O'REILLY®

数据结构与算法 Java语言描述

Think Data Structures

Allen B. Downey 著
李新叶 李楠楠 译

中国电力出版社

数据结构与算法
Java语言描述

Allen B. Downey 著
李新叶 李楠楠 译

Beijing · Boston · Farnham · Sebastopol · Tokyo O'REILLY®
O'Reilly Media, Inc. 授权中国电力出版社出版

中国电力出版社

图书在版编目（CIP）数据

数据结构与算法Java语言描述 / (美) 艾伦（Allen B.Downey）著；李新叶，李楠楠译. —北京：中国电力出版社，2018.8

书名原文：Think Data Structures

ISBN 978-7-5198-2194-4

I. ①数… II. ①艾… ②李… ③李… III. ①数据结构 ②JAVA语言－程序设计
IV. ①TP311.12 ②TP312.8

中国版本图书馆CIP数据核字(2018)第141905号

北京市版权局著作权合同登记 图字：01-2018-3071号

出版发行：中国电力出版社
地　　址：北京市东城区北京站西街19号（邮政编码100005）
网　　址：http://www.cepp.sgcc.com.cn
责任编辑：刘 炽（liuchi1030@163.com）
责任校对：黄 蓓，李 楠
装帧设计：Karen Montgomery，张 健
责任印制：杨晓东

印　刷：三河市航远印刷有限公司
版　次：2018年8月第一版
印　次：2018年8月北京第一次印刷
开　本：750毫米×980毫米　16开本
印　张：10.5
字　数：196千字
印　数：0001—3000册
定　价：38.00元

O'Reilly Media, Inc.介绍

O'Reilly Media通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自1978年开始，O'Reilly一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了Make杂志，从而成为DIY革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly现在还将先锋专家的知识传递给普通的计算机用户。无论是通过书籍出版，在线服务或者面授课程，每一项O'Reilly的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar博客有口皆碑。”

——Wired

“O'Reilly凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——Business 2.0

“O'Reilly Conference是聚集关键思想领袖的绝对典范。”

——CRN

“一本O'Reilly的书就代表一个有用、有前途、需要学习的主题。”

——Irish Times

“Tim是位特立独行的商人，他不光放眼于最长远、最广阔的视野并且切实地按照Yogi Berra的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去Tim似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——Linux Journal

目录

前言

本书背后的哲理

数据结构与算法是过去几十年来最重要的创新之一，是软件工程师需要掌握的基础工具。但在我看来，大部分有关数据结构与算法的书籍都太过于理论化、太厚而且太“自底向上”化了：

太过于理论化

算法的数学分析基于简化的假设，这些假设限制了它在实践中的实用性。很多关于数据结构与算法的讲解都忽略了其简单化而关注其数学基础。在本书中，我对这部分提出了最实用的讲解而省略或不再强调其他方面内容。

太厚

许多有关数据结构与算法的书籍至少有500页，一些甚至超过1000页。我重点关注的是我认为对软件工程师最有用的话题，这样使得本书厚度很薄。

太“自底向上”化

许多讲述数据结构的书关注于它是怎样工作的（实现），而很少涉及怎样使用它们（接口）。在本书中，我采取了自顶向下策略，从接口开始讲起。你在学习Java集合框架中结构的工作原理细节之前，先学到怎么使用它们。

最后，有些书在讲解这一话题时脱离了上下文而且没有吸引力：只是一个又一个烦人的数据结构！我尝试结合一个 Web 搜索应用来组织这个话题，从而使其变得生动有趣。Web 搜索应用广泛使用了数据结构，并且是一个有趣而重要的话题。

Web 搜索应用还带来了一些通常不在初学数据结构类的部分涉及的主题，包括 Redis 的持久性数据结构。

对于该放弃什么内容，我已经做出了艰难的决定，但我已经做出了一些妥协。在本书中包括几个大多数人永远不会使用的主题，但他们可能在技术面试中需要知道这些内容。对于这些主题，我既提出了传统的做法，同时也给出了我怀疑的理由。

本书还介绍了软件工程实践的基本知识，包括版本控制和单元测试。大多数章节包括一个练习，你可实践所学的知识。每个练习都提供了检查解决方案的自动测试。对于大多数练习，我在下一章的开头都介绍了我的解决方法。

必备知识

本书适用的读者包括计算机科学及相关领域的大学生、专业的软件工程师、软件工程培训师以及正在准备技术面试的相关人员。

在读本书之前，你需要有很好的 Java 基础。具体说，你应当知道怎么定义一个从已有类扩展的新类或怎么实现一个接口。如果你对 Java 编程已经不熟悉了，这里有两本书你需要先学习一下：

- Downey 和 Mayfield 编写的《Think Java》(O'Reilly Media，2016)，适用于没有一点编程基础的人。
- Sierra 和 Bates 编写的《Head First Java》(O'Reilly Media，2005)，适用于学过另一种编程语言的人。

