

# 前言

我遇到的大多数专业程序员都不太愿意去解决算法设计问题。这点很遗憾，因为算法设计已成长为计算机科学的核心实用技术之一。若想设计出正确、高效和易于实现的算法去求解真实世界的问题，需要了解两种不同的知识体系：

- 技术——优秀的算法设计师懂好几种基本算法设计技术，包括数据结构、动态规划、深度优先搜索、回溯以及启发式方法。也许最重要的设计技术应该就是建模了，它能将杂乱现实世界中的应用问题提炼精化以便于用算法攻破，这可称得上是门艺术。
- 资源——优秀的算法设计师都站在巨人的肩膀上。他们不是每次都从一张白纸开始费尽心思最后创造出新算法来解决问题，他们会先弄清楚这个问题目前的研究现状。他们不是从零开始重新实现那些广为流传的算法，他们会去寻找现有的程序实现并以此作为起点。他们对许多经典算法问题都非常熟悉，这些问题为大多数应用问题的建模提供了充足的素材。

本书意在作为算法设计的一本指南，从而让学生和计算机从业人员能走进组合算法技术的殿堂。全书分为两卷：技术和资源。前者是对计算机算法设计和分析技术的一般性指引。后者则可让你进行查阅和参考，它是由多条简介构成的算法问题便览，<sup>1</sup> 其中每一条都包含了算法资料、程序实现以及大量的参考书目。

## 致读者

自从1997年本书第1版初次面世以来，它所受到的热烈欢迎让我一直倍感欣慰。这本书被视为一本独一无二的指南，它能教你用算法技术解决实际中的许多常见问题。不过，从本书十年前问世至今，这个世界有了许多改变。实际上，如果我们将现代算法设计和分析的起源定在1970年左右，那么从这本书诞生到现在的这个时间段在整个现代算法的发展历史中占了大约30%之多。

这本书有三个方面尤为受人钟爱：(1) 算法问题便览，(2) *War Story*，(3) 本书的电子资源部分。这些特色在新版中得到了保留并有所加强：

- 算法问题便览——由于找出一个算法问题的现有进展可能会是一项艰巨的任务，因此我提供了在实际中最重要的75个问题的简要介绍并汇集成“算法问题便览”。通过查阅这份便览，学生或从业人员可以很快地确定他们的问题名称和该问题的研究现状，以及如何利用现有工作的基础去解决它。为了帮你找到并确认问题，每条问题简介都包含了“处理前”和“处理后”的对比示意图，<sup>2</sup> 它们描述了输入和输出应符合

---

<sup>1</sup> 译者注：在本书中，如无特殊说明，便览通常指卷II中的算法问题便览，每条简介都是一个关于算法问题的词条。

<sup>2</sup> 译者注：这种示意图在算法问题便览中已经明确地标为“输入”和“输出”。

的规格。有位颇具洞察力的评论家因为这份便览非常强大的缘故，将我的这本书称作“算法世界搭车客手册”(*The Hitchhiker's Guide to Algorithms*)。<sup>1</sup>

算法问题便览是本书最重要的部分。为了在这一版中更好地改进它，我对每个问题都向相关的世界级权威专家征求了反馈。而且新版尤为关注对每个问题的现有软件实现讨论部分的更新。

- *War Story*——在实际中，算法问题不会出现在大项目的初始阶段。相反地，它们通常会以子问题形式出现，而此时你肯定能看到如下情景：要么程序员不知道该如何继续下去，要么目前的设计方案不足以完成该任务。

为了让你更好地去观察了解算法问题在实际中究竟是如何出现的，我们在书中纳入了一系列 *War Story*，或者可称作“我们与真实问题交战所经历的故事”。这些故事的寓意是，算法设计和分析不仅仅是理论，它更是在你真有需要的时候可以派上用场的一个重要工具。

这一版保留了上一版中的所有 *War Story* (适当进行了更新)，新增的 *War Story* 覆盖了外部排序、图算法和模拟退火以及一些其他主题。

- 电子资源部分——由于从事实际工作的人通常想找到一个程序而不光是算法，我们提供了链接让你可以获取那些可靠的算法实现(只要原站点仍有效)。同时我们也已将这些实现汇集在网站(<http://www.cs.sunysb.edu/~algorithm>)上，读者可以很方便地检索。本书初版刚问世之后没多久，我们这个站点就一直稳居Google上的“算法”网站排名首位。

