Ким, Кей и Ханне
с любовью и восхищением.

И Ирин
с благодарностью за нарушение ее обещания.

Оглавление

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 прямо на нашем сайте **www.dmkpress.com**, зайдя на страницу книги, и оставить комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу **dmkpress@gmail.com**, при этом напишите название книги в теме письма.

Если есть тема, в которой вы квалифицированы, и вы 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 удостовериться в качестве наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в тексте или в коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от расстройств и поможете нам улучшить последующие версии 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 незаконно выполненной копией любой нашей книги, пожалуйста, сообщите нам адрес копии или веб-сайта, чтобы мы могли применить санкции.

Пожалуйста, свяжитесь с нами по адресу электронной почты **dmkpress@gmail.com** со ссылкой на подозрительные материалы.

Мы высоко ценим любую помощь по защите наших авторов, помогающую нампредоставлять вам качественные материалы.

Предисловие

Начнем с предисловия к книге алгоритмов практической арифметики.

— Иоанн Севильский (Ioannis Hispalensis),
Книга алгоритмов практической арифметики (около 1135 г.)

Должен ли я объяснять тебе, друг мой, как ты сможешь это понять?
Напиши об этом книгу.

— Генри Хоум, лорд Камес (Henry Home, Lord Kames) (1696–1782),
в письме сэру Гилберту Элиотту (Gilbert Eliott)

Человек всегда ошибается. Он строил множество планов и считал других людей своими помощниками, спорил с некоторыми или со всеми, много ошибался и кое-что сделал;
все это позволило продвинуться немного вперед, но человек всегда ошибается.
Оказалось, что получилось нечто новое, совсем не похожее на то, что он себе представлял.

— Ральф Уолдо Эмерсон (Ralph Waldo Emerson), Experience, эссе, второй сборник (1844 г.)

То, что я кратко описал выше, является содержанием книги, в которой реализация основной темы с включением подробностей, вероятно, стала бы невозможной; то, что я написал, является вторым или третьим черновым вариантом предварительной версии этой книги.

— Майкл Спивак (Michael Spivak), предисловие к первому изданию
книги «Дифференциальная геометрия», том 1 (1970 г.)

Об этой книге

Основой этой книги стали конспекты лекций, написанных мною для разнообразных курсов по алгоритмам в университете Иллинойса в Урбана-Шампейне (University of Illinois at Urbana-Champaign), где я преподаю примерно раз в год с января 1999 г. Из-за изменений в теоретической программе бакалавриата я существенно переработал основную редакцию этих конспектов лекций в 2016 г. Эта книга состоит из некоторого подмножества переработанных конспектов лекций в основном по материалам основного базового курса и главным образом отображает алгоритмическое содержимое новой университетской теоретической программы, обязательной для младших курсов.

Обязательный минимум

Для тех курсов по алгоритмам, которые я преподаю в университете Иллинойса, существуют два весьма важных предварительных требования к

обязательному образовательному минимуму: изучение курса дискретной математики и курса основных структур данных. Таким образом, эта книга, вероятнее всего, не подойдет для большинства студентов как начальный курс по структурам данных и алгоритмам. В частности, я предполагаю, что читатель как минимум хорошо знаком со специальными темами, такими как:

- **дискретная математика:** элементарная алгебра, логарифмические тождества, простейшая теория множеств, алгебра логики, логика первого порядка, множества, функции, эквивалентности (тождества), частичная упорядоченность, модульная арифметика (операции с остатками чисел по модулю), рекурсивные определения, деревья (как абстрактные объекты, а не структуры данных), графы (из вершин и ребер, а не графики функций);
- **методы доказательств:** прямой, косвенный, от противного (контрадикция), исчерпывающий (полный) анализ вариантов и индукция (особенно «строгая» и «структурная» индукция). В главе 0 используется индукция, и во всех случаях, когда в главе n–1 используется индукция, она используется и в главе n;
- **итеративные концепции программирования:** переменные, условные выражения, циклы, записи, косвенная адресация (адреса, указатели, ссылки), подпрограммы, рекурсия. Я не требую свободного владения каким-либо конкретным языком программирования, но предполагаю наличие у читателя практического опыта работы хотя бы с одним языком, который поддерживает косвенную адресацию и рекурсию;
- **основные (базовые) абстрактные типы данных:** скалярные значения, последовательности, векторы, множества, стеки, очереди, отображения/словари, упорядоченные отображения/словари, очереди с приоритетами;
- **основные (базовые) структуры данных:** массивы, связные списки (односвязные и двусвязные, линейные и кольцевые (циклические)), деревья двоичного поиска, как минимум одна форма сбалансированного дерева двоичного поиска (например, AVL-деревья, красно-черные деревья, декартовы деревья («дуча»), списки с пропусками или косые деревья), хеш-таблицы, двоичные кучи, а также, что наиболее важно, понимание различий между этим списком и предыдущим;
- **основные вычислительные задачи:** элементарная арифметика, сортировка, поиск, перечисление, обход вершин дерева (прямой (упорядоченный), центрированный (с порядковой выборкой), в обратном порядке (с отложенной выборкой), поиск в ширину и т. д.);
- **основные алгоритмы:** элементарные (арифметические) алгорифмы, последовательный поиск, двоичный (бинарный) поиск, сортировка (выбором, вставкой, слиянием, пирамидальная, быстрая, по-

разрядная (цифровая) и т. д.), поиск в ширину и в глубину в деревьях (как минимум в двоичных), а также, что наиболее важно, понимание различий между этим списком и предыдущим;

- **элементарный анализ алгоритмов:** асимптотическая нотация (o, O, Θ, Ω, ω), преобразование циклов в суммы и рекурсивных вызовов в рекуррентные соотношения, определение (вычисление) оценки простых сумм и рекуррентных (рекурсивных) соотношений;
- **математическая зрелость:** способность к абстракции, к формальным (особенно рекурсивным) определениям и к выполнению доказательств (особенно индуктивных); запись и прослеживание математических доказательств; распознавание и устранение синтаксических, смысловых (семантических) и/или логических элементов, не имеющих никакого смысла.

В этой книге кратко рассматриваются некоторые из перечисленных выше предварительно требуемых тем, когда этого требует контекст, но это в большей степени напоминание, нежели подробное введение. Для более глубокого изучения настоятельно рекомендуются следующие свободно распространяемые материалы:

- Margaret M. Fleck. «Building Blocks for Theoretical Computer Science». Версия 1.3 (January 2013) или более поздняя доступна здесь: **http://mfleck.cs.illinois.edu/building-blocks/**;
- Eric Lehman, F. Thomson Leighton, and Albert R. Meyer. «Mathematics for Computer Science». Версия июня 2018 г. доступна здесь: **https://courses.csail.mit.edu/6.042/spring18/**. (Настоятельно рекомендуется найти самую последнюю версию.);
- Pat Morin. «Open Data Structures». Редакция 0.1Gβ (январь 2016 г.) или более поздняя доступна здесь: **http://opendatastructures.org/**;
- Don Sheehy. «A Course in Data Structures and Object-Oriented Design». Версия февраля 2019 г. или более поздняя доступна здесь: **https://donsheehy.github.io/datastructures/**.

Дополнительные ссылки

Рекомендуется не ограничиваться одной из приведенных выше ссылок или каким-либо другим единственным источником. Авторы и читатели привносят собственные точки зрения и перспективные взгляды в любой учебно-просветительский материал. «Универсального преподавателя» для всех обучающихся подобрать невозможно, даже для очень умных студентов. Поиск «своего» автора, который наиболее эффективно вложит собственные знания и восприятие в ваш разум, требует определенных усилий, но эти усилия многократно окупаются в дальнейшем.

Приведенные ниже ссылки представляют собой особо ценные источники знаний и интуитивных представлений с примерами, упражнениями и развивающими заданиями, но этот список нельзя считать абсолютно полным.

