

Оглавление

Предисловие	8
Предисловие от издательства	10
Отзывы и пожелания.....	10
Список опечаток.....	10
Нарушение авторских прав	10
Глава 0. Введение	11
0.1. О книге.....	12
0.2. Как использовать эту книгу	15
0.3. Резюме.....	23
Глава 1. Введение в алгоритмы	24
1.1. Алгоритмы повсюду	25
1.2. Что такое алгоритм.....	30
1.3. Резюме.....	32
Глава 2. Анализ сложности	34
2.1. Оценка эффективности алгоритмов	35
2.2. Итерация и рекурсия	37
2.3. Временная сложность.....	46
2.4. Пространственная сложность	61
2.5. Резюме.....	67
Глава 3. Структуры данных	70
3.1. Классификация структур данных	71
3.2. Основные типы данных	74
3.3. Кодирование чисел*	76
3.4. Кодирование символов*	80
3.5. Резюме.....	89
Глава 4. Массивы и списки	92
4.1. Массивы.....	93
4.2. Связные списки.....	99
4.3. Списки	105
4.4. Память и кеш*	109
4.5. Резюме.....	113

Глава 5. Стек и очередь	118
5.1. Стек	119
5.2. Очередь	126
5.3. Двусторонняя очередь	133
5.4. Резюме	144
Глава 6. Хеш-таблицы	146
6.1. Хеш-таблицы	147
6.2. Хеш-коллизии	152
6.3. Алгоритмы хеширования	161
6.4. Резюме	167
Глава 7. Деревья	170
7.1. Двоичные деревья	171
7.2. Обход двоичного дерева	177
7.3. Представление двоичного дерева с помощью массива	184
7.4. Двоичное дерево поиска	189
7.5. AVL-дерево*	199
7.6. Резюме	210
Глава 8. Куча	213
8.1. Куча	214
8.2. Построение кучи	226
8.3. Поиск k наибольших элементов	229
8.4. Резюме	234
Глава 9. Графы	236
9.1. Графы	237
9.2. Основные операции с графами	241
9.3. Обход графа	249
9.4. Резюме	262
Глава 10. Поиск	264
10.1. Двоичный поиск	265
10.2. Вставка с использованием двоичного поиска	270
10.3. Двоичный поиск границ	276
10.4. Стратегии оптимизации хеширования	279
10.5. Переосмысление алгоритмов поиска	282
10.6. Резюме	286
Глава 11. Сортировка	287
11.1. Алгоритмы сортировки	288
11.2. Сортировка выбором	289
11.3. Сортировка пузырьком	294
11.4. Сортировка вставками	298

11.5. Быстрая сортировка.....	301
11.6. Сортировка слиянием.....	310
11.7. Пирамидальная сортировка.....	316
11.8. Блочная сортировка.....	324
11.9. Сортировка подсчетом.....	327
11.10. Поразрядная сортировка.....	334
11.11. Резюме.....	337
Глава 12. Разделяй и властвуй.....	342
12.1. Стратегия «разделяй и властвуй».....	343
12.2. Применение стратегии «разделяй и властвуй» для поиска.....	347
12.3. Задача построения двоичного дерева.....	350
12.4. Задача о Ханойских башнях.....	357
12.5. Резюме.....	363
Глава 13. Поиск с возвратом.....	365
13.1. Алгоритмы поиска с возвратом.....	366
13.2. Задача о перестановках.....	377
13.3. Задача о сумме подмножеств.....	383
13.4. Задача об n ферзях.....	391
13.5. Резюме.....	395
Глава 14. Динамическое программирование.....	397
14.1. Введение в динамическое программирование.....	398
14.2. Особенности задач динамического программирования.....	404
14.3. Подход к решению задач динамического программирования.....	410
14.4. Задача о рюкзаке 0-1.....	422
14.5. Задача о полном рюкзаке.....	434
14.6. Задача расстояния редактирования.....	448
14.7. Резюме.....	457
Глава 15. Жадность.....	460
15.1. Жадные алгоритмы.....	461
15.2. Задача о дробном рюкзаке.....	466
15.3. Задача о максимальной вместимости.....	469
15.4. Задача о максимальном произведении разбиения.....	478
15.5. Резюме.....	482
Глава 16. Приложение.....	483
16.1. Установка программной среды.....	484
16.2. Совместная разработка.....	486
16.3. Глоссарий.....	488

Предисловие



Два года назад я опубликовал на платформе LeetCode сборник задач по книге Coding Interviews, который получил хорошие отклики от читателей. В ходе общения с ними мне чаще всего задавали вопрос: «С чего начать изучение алгоритмов?» Постепенно этот вопрос начал все больше меня занимать.