此外，我们所提供的推荐建议能让你更容易地找出最适合于某项任务的代码。有了这些程序实现之后，算法设计的关键问题就变成了对你的应用问题正确地建模，而不是花时间去熟悉某个实际算法的细节。这个重要观点渗透于整本书之中。

我们在本书中没有讲到的内容也同样重要。我们不强调数学形式的算法分析，所以大多数的分析论证以较为直观的方式给出。你在这本书中找不到一个定理。如果需要了解更多的细节，读者应该去研究本书中所提到的程序和参考文献。一言以蔽之，这本指南的目标是让你尽可能快地走上正确的方向。

## 致教师

本书所涵盖的素材足以开设一门标准的“算法导论”课程。我们假设读者已学完相当于中级编程的课程(通常名叫“数据结构”或“计算机科学(II)”)。

你可在<http://www.algorist.com>下载用于讲授这门课程的一整套课程幻灯片。此外，我还基于上述幻灯片录制了在线音频和视频讲座，可供一个完整学期的算法课程使用。通过神奇的互联网，让我来帮你教这门课！

本书强调的是设计而非分析。它既适用于传统的课堂授课，也适应于新兴的“主动式学习”方法，也就是教授不再讲课而是去引导学生分组来解决实际问题。实际上，本书的 *War Story* 就是主动式学习方法的鲜活实例。

我在全书中做了很多教学法方面的改进。体现“教材导向”教学法的特征包括：

<sup>1</sup> 译者注：脱胎于科幻名著《银河系搭车客指南》(*The Hitchhiker's Guide to the Galaxy*)。

- 更轻松的讨论——本书卷I的讲解材料是上一版的两倍. 新增部分注重对基础材料给出更细致全面的阐述, 而并不是去增加更多的专题.
- 以错误开始——在算法教材中重要的算法通常会以既定事实的面貌出现, 而这样会掩盖其中的设计理念以及为何其他方案会失败的微妙原因. *War Story*通过一些应用问题来讲解藏于解决方案背后的风起云涌, 不过我还把这种讲授模式扩展到了经典算法设计的内容中.
- 停下来想想——这里会举例讲解我自己求解某个具体作业题时的思路演进——从错误开始一直到最后解决的全过程. 我把这类问题模块散布于整本教材中以提升读者对解题活动的积极性. 每个问题之后马上给出了解答.
- 更多且更精良的课后习题——这一版的习题是前一版的两倍. 那些在上一版中已被发现不易理解或模棱两可的习题已作改进或替换. 书中所有问题都标有难度等级(从1到10).
- 自我激励式考试设计——在我的算法课中, 我向学生保证所有的期中和期末考试题都直接取自本书的习题. 这样规定了一种“促学习型考试”(student-motivated exam), 因此学生完全明白应该如何学习才能在考试中有好的表现. 习题由我精心挑选而得, 可保证总数量、广度及难度都能满足这种考试的需求, 此外备选问题不会过少也不会过多.
- 所得教益——我们对这部分内容给出显著的阴影标记, 以强调这是你需要从对应章节中所汲取的大局观.
- 编程挑战赛的相关链接——每章的习题都包含三到五个“编程挑战赛”问题的链接, 它们来自<http://www.programming-challenges.com>. 这些链接可用于补充纸笔算法课程所缺乏的实际编程教学环节.
- 更多真代码, 更少伪代码——本书中的算法更多是以代码(用C语言写成)而不是伪代码形式出现. 我认为对于较简单的算法而言, 经过测试的真实程序实现不但正确而且可靠, 这可比那些形式化稍弱的表述<sup>1</sup>强多了. 完整实现可于<http://www.algorist.com>下载, 你可以深入学习研究这些程序.
- 章节注释——卷I的每章都包含一个简短的章节注释, 它们为读者指明了基本书目和补充参考文献.

