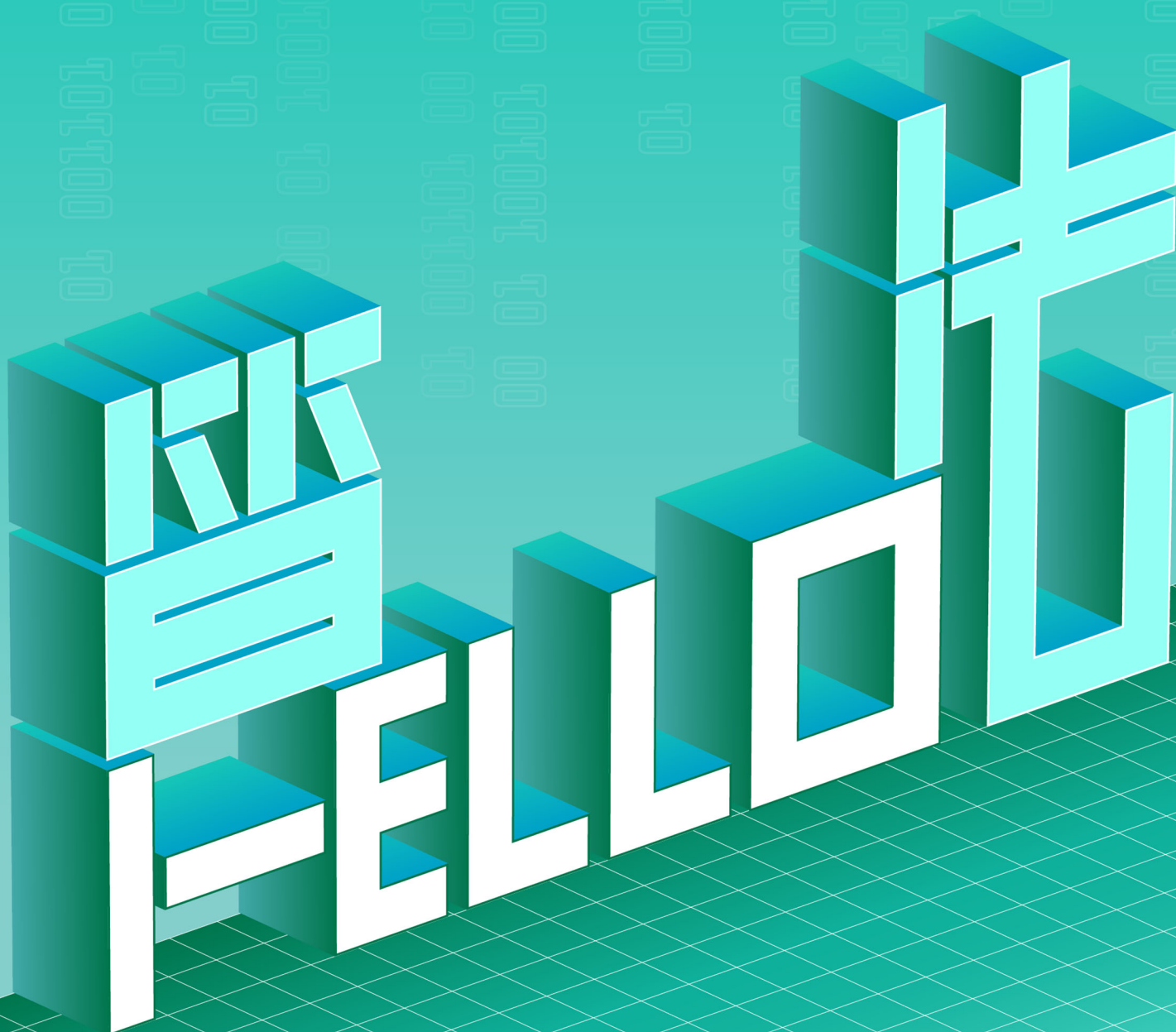


HELLO HELLO HELLO HELLO HELLO HELLO HELLO HELLO HELLO ➤ Hello 算法 |

动画图解、一键运行的数据结构与算法教程

靳宇栋 (@krahets)

著



Hello 算法

C 语言版

作者：靳宇栋 (@krahets)

代码审阅：宫兰景 (@Gonglja)



Release 1.2.0
2024-12-06

序

两年前，我在力扣上分享了“剑指 Offer”系列题解，受到了许多读者的鼓励和支持。在与读者交流期间，我最常被问的一个问题是“如何入门算法”。逐渐地，我对这个问题产生了浓厚的兴趣。

两眼一抹黑地刷题似乎是最受欢迎的方法，简单、直接且有效。然而刷题就如同玩“扫雷”游戏，自学能力强的人能够顺利将地雷逐个排掉，而基础不足的人很可能被炸得满头是包，并在挫折中步步退缩。通读教材也是一种常见做法，但对于面向求职的人来说，毕业论文、投递简历、准备笔试和面试已经消耗了大部分精力，啃厚重的书往往变成了一项艰巨的挑战。

