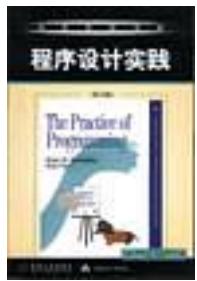


程序设计实践



[程序设计实践 下载链接1](#)

著者:Brian W. Kernighan

出版者:人民邮电出版社

出版时间:2016-1-1

装帧:平装

isbn:9787115407863

本书是计算机科学方面的经典名著，由计算机界极具影响力的两位专家Brian W. Kernighan和Rob Pike合著。书的内容围绕程序设计实践中的一系列问题展开，讲述对于程序员有共性的知识，以帮助各程序员写出更高效的程序。本书从排错、测试、性能、可移植性、设计、界面、风格和记法等方面，讨论了程序设计中既具有实际意义又具有广泛意义的思想、技术和方法。

本书值得每位梦想并努力成为程序员的人参考，值得每位计算机专业的学生和计算机工作者阅读，也适合作为程序设计高级课程的教材或参考书。

作者介绍:

作者:[美] 布莱恩 W. 克尼汉 (Brian W. Kernighan) 罗勃 · 派克 (Rob Pike) 译者:无

Brain Kernighan计算机科学家，曾与UNIX的缔造者Ken Thompson和Dennis Ritchie一起在贝尔实验室工作。他也是AWK和AMPL程序设计语言的共同作者。“K&R C”和“AWK”中的“K”都是指“Kernighan”。2000年起，他在普林斯顿大学计算机科学系任教授，并任本科部代表。

