ОГЛАВЛЕНИЕ

От издателей русского перевода

На мировом рынке компьютерной литературы существует множество книг, предназначенных для обучения основным алгоритмам и используемых при программировании. Их довольно много, и они в значительной степени конкурируют между собой. Однако среди них есть особая книга. Это трехтомник "Искусство программирования" Д. Э. Кнута, который стоит вне всякой конкуренции, входит в золотой фонд мировой литературы по информатике и является настольной книгой практически для всех, кто связан с программированием.

Мы как издатели видим ценность книги в том, что она предназначена не столько для обучения технике программирования, сколько для обучения, если это возможно, "искусству" программирования, предлагает массу рецептов усовершенствования программ и, что самое главное, учит самостоятельно находить эти рецепты.

Ни для кого не секрет, что наши программисты являются одними из наиболее высококвалифицированных специалистов в мире. Они достойно представляют за рубежом отечественную школу программирования и информатики, которая внесла значительный вклад в формирование фундаментальных основ компьютерных наук. Для сохранения такого уровня и продвижения вперед необходимо своевременное издание на русском языке книг, отражающих основные мировые достижения в этой области. Трехтомник "Искусство программирования" Д. Э. Кнута — одна из таких книг.

Мы гордимся тем, что библиотеки программистов, преподавателей, студентов, старшеклассников и многих других пополнятся этой классической книгой и что тем самым мы внесем свой вклад в формирование более глубокого понимания основ компьютерных наук. Мы глубоко убеждены, что книга "Искусство программирования" Д. Э. Кнута способна приблизить человека к совершенству. Надеемся, наше издание на русском языке этой замечательной книги еще раз подтвердит, что истинные ценности с годами не устаревают.

— *Виктор Штонда, Геннадий Петриковец, Алексей Орлович, издатели*

О книге “Искусство программирования”

У каждой книги своя судьба. Одни появляются незаметно и так же незаметно исчезают в потоке времени, покрываясь пылью на полках библиотек. Другие в определенный период пользуются спросом у узкого круга специалистов, пока им на смену не приходят новые справочники. Третьи, поднимаясь над временем, оказывают мощное влияние на технологическое развитие общества. Книг, относящихся к последней категории, не так уж и много. Их выход в свет — всегда праздник. Проходят годы, изменяются технологии, но новые поколения с постоянным интересом перечитывают их страницы. Именно к таким книгам относится предлагаемый читателю многотомный труд известного американского ученого Дональда Эрвина Кнута “Искусство программирования”.

Прошло почти 30 лет со времени первого издания в 1972 году в США этой книги. Она была переведена на большинство языков мира, в том числе и на русский. К настоящему времени на территории стран СНГ трехтомник Д. Э. Кнута стал библиографической редкостью. В 1998 году в США вышло третье издание “Искусства программирования”. В нем сохранена последовательность изложения материала прежних версий, но значительно расширен список ссылок, в который включены свежие и наиболее важные результаты, добавлены новые упражнения и комментарии, устранены неточности. Учитывая популярность во всем мире “Искусства программирования”, давно следовало ожидать появления нового переводного издания на русском языке, которое вы и держите в руках.

В чем же успех “Искусства программирования” Д. Э. Кнута?

Во-первых, эта книга — великолепное учебное пособие по составлению и анализу компьютерных алгоритмов. Ее разделы могут быть включены во многие университетские курсы по технологиям программирования, теории алгоритмов, дискретной математике. Книгу могут изучать и школьники старших классов, знакомые с основами программирования. В качестве основного языка записи алгоритмов автор выбрал язык машинных команд гипотетического универсального компьютера `MIX`. Это позволяет строить оптимальные программы с учетом особенностей вычислительных машин. Перенести `MIX`-программы на реальные ЭВМ или переписать их на языках высокого уровня не составляет особого труда. Логика работы программ почти всегда поясняется простыми блок-схемами.

Во-вторых, тщательно подобранный материал, вошедший в книгу, включает в себя основные фундаментальные классы алгоритмов, которые в том или ином виде наиболее часто встречаются в практике программирования.