如果你也面临类似的困扰，那么很幸运这本书“找”到了你。本书是我对这个问题给出的答案，即使不是最优解，也至少是一次积极的尝试。本书虽然不足以让你直接拿到 Offer，但会引导你探索数据结构与算法的“知识地图”，带你了解不同“地雷”的形状、大小和分布位置，让你掌握各种“排雷方法”。有了这些本领，相信你可以更加自如地刷题和阅读文献，逐步构建起完整的知识体系。

我深深赞同费曼教授所言：“Knowledge isn’t free. You have to pay attention.”从这个意义上看，这本书并非完全“免费”。为了不辜负你为本书所付出的宝贵“注意力”，我会竭尽所能，投入最大的“注意力”来完成本书的创作。

本人自知学疏才浅，书中内容虽然已经过一段时间的打磨，但一定仍有许多错误，恳请各位老师和同学批评指正。

本书中的代码附有可一键运行的源文件，托管于 github.com/krahets/hello-algo 仓库。
动画在 PDF 内的展示效果受限，可访问 www.hello-algo.com 网页版以获得更优的阅读体验。

推荐语

“一本通俗易懂的数据结构与算法入门书，引导读者手脑并用地学习，强烈推荐算法初学者阅读！”
——邓俊辉，清华大学计算机系教授

“如果我当年学数据结构与算法的时候有《Hello 算法》，学起来应该会简单 10 倍！”
——李沐，亚马逊资深首席科学家



计算机的出现给世界带来了巨大变革，它凭借高速的计算能力和出色的可编程性，成为了执行算法与处理数据的理想媒介。无论是电子游戏的逼真画面、自动驾驶的智能决策，还是 AlphaGo 的精彩棋局、ChatGPT 的自然交互，这些应用都是算法在计算机上的精妙演绎。

事实上，在计算机问世之前，算法和数据结构就已经存在于世界的各个角落。早期的算法相对简单，例如古代的计数方法和工具制作步骤等。随着文明的进步，算法逐渐变得更加精细和复杂。从巧夺天工的匠人技艺、到解放生产力的工业产品、再到宇宙运行的科学规律，几乎每一件平凡或令人惊叹的事物背后，都隐藏着精妙的算法思想。

同样，数据结构无处不在：大到社会网络，小到地铁线路，许多系统都可以建模为“图”；大到一个国家，小到一个家庭，社会的主要组织形式呈现出“树”的特征；冬天的衣服就像“栈”，最先穿上的最后才能脱下；羽毛球筒则如同“队列”，一端放入、另一端取出；字典就像一个“哈希表”，能够快速查找目标词条。

本书旨在通过清晰易懂的动画图解和可运行的代码示例，使读者理解算法和数据结构的核心概念，并能够通过编程来实现它们。在此基础上，本书致力于揭示算法在复杂世界中的生动体现，展现算法之美。希望本书能够帮助到你！

