

КРАТКОЕ ОГЛАВЛЕНИЕ

Оглавление	6
Об авторе	16
Благодарности	16
Введение	17
Глава 1. Основы алгоритмизации	24
Глава 2. Численные алгоритмы	45
Глава 3. Связные списки	72
Глава 4. Массивы.....	96
Глава 5. Стеки и очереди	119
Глава 6. Сортировка.....	136
Глава 7. Поиск	163
Глава 8. Хеш-таблицы	168
Глава 9. Рекурсия	181
Глава 10. Деревья	215
Глава 11. Сбалансированные деревья.....	257
Глава 12. Деревья принятия решений	274
Глава 13. Основные сетевые алгоритмы.....	296
Глава 14. Дополнительные сетевые алгоритмы.....	324
Глава 15. Строковые алгоритмы.....	345
Глава 16. Криптография.....	365
Глава 17. Теория вычислительной сложности.....	387
Глава 18. Распределенные алгоритмы	402
Глава 19. Головоломки, встречающиеся на собеседованиях	432
Приложение А.....	442
Приложение Б.....	453
Глоссарий	522
Указатель	536

ОГЛАВЛЕНИЕ

Об авторе	16
Благодарности	16
Введение	17
Выбор алгоритма.....	18
Для кого предназначена книга	19
Как извлечь наибольшую пользу из книги.....	19
Сайты с материалами книги	20
Структура книги	20
Что нужно для работы с книгой	22
Условные обозначения.....	22
Обратная связь	23
Глава 1. Основы алгоритмизации	24
Метод	24
Алгоритм и структура данных.....	25
Псевдокод.....	25
Свойства алгоритма	28
Асимптотическая сложность алгоритма.....	29
Обычные функции рабочего цикла.....	33
Визуализация функций.....	38
Практические рекомендации.....	39
Резюме	41
Упражнения	41

Глава 2. Численные алгоритмы	45
Рандомизация данных.....	45
Генерирование случайных величин.....	45
Рандомизация массивов.....	50
Генерирование неравномерных распределений.....	52
Нахождение наибольшего общего делителя.....	52
Возведение в степень.....	54
Работа с простыми числами.....	56
Нахождение простых множителей.....	56
Нахождение простых элементов.....	58
Проверка на простоту.....	59
Численное интегрирование.....	60
Формула прямоугольников.....	61
Формула трапеций.....	62
Адаптивная квадратура.....	63
Интеграция Монте-Карло.....	66
Нахождение нулей.....	67
Резюме.....	69
Упражнения.....	70
Глава 3. Связные списки	72
Основные положения.....	72
Однонаправленные связные списки.....	73
Передвижение по спискам.....	73
Нахождение ячеек.....	74
Использование ограничителей.....	75
Добавление ячеек в начало списка.....	76
Добавление ячеек в конец списка.....	77
Вставка ячеек.....	77
Удаление ячеек.....	78
Двунаправленные связные списки.....	79
Сортированные списки.....	80
Алгоритмы для работы со связными списками.....	81
Копирование.....	82

Сортировка вставкой.....	82
Сортировка методом выбора	84
Многопоточковые связные списки.....	85
Связные списки с циклами.....	86
Маркировка ячеек	87
Использование хеш-таблиц	88
Повторная трассировка списка.....	89
Реверсирование списка.....	90
Черепаша и кролик.....	92
Циклы в двунаправленных связных списках	94
Резюме	94
Упражнения	94
Глава 4. Массивы.....	96
Основные положения.....	96
Одномерные массивы.....	98
Нахождение элементов.....	98
Нахождение минимальной, максимальной и средней величин	98
Вставка элементов.....	100
Удаление элементов	101
Ненулевые нижние пределы.....	101
Двумерные массивы.....	101
Массивы высокой размерности	102
Треугольные массивы.....	105
Массивы с разрывом.....	108
Нахождение строки и столбца	110
Получение значения.....	111
Установка значения.....	111
Удаление значения	113
Матрицы	115
Резюме	117
Упражнения	117
Глава 5. Стеки и очереди	119
Стеки	119