В-третьих, немаловажным фактором успеха книги Д. Э. Кнута является энциклопедичность изложения. Профессор Кнут отличается уникальной способностью отслеживать проблему от исторических предпосылок ее зарождения до современного состояния. Многочисленные ссылки на работы старых мастеров (вплоть до времен античности), заключенные в современный контекст, создают у читателя особое чувство причастности к историческому развитию научных идей и методов.

В-четвертых, следует отметить мастерство изложения. Книга рассчитана на широкий круг читателей — от начинающих студентов до программистов-профессионалов. Каждому будет интересно изучать компьютерные алгоритмы на своем уровне. Материал

самодостаточен. Для понимания сути методов не требуется знания особых разделов математики или специальных технологий программирования. Прослеживается определенная “музыкальная” композиция сюжетного построения (дома у Д. Э. Кнута есть небольшой орган, на котором он играет).

Список составляющих успеха “Искусства программирования” можно легко продолжить.

Автор этих строк прослушал курс “Искусство программирования” в изложении профессора Кнута в 1976–1977 годах во время стажировки в Станфордском университете. Тогда формировалась алгоритмическая основа технологий программирования, у истоков которой стоял Д. Э. Кнут. Было много обсуждений, семинаров, творческих замыслов.

Значительные книги всегда связаны с судьбой автора. Дональд Эрвин Кнут начал работу над “Искусством программирования” в 1962 году. Продолжает ее и сейчас. У него много планов. Впереди новые тома “Искусства программирования”, которых с нетерпением ждут читатели.

— *Профессор Анатолий Анисимов*

От редактора перевода

Со времени первого издания книги “Искусство программирования” Д. Э. Кнута прошло около 25 лет. Тем не менее книга не только не устарела, но по-прежнему остается основным руководством по искусству программирования, книгой, по которой учатся понимать суть и особенности этого искусства.

За эти годы на английском языке вышло уже третье издание 1- и 2-го томов, а также второе издание 3-го тома. Автор внес в них значительные изменения и существенные дополнения. Достаточно сказать, что число упражнений практически удвоилось, а многие упражнения, включенные в предыдущие издания (особенно ответы к ним), модифицированы. Существенно дополнены и переделаны многие главы и разделы, исправлены неточности и опечатки, добавлены многочисленные новые ссылки на литературу, использованы теоретические результаты последних лет.

Значительно преобразилась глава 3, особенно разделы 3.5 и 3.6, а также разделы 4.5.2, 4.7, 5.1.4, 5.3, 5.4.9, 6.2.2, 6.4, 6.5 и др.

Естественно, возникла необходимость в новом издании книги.

Перевод выполнен по третьему изданию 1- и 2-го томов и второму изданию 3-го тома. Кроме того, учтены дополнения и исправления, любезно предоставленные автором.

При переводе мы старались сохранить стиль автора, обозначения и манеру изложения материала. В большинстве случаев использовались термины, принятые в научной литературе на русском языке. При необходимости приводились английские эквиваленты. По многим причинам, в частности из-за сложности некоторых разделов, читать книгу “Искусство программирования” далеко непросто. Одной из причин, которые затрудняют понимание книги, является манера изложения автора; привыкнув к ней, можно существенно облегчить чтение.

Из-за обилия материала (часто мало связанного между собой) невозможно построить книгу так, чтобы различные понятия и определения вводились сразу же при первом упоминании о них. Поэтому в главе 1 могут обсуждаться без ссылок понятия, строгие определения которых приводятся в 3-м томе. Именно поэтому так велика роль предметного указателя, без которого понимание книги было бы существенно затруднено. Надеемся, что читатель не будет удивлен, найдя ссылки на главы 7, 8 и последующие не вошедшие в предлагаемые три тома главы. Мы вместе с автором надеемся, что очень скоро они будут опубликованы и, безусловно, сразу же появятся в русском переводе в качестве продолжения этого издания.

Следует также обратить внимание на далеко не всегда стандартные обозначения, которыми пользуется автор. Так же, как и определения, эти обозначения могут появиться в 1-м томе, а вводиться во 2-м. Поэтому без указателя обозначений пользоваться книгой было бы чрезвычайно трудно. Хочу также обратить внимание на запись `[A]`, где `A` — некоторое высказывание. Эта запись встречается в формулах, а иногда и в тексте, и обозначает величину, равную индикатору `A`.