Решение задач наугад, вслепую является, пожалуй, самым распространенным методом: он прост, прямолинеен и результативен. Однако этот процесс напоминает компьютерную игру «Сапер»: люди с высокой способностью к самообучению могут без особых проблем разминировать все клетки. А те, кому

не хватает базовой подготовки, могут легко подорваться и быстро потерять мотивацию. Другой распространенный подход – читать учебники от корки до корки. Но у тех, кто готовится к устройству на работу, значительная часть времени и сил уходит на получение диплома, составление резюме, подготовку к тестам и собеседованиям. В таких условиях чтение объемных книг превращается в серьезное испытание.

Если вы сталкиваетесь с подобными трудностями, то считайте большой удачей, что эта книга «нашла» вас. Она представляет собой мой ответ на вопрос «Как войти в мир алгоритмов?». Возможно, это не самый совершенный ответ, но это точно искренняя попытка. Эта книга не гарантирует вам быстрого получения хорошего офера, но она поможет построить «карту знаний» по структурам данных и алгоритмам, познакомит с различными «минами», их формой, размером и расположением, а также научит «методам разминирования». Овладев этими знаниями, вы сможете увереннее решать задачи и читать научные статьи, постепенно формируя целостную систему знаний.

Я полностью согласен со словами Ричарда Фейнмана: «Знания не бесплатны. За них нужно платить вниманием». В этом смысле эта книга вовсе не бесплатна. И чтобы оправдать ваше драгоценное внимание, я вложу максимум усилий в ее создание.

Осознавая ограниченность собственных знаний и опыта, я понимаю, что, несмотря на многократные доработки, в книге все же могут оставаться ошибки. Буду искренне признателен за критику и замечания от преподавателей и студентов.

Исходные коды из книги доступны в виде исполняемых файлов в репозитории по адресу github.com/krahets/hello-algo.

Формат PDF не очень хорошо подходит для анимированных изображений, поэтому для лучшего восприятия можно обратиться к веб-версии по адресу <https://www.hello-algo.com/en> (на англ.).

Предисловие от издательства

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв на нашем сайте www.dmkpress.com, зайдя на страницу книги и оставив комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com; при этом укажите название книги в теме письма.

Если вы являетесь экспертом в какой-либо области и заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы обеспечить высокое качество наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в основном тексте или программном коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от недопонимания и поможете нам улучшить последующие издания этой книги.

Если вы найдете какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу dmkpress@gmail.com, и мы исправим это в следующих тиражах.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательство «ДМК Пресс» очень серьезно относится к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконной публикацией какой-либо из наших книг, пожалуйста, пришлите нам ссылку на интернет-ресурс, чтобы мы могли применить санкции.

Ссылку на подозрительные материалы можно прислать по адресу dmkpress@gmail.com.

Мы высоко ценим любую помощь по защите наших авторов, благодаря которой мы можем предоставлять вам качественные материалы.

Глава 0

Введение



Абстракция

Алгоритм подобен прекрасной симфонии, где каждая строка кода звучит как ритм.

Пусть эта книга зазвучит в твоём сознании легкой мелодией и оставит особый, глубокий след.

0.1. О КНИГЕ

Этот проект задуман как открытое, бесплатное и дружелюбное к новичкам введение в структуры данных и алгоритмы.

1. В книге используются анимационные иллюстрации. Материал изложен ясно и последовательно, что облегчает освоение и помогает начинающим построить «карту знаний» по структурам данных и алгоритмам.
2. Исходный код можно запустить одним кликом, что позволяет тренироваться, развивать навыки программирования и формировать понимание принципов работы алгоритмов и реализации структур данных на фундаментальном уровне.
3. Мы призываем к взаимопомощи читателей: задавайте вопросы и делитесь идеями в комментариях. Обсуждения помогают двигаться вперед всем вместе.