Стеки связанных списков.....	120
Стеки массивов.....	121
Двойные стеки.....	123
Алгоритмы с использованием стеков.....	124
Очереди.....	129
Очереди связанных списков.....	130
Очереди массивов.....	130
Специализированные очереди.....	133
Резюме.....	134
Упражнения.....	134
Глава 6. Сортировка.....	136
Алгоритмы $O(N^2)$	136
Сортировка вставкой в массивах.....	136
Сортировка выбором в массивах.....	138
Пузырьковая сортировка.....	139
Алгоритмы $O(N \times \log N)$	142
Пирамидальная сортировка.....	142
Быстрая сортировка.....	148
Сортировка слиянием.....	154
Алгоритмы быстрее $O(N \times \log N)$	157
Сортировка подсчетом.....	157
Блочная сортировка.....	158
Резюме.....	160
Упражнения.....	161
Глава 7. Поиск.....	163
Линейный поиск.....	163
Бинарный поиск.....	164
Интерполяционный поиск.....	165
Резюме.....	166
Упражнения.....	167
Глава 8. Хеш-таблицы.....	168
Основы хеш-таблиц.....	168
Прямое связывание.....	170

Открытая адресация	171
Удаление элементов	172
Линейное пробирование	173
Квадратичное пробирование	174
Псевдослучайное пробирование	176
Двойное хеширование.....	176
Упорядоченное хеширование	176
Резюме	178
Упражнения	179
Глава 9. Рекурсия	181
Базовые алгоритмы	181
Факториал.....	181
Числа Фибоначчи.....	183
Ханойская башня	184
Графические алгоритмы	187
Кривые Коха.....	187
Кривая Гильберта	189
Кривая Серпинского	190
Салфетки	193
Алгоритмы с возвратом	194
Задача о восьми ферзях	195
Ход коня.....	198
Сочетания и размещения	200
Сочетания с циклами	201
Сочетания с повторениями	202
Сочетания без повторений	204
Размещения с повторениями.....	204
Размещения без повторений.....	205
Удаление рекурсии	206
Удаление хвостовой рекурсии.....	206
Хранение промежуточных значений	208
Удаление общей рекурсии	209
Резюме	212
Упражнения	212

Глава 10. Деревья	215
Терминология.....	215
Свойства бинарного дерева.....	219
Представление деревьев	220
Общие правила построения деревьев.....	220
Построение завершенных деревьев.....	223
Обход дерева.....	223
Обход в прямом порядке.....	224
Симметричный обход.....	226
Обход в обратном порядке	227
Обход в ширину	228
Время выполнения обхода	229
Упорядоченные деревья	229
Добавление вершин	230
Поиск вершин	231
Удаление вершин	232
Связные деревья.....	235
Построение связных деревьев.....	236
Использование связных деревьев.....	238
Специализированные алгоритмы	240
Игра «Животные»	240
Расчет математических выражений.....	241
Деревья квадрантов.....	243
Префиксные деревья	248
Резюме	252
Упражнения	253
Глава 11. Сбалансированные деревья.....	257
AVL-деревья.....	257
Добавление значений.....	258
Удаление значений.....	260
2-3-деревья	261
Добавление значений	262
Удаление значений.....	264
B-деревья	266
Добавление значений.....	267

Удаление значений	268
Разновидности сбалансированных деревьев.....	270
Иерархически организованные В-деревья	270
В+-деревья	270
Резюме	272
Упражнения	272
Глава 12. Деревья принятия решений	274
Поиск по деревьям игры.....	274
Минимакс	275
Начальные ходы и реакции.....	279
Эвристика дерева игры.....	279
Поиск по деревьям принятия решений.....	281
Задачи оптимизации.....	282
Метод полного перебора	282
Метод ветвей и границ.....	284
Эвристика дерева принятия решений.....	285
Другие задачи дерева принятия решений	290
Резюме	294
Упражнения	295
Глава 13. Основные сетевые алгоритмы.....	296
Сетевая терминология	296
Разные представления сети.....	299
Обход сети	302
Обход в глубину	302
Обход в ширину	304
Проверка связности	305
Остовные деревья.....	307
Минимальные остовные деревья	308
Поиск путей	309
Поиск произвольного пути	309
Поиск кратчайшего пути с помощью установки меток.....	310
Поиск кратчайшего пути с помощью коррекции меток.....	313
Поиск кратчайшего пути между всеми парами вершин	315
Резюме	320
Упражнения	320

