了能够“提高质量、提高生产率，并且降低成本”的软件开发方法。

## 1.1　软件质量基本概念

### 1.1.1　如何理解软件的质量

什么是质量？

词典的定义是：① 典型的或本质的特征；② 事物固有的或区别于其他事物的特征或本质；③ 优良或出色的程度。

CMM对质量的定义是：① 一个系统、组件或过程符合特定需求的程度；② 一个系统、组件或过程符合客户或用户的要求或期望的程度。

上述定义很抽象，软件开发人员看了准会一脸迷惘。软件的质量不容易说清楚，但我们今天非得把它搞个水落石出不可。

就以健康做类比吧。早先人们以为长得结实、饭量大就是健康，这显然是不科学的。现代人总是通过考察多方面的生理因素来判断是否健康，如测量身高、体重、心跳、血压、血液、体温等。如果上述因素都合格，那么表明这人是健康的。如果某个因素不合格，则表明此人在某个方面不健康，医生会对症下药。同理，我们也可以通过考察软件的质量属性来评价软件的质量，并给出提高软件质量的方法。

一提起软件的质量属性，人们首先想到的是“正确性”。“正确性”的确很重要，但运行正确的软件就是高质量的软件吗？不见得。

这个软件也许运行速度很低，并且浪费内存，甚至代码写得一塌糊涂，除了开发者本人谁也看不懂，也不会使用。可见正确性只是反映软件质量的一个因素而已。

不贪污的官就是好官吗？不见得。

时下老百姓对一些腐败的地方政府深恶痛绝，对“官”不再有质量期望。只要当官的不贪污，哪怕毫无政绩也算是“好官”了。过去有一个县官，在上级面前标榜自己的清廉：“我下去视察最多只喝老百姓的一碗水”，上级回敬他：“那我不如放

此为试读，需要完整PDF请访问：www.ertongbook.com