0.1.1. Целевая аудитория

Если вы новичок в алгоритмах, никогда с ними не сталкивались или у вас есть некоторый опыт решения задач, но еще нет четкого понимания структур данных и алгоритмов, эта книга создана специально для вас!

Если у вас уже есть определенный опыт решения задач и вы знакомы с большинством типов задач, эта книга поможет вам освежить и систематизировать знания об алгоритмах. Исходный код может служить набором инструментов для решения задач или алгоритмическим словарем.

Если вы владеете алгоритмами на экспертном уровне, мы будем рады вашим ценным советам или совместному участию в создании книги.

Предварительные требования

Вам необходимо иметь базовые навыки программирования на любом языке, а также уметь читать и писать простой код.

0.1.2. Структура книги

Основное содержание книги представлено на рис. 0.1.

1. **Анализ сложности:** критерии и методы оценки структур данных и алгоритмов. Методы расчета временной и пространственной сложности, распространенные типы, примеры и т. д.



Рис. 0.1. Структура книги

2. **Структуры данных:** классификация основных типов данных и структур данных. Определение, преимущества и недостатки, основные операции, распространенные типы, типичные приложения и методы реализации массивов, списков, стеков, очередей, хеш-таблиц, деревьев, куч, графов и т. д.
3. **Алгоритмы:** определение, преимущества и недостатки, эффективность, области применения, этапы решения и примеры задач для поиска, сортировки, алгоритма «разделяй и властвуй», обратного поиска, динамического программирования, жадных алгоритмов и т. д.

0.1.3. Благодарности

Эта книга постоянно совершенствуется благодаря совместным усилиям множества участников открытого сообщества. Благодарим каждого автора, вложившего свое время и силы. Имена перечислены в порядке, автоматически сгенерированном GitHub: 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, Jefferson-Huang, Transmigration-zhou, magentaqin, Wonderdch, malone6, xiaomiusa87, gaofer, bluebean-cloud, a16su, Shyam-Chen, nanlei, hongyun-robot, Phoenix0415, MolDuM, Nigh, he-weilai, junminhong, mgisir, iron-irax, yd-j, XiaChuerwu, XC-Zero, seven1240, SamJin98, wodray, reeswell, NI-SW, Horbin-Magician, Enlightenus, xjr7670, YangXuan-yi, DullSword, boloboloda, iStig, qq909244296, jiaxianhua, wenjianmin, keshida, kilikilikid, lclc6, lwbaptx, liuxjerry, lucaswangdev, lyl625760, hts0000, gledfish, fbigm, echo1937, szu17dmy, dshlstar, Yucuo-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, iF-leey, fanchenggang, gltianwen, goerll, Dr-XYZ, nedchu, curly210102, CuB3yOnd, 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, nidhoggfgg, 1ch0, MwumLi, martinx, ZnYang2018, hugtyftg, logan-qiu, psychelzh, Keynman, KeiichiKasai и 0130w.

Рецензирование кода книги выполнили coderonion, curtishd, Gonglja, gvenusleo, hpstory, justin-tse, khoaxuantu, krahets, night-cruise, nuomi1, Reanon и rongyi (в алфавитном порядке). Благодарим их за потраченное время и усилия, которые обеспечили стандартизацию и единообразие кода на различных языках.

В процессе создания этой книги мне помогало много людей.

1. Благодарю моего наставника в компании, доктора Ли Си (Li Xi), который в одной из бесед вдохновил меня «действовать быстро», что укрепило мою решимость написать эту книгу.
2. Благодарю мою девушку Пао Пао (Pao Pao), которая, будучи первым читателем книги, дала множество ценных советов с точки зрения новичка в алгоритмах, что сделало книгу более понятной и доступной.

3. Благодарю Тен Бао (Teng Bao), Ци Бао (Qi Bao) и Фей Бао (Fei Bao) за креативное название книги, которое навевает приятные воспоминания о написании первой строки кода «Hello World!».
4. Благодарю Сяо Цюань (Xiao Quan) за профессиональную помощь в вопросах интеллектуальной собственности, что сыграло важную роль в совершенствовании этой открытой книги.
5. Благодарю Су Тун (Su Tong) за дизайн обложки и логотипа книги, а также за терпение при многократных исправлениях по моим просьбам.
6. Благодарю @squidfunk за советы по оформлению и за разработку открытой темы документации Material-for-MkDocs.