— Профессор Ю. В. Козаченко

ПРЕДИСЛОВИЕ

Дорогая Офелия!
Мне плохо от этих чисел:
Я не способен сосчитать мои стоны!
— Гамлет (акт II, сцена 2, строка 120)

Алгоритмы, описываемые в этой книге, имеют непосредственное отношение к числам. Я считаю, что их справедливо называют *получисленными*, так как они лежат на границе между численными и символьными методами. Каждый алгоритм должен не только находить числовое решение проблемы, но и хорошо сочетаться с внутренними операциями цифрового компьютера. В большинстве случаев человек не может оценить всю красоту подобных алгоритмов, если только он не владеет машинным языком компьютера. Эффективность соответствующей компьютерной программы — это жизненно важный фактор, который нельзя отделить от самого алгоритма. Проблема заключается в том, чтобы найти оптимальные способы работы компьютеров с числами, а это включает вопросы тактики и численного анализа. Поэтому предмет данной книги, без сомнения, относится к компьютерной науке так же, как к разделам математики, занимающимся численным анализом.

Некоторые математики, работающие в "высоких сферах" численного анализа, будут считать, что предлагаемые в этом томе темы относятся к сфере влияния системных программистов. А специалисты, работающие в "высоких сферах" системного программирования, наоборот, решат, что изучать рассматриваемые темы — дело численных аналитиков. Но я все-таки надеюсь, что найдутся читатели, которые захотят внимательно изучить эти фундаментальные методы. Хотя данные методы можно отнести, скорее всего, к нижнему уровню, на них основаны все грандиозные компьютерные приложения, предназначенные для решения числовых задач. Отсюда следует, насколько важно хорошо в них разобраться. В настоящем томе будет рассмотрена область, которая находится на стыке численного анализа и программирования; именно это и делает предмет книги весьма интересным.

В данном томе по сравнению с другими содержится значительно больший объем математического материала; это обусловлено спецификой изучаемых тем. Причем в большинстве случаев нужные математические темы раскрываются прямо на страницах книги практически с нуля (или на основании результатов, доказанных в томе 1). Но в некоторых разделах предполагается, что читатель знаком с материалом.

В этом томе содержатся главы 3 и 4. Глава 3 посвящена случайным числам: здесь изучаются не только различные методы генерирования случайных чисел, но

и статистические критерии случайности, а также преобразование равномерно распределенных случайных чисел в другие типы случайных величин. Последняя тема позволяет проиллюстрировать практическое применение случайных чисел. В эту главу также включен раздел, повествующий о природе самой случайности. Глава 4 — это захватывающая история о том, какие открытия были сделаны в арифметике в результате многовекового прогресса. В ней обсуждаются различные системы представления чисел и способы преобразования одной системы в другую; рассматриваются арифметика чисел с плавающей точкой, целых чисел высокой точности, рациональных дробей, полиномов и степенных рядов, а также вопросы разложения на множители и нахождения наибольших общих делителей.

Каждая из глав 3 и 4 может использоваться в качестве основы для семестрового университетского курса, причем изложение материала можно построить так, чтобы его можно было излагать на разных уровнях: как для первокурсников, так и для выпускников. В настоящее время курсы “Случайные числа” и “Арифметика” не входят в программы многих университетов. Но я надеюсь, читатель увидит, что в этих главах в едином ключе освещается материал, имеющий реальную образовательную ценность. Мой собственный опыт подтверждает, что он служит прекрасным способом ознакомления студентов с элементарной теорией вероятности и теорией чисел. Почти все темы, которые обычно рассматривают в таких вводных курсах, естественно возникают в связи с вопросами практического применения теории. Кроме того, обсуждение на лекциях вопросов практического использования результатов может стать той движущей силой, которая вызовет у студентов интерес к учебе и поможет понять важность и значение теории. Более того, в каждой главе содержатся упоминания о более сложных темах, которые у многих студентов вызовут интерес к дальнейшему изучению математики.