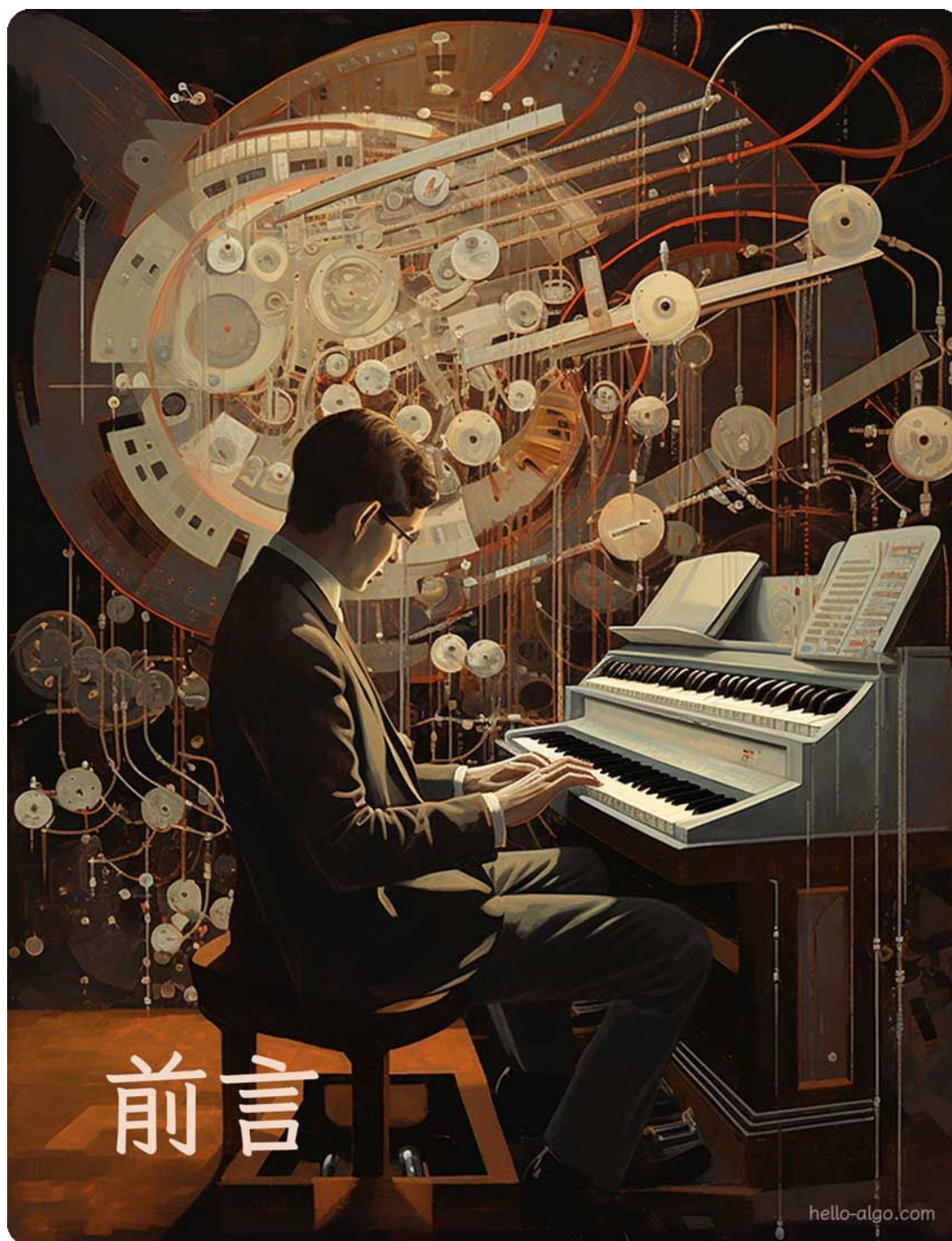
目 录

第 0 章 前言	1
0.1 关于本书	2
0.2 如何使用本书	4
0.3 小结	9
第 1 章 初识算法	10
1.1 算法无处不在	11
1.2 算法是什么	13
1.3 小结	15
第 2 章 复杂度分析	17
2.1 算法效率评估	18
2.2 迭代与递归	19
2.3 时间复杂度	28
2.4 空间复杂度	41
2.5 小结	49
第 3 章 数据结构	52
3.1 数据结构分类	53
3.2 基本数据类型	55
3.3 数字编码 *	56
3.4 字符编码 *	61
3.5 小结	65
第 4 章 数组与链表	67
4.1 数组	68
4.2 链表	73
4.3 列表	80
4.4 内存与缓存 *	84
4.5 小结	87
第 5 章 栈与队列	90
5.1 栈	91
5.2 队列	97
5.3 双向队列	103
5.4 小结	113
第 6 章 哈希表	114
6.1 哈希表	115
6.2 哈希冲突	121
6.3 哈希算法	132
6.4 小结	136
第 7 章 树	139
7.1 二叉树	140
7.2 二叉树遍历	146
7.3 二叉树数组表示	151
7.4 二叉搜索树	155
7.5 AVL 树 *	163
7.6 小结	174

第 8 章 堆	176
8.1 堆	177
8.2 建堆操作	184
8.3 Top-k 问题	187
8.4 小结	191
第 9 章 图	192
9.1 图	193
9.2 图的基础操作	197
9.3 图的遍历	205
9.4 小结	212
第 10 章 搜索	214
10.1 二分查找	215
10.2 二分查找插入点	218
10.3 二分查找边界	222
10.4 哈希优化策略	225
10.5 重识搜索算法	228
10.6 小结	230
第 11 章 排序	231
11.1 排序算法	232
11.2 选择排序	233
11.3 冒泡排序	236
11.4 插入排序	238
11.5 快速排序	241
11.6 归并排序	246
11.7 堆排序	250
11.8 桶排序	253
11.9 计数排序	256
11.10 基数排序	260
11.11 小结	263
第 12 章 分治	266
12.1 分治算法	267
12.2 分治搜索策略	270
12.3 构建二叉树问题	272
12.4 汉诺塔问题	277
12.5 小结	281
第 13 章 回溯	283
13.1 回溯算法	284
13.2 全排列问题	292
13.3 子集和问题	298
13.4 n 皇后问题	305
13.5 小结	309
第 14 章 动态规划	311
14.1 初探动态规划	312
14.2 动态规划问题特性	318
14.3 动态规划解题思路	323
14.4 0-1 背包问题	332
14.5 完全背包问题	339
14.6 编辑距离问题	349
14.7 小结	355

第 15 章 贪心	357
15.1 贪心算法	358
15.2 分数背包问题	361
15.3 最大容量问题	365
15.4 最大切分乘积问题	370
15.5 小结	373
第 16 章 附录	375
16.1 编程环境安装	376
16.2 一起参与创作	379
16.3 术语表	380

第 0 章 前言



Abstract

算法犹如美妙的交响乐，每一行代码都像韵律般流淌。
愿这本书在你的脑海中轻轻响起，留下独特而深刻的旋律。

0.1 关于本书

本项目旨在创建一本开源、免费、对新手友好的数据结构与算法入门教程。

- 全书采用动画图解，内容清晰易懂、学习曲线平滑，引导初学者探索数据结构与算法的知识地图。
- 源代码可一键运行，帮助读者在练习中提升编程技能，了解算法工作原理和数据结构底层实现。
- 提倡读者互助学习，欢迎大家在评论区提出问题与分享见解，在交流讨论中共同进步。