Глава 14. Дополнительные сетевые алгоритмы.....	324
Топологическая сортировка.....	324
Поиск циклов.....	327
Раскрашивание карты.....	328
Закрашивание двумя цветами.....	328
Закрашивание тремя цветами.....	330
Закрашивание четырьмя цветами.....	331
Закрашивание пятью цветами.....	331
Другие алгоритмы закрашивания карт.....	335
Максимальный поток.....	336
Распределение рабочих мест.....	338
Минимальный разрез в потоке.....	340
Резюме.....	342
Упражнения.....	343
Глава 15. Строковые алгоритмы.....	345
Парные скобки.....	345
Вычисление арифметических выражений.....	347
Синтаксические деревья.....	347
Сопоставление с шаблоном.....	348
Детерминированные конечные автоматы.....	349
Построение ДКА для регулярных выражений.....	351
Недетерминированные конечные автоматы.....	354
Поиск строк.....	355
Вычисление редакционного расстояния.....	359
Резюме.....	361
Упражнения.....	362
Глава 16. Криптография.....	365
Терминология.....	366
Перестановочные шифры.....	367
Перестановка строк/столбцов.....	367
Перестановка столбцов.....	369
Маршрутные шифры.....	371
Шифры подстановки.....	372

Шифр Цезаря.....	372
Шифр Виженера	373
Простая подстановка.....	375
Схема одноразовых блокнотов	375
Блочные шифры.....	376
Подстановочно-перестановочные сети	377
Шифры Фейстеля	378
Шифрование с открытым ключом и RSA	380
Функция Эйлера.....	381
Обратные величины	381
Пример использования RSA.....	382
Практические соображения.....	383
Другие области применения криптографии	383
Резюме	384
Упражнения	385
Глава 17. Теория вычислительной сложности.....	387
Обозначения.....	388
Классы сложности	388
Сведение	391
3SAT	393
Паросочетание в двудольном графе.....	393
NP-сложность.....	394
Задачи обнаружения, сообщения и оптимизации.....	394
Обнаружение \leq_p Сообщение	395
Обнаружение \leq_p Оптимизация	395
Сообщение \leq_p Обнаружение	396
Оптимизация \leq_p Сообщение	396
NP-полные задачи	397
Резюме	399
Упражнения	400
Глава 18. Распределенные алгоритмы	402
Виды параллелизма.....	402
Систолические массивы	403
Распределенные вычисления	405

Многопроцессорные вычисления	407
Состояние гонки	407
Взаимная блокировка.....	411
Квантовые вычисления	412
Распределенные алгоритмы.....	413
Отладка распределенных алгоритмов	413
Чрезвычайно параллельные алгоритмы	414
Сортировка слиянием	416
Проблема обедающих философов	416
Задача двух генералов.....	419
Задача византийских генералов	420
Согласование.....	423
Выбор лидера	426
Снимок	427
Синхронизация часов.....	428
Резюме	429
Упражнения	429
Глава 19. Головоломки, встречающиеся на собеседованиях	432
Как задавать вопросы с подвохом	433
Как отвечать на вопросы с подвохом	435
Резюме	439
Упражнения	440
Приложение А.....	442
Приложение Б.....	453
Глоссарий.....	522
Алфавитный указатель	536

ОБ АВТОРЕ

Род Стивенс начинал свою карьеру как математик, однако во время учебы в Массачусетском технологическом институте он всерьез увлекся алгоритмами, прошел все связанные с ними курсы и занялся написанием довольно сложных программ. Ему довелось применить свои знания об алгоритмах во многих сферах: коммутация телефонов, учет вызовов и времени разговора абонента, оперативное управление ремонтными работами, налогообложение, водоочистка, картография и даже подготовка профессиональных футболистов.