Rob Pike 软件工程师。他在贝尔实验室任职期间，作为UNIX小组成员参与开发了Plan

9和Inferno操作系统以及Limbo程序设计语言。目前他在Google公司工作，参与了Go和Sawzall程序设计语言的开发。

目录: 目录

Chapter 1: Style / 风格 1

1.1 Names / 名字 3

1.2 Expressions and Statements / 表达式和语句 6

1.3 Consistency and Idioms / 一致性和习惯用语 10

1.4 Function Macros / 函数宏 17

1.5 Magic Numbers / 幻数 19

1.6 Comments / 注释 23

1.7 Why Bother? / 为何要在风格方面费心 27

Chapter 2: Algorithms and Data Structures / 算法与数据结构 29

2.1 Searching / 检索 30

2.2 Sorting / 排序 32

2.3 Libraries / 库 34

2.4 A Java Quicksort / 一个Java快速排序实现 37

2.5 O-Notation / 大O记法 40

2.6 Growing Arrays / 自增长数组 41

2.7 Lists / 表 44

2.8 Trees / 树 50

2.9 Hash Tables / 散列表 55

2.10 Summary / 小结 58

Chapter 3: Design and Implementation / 设计与实现 61

3.1 The Markov Chain Algorithm / 马尔可夫链算法 62

3.2 Data Structure Alternatives / 在多种数据结构之间选择 64

3.3 Building the Data Structure in C / 使用C语言构建数据结构 65

3.4 Generating Output / 生成输出 69

3.5 Java 71

3.6 C++ 76

3.7 Awk and Perl / Awk和Perl 78

3.8 Performance / 性能 80

3.9 Lessons / 经验教训 82

Chapter 4: Interfaces / 接口 85

4.1 Comma-Separated Values / 逗号分隔值 86

4.2 A Prototype Library / 一个原型库 87

4.3 A Library for Others / 一个给他人用的库 91

4.4 A C++ Implementation / 一个C++实现 99

4.5 Interface Principles / 接口原则 103

4.6 Resource Management / 资源管理 106

4.7 Abort, Retry, Fail? 109

4.8 User Interfaces / 用户界面 113

Chapter 5: Debugging / 调试 117

5.1 Debuggers / 调试器 118

5.2 Good Clues, Easy Bugs / 线索明显、易于发现的错误 119

5.3 No Clues, Hard Bugs / 线索不明、难以发现的错误 123

5.4 Last Resorts / 最后的手段 127

5.5 Non-reproducible Bugs / 不可重现的错误 130

5.6 Debugging Tools / 调试工具 131

5.7 Other People's Bugs / 他人引入的错误 135

5.8 Summary / 小结 136

Chapter 6: Testing / 测试 139

6.1 Test as You Write the Code / 一边编码，一边测试 140

6.2 Systematic Testing / 系统化测试	145
6.3 Test Automation / 测试自动化	149
6.4 Test Scaffolds / 测试脚手架	151
6.5 Stress Tests / 压力测试	155
6.6 Tips for Testing / 测试心得	158
6.7 Who Does the Testing? / 谁来测试	159
6.8 Testing the Markov Program / 马尔可夫程序的测试	160
6.9 Summary / 小结	162
Chapter 7: Performance / 性能	165
7.1 A Bottleneck / 瓶颈	166
7.2 Timing and Profiling / 计时和剖析	171
7.3 Strategies for Speed / 加速策略	175
7.4 Tuning the Code / 代码调优	178
7.5 Space Efficiency / 空间利用率	182
7.6 Estimation / 评估	184
7.7 Summary / 小结	187
Chapter 8: Portability / 可移植性	189
8.1 Language / 语言	190
8.2 Headers and Libraries / 头文件和库	196
8.3 Program Organization / 程序架构	198
8.4 Isolation / 隔离	202
8.5 Data Exchange / 数据交换	203
8.6 Byte Order / 字节序	204
8.7 Portability and Upgrade / 可移植性和升级	207
8.8 Internationalization / 国际化	209
8.9 Summary / 小结	212
Chapter 9: Notation / 记法	215
9.1 Formatting Data / 数据格式化	216
9.2 Regular Expressions / 正则表达式	222
9.3 Programmable Tools / 可编程工具	228
9.4 Interpreters, Compilers, and Virtual Machines / 解释器、编译器和虚拟机	231
9.5 Programs that Write Programs / 写程序的程序	237
9.6 Using Macros to Generate Code / 用宏生成代码	240
9.7 Compiling on the Fly / 运行中编译	241
Epilogue / 后记	247
Appendix: Collected Rules / 规则汇编	249
· · · · · (收起)	

[程序设计实践](#) [下载链接1](#)

标签

程序设计

计算机

编程

计算机科学

软件开发

CS软件工程

计算机科学与编程

无index!

评论

养成良好的编程习惯, 学习优秀的编程手法。
本书可以说是涉及编程各阶段的技术与思路的引子,
可以按图索骥找具体的技术进行研究.

这本书或许会成为今年学过的最棒的CS书籍了 (第二或许是SCIP) 。全书虽然不厚
但包含了programming的方方面面。其中C2的Data Structure and Algorithm简洁明晰
C3和C4的Markov
Chain和CSV值得反复琢磨。不同层册的prgrammer可以领悟到不同内容。期待《All of
Programming》

Simplicity & Clarity; Generality; Evolution; Interface; Automation; Notation;

思想不多, 读不进去。

一般

这本书是作者多年编程经历总结出来的很多经验。涉及编程的很多方面，比如编程风格，数据结构，性能调优，debug等等。虽然涉及了这么多方面，但是每一方面都写得很到位，不会让人觉得浅尝辄止。这个也推荐作为编程的入门读物。

还好

之前发言过于不理性，现在重新来过，对之前的发言深表遗憾。与出版社工作人员沟通得知，这次封面设计不用原书封面，删减index是版权方要求所致。对index要求高的同学有准备就好了。仅此…… 3.17

[程序设计实践 下载链接1](#)

书评

我看过很多类似#程序员必读XX本书#的书单，似乎在几乎所有的推荐书单里面，《程序设计实践》（英文名The Practice of Programming）都没有出现作为一本被推荐的书。我也是在今年暑假突然想开始读一些英文版的技术书籍，然后豆瓣上大致浏览了一下，机缘巧合的买下了这本书。在...

这是一本牢牢占据经典书架最起眼位置的小册子，也是最符合KISS之道的神作之一。抛开已有的业界的best practices，这本书还修正了我记忆中的很多错误认识，Brian Kernighan、Rob Pike的大名自然是本书最好的招牌；但作者的求实态度更值得每一个认真的programmer学习。整体上...

英语不好，第一次买英文书，没信心，正好发现这个评注版，欣然买之。然后发现所谓的评注不是针对难点的解释，大都是没用的，该解释的作者都已解释清楚了，而这本书又是写思想的不是技术，没有什么难点，评注看的反而碍眼，影响书原汁原味的感觉，这本书40多块329页，而原版的才...