0.1.1 读者对象

若你是算法初学者，从未接触过算法，或者已经有一些刷题经验，对数据结构与算法有模糊的认识，在会与不会之间反复横跳，那么本书正是为你量身定制的！

如果你已经积累一定的刷题量，熟悉大部分题型，那么本书可助你回顾与梳理算法知识体系，仓库源代码可以当作“刷题工具库”或“算法字典”来使用。

若你是算法“大神”，我们期待收到你的宝贵建议，或者[一起参与创作](#)。

前置条件

你需要至少具备任一语言的编程基础，能够阅读和编写简单代码。

0.1.2 内容结构

本书的主要内容如图 0-1 所示。

- **复杂度分析**：数据结构和算法的评价维度与方法。时间复杂度和空间复杂度的推算方法、常见类型、示例等。
- **数据结构**：基本数据类型和数据结构的分类方法。数组、链表、栈、队列、哈希表、树、堆、图等数据结构的定义、优缺点、常用操作、常见类型、典型应用、实现方法等。
- **算法**：搜索、排序、分治、回溯、动态规划、贪心等算法的定义、优缺点、效率、应用场景、解题步骤和示例问题等。

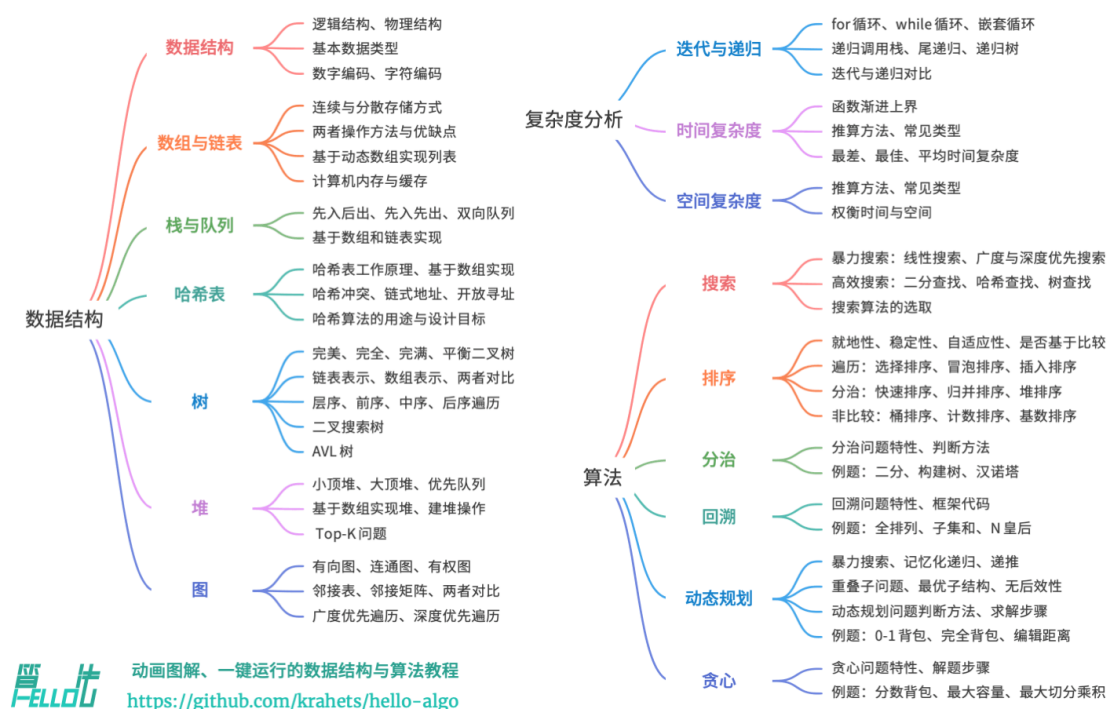


图 0-1 本书主要内容

0.1.3 致谢

本书在开源社区众多贡献者的共同努力下不断完善。感谢每一位投入时间与精力的撰稿人，他们是（按照 GitHub 自动生成的顺序）：krahets、coderonion、Gonglja、nuomi1、Reanon、justin-tse、hpstory、danielsss、curtishd、night-cruise、S-N-O-R-L-A-X、msk397、gvenusleo、khoaxuantu、RiverTwilight、rongyi、gyt95、zhuoqinyue、K3v123、Zuoxun、mingXta、hello-ikun、FangYuan33、GN-Yu、yuelinxin、longsizhuo、Cathay-Chen、guowei-gong、xBLACKICE、IsChristina、JoseHung、qualifier1024、QiLOL、pengchzn、Guanngxu、L-Super、WSL0809、Slone123c、lhxsm、yuan0221、what-is-me、theNefelibatas、longranger2、cy-by-side、xiongsp、JeffersonHuang、Transmigration-zhou、magentaqin、Wonderdch、malone6、xiaomiusa87、gaofer、bluebean-cloud、a16su、Shyam-Chen、nanlei、hongyun-robot、Phoenix0415、MolDuM、Nigh、he-weilai、junminhong、mgisr、iron-irax、yd-j、XiaChuerwu、XC-Zero、seven1240、SamJin98、wodray、reeswell、NI-SW、Horbin-Magician、Enlightenus、xjr7670、YangXuanyi、DullSword、boloboloda、iStig、qq909244296、jiaxianhua、wenjianmin、keshida、kilikilikid、lclc6、lwbaptx、liuxjerry、lucaswangdev、lyl625760、hts0000、gledfish、fbigm、echo1937、szu17dmy、dshlstarr、Yucao-cy、coderlef、czruby、bongbongbakudan、beintentional、ZongYangL、ZhongYuuu、luluxia、xb534、bitsmi、ElaBosak233、baagod、zhouLion、yishangzhang、yi427、yabo083、weibk、wangwang105、th1nk3r-ing、tao363、4yDX3906、syd168、steventimes、sslmj2020、smilelsb、siqyka、selear、sdshaoda、Xi-Row、popozhu、nuquist19、noobcodemaker、XiaoK29、chadyi、ZhongGuanbin、shanghai-Jerry、JackYang-hellobobo、Javesun99、lipusheng、BlindTerran、ShiMaRing、FreddieLi、FloranceYeh、iFleey、fanchenggang、gltianwen、goerll、Dr-XYZ、nedchu、curly210102、CuB3y0nd、