- *Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman*. The Design and Analysis of Computer Algorithms. Addison-Wesley, 1974. (Я использовал эту книгу как студент в университете Райса – Rice University, потом как студент магистратуры в Калифорнийском университете в Ирвайне – UC Irvine.)
- *Boaz Barak*. Introduction to Theoretical Computer Science. Предварительная редакция книги, последняя редакция в июне 2019 г. (Это не книга по теории ИТ, доставшаяся вам от дедушки, это гораздо более качественный материал, а бесплатный доступ является дополнительным существенным преимуществом.)
- *Thomas Cormen, Charles Leiserson, Ron Rivest, and Cliff Stein*. Introduction to Algorithms, third edition. MIT Press/McGraw-Hill, 2009. (Я использовал первое издание этой книги, когда был ассистентом кафедры в университете Беркли – Berkeley.)
- *Sanjoy Dasgupta, Christos H. Papadimitriou, and Umesh V. Vazirani*. Algorithms. McGraw-Hill, 2006. (Вероятно, эта книга по содержанию ближе всего к моей, но она значительно менее подробна.)
- *Jeff Edmonds*. How to Think about Algorithms. Cambridge University Press, 2008.
- *Michael R. Garey and David S. Johnson*. Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman, 1979.
- *Michael T. Goodrich and Roberto Tamassia*. Algorithm Design: Foundations, Analysis, and Internet Examples. John Wiley & Sons, 2002.
- *Jon Kleinberg and Éva Tardos*. Algorithm Design. Addison-Wesley, 2005. Найдите эту книгу в библиотеке, если сможете.
- *Donald Knuth*. The Art of Computer Programming, volumes 1–4A. Addison-Wesley, 1997 and 2011. (Мои родители подарили мне первые три тома на Рождество, когда мне было 14 лет. Увы, до сих пор так и не удалось прочитать эти тома полностью.)
- *Udi Manber*. Introduction to Algorithms: A Creative Approach. Addison-Wesley, 1989. (Я использовал эту книгу, когда был ассистентом кафедры в университете Беркли – Berkeley.)
- *Ian Parberry*. Problems on Algorithms. Prentice-Hall, 1995 (не выпускалась в печатном виде). Книгу можно загрузить (**https://larc.unt.edu/ian/books/free/license.html**) после внесения небольшой суммы пожертвования. Соблюдайте соглашение о нераспространении.
- *Robert Sedgewick and Kevin Wayne*. Algorithms. Addison-Wesley, 2011.
- *Robert Endre Tarjan*. Data Structures and Network Algorithms. SIAM, 1983.
- Записи лекций по моему курсу алгоритмов в университете Беркли, особенно прочитанные Диком Карпом (Dick Karp) и Раймондом Зайделем (Raimund Seidel).

- Записи лекций, слайды, подготовительные материалы, экзаменационные материалы, видеолекции, отчеты об исследованиях, посты в блогах, вопросы и ответы на сайте StackExchange, подкасты и полнофункциональные массовые открытые онлайн-курсы, свободно распространяемые в веб-среде многочисленными учебными заведениями по всему миру.

Упражнения в этой книге

В конце каждой главы вниманию читателя предлагается несколько упражнений. большинство из которых я использовал как минимум один раз для домашних заданий, на семинарах и в лабораторных работах или на экзаменах. Упражнения не упорядочены по возрастанию сложности, но (в общем случае) сгруппированы по общему применяемому методу или по определенной теме. Некоторые задачи помечены символами, описанными ниже:

- символы красного сердечка (карточная масть «черви») ♥ обозначают особенно сложные задачи. Многие из этих задач предлагались студентам на экзаменах на получение степени доктора философии (Ph.D.) в университете Иллинойса. Некоторые (немногие) действительно сложные задачи обозначены увеличенным символом ♥;
- синие ромбики (карточная масть «бубны») ♦ обозначают задачи, для которых требуется изучение материала из следующих глав, но относящиеся к теме текущей главы. Но задачи, для которых требуется знание материала предыдущих глав, никак не помечены. В этой книге, как и в жизни, знания накапливаются постепенно;
- зеленые символы карточной масти «трефы» ♣ обозначают задачи, для которых требуется знание материала, не относящегося к тематике данной книги, например конечные автоматы, линейная алгебра, теория вероятностей или плоские графы. Но такие задачи встречаются редко;
- черные символы карточной масти «пики» ♠ обозначают задачи, требующие значительного объема рутинной работы и/или написания программного кода. Такие задачи также редко встречаются;
- оранжевые звездочки ★ означают, что вы едите «Лаки чармс» (Lucky Charms – детские сухие завтраки в виде глазированных фигурок-талисманов), произведенные до 1998 г., т. е. имеете дело с очень старыми продуктами. Брр.

Эти упражнения предоставляют возможность применить полученные знания на практике, а не являются «задачами ради самих задач». Цель каждого упражнения не решение конкретно сформулированной задачи, а развитие определенного набора навыков и умений или получение практического решения определенного типа задач. Отчасти по этой причине я не привожу решения упражнений, здесь решения не самое важное. Заметьте,

это не «учебное руководство», и если вы сами не можете решить задачу, то, вероятнее всего, вы не должны предлагать ее своим подопечным. Но при этом, возможно, вы сможете найти решения некоторых домашних заданий, которые опубликованы в этом семестре на веб-странице курса, который я читаю. Кроме того, что может помешать вам написать собственное учебное руководство.

Стащите эту книгу

 Полная электронная версия книги, доступная свободно (бесплатно), размещена на сайтах, таких как:

- сайт книги: **http://jeffe.cs.illinois.edu/teaching/algorithms/**;
- мнемоническое сокращенное имя: **http://algorithms.wtf**;
- сайт отчетов об ошибках: **https://github.com/jeffgerickson/algorithms**;
- глобальный архив Internet Archive: **https://archive.org/details/Algorithms-Jeff-Erickson**.

На сайте книги также размещено несколько сотен страниц дополнительных конспектов лекций, содержащих материалы, связанные с темой этой книги, а также расширенные и дополненные материалы. Кроме того, на сайте находится почти полный архив последних домашних заданий, задач для экзаменов, семинаров и лабораторных работ и прочие учебные ресурсы. В процессе чтения курса по алгоритмам я пересматриваю, обновляю и иногда исключаю некоторые учебные материалы, поэтому на веб-странице текущего курса можно обнаружить более свежие версии.

Вне зависимости от того, являетесь вы студентом или преподавателем, использование любой части этой книги или какого-либо другого конспекта лекций не только допускается, но и поощряется – без моего специального разрешения, – именно поэтому я разместил все эти материалы на открытых веб-сайтах. Но при цитировании этой книги убедительно прошу указывать ее название или хотя бы сокращенную ссылку на **http://algorithms.wtf**. Это особенно важно, если вы являетесь студентом и используете мои материалы учебного курса как помощь при выполнении домашнего задания. (Уточните подробности у своего преподавателя.)

Но если вы преподаватель, то я настоятельно рекомендую расширить курс с помощью дополнительных материалов, *написанных вами самостоятельно*. Написание собственных учебных материалов улучшит ваши преподавательские навыки, а также представление учебного материала в аудитории, что в свою очередь способствует более эффективному усвоению материала студентами. Кроме того, стимулом к самостоятельной работе послужит разочарование после ознакомления с теми частями книги, кото-

рые вам не понравились. Все книги ~~плохи~~ несовершенны, и эта не является исключением.

Последнее замечание: **рекомендую все, что вы пишете, открыто размещать в свободном доступе в глобальной веб-среде** – не скрывайте это «за семью замками» системы управления обучением или какой-либо другой системы ограничения доступа к информации, – чтобы студенты и преподаватели везде и всюду могли воспользоваться преимуществами ваших достижений и идей. В особенности если вы разрабатываете полезные ресурсы, которые непосредственно дополняют эту книгу, например слайды, видеоматериалы или руководства по решениям задач, то убедительно прошу сообщить мне об этом, чтобы я мог добавить ссылки на ваши ресурсы на веб-сайте книги.

