

TURING

图灵程序设计丛书

[PACKT]
PUBLISHING

Learning JavaScript Data Structures and Algorithms Second Edition

学习JavaScript 数据结构与算法 (第2版)

[巴西] Loiane Groner 著 邓钢 孙晓博 吴双 陈迪 袁源 译

用JavaScript学习常用的数据结构和算法，
高效解决计算机科学中的常见问题。



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

学习JavaScript 数据结构与算法

(第2版)



[巴西] Loiane Groner 著

邓钢 孙晓博 吴双 陈迪 袁源 译

人民邮电出版社

北京

图书在版编目 (C I P) 数据

学习JavaScript数据结构与算法：第2版 / (巴西)
洛伊安妮·格罗纳 (Loiane Groner) 著；邓钢等译。—
2版。—北京：人民邮电出版社，2017.9
(图灵程序设计丛书)
ISBN 978-7-115-46753-9

I. ①学… II. ①洛… ②邓… III. ①JAVA语言—程
序设计 IV. ①TP312.8

中国版本图书馆CIP数据核字(2017)第213790号

内 容 提 要

本书首先介绍了 JavaScript 语言的基础知识以及 ES6 和 ES7 中引入的新功能，接下来讨论了数组、栈、队列、链表、集合、字典、散列表、树、图等数据结构，之后探讨了各种排序和搜索算法，包括冒泡排序、选择排序、插入排序、归并排序、快速排序、堆排序、计数排序、桶排序、基数排序、顺序搜索、二分搜索，然后介绍了动态规划和贪心算法等常用的高级算法以及函数式编程，最后还介绍了如何计算算法的复杂度。

本书适用于前端 Web 开发人员，以及所有对 JavaScript 数据结构与算法感兴趣的读者。

-
- ◆ 著 [巴西] Loiane Groner
 - 译 邓 钢 孙晓博 吴 双 陈 迪 袁 源
 - 责任编辑 岳新欣
 - 责任印制 彭志环
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
 - 邮编 100164 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：800×1000 1/16
 - 印张：14.5
 - 字数：342千字 2017年9月第2版
 - 印数：5 601-9 100册 2017年9月河北第1次印刷
 - 著作权合同登记号 图字：01-2017-5589号
-

定价：49.00元

读者服务热线：(010)51095186转600 印装质量热线：(010)81055316

反盗版热线：(010)81055315

广告经营许可证：京东工商广登字 20170147 号

站在巨人的肩上
Standing on Shoulders of Giants



iTuring.cn

站在巨人的肩上
Standing on Shoulders of Giants



iTuring.cn

版权声明

Copyright © 2016 Packt Publishing. First published in the English language under the title *Learning JavaScript Data Structures and Algorithms, Second Edition*.

Simplified Chinese-language edition copyright © 2017 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由Packt Publishing授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

致 谢

我要感谢父母这些年来对我的教育、指导和建议，帮助我成为一个更好、更专业的人。特别要感谢我的丈夫给予我的耐心、支持和鼓励。我要感谢Packt出版社给我机会，让我能够就自己真正钟爱的话题写书。感谢在图书创作、审阅和出版过程中帮助过我的所有人。

我还要感谢这本书以及我撰写的其他书的读者，感谢你们的支持和反馈。你们的反馈非常宝贵，帮助我成长为一名更优秀的作者和专业人士。非常感谢！

前 言

JavaScript是当下最流行的编程语言之一。由于浏览器的原生支持（无需安装任何插件），JavaScript也被称作“互联网语言”。JavaScript的应用非常广泛，不仅被用于前端开发，也被用到服务器（Node.js）和数据库（MongoDB）环境中。

对任何专业技术人员来说，理解数据结构都非常重要。作为软件开发者，我们要能够借助编程语言来解决问题，而数据结构是这些问题的解决方案中不可或缺的一部分。如果选择了不恰当的数据结构，可能会影响所写程序的性能。因此，了解不同数据结构和它们的适用范围十分重要。

