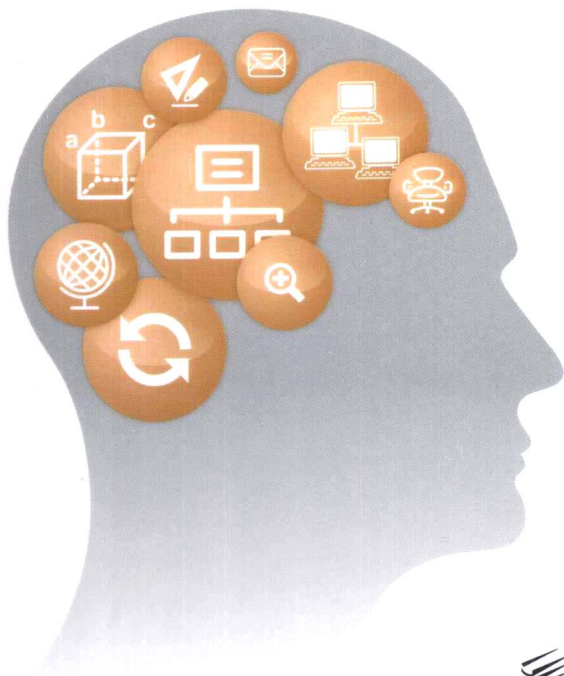


# 像计算机科学家 一样思考 Java

[美] Allen B. Downey 著  
滕云 周哲武 译

Think Java  
How to Think  
Like a Computer Scientist

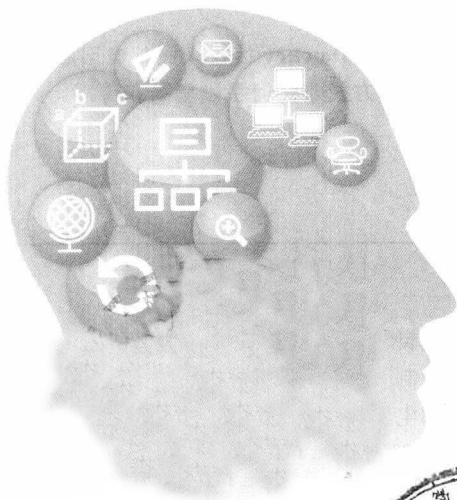
语言本身并不重要，解决问题的创新方法才是王道  
学会“像计算机科学家一样思考”



 人民邮电出版社  
POSTS & TELECOM PRESS

# 像计算机科学家 一样思考 Java

[美] Allen B. Downey 著  
滕云 周哲武 译



人民邮电出版社  
北京

## 图书在版编目 (C I P) 数据

像计算机科学家一样思考Java / (美) 唐尼  
(Downey, A. B.) 著 ; 滕云, 周哲武译. — 北京 : 人民  
邮电出版社, 2013. 6  
ISBN 978-7-115-31322-5

I. ①像… II. ①唐… ②滕… ③周… III. ①  
JAVA语言—程序设计 IV. ①TP31

中国版本图书馆CIP数据核字(2013)第060775号

## 版 权 声 明

Simplified Chinese translation copyright ©2013 by Posts and Telecommunications Press  
ALL RIGHTS RESERVED

Think Java How to Think Like a Computer Scientist, by Allen B. Downey

ISBN-13: 978-1441419200

Copyright © 2012 by Allen B. Downey

本书中文简体版由作者 **Allen B. Downey** 授权人民邮电出版社出版。未经出版者书面许可,  
对本书的任何部分不得以任何方式或任何手段复制和传播。

版权所有, 侵权必究。

- 
- ◆ 著 [美] Allen B. Downey
  - 译 滕 云 周哲武
  - 责任编辑 陈冀康
  - 责任印制 程彦红 杨林杰
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号  
邮编 100061 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
三河市海波印务有限公司印刷
  - ◆ 开本: 787×1000 1/16  
印张: 14.75  
字数: 272千字 2013年6月第1版  
印数: 1-3000册 2013年6月河北第1次印刷

---

著作权合同登记号 图字: 01-2012-5790 号

定价: 39.00 元

读者服务热线: (010)67132689 印装质量热线: (010)67129223  
反盗版热线: (010)67171154

---

# 内容提要

本书作者基于自己在美国各所大学和学院讲授计算机科学导论课程的经验，开创了“像计算机科学家一样思考 (How to Think Like a Computer Scientist)”的教学理念和方法。本书正是基于这样的方法，用全新的角度、丰富的实例全面地讲解了 Java 语言。