Кроме того, Род Стивенс — один из ведущих специалистов по Microsoft Visual Basic и преподаватель вводного курса программирования в технологическом институте ИТТ, а также автор более 20 книг, переведенных на несколько иностранных языков, и свыше 250 журнальных статей о C#, Visual Basic, Visual Basic for Applications, Delphi и Java.

У Стивенса есть несколько сайтов: VB Helper (vb-helper.com)¹, куда ежемесячно заходят несколько миллионов пользователей и где содержатся подсказки, секреты, а также примеры программ на Visual Basic, а также C# Helper (csharpHelper.com) с аналогичного рода материалами для программистов на C#. Связаться с ним можно по электронному адресу RodStephens@vb-helper.com или RodStephens@csharpHelper.com.

БЛАГОДАРНОСТИ

Автор выражает свое признание Бобу Элиоту, Тому Динсу, Гейлу Джонсону и Дэниэлу Скрибнеру за те усилия, которые они приложили, чтобы эта книга вышла в свет. Большое спасибо хочется сказать техническим редакторам Джорджу Кокуру, Дэйву Колмену и Джеку Жинхиу Хао за точность изложенной здесь информации. В любых ошибках, которые могли остаться в книге, виноват я сам, а не эти люди.

¹Все указанные в книге сайты англоязычные. Издательство не несет ответственности за их содержимое и напоминает, что со времени написания книги сайты могли измениться или вовсе исчезнуть. — *Примеч. ред.*

ВВЕДЕНИЕ

Алгоритмы — это наборы команд, способствующие эффективному программированию. Они объясняют, как сортировать записи, искать элементы, рассчитывать числовые значения (например, простые множители), находить кратчайший путь между двумя точками на карте, определять максимально возможный поток информации по сети и т. д. Хороший алгоритм способен решить задачу в считанные секунды, плохой потребует на это часы или же не решит ее вовсе.

Изучение алгоритмов поможет вам овладеть различными полезными методами и научиться выбирать те из них, что наилучшим образом подходят к вашему сценарию. Ведь для решения какой-либо задачи можно использовать разную методику, но только определенная окажется идеальной. Кроме того, действия, показавшие отличные результаты с одним набором данных, с другим могут просто не сработать.

И, что еще более важно, вы познакомитесь с некоторыми общими способами решения тех или иных задач. И даже если приведенный алгоритм не сможет быть полезен в конкретном случае, это позволит по-другому взглянуть на стоящую перед вами проблему, а также даст возможность проанализировать собственные алгоритмы и найти неожиданный способ их применения.

Изученные методы будут нелишними и в вашей профессиональной деятельности. Многие крупные компании, занимающиеся информационными технологиями, такие как Microsoft, Google, Yahoo!, IBM и др., хотят, чтобы их программисты понимали алгоритмы и связанные с ними способы решения задач. На собеседовании в подобных корпорациях вам вполне могут предложить похожее тестовое задание. Даже если у вас не получится его выполнить, огромное значение будет иметь то, как вы попытаетесь получить результат и как станете решать незнакомую проблему. Не исключено, что в процессе рассуждений вам откроется, какие алгоритмические подходы следует использовать по отношению к задаче. «Я не знаю... Может быть, поищу что-то похожее в Интернете» — плохой ответ. А вот пример хорошего ответа: «Возможно, здесь мог бы помочь рекурсивный метод разобобщения».

Данная книга — своеобразная инструкция, изложенная простым языком. В ней рассматривается множество важнейших классических алгоритмов; рассказывается, в каких случаях и какие из них стоит использовать; объясняется, на каких принципах строится их работа; сообщается о методах, которые можно применять для поиска самостоятельных решений.

Вот часть той полезной информации, которая приводится в этом издании:

- численные алгоритмы (рандомизация, разложение на множители, работа с простыми числами, численное интегрирование);

- методы работы с общими структурами данных (массивами, связными списками, деревьями, сетями);
- использование более совершенных структур данных (неупорядоченных массивов, сбалансированных деревьев, *B*-деревьев);
- сортировка и поиск;
- сетевые алгоритмы (кратчайший путь, связующее дерево, топологическая сортировка, расчеты потока).