书非常好。但翻译却不能令人满意，就好像一个外行在翻译。

裘宗燕是计算机书籍翻译的前辈，也翻译了大量经典，但可能是由于翻译时年代久远，很多词汇都与现今流行的翻译方式不同。

另外这本书已经绝版了，所以只能找电子版来看。期待本书再版。

这里只举几个例子：本书的翻译...

首先这个书评和原书内容几乎没有关系，针对这本评注版。

当初选择买评注版的原因在于，它是英文版的，就这么简单。但是看完后发现这版有一两个说小不小，但是确实恼人的缺陷。

其一是校对问题。书中出现很多单词排版错误，即两个单词连到一块去了，而且从第三章开始几乎每章...

足足看了一整天才翻到了最后的后记。

昨天告诉朋友在读此书，朋友看了目录说这个大而全，感觉不到有东西在里面。

说说自己的感受吧。风格部分最简单，类似于代码大全中的一些。

算法，数据结构很简略，把树拿出来大概能吓跑一堆没经过学习的程序员。

马可夫链很好，立马去翻编...

推荐章节：第一，二，六，七章。

尤其是第一章，值得熟记并在以后的编程过程中加以运用。

其他章节可以以后再来看。

注意有些部分的技巧可能过时了，不过大部分地方还是值得参考。

书的最后还有各章节的要点汇总，方便查询。

你可曾……花费了大量时间编写错误的算法？使用了过于复杂的数据结构？

测试程序却忽略了明显的错误？需要让程序使用更少的内存，但运行快三倍？

纠结地在工作站和PC间移植程序？试图对他人的程序做适当的修改？

因无法理解程序而重写？这有趣吧？以上种种，程序...

同样是出自大师之手，同样都是珠玑，这本书的名气实在太小。这就像做生意一样，生意好的店会越来越好，生意差的店只会越来越冷清。

但是，没有看过这本书，是自己的损失。

这本书有三好：1，实践性强；2，实践性强；3，实践性强。例如：第四章，将库函数的界面，作者首先通过...

这本书都是一些经验介绍，很好，如C的尽量少用函数宏等，很值得一看，有经验的程序员可以发现你以前的一些习惯可能会有问题，自己体会吧。

被大多数人推崇的书是不会让人失望的，虽然书中的有的思想，在看之前就已经有在应用了，但是看了之后仍觉得收获很大。书本的内容并不算多，但是仍然能感觉到一个集大成者在程序设计中的严谨分析，作者引据论典，以经典的实例层层剖析，让我们在实践中强化理论，印证理论。毕业以...

个人认为比《设计模式》好，不论是否使用OO，都会有所收获。
最近又在重读，当思路遇到死胡同时，看一看往往会有新的想法。
文笔够精炼，译笔又够流畅。总体来说相得益彰，不过个人不太喜欢把interface翻译成“界面”。

本书涉及到编程的各个环节，结合这些书可以制定出一个软件研发的各环节checklist。
从团队的代码规范(命名,结构与组织)到设计阶段的要求，
再到后面开发/调试/优化阶段的技术方法。
软件研发人员可以按这个图谱去加强各方面的技艺。第一章命名部分，
值得学习与借鉴并写出规范。并...

作者不是个美国人吗？怎么成中国科学院高级工程师了？译者和作者分不清吗？？？？
？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？
？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？
？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？？

现在看完了前面的第一章，对于第一章中的style非常有感触。程序毕竟是给人看的，如果风格不好，就没与人愿意去读。当然风格好不好是仁者见仁，智者见智的事情。自己由于也看过一些见教你如何写出一些优美代码的书籍，书看多了，结果为了优美而优美，出现风格不一致的情况。这应...

推荐所有有一年编程以上的人看 真是句句血泪呀 不论语言不论平台 总有适合的部分
而且要做为备查的工具书 时不时看一次不够还要多次看 嗯 考虑是不是买一本了
图书馆的几天就被人recall回去了

简短，精炼却又全面的程序员必读，涉及编码的各个方面，建议每个有一年左右编程经验的人都读一下，可以拓展视野，纠正问题。其中的例子十分经典，易于理解又不乏实

用性。

就是最近流行的敏捷编程，很多概念这本书也体现得很清晰了。文档寓于代码之中。测试先行，渐进式开发和测试。甚至结对编程都有一个例子。

这本书从排错、测试、性能、可移植性、设计、界面、风格和记法等方面，讨论了程序设计中实际的、又是非常深刻和具有广泛意义的思想、技术和方法。

[程序设计实践 下载链接1](#)