В процессе написания я ознакомился с множеством учебников и статей по структурам данных и алгоритмам. Эти работы послужили отличным образцом для этой книги, обеспечив ее точность и качество. Выражаю благодарность всем преподавателям и предшественникам за их выдающийся вклад!

Настоящая книга пропагандирует метод обучения, сочетающий умственную и практическую деятельность, на который меня вдохновила книга Dive into Deep Learning («Погружение в глубокое обучение», на англ. языке). Настоятельно рекомендую эту замечательную работу всем читателям.

Сердечно благодарю своих родителей, ведь именно ваша постоянная поддержка и ободрение дали мне возможность заняться этим увлекательным делом.

0.2. Как использовать эту книгу

Совет

Для наилучшего восприятия материала рекомендуется внимательно прочитать данный раздел.

0.2.1. Стиль изложения

- Главы, имеющие символ * после заголовка, являются дополнительными и содержат более сложный материал. Если у вас ограничено время, можно их пропустить.
- Профессиональные термины выделяются полужирным шрифтом (в печатной и PDF-версии) или подчеркиванием (в веб-версии), например **массив** (array). Рекомендуется запоминать их для удобства чтения литературы.
- Важные моменты и обобщающие фразы выделяются **полужирным шрифтом**, на такие тексты следует обращать особое внимание.
- При упоминании терминов, различающихся в разных языках программирования, в качестве стандарта используется Python, например `None` для обозначения «пустого значения».
- В некоторых местах книга отходит от стандартов комментирования программного кода ради более компактного оформления. Комментарии делятся на три типа: заголовочные, содержательные и многострочные.

""" Заголовочные комментарии, используются для обозначения функций, классов, тестовых примеров и т.д. """

Содержательные комментарии, используются для пояснения кода.

"""

Многострочные
комментарии.

"""

0.2.2. Эффективное обучение с помощью анимированных иллюстраций

Видео и изображения обладают более высокой плотностью информации и структурированностью по сравнению с текстом, что облегчает понимание. В этой книге **ключевые и сложные моменты в основном представлены в виде анимированных иллюстраций**, а текстовая информация служит пояснением и дополнением.

Если какой-либо раздел в книге сопровождается анимационной иллюстрацией, как на рис. 0.2, **используйте иллюстрацию в качестве основного источника информации, а текст – в качестве вспомогательного.**