KraHsu、CarrotDLaw、youshaoXG、bubble9um、fanenr、eagleanurag、LifeGoesOnionOnionOnion、52coder、foursevenlove、KorsChen、hezhizhen、linzeyan、ZJKung、GaochaoZhu、hopkings2008、yang-le、Evilrabbit520、Turing-1024-Lee、thomasq0、Suremotoo、Allen-Scai、Risuntsy、Richard-Zhang1019、qingpeng9802、primexiao、nidhogfgg、lch0、MwumLi、martinx、ZnYang2018、hugtyftg、logan-qiu、psychelzh、Keynman、KeiichiKasai 和 0130w。

本书的代码审阅工作由 coderonion、curtishd、Gonglja、gvenusleo、hpstory、justin-tse、khoaxuantu、krahets、night-cruise、nuomi1、Reanon 和 rongyi 完成（按照首字母顺序排列）。感谢他们付出的时间与精力，正是他们确保了各语言代码的规范与统一。

在本书的创作过程中，我得到了许多人的帮助。

- 感谢我在公司的导师李汐博士，在一次畅谈中你鼓励我“快行动起来”，坚定了我写这本书的决心；
- 感谢我的女朋友泡泡作为本书的首位读者，从算法小白的角度提出许多宝贵建议，使得本书更适合新手阅读；
- 感谢腾宝、琦宝、飞宝为本书起了一个富有创意的名字，唤起大家写下第一行代码“Hello World!”的美好回忆；
- 感谢校铨在知识产权方面提供的专业帮助，这对本开源书的完善起到了重要作用；
- 感谢苏瑾为本书设计了精美的封面和 logo，并在我的强迫症的驱使下多次耐心修改；
- 感谢 @squidfunk 提供的排版建议，以及他开发的开源文档主题 [Material-for-MkDocs](#)。

在写作过程中，我阅读了许多关于数据结构与算法的教材和文章。这些作品为本书提供了优秀的范本，确保了本书内容的准确性与品质。在此感谢所有老师和前辈的杰出贡献！

本书倡导手脑并用的学习方式，在这一点上我深受《[动手学深度学习](#)》的启发。在此向各位读者强烈推荐这本优秀的著作。

衷心感谢我的父母，正是你们一直以来的支持与鼓励，让我有机会做这件富有趣味的事。

0.2 如何使用本书

Tip

为了获得最佳的阅读体验，建议你通读本节内容。

0.2.1 行文风格约定

- 标题后标注 * 的是选读章节，内容相对困难。如果你的时间有限，可以先跳过。
- 专业术语会使用黑体（纸质版和 PDF 版）或添加下划线（网页版），例如数组（array）。建议记住它们，以便阅读文献。
- 重点内容和总结性语句会 **加粗**，这类文字值得特别关注。
- 有特指含义的词句会使用“引号”标注，以避免歧义。
- 当涉及编程语言之间不一致的名词时，本书均以 Python 为准，例如使用 `None` 来表示“空”。
- 本书部分放弃了编程语言的注释规范，以换取更加紧凑的内容排版。注释主要分为三种类型：标题注释、内容注释、多行注释。

```
/* 标题注释，用于标注函数、类、测试样例等 */

// 内容注释，用于详解代码

/**
 * 多行
 * 注释
 */
```

0.2.2 在动画图解中高效学习

相较于文字，视频和图片具有更高的信息密度和结构化程度，更易于理解。在本书中，**重点和难点知识将主要通过动画以图解形式展示**，而文字则作为解释与补充。

如果你在阅读本书时，发现某段内容提供了如图 0-2 所示的动画图解，**请以图为主、以文字为辅**，综合两者来理解内容。

9.2 图基础操作

- 删除顶点：在邻接矩阵中删除一行一列。当删除首行首列时达到最差情况，需要将 $(n-1)^2$ 个元素“向左上移动”，从而使用 $O(n^2)$ 时间。
- 初始化：传入 n 个顶点，初始化长度为 n 的顶点列表 `vertices`，使用 $O(n)$ 时间；初始化 $n \times n$ 大小的邻接矩阵 `adjMat`，使用 $O(n^2)$ 时间。