Благодарности

Создание этой книги в немалой степени зависело от вклада многочисленных студентов, изучающих курсы по алгоритмам, преподавателей таких курсов и исследователей в этой области. В особенности я безмерно благодарен более чем трем тысячам студентов Иллинойского университета, которые пользовались моими конспектами лекций как основным источником, высказывали полезные (иногда нелицеприятные) критические замечания и терпеливо штудировали ранние версии этих конспектов, которые были действительно ужасными. Также благодарю многих коллег и студентов по всему миру, которые использовали мои конспекты в собственных учебных курсах и присылали полезные отзывы и отчеты об ошибках.

Особая благодарность за постоянное двустороннее взаимодействие и вклад (особенно в создание упражнений) моим великолепным ассистентам. Это:

Адитья Рамани (Aditya Ramani), Акаш Гаутам (Akash Gautam), Алекс Штайгер (Alex Steiger), Алина Эне (Alina Ene), Амир Найери (Amir Nayyeri), Аша Ситхарам (Asha Seetharam), Ашиш Вулимири (Ashish Vulimiri), Бен Мозли (Ben Moseley), Брэд Стёрт (Brad Sturt), Брайан Энсинк (Brian Ensink), Чао Сюй (Chao Xu), Чарли Карлсон (Charlie Carlson), Крис Нейгенген (Chris Neihengen), Коннор Кларк (Connor Clark), Дэн Буллок (Dan Bullok), Дэн Крэнстон (Dan Cranston), Даниэль Хашаби (Daniel Khashabi), Дэвид Моррисон (David Morrison), Экта Манактала (Ekta Manaktala), Эрин Вольф Чемберс (Erin Wolf Chambers), Гейл Штайц (Gail Steitz), Чжо Као (Gio Kao), Грант Чайковски (Grant Czajkowski), Сянь-Чжи Чан (Hsien-Chih Chang), Игор Гаммер (Igor Gammer), Джейкоб Лорел (Jacob Laurel), Джон Ли (John Lee), Джонатон Фишер (Johnathon Fischer), Цзюньцин Дэн (Junqing Deng), Кент Куанруд (Kent Quanrud), Кевин Миланс (Kevin Milans), Кевин Смолл (Kevin Small), Константинос Койлиарис (Konstantinos Koiliaris), Кайл Фокс (Kyle Fox), Кайл Цзяо (Kyle Jao), Лан Чень (Lan Chen),

Марк Идельман (Mark Idleman), Майкл Бонд (Michael Bond), Митч Харрис (Mitch Harris), Навин Ариважаген (Naveen Arivazhagen), Ник Бахмеер (Nick Bachmair), Ник Херлберт (Nick Hurlburt), Нирман Кумар (Nirman Kumar), Нитиш Корула (Nitish Korula), Патрик Лин (Patrick Lin), Филип Ши (Phillip Shih), Рачит Агравал (Rachit Agarwal), Реза Замани-Насаб (Reza Zamani-Nasab), Риши Талрейя (Rishi Talreja), Роб МакКенн (Rob McCann), Саханд Мозаффари (Sahand Mozaffari), Шалан Накви (Shalan Naqvi), Шрипад Тите (Shripad Thite), Спенсер Гордон (Spencer Gordon), Срихита Ватсавайя (Srihita Vatsavaya), Субро Рой (Subhro Roy), Тана Ваттанавароон (Tana Wattanawaroon), Уманг Матхур (Umang Mathur), Випул Гоял (Vipul Goyal), Ясу Фуракава (Yasu Furakawa) и Йипу Ван (Yipu Wang).