全书共 15 章。第 1 章介绍了编程的基本知识，即什么是编程以及如何编程。第 2 章到第 4 章分别介绍了变量和类型、无返回值方法、条件语句和递归等基础知识；第 6 章到第 9 章分别介绍了带返回值的方法、迭代、循环、字符串和可变对象；第 11 章到第 15 章则分别介绍了对象、数组、面向对象编程等相对高级的话题。全书贯穿了一个 GridWorld 的程序示例，分三个部分逐步扩展，分别在第 5 章、第 10 章和第 16 章介绍，用以巩固前面相应各章所介绍的知识和概念。附录 A 介绍了 Java 图形编程，附录 B 介绍了 Java 中的输入输出，附录 C 和 D 分别从程序开发和调试的角度给出了有用的建议。

本书适合 Java 的初学者和初级程序员阅读，也可以作为相关专业或培训的教程使用。通过学习本书，读者不仅可以在 Java 方面达到初窥门径的效果，同时对于如何掌握和应用计算机编程这门技艺也会有一个全面而科学的认识。

---

# 作者简介

Allen B. Downey 是美国 Olin 工程学院的计算机科学副教授。他曾经在 Wellesley College、Colby College 和 U.C. Berkeley 教授计算机科学课程。他在 MIT 获得学士和硕士学位，并且从 U.C. Berkeley 获得计算机科学博士学位。Allen 基于自己教授计算机程序设计课程的经验，开创了“像计算机科学家一样思考 (How to Think Like a Computer Scientist)”的教学理念和方法，并由此编写了几本程序设计语言的图书。其中，《Think Python》、《Think Complexity》由 O'Reilly 出版；《Think Java》、《Think C++》也广受关注和好评。

---

# 译者简介

滕云, ThoughtWorks 软件工程师, 目前主要从事企业级软件开发, 擅长技术包括 C/C++、Java、Ruby、Linux 和测试自动化等。爱听古典音乐, 尤其是贝多芬; 爱写诗词, 尤其是长短句; 爱吃川菜, 尤其是回锅肉。个人博客: <http://www.davenkin.me>。腾讯微博: @无知者云

周哲武, 码农, 躬耕于 ThoughtWorks, 不求闻达于诸侯, 但求程序之乐。性喜静、好宅、好酒, 更好书。尝以为不写代码即解决客户问题为善之善者, 干净优雅代码解决问题次之。近年专注于企业级应用开发, 擅长各种自动化测试策略, 以及遗留系统代码重构。新浪微博: @牧野飞霜

---

# 前言

我们从别人的发明中享受了很大益处，我们也应该乐于以我们的任何一种发明为别人服务，并且这种事我们应该自愿地和慷慨地去做。

—— 本杰明·富兰克林，引用自 Edmund S. Morgan 所著的《本杰明·富兰克林传》

## 我为什么写这本书

1999 年，当我还在科尔比学院教书的时候，我便开始写这本书了。现在，这已经是第 5 个版本了。那时，我正使用 Java 程序设计语言来教授计算机科学导论课程，但始终没有找到一本合适的教材。一方面，这些书的内容太多，让学生去阅读一本 800 页的技术书籍显然不可能，即便是我硬叫他们去读也不可能，更何况我并不想他们这么做。另外，书中多数内容的针对性都很强，比如 Java 语言的细节和 Java 类库等内容，而这些类库在学期结束时很有可能就已经过时了。这样的书籍显然不是我需要的。

另一方面，我发现一开始就讲面向对象编程并不合适。很多在其他方面表现得很好的学生，一旦接触到对象就一头雾水了。于是我开始写这本书，每天写一章，一共写了 13 天，第 14 天编辑，然后是复印和装订。上课第一天我便把书分发给学生们，并叫他们每周阅读一章，这样一来，他们阅读的速度比我书写的速度慢了 7 倍。

## 背后的哲学

以下是我写这本书的一些想法：

- 词汇很重要。学生应该有去讨论程序并且能够听懂我所讲的话。我尝试着引入尽可能少的术语，并在第一次使用这些术语时精心地给出定义，然后在每章的最后附上术语表。在我的课程中，我会在小测验和考试中包含一些和词汇相关的题目，然后要求学生使用恰当的术语进行回答。