Этот том в основном представляет собой полную и самостоятельную книгу; исключение составляют только вопросы, касающиеся компьютера `MIX`, который описывался в томе 1. В приложении Б приведены использованные в данной книге математические обозначения, которые иногда отличаются от принятых в традиционной математической литературе.

Предисловие к третьему изданию

Когда в 1980 году второе издание этой книги было закончено, в ней впервые были использованы системы компьютерного набора TEX и METAFONT. А теперь я рад отметить завершение разработки этих систем возвратом к книге, которая вдохновила меня на их создание. Наконец-то мне удалось внести все тома в персональный компьютер и таким образом получить ее электронную версию, что позволит в дальнейшем вносить любые изменения в технологию печати и отображения на экране. Такой способ работы предоставил мне возможность сделать буквально тысячи улучшений, и я добился того, о чем так долго мечтал.

В этом новом издании я смог проверить каждое слово в тексте, стараясь сохранить юношеский задор оригинальных предложений и в то же время внести бóльшую зрелость суждений. Были добавлены десятки новых упражнений, а на десятки старых даны новые или улучшенные ответы. Изменения коснулись всего текста,

но особенно это относится к разделам 3.5 (теоретические основы случайности), 3.6 (универсальные генераторы случайных чисел), 4.5.2 (двоичный алгоритм нахождения наибольшего общего делителя) и 4.7 (композиция и итерация степенных рядов).

Таким образом, работа над книгой *Искусство программирования* продолжается. Исследования получисленных алгоритмов продвигаются с феноменальной скоростью. Именно поэтому некоторые части данной книги начинаются пиктограммой "В процессе построения" (это своеобразное извинение за то, что приведены не самые новые данные). Мои файлы переполнены важными материалами, которые я планирую включить в окончательное, знаменательное четвертое издание тома 2 (оно выйдет, вероятно, через 16 лет). Но сначала я должен закончить тома 4 и 5. Я хочу, чтобы они были опубликованы сразу же, как только будут готовы к печати.

Я чрезвычайно благодарен сотням людей, которые помогали мне собирать материал в течение последних 35 лет. Большая часть тяжелой работы по подготовке этого нового издания была выполнена Сильвио Леви (Silvio Levi), который профессионально отредактировал электронную версию текста, а также Джеффри Олдхэмом (Jefferey Oldham), который конвертировал почти все оригинальные иллюстрации в формат METAPOST. Я исправил все ошибки, которые бдительные читатели обнаружили во втором издании (а также ошибки, которых, увы, не заметил никто), и постарался избежать появления новых ошибок. Тем не менее я допускаю, что некоторые огрехи все же остались, и хотел бы исправить их как можно скорее. Поэтому за каждую опечатку*, а также ошибку, относящуюся к сути излагаемого материала или к приведенным историческим сведениям, я охотно заплачу $2,56 тому, кто первым ее найдет. На web-странице, адрес которой приведен на обложке книги, содержится текущий список всех ошибок, о которых мне сообщили.

Станфорд, Калифорния Д. Э. К.
Июль 1997

Когда работа над книгой продолжается в течение восьми лет, то появляется очень много людей — коллег, наборщиков, студентов, преподавателей и друзей, которых нужно поблагодарить. Но я не собираюсь освобождать их от ответственности за ошибки, которые остались в тексте. Они должны были их исправить! Иногда они даже несут ответственность за идеи, которые в конце концов оказываются ошибочными. Но в любом случае я благодарен всем своим сотрудникам.

— Эдвард Ф. КЭМПБЕЛЛ (мл.) EDWARD F. CAMPBELL, JR.) (1975)

Defendit numerus, [В числах ты найдешь покой] — *это истина дураков;*
Deperdit numerus, [В числах та найдешь погибель] — *истина мудрых.*

— Ч. К. КОЛТОН (C. C. COLTON) (1820)

* Имеется в виду оригинал настоящего издания. — *Примеч. ред.*

ПРИМЕЧАНИЯ К УПРАЖНЕНИЯМ

Упражнения, приведенные в этой серии книг, предназначены как для самостоятельной проработки, так и для семинарских занятий. Очень трудно и, наверное, просто невозможно выучить предмет, только читая теорию и не применяя ее для решения конкретных задач, которые заставляют задуматься о прочитанном. Более того, мы лучше всего заучиваем то, до чего дошли самостоятельно, своим умом. Поэтому упражнения занимают важное место в данном издании. Я приложил немало усилий, чтобы сделать их как можно более информативными, а также отобрать задачи, которые были бы не только поучительны, но и позволяли читателю получить удовольствие от их решения.