如果你不熟悉 Java 的接口，你可能需要学习 *http://thinkdast.com/interface* 上提供的教程“What Is an Interface?”。

词汇方面的注释：Interface（接口）一词比较容易使人混淆。在应用程序接口（API）中，它指提供一定功能的一组类和方法。

在 Java 语言中，Interface（接口）还指一种语言特征，它与类相似，定义了一组方法。

你也应该熟悉类型参数与通用类型。比如，你应知道怎样创建一个有类型参数的对象，如 `ArrayList<Integer>`。如果你对类型参数不熟悉，请参阅 *http://thinkdast.com/types*。

你应该熟悉 Java 集合框架（JCF），可参阅 *http://thinkdast.com/collections*。特别是，你应熟悉 `List` 接口、`ArrayList` 类和 `LinkedList` 类。

你最好也应该熟悉 Apache 的 Ant，它是 Java 的自动编译工具，可参阅 *http://thinkdast.com/anttut*。

你也应该熟悉 JUnit，它是 Java 的单元测试框架，可参阅 *http://thinkdast.com/junit*。

使用书中的代码

本书的代码在一个 Git 库中，参见 *http://thinkdast.com/repo*。

Git 是一个版本控制系统，它允许你跟踪组成项目的文件。Git 控制下的一个文件集称为一个仓库（repository）。

GitHub 是一个主机服务，为 Git 仓库提供了存储空间和便捷的 Web 接口。它提供了以下几种使用代码的方式：

- 可以按下 Fork 按钮创建一个 GitHub 仓库的备份。如果你还没有一个 GitHub 账户，需要注册一个。备份后，你将在 GitHub 上拥有自己的仓库，可以使用它跟踪你编写的代码。然后你可以克隆这个仓库，将文件的一个备份下载到你的计算机上。

- 另一个做法是，可以不使用 Fork 备份而直接克隆仓库。如果你选择这么做，就不需要 GitHub 账户，但不能在 GitHub 上保存你的修改。

- 如果你一点也不想使用 Git，可以按下 GitHub 页或链接 http://think dast.com/zip 上的 Download 按钮下载代码的 ZIP 压缩包。

当你克隆仓库或对 ZIP 文件解压后，就会找到一个 `ThinkDataStructures` 目录，其下有一个子目录 `code`。

本书中的例子使用 Java 开发工具包 7（Java SE Development Kit 7）开发和测试。如果你使用的是旧开发版本，有些例子将不能运行。如果你在使用一个更新的版本，这些代码都应当能正常运行。

本书使用约定

本书使用以下排版约定：

斜体（Italic）
: 表示强调、按键、菜单选项、URL 网址和 email 地址。

黑体（Bold）
: 表示首次定义的新术语。

`等宽字体（Constant width）`
: 表示程序代码清单，以及段落内的文件名、文件扩展，程序中的元素如变量、函数名、数据类型、语句和关键字。

`等宽黑体 (Constant width bold)`
: 表示由用户输入的命令或其他文本。

Safari 图书在线

Safari 图书在线（*www.safaribooksonline.com*）是一个应需而变的数字图书馆，它以书籍和视频的形式提供了来自全球范围内技术和企业领域资深作者撰写的专业内容。

专业技术人员、软件开发人员、网页设计师、商业和创意专业人士使用 Safari 图书在线作为研究解决问题、学习和认证培训的首要资源。

Safari 图书在线为企业、政府机构、教育机构和个人提供了一系列计划和定价方案。

订阅者可在一个快捷搜索的数据库中获得成千上万的书籍、培训视频和出版前的手稿，如 O'Reilly Media，Prentice Hall Professional，Addison-Wesley Professional，Microsoft Press，Sams，Que，Peachpit Press，Focal Press，Cisco Press，John Wiley & Sons，Syngress，Morgan Kaufmann，IBM Redbooks，Packt，Adobe Press，FT Press，Apress，Manning，New Riders，McGraw-Hill，Jones & Bartlett，Course Technology 以及其他数百家出版社。有关 Safari 图书在线的更多信息，请访问我们的网站。

联系方式

请将您发现的问题以及建议及时报告给出版商：

美国：

O'Reilly Media，Inc.
1005 Gravenstein Highway North
Sebastopol，CA 95472

中国：

北京市西城区西直门南大街2号成铭大厦C座807室（100035）
奥莱利技术咨询（北京）有限公司

发表评论或咨询有关本书的技术问题，请发送电子邮件至 *bookquestions@oreilly.com*。

关于我们的书籍、课程、会议和新闻的更多信息，请参阅我们的网站：

http://www.oreilly.com
http://www.oreilly.com.cn

我们的 Facebook：*http://facebook.com/oreilly*。

我们的 Twitter：*http://twitter.com/oreillymedia*。

我们的 YouTube：*http://www.youtube.com/oreillymedia*。

致谢

这本书是我在纽约 Flatiron 学校编写的全部课程内容的一个改编版，这所学校提供了各种与编程和 Web 开发相关的在线课堂。他们基于这个材料有一个课堂，提供在线开发环境、来自讲师和其他学生的帮助，以及结业证书。你可在网站 *http://flatironschool.com* 上找到更多信息。