算法在计算机科学中扮演着非常重要的角色。解决一个问题有很多种方法，但有些方法会比其他方法更好。因此，了解一下最著名的算法也很重要。

本书为数据结构和算法初学者所写，也为熟悉数据结构和算法，但想在JavaScript语言中使用它们的人所写。

快乐地编码吧！

本书结构

第1章“JavaScript简介”，讲述了JavaScript的基础知识，它们可以帮助你更好地学习数据结构和算法，同时还介绍了如何搭建开发环境来运行书中的代码示例。

第2章“数组”，介绍了如何使用数组这种最基础且最常用的数据结构。这一章演示了如何对数组声明、初始化、添加和删除其中的元素，还讲述了如何使用JavaScript语言本身支持的数组方法。

第3章“栈”，介绍了栈这种数据结构，演示了如何创建栈以及怎样添加和删除元素，还讨论了如何用栈解决计算机科学中的一些问题。

第4章“队列”，详述了队列这种数据结构，演示了如何创建队列，以及如何添加和删除队列中的元素，还讨论了如何用队列解决计算机科学中的一些问题，以及栈和队列的主要区别。

第5章“链表”，讲解如何用对象和指针从头创建链表这种数据结构。这一章除了讨论如何声

明、创建、添加和删除链表元素之外，还介绍了不同类型的链表，例如双向链表和循环链表。

第6章“集合”，介绍了集合这种数据结构，讨论了如何用集合存储非重复性的元素。此外，还详述了对集合的各种操作以及相应代码的实现。

第7章“字典和散列表”，深入讲解字典、散列表及它们之间的区别。这一章介绍了这两种数据结构是如何声明、创建和使用的，还探讨了如何解决散列冲突，以及如何创建更高效的散列函数。

第8章“树”，讲解了树这种数据结构和它的相关术语，重点讨论了二叉搜索树，以及如何在树中搜索、遍历、添加和删除节点。如果想更深入地学习树（包括相关的算法），这一章还给出了一些建议。

第9章“图”，介绍了图这种数据结构和它的适用范围。这一章讲述了图的常用术语和不同表示方式，探讨了如何使用深度优先算法和广度优先算法遍历图，以及它们的适用范围。

第10章“排序和搜索算法”，探讨了常用的排序算法，如冒泡排序（包括改进版）、选择排序、插入排序、归并排序和快速排序。另外还介绍了搜索算法中的顺序搜索和二分搜索。

第11章“算法模式”，介绍了一些算法技巧和一些著名的算法。这一章讲解了什么是递归，介绍了一些高级算法，如动态规划和贪心算法。

第12章“算法复杂度”，介绍了大 O 表示法的概念，以及本书实现算法的复杂度列表。这一章还介绍了NP完全问题和启发式算法。最后，讲解了提升算法能力的诀窍。

准备工作

为学习本书，你可以设置三种不同的开发环境。你不需要设置所有这三种环境，可以选择其一，也可以逐一尝试。

方法一，你需要一个浏览器，请从下面二选一：

- Chrome (<https://www.google.com/chrome/browser/>)
- Firefox (<https://www.mozilla.org/en-US/firefox/new/>)

方法二，你需要：

- 安装方法一中的任意一个浏览器；
- 安装一个Web服务器。如果你的电脑里没有安装过Web服务器，推荐安装XAMPP (<https://www.apachefriends.org>)。

方法三，如果想安装一个纯JavaScript的环境，你需要完成下面几步。

- 安装步骤一中的任意浏览器
- 安装Node.js (<http://nodejs.org/>)
- 安装好Node.js后, 安装http-server开发包:

```
npm install http-server -g
```

第1章还会对此进行更详细的介绍。

读者对象

本书的目标读者包括计算机科学专业的学生、刚刚开启职业生涯的技术人员, 以及想探索JavaScript最优能力的朋友。要想学好书中的算法, 需要对JavaScript和编程逻辑有基本的了解。

排版约定

在本书中, 你会发现一些不同的文本样式, 用以区别不同种类的信息。下面举例说明。

正文中的代码、用户输入这样表示: “用isEmpty方法就可以判断内部数组的长度是否为0。”