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 添加边 删除边 添加顶点 删除顶点

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
```

Рис. 0.2. Пример анимационной иллюстрации. Шаг 1

9.2 图基础操作

目录

- 9.2.1 基于邻接矩阵的实现
- 9.2.2 基于邻接表的实现
- 9.2.3 效率对比

初始化邻接矩阵 添加边 删除边 添加顶点 删除顶点

图

邻接矩阵

Step 2

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

9.2 图基础操作

目录

- 9.2.1 基于邻接矩阵的实现
- 9.2.2 基于邻接表的实现
- 9.2.3 效率对比

初始化邻接矩阵 添加边 删除边 添加顶点 删除顶点

图

邻接矩阵

Step 3

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

Рис. 0.2. Продолжение. Шаги 2–3

9.2 图基础操作

Hello 算法

- 第 0 章 前言
- 第 1 章 初识算法
- 第 2 章 复杂度分析
- 第 3 章 数据结构
- 第 4 章 数组与链表
- 第 5 章 栈与队列
- 第 6 章 哈希表
- 第 7 章 树
- 第 8 章 堆
- 第 9 章 图
 - 9.1 图
 - 9.2 图基础操作
 - 9.3 图的遍历
 - 9.4 小结
 - 第 10 章 搜索
 - 第 11 章 排序
 - 第 12 章 分治
 - 第 13 章 回溯
 - 第 14 章 动态规划
 - 第 15 章 贪心
 - 第 16 章 附录
- 参考文献

- 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 添加边 删除边 **添加顶点** 删除顶点

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`

9.2 图基础操作

Hello 算法

- 第 0 章 前言
- 第 1 章 初识算法
- 第 2 章 复杂度分析
- 第 3 章 数据结构
- 第 4 章 数组与链表
- 第 5 章 栈与队列
- 第 6 章 哈希表
- 第 7 章 树
- 第 8 章 堆
- 第 9 章 图
 - 9.1 图
 - 9.2 图基础操作
 - 9.3 图的遍历
 - 9.4 小结
 - 第 10 章 搜索
 - 第 11 章 排序
 - 第 12 章 分治
 - 第 13 章 回溯
 - 第 14 章 动态规划
 - 第 15 章 贪心
 - 第 16 章 附录
- 参考文献

- 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 添加边 删除边 添加顶点 **删除顶点**

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`

Рис. 0.2. Окончание. Шаги 4–5

0.2.3. Углубление понимания через практику написания кода

Сопроводительный код размещен в репозитории GitHub, он содержит тестовые примеры и может быть запущен одним нажатием кнопки, как показано на рис. 0.3.