初始化邻接矩阵

图

邻接矩阵

图 9-7 邻接矩阵的初始化、增删边、增删顶点

以下是基于邻接矩阵表示图的实现代码：

Python C++ Java C# Go Swift JS TS Dart Rust C Zig

graph_adjacency_matrix.py

图 0-2 动画图解示例

0.2.3 在代码实践中加深理解

本书的配套代码托管在 [GitHub 仓库](#)。如图 0-3 所示，源代码附有测试样例，可一键运行。

如果时间允许，建议你参照代码自行敲一遍。如果学习时间有限，请至少通读并运行所有代码。

与阅读代码相比，编写代码的过程往往能带来更多收获。**动手学，才是真的学。**

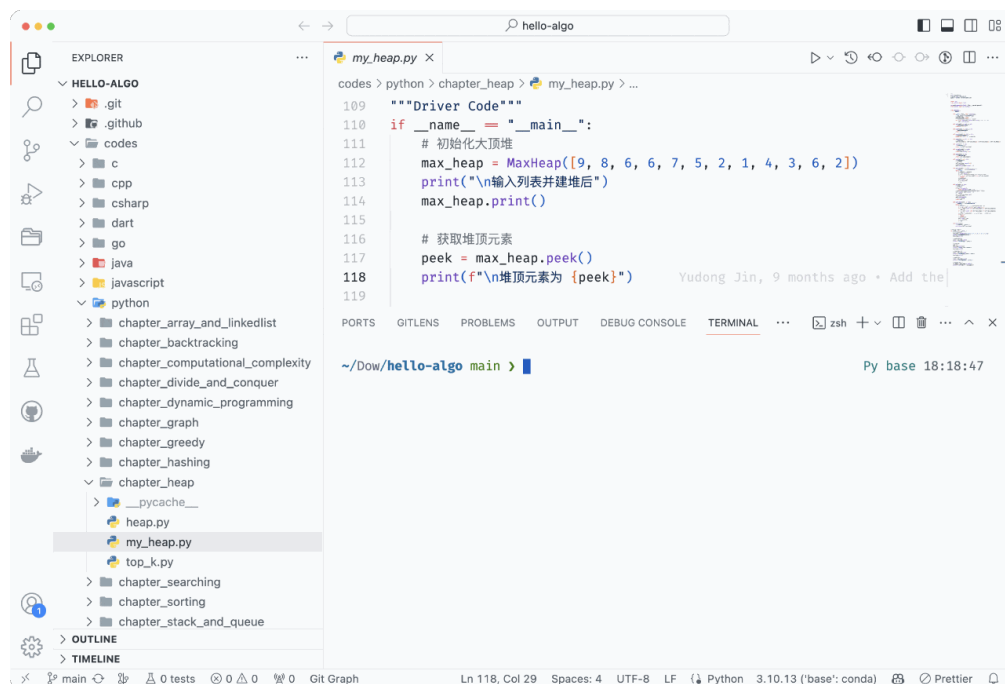


图 0-3 运行代码示例

运行代码的前置工作主要分为三步。

第一步：安装本地编程环境。请参照附录所示的教程进行安装，如果已安装，则可跳过此步骤。

第二步：克隆或下载代码仓库。前往 [GitHub 仓库](https://github.com/krahets/hello-algo)。如果已经安装 [Git](https://git-scm.com/)，可以通过以下命令克隆本仓库：

```
git clone https://github.com/krahets/hello-algo.git
```