- 在 Flatiron 学校，Joe Burgess、Ann John 和 Charles Pletcher 提供了指导、建议以及对初始版本从实现到测试整个过程的修改。感谢你们！
- 非常感谢技术评论员 Barry Whitman、Patrick White 以及 Chris Mayfield，他们发现了许多错误并提供了许多有帮助的建议。当然，书中还有错误的话，那是我的过失，与他们无关！
- 感谢 OlinCollege 学院数据结构与算法课程的指导教师与学生们，他们阅读了本书并给出了有益的反馈意见。
- Charles Roumeliotis 为 O'Reilly 公司编辑了本书并作出了许多改进。

如果你对本书有看法或评论，请发邮件到 *feedback@greenteapress.com*。

第1章

接口

本书包括三个主题：

数据结构

从 Java 集合框架（JCF）中的结构开始，你将学习如何使用列表和映射等数据结构，并将看到它们是如何工作的。

算法分析

介绍了分析代码和预测代码运行速度以及所需内存空间的技术。

信息检索

为了激发对前两个主题的学习兴趣，并使练习更有趣，我们将使用数据结构和算法来构建一个简单的 Web 搜索引擎。

以下是主题顺序的概述：

- 我们将从列表接口开始，你将用两种不同方法编写实现该接口的类。然后，我们将你的实现与 Java 类 ArrayList 和 LinkedList 进行比较。
- 接下来将介绍树形数据结构，你将使用第一个应用程序：一个从 Wikipedia 读取页面、分析内容并导航结果树以找到连接和其他特征的程序。我们将使用这些工具来测试“Getting to Philosophy”（到达哲学）猜测（可以通过读取 *http://Thinkdast.com/GetPil* 获得预览）。

- 我们将了解 Map 接口和 Java 的 HashMap 实现。然后，你将编写使用哈希表和二叉搜索树实现此接口的类。

- 最后，你将使用这些类（以及我将介绍的其他一些类）来实现一个 Web 搜索引擎，包括一个查找和读取页面的爬虫、一个存储网页内容并使搜索网页变得有效的索引器，以及一个从用户那里获取查询并返回相关结果的检索器。

让我们现在开始吧！

为什么有两种列表？

当人们开始使用 Java 集合框架时，他们有时会对 ArrayList 和 LinkedList 感到困惑。Java 为什么要提供列表接口的两个实现？你应当选择哪一个来用？我会在接下来的几章中回答这些问题。

首先，我将回顾 Java 接口和实现它们的类，并介绍“面向接口编程”的想法。

在前几个练习中，将实现类似 ArrayList 和 LinkedList 的类，了解它们是如何工作的，我们将看到它们各有优缺点。有些操作使用 ArrayList 时更快或使用更少的空间，其他操作则在使用 LinkedList 时更快或使用更少的空间。对于特定的应用程序来说，哪个操作更好取决于它执行的最频繁的操作是什么。

Java 接口

一个 Java 接口定义了一组方法，实现该接口的任何类都必须提供这些方法的实现。例如，下面是 Comparable 的源代码，它是 java.lang 包中定义的一个接口：

```
public interface Comparable<T> {
    public int compareTo (T o) ;
}
```

此接口定义使用了一个类型参数 T，使得 Comparable 是通用型的。为了实现这个接口，一个类必须：

- 定义 T 引用的类型。
- 提供一个名为 compareto 的方法，该方法将对象作为参数并返回一个 int。

例如，以下是用于 java.lang.Integer 的 Comparable 的源代码：

```
public final class Integer extends Number implements Comparable<Integer> {

    public int compareTo (Integer anotherInteger)  {
       int thisVal = this.value;
    int anotherVal = anotherInteger.value;
return  (thisVal<anotherVal ? -1 :  (thisVal==anotherVal ? 0 : 1) ) ;
}
// 省略的其他方法
}
```

这个类扩展了 Number，它继承来自 Number 的方法和实例变量，并实现了 Comparable<Integer>，因此它提供了一个名为 compareto 的方法，该方法接受一个 Integer 参数并返回一个 int。

当一个类声明它实现了一个接口时，编译器会检查它是否提供了接口定义的所有方法。

顺便说一句，这里 Compareto 的实现使用了“三元运算符”，有时会写作？：。如果你不熟悉它，你可以在 *http://Thinkdast.com/ternary* 上查询。

List 接口

Java 集合框架（JCF）定义了一个名为 List 的接口，并提供了两个实现：ArrayList 和 LinkedList。

该接口定义了使其成为一个列表的全部。任何实现该接口的类都必须提供一组特定的方法，包括 add、get、remove 以及其余大约 20 个方法。