代码段的格式如下:

```
function Stack() {  
    //这里是属性和方法  
}
```

如果我们想让你重点关注代码段中的某个部分, 会加粗显示:

```
class Stack {  
    constructor () { this.items = []; //{1} }  
    push(element){  
        this.items.push(element);  
    }  
    //其他方法  
}
```

所有的命令行输入或输出的格式如下:

```
stack.push(5);  
stack.push(8);
```

新术语和重点词汇以黑体标示。



这个图标表示警告或需要特别注意的内容。



这个图标表示提示或者技巧。

读者反馈

欢迎提出反馈。如果你对本书有任何想法，喜欢它什么，不喜欢它什么，请让我们知道。要写出真正对大家有帮助的书，了解读者的反馈很重要。

一般的反馈，请发送电子邮件至feedback@packtpub.com，并在邮件主题中包含书名。

如果你有某个主题的专业知识，并且有兴趣写成或帮助促成一本书，请参考我们的作者指南www.packtpub.com/authors。

客户支持

现在，你是一位自豪的Packt图书的拥有者，我们会尽全力帮你充分利用你手中的书。

下载示例代码

你可以用你的账户从<http://www.packtpub.com>下载所有已购买Packt图书的示例代码文件。如果你从其他地方购买的本书，可以访问<http://www.packtpub.com/support>并注册，我们将通过电子邮件把文件发送给你。

下载代码文件的步骤如下：

- (1) 访问<http://www.packtpub.com>，用你的邮箱和密码登录或注册；
- (2) 将鼠标指针悬停在顶部的SUPPORT标签上；
- (3) 点击Code Downloads & Errata；
- (4) 在搜索框输入书名；
- (5) 选择你要下载代码文件的图书；
- (6) 在下拉菜单中选择购书方式；
- (7) 点击Code Download。

你还可以访问本书在Packt Publishing网站上的页面，点击Code Files按钮下载代码文件。在搜索框输入书名可以访问此页。请注意，你需要登录Packt账户。

文件下载后，请使用以下软件的最新版本解压：

- Windows系统请使用WinRAR或7-Zip

- Mac系统请使用Zipeg、iZip或UnRarX
- Linux系统请使用7-Zip或PeaZip

本书的代码包在GitHub的托管地址是<https://github.com/loiane/javascript-datastructures-algorithms>。其他图书或视频的代码包也可以到<https://github.com/PacktPublishing/>查阅。别错过！

下载彩色插图

我们还提供了一份PDF文档，里面是本书中用到的彩色截图和图表。这些彩图能帮你更好地理解输出的变化。你可以从https://www.packtpub.com/sites/default/files/downloads/Learning-JavaScriptDataStructuresandAlgorithmsSecondEdition_ColorImages.pdf下载。

勘误表

虽然我们已尽力确保本书内容正确，但出错仍旧在所难免。如果你在我们的书中发现错误，不管是文本还是代码，希望能告知我们，我们不胜感激。这样做可以减少其他读者的困扰，帮助我们改进本书的后续版本。如果你发现任何错误，请访问<http://www.packtpub.com/submit-errata>提交，选择你的书，点击勘误表提交表单的链接，并输入详细说明。^①勘误一经核实，你的提交将被接受，此勘误将上传到本公司网站或添加到现有勘误表。从<http://www.packtpub.com/books/content/support>选择书名就可以查看现有的勘误表。

侵权行为

互联网上的盗版是所有媒体都要面对的问题。Packt非常重视保护版权和许可证。如果你发现我们的作品在互联网上被非法复制，不管以什么形式，都请立即为我们提供位置地址或网站名称，以便我们可以寻求补救。