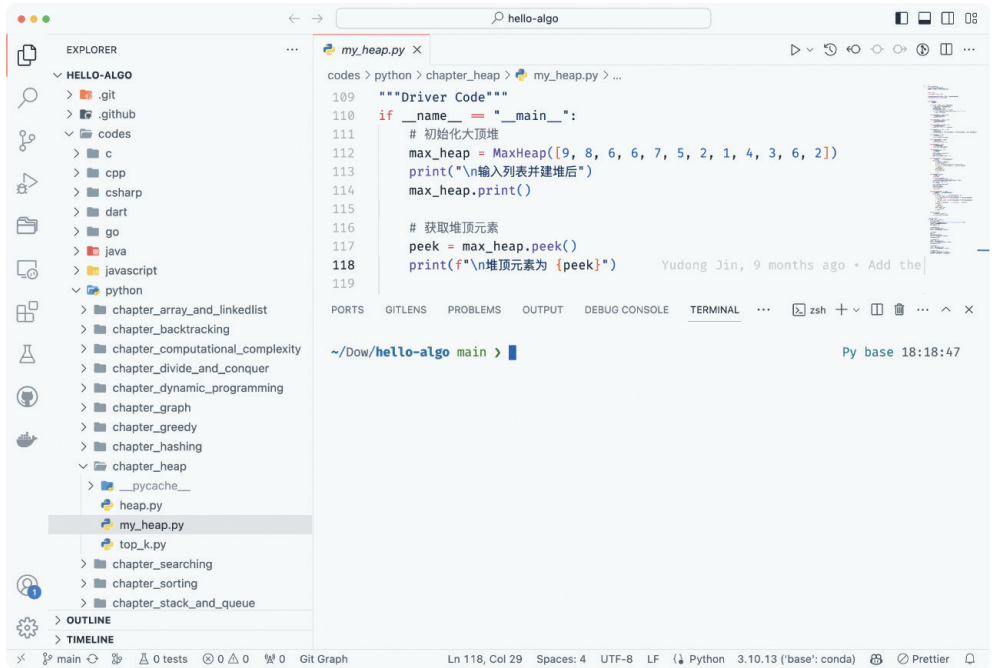


Рис. 0.3. Запуск примера кода

Если позволяет время, **рекомендуется самостоятельно набирать код**. Если время ограничено, по крайней мере **просмотрите и выполните весь код**.

Процесс написания кода приносит больше пользы, чем его чтение. **Настоящее обучение – это обучение на практике**.

Предварительная подготовка для запуска кода включает три этапа.

1. **Установка локальной среды программирования**. Следуйте инструкциям программы установки. Если среда уже установлена, этот шаг можно пропустить.
2. **Клонирование или загрузка репозитория кода**. Перейдите в репозиторий GitHub. Если у вас установлена утилита Git, можно клонировать репозиторий с помощью следующей команды:

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

Либо можно нажать кнопку **Download ZIP** (Скачать ZIP-архив), как показано на рис. 0.4, загрузить архив с кодом и затем распаковать его на локальном компьютере.

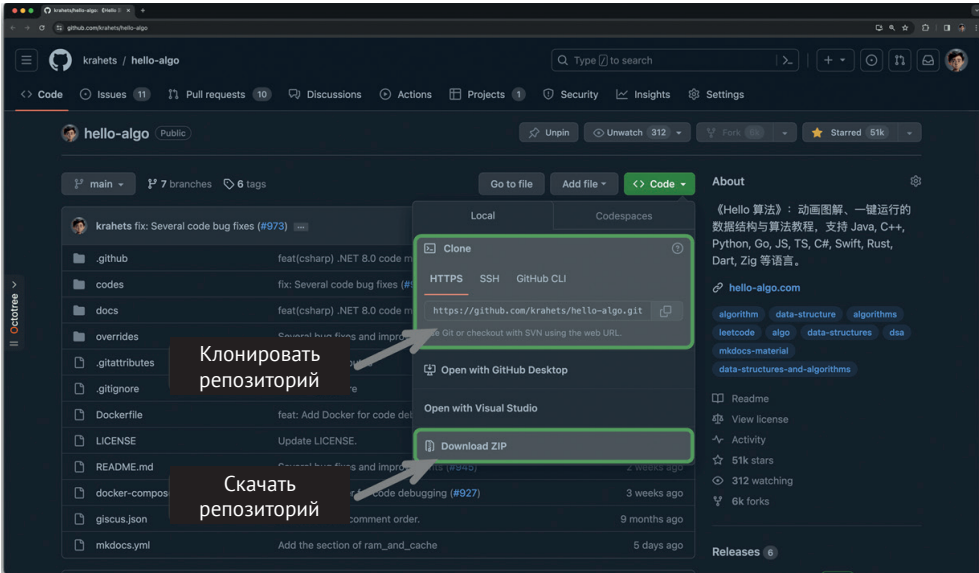


Рис. 0.4. Клонирование репозитория и загрузка кода

3. **Запуск исходного кода.** Если для блока кода в книге указано имя файла, этот файл можно найти в папке codes репозитория, как показано на рис. 0.5. Исходный код можно запустить одним нажатием, что поможет вам сэкономить время на отладку и сосредоточиться на изучении материала.

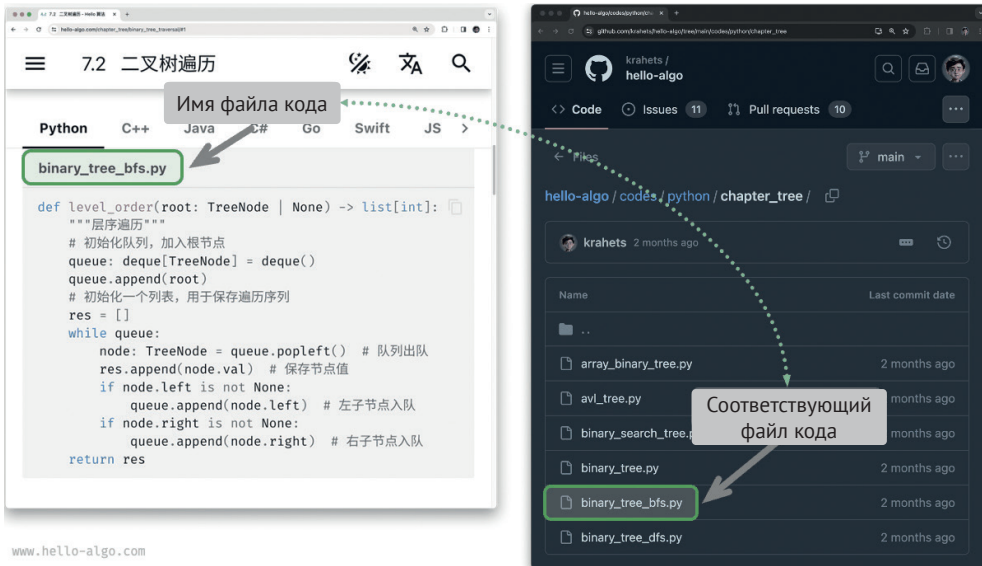


Рис. 0.5. Блоки кода и соответствующие файлы исходного кода

Конец ознакомительного фрагмента.

Приобрести книгу можно

в интернет-магазине

«Электронный универс»

e-Univers.ru