Во многих книгах простые упражнения даются вместе с исключительно сложными. Это не всегда удобно, так как читателю хочется знать заранее, сколько времени ему придется затратить на решение задач (иначе в лучшем случае он их только просмотрит). В качестве классического примера подобной ситуации можно привести книгу Ричарда Беллмана (Richard Bellman) *Динамическое программирование* (М.: Изд-во иностр. лит., 1960). Это очень важная, новаторская работа, но у нее есть один недостаток: в конце некоторых глав в разделе "Упражнения и научные проблемы" среди серьезных, еще нерешенных проблем приводятся простейшие вопросы. Говорят, что кто-то однажды спросил д-ра Беллмана, как отличить упражнения от научных проблем, и он ответил: "Если вы можете решить задачу, значит, это упражнение; в противном случае это научная проблема".

Совершенно очевидно, что в книге, подобной этой, должны быть приведены и сложные научные проблемы, и простейшие упражнения. Поэтому, чтобы читатель не ломал голову, пытаясь отличить одно от другого, были введены рейтинги, которые определяют степень сложности каждого упражнения. Эти рейтинги имеют следующее значение.

Рейтинг	*Объяснение*
00	Чрезвычайно простое упражнение, на которое можно ответить сразу же, если прочитанный материал понят. Упражнения подобного типа почти всегда можно решить "в уме".
10	Простая задача, которая заставляет задуматься над прочитанным, но не представляет особых трудностей. На ее решение вы затратите не больше минуты; в процессе решения могут понадобиться карандаш и бумага.
20	Средняя задача, которая позволяет проверить, понял ли читатель основные положения изложенного материала. Чтобы получить исчерпывающий ответ, может понадобиться примерно 15–20 минут.
30	Задача умеренной сложности. Для ее решения может понадобиться более двух часов (а если одновременно вы смотрите телевизор, то еще больше).

40 Достаточно сложная или трудоемкая задача, которую вполне можно включить в план семинарских занятий. Предполагается, что студент должен справиться с ней, затратив не слишком много времени, и решение будет нетривиальным.

50 Научная проблема, которая (насколько известно автору в момент написания книги) пока еще не получила удовлетворительного решения, хотя найти его пытались очень многие. Если вы нашли решение подобной проблемы, то опубликуйте его; более того, автор данной книги будет очень признателен, если ему сообщат решение как можно скорее (при условии, что оно правильно).

Интерполируя по этой "логарифмической" шкале, можно понять, что означает любой промежуточный рейтинг. Например, рейтинг *17* говорит о том, что упражнение немного проще, чем задача средней сложности. Если задача с рейтингом *50* будет впоследствии решена каким-либо читателем, то в следующих изданиях данной книги и в списке ошибок, опубликованных в Internet, она может иметь рейтинг *45* (адрес Web-страницы приводится на обложке книги).

Остаток от деления рейтинга на 5 показывает, какой объем рутинной работы потребуется для решения данной задачи. Таким образом, для выполнения упражнения с рейтингом *24* может потребоваться больше времени, чем для упражнения с рейтингом *25*, но для последнего необходим более творческий подход.

Автор очень старался правильно присвоить рейтинги упражнениям, но тому, кто составляет задачи, трудно предвидеть, насколько сложными они окажутся для кого-то другого. К тому же одному человеку некая задача может показаться простой, а другому — сложной. Таким образом, определение рейтингов — дело достаточно субъективное и относительное. Я надеюсь, что рейтинги помогут вам получить правильное представление о степени трудности задач, но их следует воспринимать в качестве ориентира, а не в качестве абсолюта.