请把可疑盗版材料的链接发到copyright@packtpub.com。

非常感谢你帮助我们保护作者，以及保护我们给你带来有价值内容的能力。

问题

如果你对本书内容存有疑问，不管是哪个方面，都可以通过questions@packtpub.com联系我们，我们将尽最大努力来解决。

^① 本书中文版的勘误请到<http://www.it-ebooks.com.cn/book/2029>查看和提交。——编者注

电子书

扫描如下二维码，即可购买本书电子版。



目 录

第 1 章 JavaScript 简介	1	2.2 创建和初始化数组	32
1.1 JavaScript 数据结构与算法	1	2.3 添加元素	33
1.2 环境搭建	2	2.3.1 使用 push 方法	33
1.2.1 最简单的环境搭建	2	2.3.2 插入元素到数组首位	34
1.2.2 使用 Web 服务器 (XAMPP)	4	2.4 删除元素	34
1.2.3 使用 Node.js 搭建 Web 服务器	5	2.5 在任意位置添加或删除元素	36
1.3 JavaScript 基础	6	2.6 二维和多维数组	36
1.3.1 变量	7	2.6.1 迭代二维数组的元素	37
1.3.2 操作符	9	2.6.2 多维数组	38
1.3.3 真值和假值	11	2.7 JavaScript 的数组方法参考	39
1.3.4 相等操作符 (==和===)	12	2.7.1 数组合并	39
1.4 控制结构	14	2.7.2 迭代器函数	40
1.4.1 条件语句	14	2.7.3 ECMAScript 6 和数组的新 功能	42
1.4.2 循环	15	2.7.4 排序元素	46
1.5 函数	16	2.7.5 搜索	48
1.6 JavaScript 面向对象编程	17	2.7.6 输出数组为字符串	49
1.7 调试工具	18	2.8 类型数组	50
1.8 ECMAScript 概述	19	2.9 小结	51
1.9 ECMAScript 6 的功能	21	第 3 章 栈	52
1.9.1 用 let 替代 var 声明变量	21	3.1 栈数据结构	52
1.9.2 常量	23	3.1.1 创建栈	53
1.9.3 模板字面量	23	3.1.2 向栈添加元素	53
1.9.4 箭头函数	24	3.1.3 从栈移除元素	53
1.9.5 函数的参数默认值	24	3.1.4 查看栈顶元素	54
1.9.6 声明展开和剩余参数	25	3.1.5 检查栈是否为空	54
1.9.7 使用类进行面向对象编程	27	3.1.6 清空和打印栈元素	54
1.10 ECMAScript 7 的功能	29	3.2 ECMAScript 6 和 Stack 类	56
1.11 小结	30	3.3 用栈解决问题	59
第 2 章 数组	31	3.4 小结	61
2.1 为什么用数组	31		

第4章 队列	62	6.3.3 差集	97
4.1 队列数据结构	62	6.3.4 子集	98
4.2 创建队列	63	6.4 ES6——Set类	99
4.2.1 向队列添加元素	63	6.5 小结	101
4.2.2 从队列移除元素	63	第7章 字典和散列表	102
4.2.3 查看队列头元素	64	7.1 字典	102
4.2.4 检查队列是否为空	64	7.1.1 创建字典	102
4.2.5 打印队列元素	64	7.1.2 使用Dictionary类	105
4.3 用ECMAScript 6语法实现 的Queue类	66	7.2 散列表	106
4.4 优先队列	66	7.2.1 创建散列表	106
4.5 循环队列——击鼓传花	68	7.2.2 使用HashTable类	108
4.6 JavaScript任务队列	70	7.2.3 散列表和散列集合	109
4.7 小结	70	7.2.4 处理散列表中的冲突	109
第5章 链表	71	7.2.5 创建更好的散列函数	117
5.1 链表数据结构	71	7.3 ES6——Map类	118
5.2 创建链表	72	7.4 ES6——WeakMap类和WeakSet类	118
5.2.1 向链表尾部追加元素	73	7.5 小结	119
5.2.2 从链表中移除元素	75	第8章 树	120
5.2.3 在任意位置插入元素	77	8.1 树数据结构	120
5.2.4 实现其他方法	79	8.2 树的相关术语	121
5.3 双向链表	82	8.3 二叉树和二叉搜索树	121
5.3.1 在任意位置插入新元素	82	8.3.1 创建BinarySearchTree类	122
5.3.2 从任意位置移除元素	85	8.3.2 向树中插入一个键	123
5.4 循环链表	87	8.4 树的遍历	126
5.5 小结	88	8.4.1 中序遍历	126
第6章 集合	89	8.4.2 先序遍历	127
6.1 构建数据集	89	8.4.3 后序遍历	128
6.2 创建集合	89	8.5 搜索树中的值	129
6.2.1 has(value)方法	90	8.5.1 搜索最小值和最大值	130
6.2.2 add方法	91	8.5.2 搜索一个特定的值	131
6.2.3 remove和clear方法	91	8.5.3 移除一个节点	133
6.2.4 size方法	92	8.6 自平衡树	137
6.2.5 values方法	93	8.6.1 Adelson-Velskii-Landi树 (AVL树)	137
6.2.6 使用Set类	93	8.6.2 更多关于二叉树的知识	143
6.3 集合操作	94	8.7 小结	143
6.3.1 并集	94	第9章 图	144
6.3.2 交集	95	9.1 图的相关术语	144