- 要编写程序。学生应该理解算法，知晓程序设计语言本身，并能够调试程序。太多的书籍都将程序调试内容给忽略掉了。本书最后的附录 D 讲述程序调试的内容，附录 C 讨论程序开发内容（可以帮你避免调试）。我建议学生们尽早地阅读这些内容，并且时常回过头来复习。
- 有些概念需要时间来消化。对于一些较难的内容，本书将反复讲到，这样可以给学生更多的机会去复习和巩固。如果第一次没有学会，可以在后面赶上。
- 我试图尽量少地使用 Java 语言来教会学生获得尽量多的编程能力。本书的目的在于教学生如何编写程序和一些计算机科学导论方面的概念，而不是 Java 语言。因此，我去除了一些 Java 语言特性方面的内容，比如 switch 语句便没有必要了，多数类库也不在本书的教授范围之内，特别是一直在变并很可能被替换掉的 AWT。

这种最小化的方式有它的优点。除了章后练习，每章大概 10 页左右。在我的课程中，我要求学生提前阅读要讲的章节，他们都愿意这么做并且理解得很好。这样一来，我们便有时间来讨论一些更抽象的内容、做一些课内练习和学习本书之外的内容。

但是，这种最小化方式也有缺点。从本质上讲，这种方式没有多少有趣的地方。书中的多数例子只是在展现基本的语言特性，并且很多练习都包含了字符串操作和数学相关的知识。我认为其中的一些内容是有趣的，但是还有很多能够激发学生兴趣的内容，比如图形、声音和网络应用只是一带而过。

问题在于，很多令人兴奋的内容都包含大量的细节，而其中的概念并不多。从教学上讲，这样浪费了太多的努力但又达不到教学目的。因此，在取悦学生和富含知识之间便出现了一个折中。我让老师们自己去解决这种平衡。作为帮助，本书的附录中包含图形、键盘输入和文件输入等内容。

## 面向对象编程

有些书一上来便介绍对象，另一些书先通过面向过程的编程方式进行引导，再逐渐讲授面向对象编程方式。本书采用后一种方式。

Java 中许多面向对象特性都是由先前语言出现的问题来驱动的，并且它的实现也受



到了这些语言发展历史的影响。在学生不了解他们需要解决的问题的情况下，有些特性是很难解释的。推迟讲面向对象编程并不是我的意图所在，相反，我会尽快地涉及到这些内容，只是我每次只讲一个概念，并且尽可能清晰，以使学生可以相对独立地练习每一个知识点，然后再加入后续的概念。但是我也承认，达到这样的目的确实需要时间。

## 计算机科学预科考试

当科尔比学院宣布预科考试将转向 Java，我顺其自然地将本书更换成 Java 版本。当看到预科的考试大纲时，我发现大纲中的内容完全就是我书中所选内容的一个子集。

2003 年 1 月，我正在写本书的第 4 个版本，该版有如下变化：

- 增加了一些章节以覆盖更多的预科大纲。
- 改进了关于调试和程序开发内容的附录。
- 将我在上课时用到的练习、小测验和考试的题目放在了相关章节的末尾。同时，针对预科考试增加了一些题目。

最终，在 2011 年的 8 月，我开始了本书第五版的写作，增加了预科考试中包含的 GridWorld 案例研究。

## 免费！

从一开始，本书便允许读者进行复制、分发和修改。读者可以下载到本书不同格式，并且既可以在计算机上阅读，也可以进行打印。老师可以免费地进行多份复印。任何人均可以按照自己需求对本书进行定制性的修改。

有人将本书翻译成其他的计算机语言（包括 Python 和 Eiffel），也有翻译成其他的自然语言（包括西班牙语、法语和德语）。这些翻译版本中的许多也是免费的。

受开源软件的驱动，我也采用了“早发布，勤更新”的方式。我尽量减少错误，但同时我也需要读者的帮助。

本书的反响是很好的。我几乎每天都收到读者的反馈信息，他们甚至不嫌麻烦将

“bug 报告”发送给我。通常，我都会在几分钟内更改书中的错误，然后发布更新。我将本书看做是一份正在进行中的工作，一旦有时间或者有读者发回反馈，我都将一点一点地进行改进。

## 关于书名

本书的书名给了我不少痛苦。不是所有人都知道，本书的题目在很大程度上只是个玩笑。阅读本书很可能不会让你像一个计算机科学家那样思考，那需要时间、经验，并且很可能还需要更多的课程。