Эта книга написана для читателей с различным уровнем математической подготовки и научного кругозора, поэтому некоторые упражнения рассчитаны исключительно на тех, кто серьезно интересуется математикой или занимается ею профессионально. Если рейтингу предшествует буква M, значит, математические понятия и обоснования используются в упражнении в большей степени, чем это необходимо тому, кто интересуется в основном программированием алгоритмов. Если же упражнение отмечено буквами *НМ*, то для его решения необходимо знание высшей математики в большем объеме, чем дается в настоящей книге. Но пометка *НМ* совсем необязательно означает, что упражнение трудное.

Перед некоторыми упражнениями стоит стрелка "▸", которая означает, что они особенно поучительны и их очень рекомендуется выполнить. Само собой разумеется, никто не ожидает, что читатель (или студент) будет решать *все* задачи, поэтому наиболее важные из них и были выделены. Но это ни в коем случае не означает, что другие упражнения выполнять не стоит! Каждый читатель должен хотя бы попытаться решить все задачи, рейтинг которых меньше или равен *10*. Стрелки помогут выбрать задачи с более высокими рейтингами, которые следует решать в первую очередь.

К большинству упражнений приведены ответы, помещенные в отдельном разделе в конце книги. Пожалуйста, пользуйтесь ими разумно: ответ смотрите только

после того, как приложите все усилия, чтобы решить задачу самостоятельно, либо если у вас совершенно нет времени на ее решение. Ответ будет поучителен и полезен для вас только в том случае, если вы ознакомитесь с ним *после* того, как найдете свое решение или изрядно потрудитесь над задачей. Ответы к задачам излагаются очень кратко и схематично, так как предполагается, что читатель честно пытался решить задачу собственными силами. Иногда в приведенном решении дается меньше информации, чем спрашивалось, но чаще бывает наоборот. Вполне возможно, что полученный вами ответ окажется лучше того, который помещен в книге, или вы найдете ошибку в ответе. В таком случае автор был бы очень признателен, если бы вы как можно скорее подробно сообщили ему об этом; тогда в последующих изданиях книги будет опубликовано более удачное решение, а также имя его автора.

Решая задачи, вы, как правило, можете пользоваться ответами к предыдущим упражнениям, за исключением случаев, когда это будет оговорено особо. Рейтинги упражнениям присваивались в расчете именно на это, и вполне возможно, что рейтинг упражнения $n + 1$ ниже рейтинга упражнения n, даже если результат упражнения n является его частным случаем.

Условные обозначения

▶ Рекомендуется
M С математическим уклоном
HM Требует знания высшей математики

00 Простейшее (ответ дать немедленно)
10 Простое (на одну минуту)
20 Средней трудности (на четверть часа)
30 Повышенной трудности
40 Высокой трудности
50 Научная проблема

УПРАЖНЕНИЯ

▶ **1.** [*00*] Что означает рейтинг *M20*?