Ниже перечислены некоторые из общих методов решения задач, объяснение которых вы найдете на страницах книги:

- поиск с помощью прямого подбора;
- метод «разделяй и властвуй»;
- перебор с возвратами;
- рекурсия;
- метод ветвей и границ;
- жадные алгоритмы и поиск экстремума;
- алгоритмы сочетания ресурсов;
- сжатие пределов;
- эвристика.

Для более легкого усвоения материала в конце каждой главы приводятся упражнения, позволяющие изменить алгоритм или использовать его в новых ситуациях. Они также помогут закрепить продемонстрированные методы.

Наконец, в книге содержатся некоторые советы, которые окажутся полезными на собеседовании. Методы алгоритмизации помогут решить многие тестовые задания. И даже если вы не справитесь с какой-то конкретной головоломкой, то сможете доказать, что знакомы со способами решения других задач.

Выбор алгоритма

Каждый алгоритм включен в книгу по нескольким причинам.

- Он полезен, поэтому опытный программист должен понимать принцип его работы и уметь использовать в программах.
- Он демонстрирует важные технологии алгоритмического программирования, которые можно применить к другим задачам.
- Он предлагается для изучения студентам ИТ-специальностей и может встретиться на техническом собеседовании.

Ознакомившись с предлагаемым материалом и поработав над упражнениями, вы будете располагать достаточной базой знаний по алгоритмам и методам, которые используются для решения многих задач программирования.

Для кого предназначена книга

Это пособие в основном рассчитано на три категории читателей: профессиональных программистов, тех, кто только готовится к собеседованию при устройстве на работу, и студентов.

Первой категории будут полезны алгоритмы и методы решения, связанные непосредственно с рабочими задачами. Даже если вы столкнетесь с проблемой, которая здесь не описана, полученные знания позволят взглянуть на нее по-новому.

Вторая категория читателей сможет отточить навыки алгоритмизации. Есть вероятность, что на собеседовании приведенные примеры вам не попадутся, но рассматриваемые методики вполне пригодятся для решения похожих вопросов.

Третьей категории следует помнить, что принципы, описанные в книге, являются простыми и эффективными. Не с каждым из них вам придется столкнуться в реальной жизни, но без таких, как, например, рекурсия, «разделяй и властвуй», метод ветвей и границ, программисту просто не обойтись.



ЗАМЕЧАНИЕ

Лично я считаю, что алгоритмы – сплошное удовольствие! Они заменяют мне кроссворды и sudoku. Я обожаю то чувство, когда сложный алгоритм составлен, в него введены данные и видно, как появляется кривая, построенная по множеству найденных точек, или какая-нибудь трехмерная фигура.

Как извлечь наибольшую пользу из книги

Конечно, вы можете просто читать это пособие, знакомясь с предлагаемыми алгоритмами и методами, но, чтобы действительно хорошо овладеть материалом, вам стоит применить все на практике, то есть попытаться использовать полученные знания в каком-либо языке программирования. Не ленитесь экспериментировать — изменять алгоритмы, находить новые варианты для решения старых задач. Не оставляйте без внимания приведенные упражнения и вопросы для собеседований — они могут натолкнуть вас на необычное использование уже знакомого метода.

Кроме того, я настоятельно рекомендую проработать как можно большее число алгоритмов и желательно на разных языках программирования. Это позволит увидеть, как разные языки влияют на решение одной и той же задачи. Тщательным образом изучайте упражнения и анализируйте принципы их решения, в идеале вы должны уделить внимание каждому из них. Часто важность включенного в книгу алгоритма становится понятной только после того, как с его помощью вам удастся решить конкретную задачу.

Еще один полезный совет. Поищите в Интернете задачи, предлагаемые на собеседованиях, и подумайте, как бы вы с ними справились. Возможно, вам не предложат что-то подобное, но иметь хотя бы общее представление о решении не помешает. Время, затраченное на практику, не пропадет зря — так вы укрепите свои знания.

Изучить алгоритмы можно только в действии, поэтому не бойтесь отложить книгу, открыть компилятор и написать код!