但本书的书名的真实含义在于：它不是关于 Java 的，并且不是完全关于编程的。如果成功的话，那么本书是关于思考方式的。计算机科学家自有一套解决问题的方法，而且这种方法是独特的、通用的和强大的。我希望本书能引领你去感知这样的方法，继而在某种程度上，你会发现自己确实是在像计算机科学家一样思考。

## 贡献者名单

当我开始写免费图书时，我并没有留意贡献者名单。当 Jeff Elkner 建议我之后，我才发现省略掉这份名单是多么的尴尬。这份贡献者名单是从第四版开始的，故许多对先前版本提出建议和改正的贡献者都被省略掉了。如果你有另外的建议，请发送邮件到：[feedback@greenteapress.com](mailto:feedback@greenteapress.com)。

- Ellen Hildreth 将本书用在威尔斯利学院的数据结构课程中，并对本书做了一系列的更正，另外还提出了一些很好的建议。
- Tania Passfield 曾指出，在第 4 章的术语表中，有些术语在该章内容中已经不存在了。
- Elizabeth Wiethoff 发现了我对于  $e^{-x^2}$  的解释是错误的，她目前正在写本书的 Ruby 版本。
- Matt Crawford 发给我了一份满是更正的补丁文件。
- Chi-Yu Li 指出了排版错误和一个代码例子错误。
- Doan Thanh Nam 更正了第 3 章中的一个例子。

- Stijn Debrouwere 发现了一个数学排版错误。
- Muhammad Saied 将本书翻译成了阿拉伯文，并发现了多处错误。
- Marius Margowski 发现了一个代码例子中的一处不一致。
- Guy Driesen 发现了多个排版错误。

最后，我想感谢 Chris Mayfield 对本书最新版的巨大贡献。他仔细审阅了本书并对书中一百多次错误做了更正。最新版中加入了一些新的变化，比如嵌入式超文本链接，交叉引用，练习题版式的一致性，代码例子中 Java 语法高亮等。

# 目录

<b>第 1 章 程序之道</b> .....	1
1.1 什么是编程语言 .....	1
1.2 什么是程序 .....	3
1.3 什么是调试 .....	3
1.3.1 语法错误 .....	4
1.3.2 运行时错误 .....	4
1.3.3 逻辑错误和语义错误 .....	4
1.3.4 实验性调试 .....	5
1.4 形式语言和自然语言 .....	5
1.5 第一个程序 .....	7
1.6 术语表 .....	8
1.7 练习 .....	10
<b>第 2 章 变量和类型</b> .....	12
2.1 更多打印 .....	12
2.2 变量 .....	13
2.3 赋值 .....	14
2.4 打印变量 .....	15
2.5 关键字 .....	16
2.6 运算符 .....	17
2.7 运算符优先级 .....	18
2.8 字符串运算符 .....	19
2.9 组合 .....	19
2.10 术语表 .....	20
2.11 练习 .....	21
<b>第 3 章 无返回值方法</b> .....	23
3.1 浮点数 .....	23
3.2 双精度型转整型 .....	24
3.3 数学函数 .....	25
3.4 组合 .....	26

3.5	添加新方法	26
3.6	类和方法	28
3.7	为程序添加多个方法	29
3.8	形参和实参	30
3.9	栈帧图	31
3.10	多参数方法	32
3.11	有返回值的方法	33
3.12	术语表	33
3.13	练习	34
<b>第 4 章</b>	<b>条件语句和递归</b>	<b>36</b>
4.1	模运算符	36
4.2	条件执行	36
4.3	选择性执行	37
4.4	条件判断链	38
4.5	嵌套条件	39
4.6	返回语句	39
4.7	类型转换	40
4.8	递归	40
4.9	递归方法的栈帧图	42
4.10	术语表	43
4.11	练习	43
<b>第 5 章</b>	<b>GridWorld (第一部分)</b>	<b>47</b>
5.1	入门	47
5.2	BugRunner	48
5.3	练习	49
<b>第 6 章</b>	<b>有返回值方法</b>	<b>52</b>
6.1	返回值	52
6.2	程序开发	54
6.3	组合	56
6.4	重载	57
6.5	布尔表达式	58
6.6	逻辑运算符	59
6.7	布尔方法	60
6.8	更多递归	61
6.9	飞跃的信心	63