## 致谢

十年后新书的题献会主要关注时间流逝中的世事变幻. 自第1版问世后, Renee已经成为了我的妻子, 然后又变成两个孩子(Bonnie和Abby)的母亲. 我的父亲离开了人世, 但我的母亲和兄弟(Len与Rob)仍然与我同在, 并且在我的生命中占据了非常重要的地位. 我将这本书献给我的家人, 不管是新成员还是老成员, 无论尚在或是离去.

---

<sup>1</sup> 译者注: 指的是以伪代码或普通语言形式描述的算法.

我要感谢以下人士, 他们为新版做出了卓越的贡献: Andrew Gaun和Betson Thomas提供了很多方面的帮助, 尤其是在构建新网站<http://www.cs.sunysb.edu/~algorithm>的基础架构和有关手稿准备的各类事项上. David Gries热心无私地为本书提供了很有价值的反馈. Himanshu Gupta和Bin Tang非常大胆地用这版的手稿在课堂讲授. 此外, 我还要感谢我在Springer出版社的编辑Wayne Wheeler和Allan Wylde.

我邀请了一批相关领域的算法大师审查了“算法世界搭车客手册”部分(即卷II)的那些章节, 他们与我分享了聪明才智, 并督促我不断进步. 感谢他们:

Ami Amir, Herve Bronnimann, Bernard Chazelle, Chris Chu, Scott Cotton, Yefim Dinitz, Komei Fukuda, Michael Goodrich, Lenny Heath, Cihat Imamoglu, Tao Jiang, David Karger, Giuseppe Liotta, Albert Mao, Silvano Martello, Catherine McGeoch, Kurt Mehlhorn, Scott A. Mitchell, Naceur Meskini, Gene Myers, Gonzalo Navarro, Stephen North, Joe O'Rourke, Mike Paterson, Theo Pavlidis, Seth Pettie, Michel Pocchiola, Bart Preneel, Tomasz Radzik, Edward Reingold, Frank Ruskey, Peter Sanders, Joao Setubal, Jonathan Shewchuk, Robert Skeel, Jens Stoye, Torsten Suel, Bruce Watson, and Uri Zwick.

很多习题都来源于我的同事或是受到其他教材的启发. 多年后要想再弄清它们来自哪里则是一项挑战, 不过每个问题的来源(我已尽全力重新搜集)都已发布在网站上.

若是不感谢为本书初版做出重要贡献的人也是很无礼的. Ricky Bradley和Dario Vlah建立了网站所需的坚实基础架构, 不但逻辑清晰而且易于扩展. Zhong Li用xfig绘制了卷II中的大部分算法示意图. Richard Crandall, Ron Danielson, Takis Metaxas, Dave Miller, Giri Narasimhan和Joe Zachary都审查过第1版的初稿——他们的反馈都是经过深思熟虑的, 也正是这些反馈使得本书不断完善, 最终呈现在读者面前.

我对算法所了解的很多东西都是和我的研究生在一起学到的. 他们中的很多人(Yaw-Ling Lin, Sundaram Gopalakrishnan, Ting Chen, Francine Evans, Harald Rau, Ricky Bradley和Dimitris Margaritis)都是相关*War Story*中的真实英雄. Estie Arkin, Michael Bender, Jie Gao和Joe Mitchell是我在石溪大学的朋友和算法领域的同事, 与他们共事与相处都很开心. 最后, 感谢我的朋友们, Michael Brochstein以及在纽约的其他友人, 在我离开石溪造访他们时, 共渡了许多美好时光.

## 说明

作者大度地接受对任何不足之处的批评乃是一个传统. 可我有所不同. 本书中所出现的任何错误、缺陷或问题都是他人的过失, 不过你要指出来我会非常感激, 这样我就可以去追究相关人士的责任了.

Steven S. Skiena

石溪大学(Stony Brook University)<sup>1</sup> 计算机科学系

纽约 石溪(Stony Brook, NY) 11794-4400

<http://www.cs.sunysb.edu/~skiena>

2008年4月

<sup>1</sup> 译者注: 以前称为“纽约州立大学石溪分校”(State University of New York at Stony Brook).