Огромную помощь оказали многочисленные обсуждения с коллегами по факультету Иллинойского университета, среди них Александра Колла (Alexandra Kolla), Синда Хеерен (Cinda Heeren), Эдгар Рамос (Edgar Ramos), Херберт Эдельсбруннер (Herbert Edelsbrunner), Джейсон Зих (Jason Zych), Ким Уиттлси (Kim Whittlesey), Ленни Питт (Lenny Pitt), Мадху Парасарати (Madhu Parasarathy), Махеш Висванатан (Mahesh Viswanathan), Маргарет Флек (Margaret Fleck), Шан-Хуа Тен (Shang-Hua Teng), Стив ЛаВалль (Steve LaValle) и особенно Чандра Чекури (Chandra Chekuri), Эд Рейнголд (Ed Reingold) и Сариел Хар-Пелед (Sariel Har-Peled).

Разумеется, эта книга обязана своим появлением работе тех людей, которые научили меня использовать алгоритмы во времена студенчества. Это Боб Биксби (Bob Bixby) и Майкл Перлман (Michael Pearlman) в университете Райса; Дэвид Эппштейн (David Eppstein), Дэн Хершберг (Dan Hirschberg) и Джордж Лекер (George Lueker) в университете Ирвайна; Эбхирам Рэнэйд (Abhiram Ranade), Дик Карп (Dick Karp), Мануэл Блум (Manuel Blum), Майк Лаби (Mike Luby) и Раймунд Зайдель (Raimund Seidel) в университете Беркли.

Я позаимствовал общую структуру курса в начальном варианте и способ подробной записи своих конспектов лекций в первую очередь у Херберта Эдельсбрюннера (Herbert Edelsbrunner). Идея превращения набора моих конспектов в книгу принадлежит Стиву ЛаВаллю (Steve LaValle), а дизайном некоторых компонентов книги занимался Роберт Грист (Robert Ghrist).

Предостережение для преподавателя

Разумеется, ни один из перечисленных выше людей не должен быть обвинен в каких-либо недостатках написанной мною книги. Несмотря на многочисленные этапы переработки и редактирования, в этой книге имеются некоторые ошибки, «баги», оплошности, упущения, запутанные моменты, нелепости, опечатки, математические, грамматические и логические ошибки, «глюки», плохо спроектированные решения, исторические неточности, анахронизмы, несоответствия, излишние обобщения, неясности,

лишняя болтовня, искажения, чрезмерные упрощения, избыточность, недержание речи, бессмыслица, всякий хлам, халтура, мусор и явная ложь – но все это полностью на совести Стива Скиены (Steve Skiena).

Я сопровождаю систему отслеживания ошибок (баг-трекер) на сайте GitHub: **https://github.com/jeffgerickson/algorithms**, где все читатели, в том числе и вы, можете представлять на рассмотрение отчеты об ошибках, запросы новых свойств и обеспечивать общую обратную связь по теме этой книги. Убедительно прошу сообщать мне обо всех найденных ошибках любого типа – математических, грамматических, исторических, типографических, культурологических и т. п. – как в основном тексте, так и в упражнениях или в других материалах учебного курса. (Вряд ли Стиву есть дело до этого.) Разумеется, приветствуются и любые другие варианты обратной связи.

Желаю успеха!

— Джефф

Обычно автор великодушно принимает на себя вину за все обнаруженные недостатки. Я не принимаю. Любые ошибки, недостатки или проблемы, найденные в этой книге, – это вина кого-то другого, но я был бы рад узнать, кто действительно виноват.

— Стивен С. Скиена (Steven S. Skiena), The Algorithm Design Manual (1997 г.)

Вне всякого сомнения, за этим утверждением последует аннотированный список всех книг с объяснением того, почему каждая из них – явная чушь.

— Адам Контини (Adam Contini), MetaFilter, 4 января 2010 г.

Глава 0

Введение

Отсюда начинается алгоризм. Эта техника называется алгоризмом, используя который индийцы дважды наслаждаются пятью разными фигурами: 0. 9. 8. 7. 6. 5. 4. 3. 2. 1.
(Hinc incipit algorismus. Haec algorismus ars praesens dicitur in qua talibus indorum fruimur bis quinque figuris 0. 9. 8. 7. 6. 5. 4. 3. 2. 1.)