当然，你也可以在图 0-4 所示的位置，点击“Download ZIP”按钮直接下载代码压缩包，然后在本地解压即可。

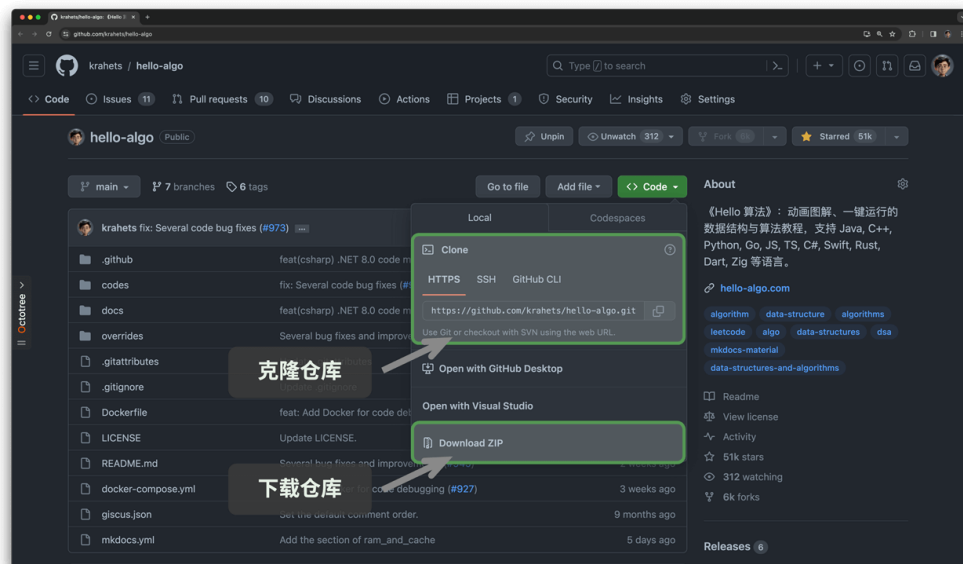


图 0-4 克隆仓库与下载代码

第三步：运行源代码。如图 0-5 所示，对于顶部标有文件名称的代码块，我们可以在仓库的 `codes` 文件夹内找到对应的源代码文件。源代码文件可一键运行，将帮助你节省不必要的调试时间，让你能够专注于学习内容。

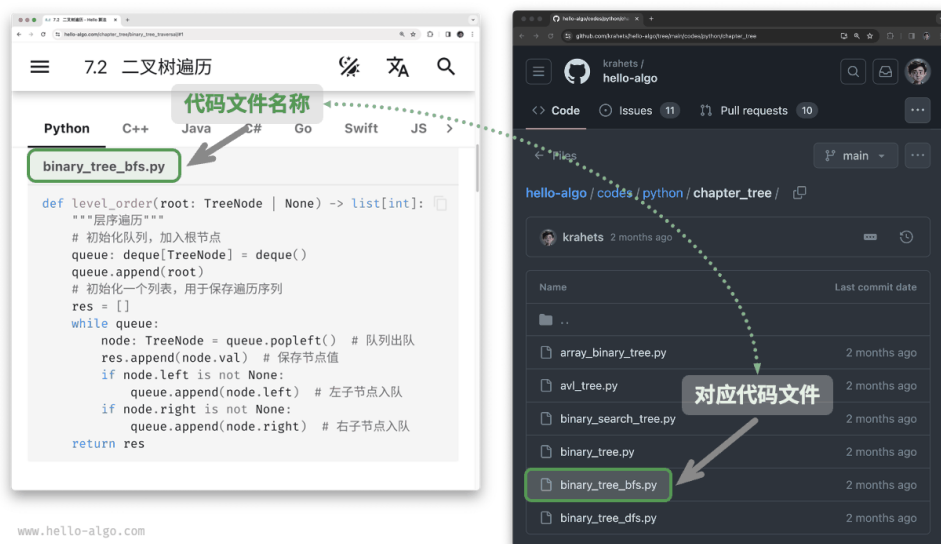


图 0-5 代码块与对应的源代码文件

除了本地运行代码，网页版还支持 Python 代码的可视化运行（基于 [pythontutor](https://pythontutor.com) 实现）。如图 0-6 所示，你可以点击代码块下方的“可视化运行”来展开视图，观察算法代码的执行过程；也可以点击“全屏观看”，以获得更好的浏览体验。

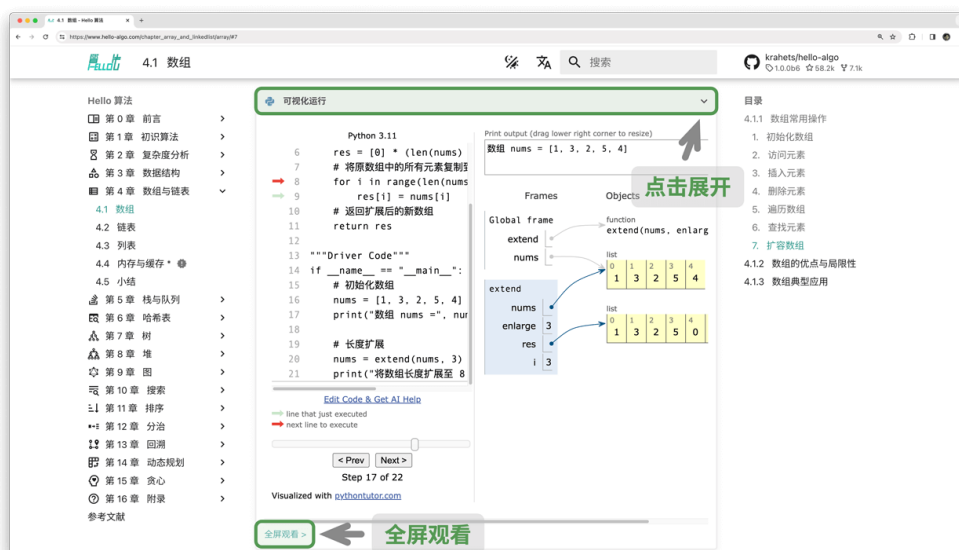


图 0-6 Python 代码的可视化运行

0.2.4 在提问讨论中共同成长

在阅读本书时，请不要轻易跳过那些没学明白的知识点。**欢迎在评论区提出你的问题**，我和小伙伴们将竭诚为你解答，一般情况下可在两天内回复。

如图 0-7 所示，网页版每个章节的底部都配有评论区。希望你能多关注评论区的内容。一方面，你可以了解大家遇到的问题，从而查漏补缺，激发更深入的思考。另一方面，期待你能慷慨地回答其他小伙伴的问题，分享你的见解，帮助他人进步。