9.2 图的表示	146	第 11 章 算法模式	190
9.2.1 邻接矩阵	146	11.1 递归	190
9.2.2 邻接表	147	11.1.1 JavaScript 调用栈大小的限制	191
9.2.3 关联矩阵	148	11.1.2 斐波那契数列	191
9.3 创建 Graph 类	148	11.2 动态规划	193
9.4 图的遍历	150	11.2.1 最少硬币找零问题	194
9.4.1 广度优先搜索	151	11.2.2 背包问题	196
9.4.2 深度优先搜索	156	11.2.3 最长公共子序列	198
9.5 最短路径算法	162	11.2.4 矩阵链相乘	200
9.5.1 Dijkstra 算法	163	11.3 贪心算法	202
9.5.2 Floyd-Warshall 算法	165	11.3.1 最少硬币找零问题	203
9.6 最小生成树	166	11.3.2 分数背包问题	204
9.6.1 Prim 算法	166	11.4 函数式编程简介	205
9.6.2 Kruskal 算法	168	11.4.1 函数式编程与命令式编程	205
9.7 小结	169	11.4.2 ES2015 和函数式编程	206
第 10 章 排序和搜索算法	170	11.4.3 JavaScript 函数式工具箱——map、filter 和 reduce	207
10.1 排序算法	170	11.4.4 JavaScript 函数式类库和数据结构	209
10.1.1 冒泡排序	171	11.5 小结	209
10.1.2 选择排序	174	第 12 章 算法复杂度	210
10.1.3 插入排序	175	12.1 大 O 表示法	210
10.1.4 归并排序	176	12.1.1 理解大 O 表示法	210
10.1.5 快速排序	179	12.1.2 时间复杂度比较	212
10.1.6 堆排序	183	12.1.3 NP 完全理论概述	214
10.1.7 计数排序、桶排序和基数排序（分布式排序）	186	12.2 用算法娱乐身心	216
10.2 搜索算法	187	12.3 小结	217
10.2.1 顺序搜索	187		
10.2.2 二分搜索	187		
10.3 小结	189		

JavaScript是一门非常强大的编程语言。它是最流行的编程语言，也是互联网上最卓越的语言之一。在GitHub(世界上最大的代码托管站点，<https://github.com>)上，托管了400 000多个JavaScript代码仓库(用JavaScript开发的项目数量也是最多的，并且还在逐年增长，参看<http://goo.gl/ZFx6mg>)。

JavaScript不仅可用于前端开发，也适用于后端开发，而Node.js就是其背后的技术。Node包(<http://www.npmjs.org/>)的数量也呈指数级增长。

要成为一名Web开发工程师，掌握JavaScript必不可少。

本章，你会学到JavaScript的语法和一些必要的基础，这样我们就可以开始开发自己的数据结构和算法了。本章内容如下：

- 环境搭建
- JavaScript基础
- 控制结构
- 函数
- JavaScript面向对象编程
- 调试工具
- ECMAScript 6和ECMAScript 7简介

1.1 JavaScript 数据结构与算法

在本书中，你将学习最常用的数据结构和算法。为什么用JavaScript来学习这些数据结构和算法呢？我们已经回答了这个问题。JavaScript非常受欢迎，作为函数式编程语言，它非常适合用来学习数据结构和算法。通过它来学习数据结构比C或Java这些标准语言更简单，学习新东西也会变得很有趣。谁说数据结构和算法只为C或Java这样的语言而生？在前端开发当中，你可能也需要实现这些语言。