— Брат Александр де Вилья Деи (Friar Alexander de Villa Dei), Песнь об алгоризме (Carmen de Algorismo) (около 1220 г.)

Требуя от художника сознательного отношения к работе, Вы правы, но Вы смешиваете два понятия: решение вопроса и правильную постановку вопроса.

— А. П. Чехов, в письме А. С. Суворину (27 октября 1888 г.)

Чем больше мы приучаем себя к механическим действиям в малозначимых случаях, тем больше сил мы освобождаем для использования в более важных случаях.

— Анна Бракетт (Anna C. Brackett), The Technique of Rest (1892 г.)

И вот я в 2:30 ночи пишу о методе, несмотря на твердое убеждение, что мгновение, когда мужчина начинает говорить о методе, является доказательством того, что он только что лишился идей.

— Реймонд Чандлер (Raymond Chandler), в письме Эрлу Стенли Гарднеру (Erle Stanley Gardner) (5 мая 1939 г.)

Хорошему человеку не нужны правила.
Сегодня не тот день, чтобы выяснять, почему я имею так много.

— Доктор [Matt Smith], «Хороший человек идет на войну» (A Good Man Goes to War), сериал «Доктор Кто» (Doctor Who, 2011 г.)

0.1. Что такое алгоритм

Алгоритм – это явно заданная, точная, однозначная, механически исполняемая последовательность элементарных инструкций, обычно предназначенная для достижения конкретной цели. Например, ниже приведен алгоритм исполнения известной навязчивой песенки «99 бутылок пива на стене» (99 Bottles of Beer on the Wall) для произвольных значений, отличающихся от 99.

```
BottlesOfBeer(n):
  For i ← n down to 1
    Петь "i бутылок пива на стене, i бутылок пива,"
    Петь "Возьми одну, пусти по кругу, i – 1 бутылок пива на стене."

  Петь "Нет бутылок пива на стене, нет бутылок пива,"
  Петь "Сбегай в магазин, купи еще, n бутылок пива на стене."
```

Слово «алгоритм» происходит не от греческого корня *arithmos*, означающего «число», и *algos*, означающего «боль», как могли бы предположить алгоритмофобы-классики. Скорее, это искажение имени персидского ученого IX века Мухаммада ибн Мусы-аль-Хва̄ризми. Аль-Хва̄ризми, возможно, наиболее известен как автор трактата «Аль-Китāб аль-мухтасар ф āих āисāб аль-габр ва'л-мукāбала», от которого происходит современное слово «алгебра». В другом трактате аль-Хва̄ризми описал современную десятичную систему записи и манипулирования числами (в частности, использование маленького круга или сифра для обозначения недостающего количества), которая была разработана в Индии несколькими столетиями ранее. Методы, описанные в этом последнем трактате, с использованием либо письменных цифр, либо счетных камней, стали известны в английском языке как алгоризм или аугрим, а его цифры стали известны в английском языке как *ciphers* (шифры).

Несмотря на то что и позиционная нотация, и работы аль-Хва̄ризми были уже известны некоторым европейским ученым, «индийско-арабская» система была популяризирована в Европе средневековым итальянским математиком и торговцем Леонардо Пизанским, более известным как Фибоначчи. Во многом благодаря его книге «Liber Abaci»[1], написанной в 1202 г., запись чисел цифровыми символами стала заменять счетные таблицы (известные под названием «абак(а)» (*abacus*)) и «арифметику на пальцах»[2] как основные предпочитаемые методики вычислений[3] в Европе в XIII в. – но не потому, что запись десятичными цифрами была проще для обучения и использования, а потому, что такой способ обеспечивал ведение бухгалтерской книги. Цифры стали широко распространенными в Западной Европе только после появления наборного типографского шрифта,

[1] Несмотря на соблазн перевода названия Liber Abaci буквально как «Книга абака», более правильный перевод труда Фибоначчи – «Книга вычислений». И до и после Фибоначчи итальянское слово *abaco* использовалось для описания всего, что связано с вычислениями – устройств, методов, школ, книг и т. д. – во многом так же, как сегодня выражение *computer science* используется в английском языке (или «информационные технологии» в русском языке) или как китайское выражение «исследование операций» переводится буквально как «изучение использования счетных палочек».