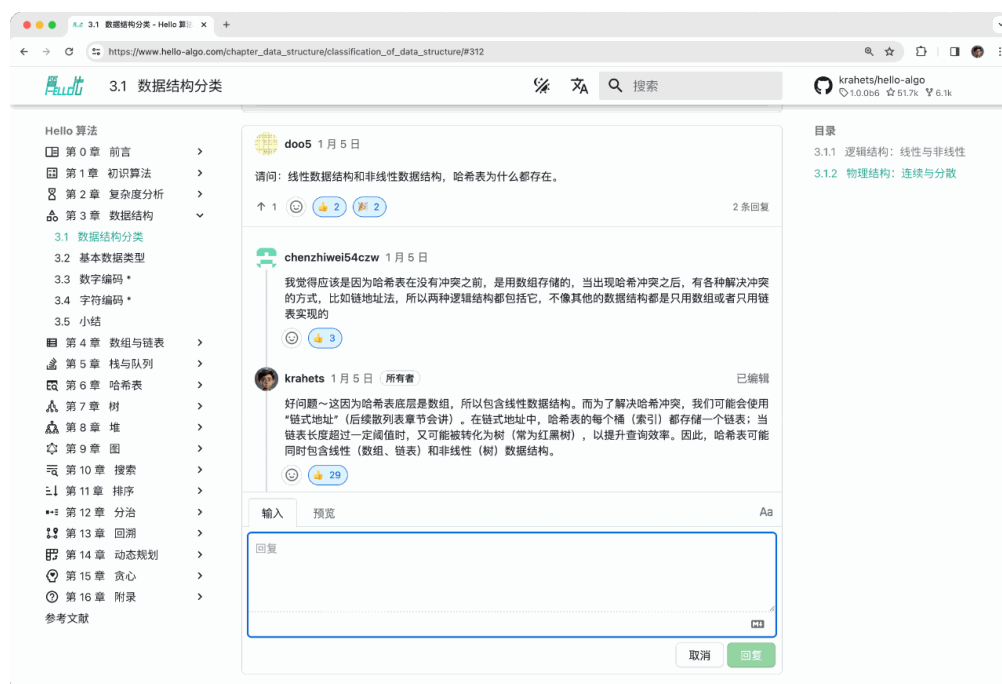


图 0-7 评论区示例

0.2.5 算法学习路线

从总体上看，我们可以将学习数据结构与算法的过程划分为三个阶段。

1. **阶段一：算法入门。**我们需要熟悉各种数据结构的特点和用法，学习不同算法的原理、流程、用途和效率等方面的内容。
2. **阶段二：刷算法题。**建议从热门题目开刷，先积累至少 100 道题目，熟悉主流的算法问题。初次刷题时，“知识遗忘”可能是一个挑战，但请放心，这是很正常的。我们可以按照“艾宾浩斯遗忘曲线”来复习题目，通常在进行 3~5 轮的重复后，就能将其牢记在心。推荐的题单和刷题计划请见此 [GitHub 仓库](#)。
3. **阶段三：搭建知识体系。**在学习方面，我们可以阅读算法专栏文章、解题框架和算法教材，以不断丰富知识体系。在刷题方面，可以尝试采用进阶刷题策略，如按专题分类、一题多解、一解多题等，相关的刷题心得可以在各个社区找到。

如图 0-8 所示，本书内容主要涵盖“阶段一”，旨在帮助你更高效地展开阶段二和阶段三的学习。

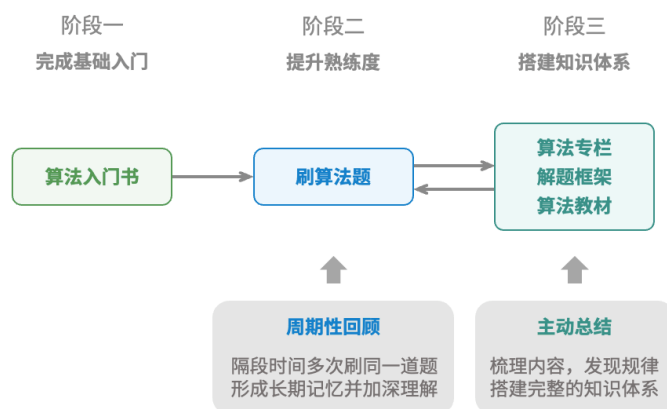


图 0-8 算法学习路线

0.3 小结

- 本书的主要受众是算法初学者。如果你已有一定基础，本书能帮助你系统回顾算法知识，书中源代码也可作为“刷题工具库”使用。
- 书中内容主要包括复杂度分析、数据结构和算法三部分，涵盖了该领域的大部分主题。
- 对于算法新手，在初学阶段阅读一本入门书至关重要，可以少走许多弯路。
- 书中的动画图解通常用于介绍重点和难点知识。阅读本书时，应给予这些内容更多关注。
- 实践乃学习编程之最佳途径。强烈建议运行源代码并亲自敲代码。
- 本书网页版的每个章节都设有评论区，欢迎随时分享你的疑惑与见解。