6.10	另一个例子	64
6.11	术语表	64
6.12	练习	65
<b>第 7 章</b>	<b>迭代和循环</b>	<b>71</b>
7.1	多次赋值	71
7.2	迭代	72
7.3	while 语句	72
7.4	表格	74
7.5	二维表格	76
7.6	封装与泛化	77
7.7	方法	78
7.8	更多的封装	78
7.9	局部变量	79
7.10	更多的泛化	79
7.11	术语表	81
7.12	练习	82
<b>第 8 章</b>	<b>字符串</b>	<b>85</b>
8.1	调用对象上的方法	85
8.2	字符串长度	86
8.3	遍历	87
8.4	运行时错误	87
8.5	阅读文档	88
8.6	indexOf 方法	89
8.7	循环和计数	90
8.8	自增和自减运算符	90
8.9	字符串是不可变的	91
8.10	字符串是不可比较的	91
8.11	术语表	92
8.12	练习	93
<b>第 9 章</b>	<b>可变对象</b>	<b>100</b>
9.1	Point 对象和 Rectangle 对象	100
9.2	包	100
9.3	Point 对象	101
9.4	实例变量	102
9.5	对象作为方法参数	102

9.6	Rectangle 对象	103
9.7	对象作为返回类型	104
9.8	对象是可变的	104
9.9	别名	105
9.10	null	106
9.11	垃圾回收	107
9.12	对象类型和原始类型	108
9.13	术语表	109
9.14	练习	109
<b>第 10 章</b>	<b>GridWorld (第二部分)</b>	<b>114</b>
10.1	Termite 类	116
10.2	Langton 白蚁	118
10.3	练习	119
<b>第 11 章</b>	<b>创建对象</b>	<b>122</b>
11.1	类定义与对象类型	122
11.2	Time 类	123
11.3	构造函数	124
11.4	更多构造函数	125
11.5	创建新对象	125
11.6	打印对象	126
11.7	操作对象	127
11.8	纯函数	128
11.9	修改器	130
11.10	填值方法	131
11.11	渐进式开发	131
11.12	泛化	133
11.13	算法	133
11.14	术语表	134
11.15	练习	134
<b>第 12 章</b>	<b>数组</b>	<b>137</b>
12.1	访问数组元素	138
12.2	拷贝数组	139
12.3	for 循环	139
12.4	数组和对象	140
12.5	数组长度	141

12.6	随机数	141
12.7	随机数数组	142
12.8	计数	143
12.9	直方图	144
12.10	单次遍历	145
12.11	术语表	145
12.12	练习	146
<b>第 13 章</b>	<b>对象数组</b>	<b>152</b>
13.1	内容前瞻	152
13.2	Card 对象	152
13.3	printCard 方法	154
13.4	sameCard 方法	155
13.5	compareCard 方法	157
13.6	扑克牌数组	158
13.7	printDeck 方法	160
13.8	查找	160
13.9	整体与部分	163
13.10	术语表	164
13.11	练习	164
<b>第 14 章</b>	<b>数组对象</b>	<b>166</b>
14.1	Deck 类	166
14.2	洗牌	168
14.3	排序	169
14.4	子集	169
14.5	洗牌和发牌	170
14.6	合并排序法	171
14.7	类变量	173
14.8	术语表	174
14.9	练习	174
<b>第 15 章</b>	<b>面向对象编程</b>	<b>176</b>
15.1	编程语言和风格	176
15.2	对象方法和类方法	177
15.3	toString 方法	178
15.4	equals 方法	178
15.5	错误	179



15.6	继承	180
15.7	类的层级结构	181
15.8	面向对象设计	182
15.9	术语表	182
15.10	练习	183
<b>第 16 章</b>	<b>GridWorld (第三部分)</b>	<b>185</b>
16.1	ArrayList	185
16.2	接口	187
16.3	public 和 private	188
16.4	生命游戏	188
16.5	LifeRunner	189
16.6	LifeRock	190
16.7	同步更新	190
16.8	初始条件	192
16.9	练习	193
<b>附录 A</b>	<b>图形</b>	<b>194</b>
A.1	Java 二维图形	194
A.2	Graphics 方法	195
A.3	坐标	196
A.4	颜色	196
A.5	米老鼠	197
A.6	练习	198
<b>附录 B</b>	<b>Java 中的输入和输出</b>	<b>200</b>
B.1	System 对象	200
B.2	键盘输入	200
B.3	文件输入	201
B.4	捕获异常	202
<b>附录 C</b>	<b>程序开发</b>	<b>203</b>
C.1	策略	203
C.2	失败的开发模式	204
<b>附录 D</b>	<b>调试</b>	<b>206</b>
D.1	语法错误	206
D.2	运行时错误	209
D.3	逻辑错误	213