[2] Счет с помощью десяти цифр-пальцев!

[3] Слово *calculate* (вычислять) произошло от латинского calculus, означающего «небольшой камешек», т. е. камешки на счетной таблице (доске), которые Джеффри Чосер (Chaucer, Geoffrey) называл *augrym stones*. В 440 г. до н. э. Геродот в своей «Истории» писал, что «греки пишут и считают (λογίζεσθαι ψήφοις, дословно: «считать камешками») слева направо, а египтяне делают это наоборот. Но они говорят, что их способ записи направлен вправо, а греческий способ записи направлен влево». (Странно, что Геродот ничего не говорит о том, с какого конца начинают разбивать яйцо египтяне.)

а действительно вездесущими – только при массовом производстве дешевой бумаги в самом начале XIX в.

В итоге слово «алгоризм» (*algorism*) превратилось в современный термин «алгоритм» (algorithm) по ложной («вульгарной») этимологии от греческого слова *arithmos* (и, возможно, от ранее упомянутого *algos*)[4]. Таким образом, до недавнего времени термин «алгоритм» относился исключительно к механическим (автоматически выполняемые) методам арифметики с использованием позиционной системы записи арабских цифр. Люди, обученные быстрому и надежному выполнению этих процедур, назывались алгоритмистами (*algorists*) или вычислителями – *computators*, упрощенно *computers*.

0.2. Умножение

Несмотря на то что как предмет в академии алгоритмы существуют лишь несколько десятилетий, они были вместе с нами с самого зарождения цивилизации. Описания пошаговых арифметических вычислений встречаются среди самых ранних образцов письменности, задолго до работ, представленных Фибоначчи и аль-Хорезми и даже до появления позиционной системы записи чисел, которую они внедряли.

Умножение методом решетки

Самый известный метод умножения больших чисел, по крайней мере для американских учащихся, – умножение методом решетки (lattice algorithm). Этот алгоритм был популяризирован Фибоначчи и описан в Liber Abaci. Фибоначчи узнал об этом алгоритме из арабских источников, в том числе из работ аль-Хорезми, который в свою очередь изучил его по индийским источникам, включая трактат VII в. «Brāhmasphuṭasiddhānta» Брахмагупты (Brahmagupta), который, возможно, обучался по китайским источникам. Старейшие сохранившиеся описания этого алгоритма встречаются в трактате «Математическая классика» Сунь Цзы, написанном в Китае между III и V вв., а также в комментариях Евтокия Аскалонского к трактату Архимеда «Об измерении круга», написанных около 500 г. н. э., но существуют свидетельства того, что этот алгоритм был известен намного раньше. Евтокий считает, что этот метод появился в последнем трактате Аполлона Пергийского (около 300 г. до н. э.) под названием «Okytokion» (Ὠκυτόκιον)[5]. Шумеры

[4] Некоторые средневековые источники утверждают, что греческая приставка *algo-* означает «искусство» или «введение». В других источниках сообщается, что алгоритмы были изобретены неким греческим философом, или королем Индии, или, возможно, королем Испании с именем Algus, или Algor, или Argus. Кое-кто, возможно, Данте Алигьери даже отождествлял изобретателя алгоритмов с мифологическим греческим кораблестроителем и одноименным аргонавтом (Арго). Совершенно неясно, претендовали ли все эти смехотворные утверждения на историческую точность или являлись лишь плодами богатой фантазии.

[5] Дословно: «лекарство, способствующее быстрым и легким родам». В папирусе из Александрийской библиотеки воспроизводились некоторые извлечения из «Okytokion» приблизительно за 200 лет до Евтокия, но его описание алгоритма умножения методом решетки (если Евтокий действительно приводил его) также утерян.