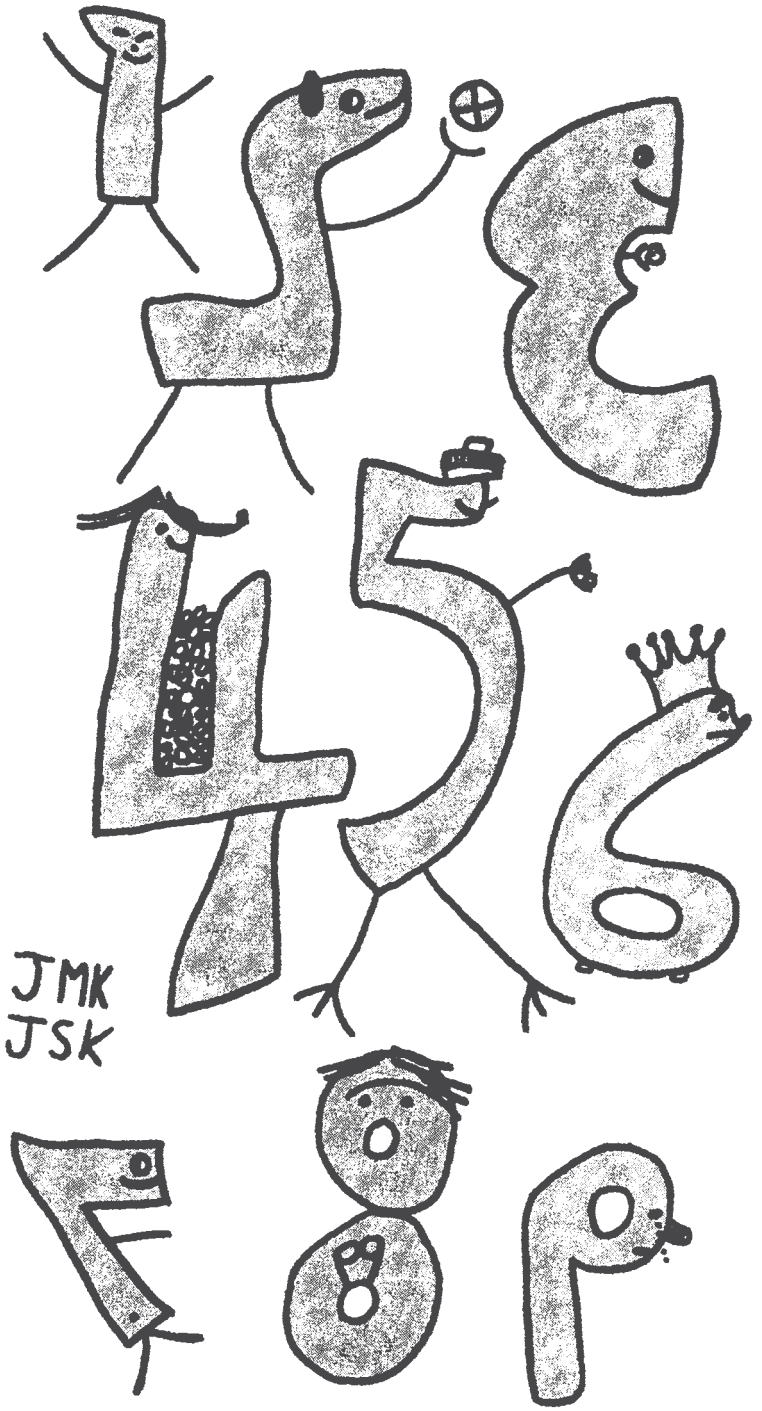
2. [*10*] Какое значение для читателя имеют упражнения, которые приводятся в учебниках?

3. [*34*] Леонард Эйлер (Leonhard Euler) в 1772 году предположил, что уравнение $w^4+x^4+y^4 = z^4$ не имеет решения в целых положительных числах, но Ноам Элкис (Noam Elkies) доказал в 1987 году, что существует бесконечное множество решений [см. *Math. Comp.* **51** (1988), 825–835]. Найдите все целочисленные решения, такие, что $0 \le w \le x \le y < z < 10^6$.

4. [*M50*] Докажите, что если n — целое число, $n > 4$, то уравнение $w^n + x^n + y^n = z^n$ неразрешимо в целых положительных числах w, x, y, z.

Упражнения — лучший инструмент познания.

— РОБЕРТ РЕКОРД (ROBERT RECORDE),
The Whetstone of Witte (1557)


JMK
JSK

ГЛАВА 3

СЛУЧАЙНЫЕ ЧИСЛА

Каждый, кто использует арифметические методы генерирования случайных чисел, безусловно, грешит.

— ДЖОН ФОН НЕЙМАН (JOHN VON NEUMANN) (1951)

О вероятности коль кто забудет, обманщиком вовек не будет.

— ДЖОН ГЕЙ (JOHN GAY) (1727)

Достаточно лишь нескольких лучей света, чтобы помочь людям в совершенствовании их "стохастических" способностей.

— ДЖОН ОУЭН (JOHN OWEN) (1662)

3.1. ВВЕДЕНИЕ

Числа, которые выбираются случайным образом, находят множество полезных применений.

a) *Моделирование.* При использовании компьютера для моделирования естественных явлений случайные числа нужны для того, чтобы сделать эти модели похожими на реальные явления. Моделирование применяется во многих областях, начиная от исследований в ядерной физике (где частицы испытывают случайные столкновения) и заканчивая исследованием операций (где люди прибывают, например, в аэропорт через случайные промежутки времени).

b) *Выборочный метод.* Часто невозможно исследовать все варианты, но случайная выборка обеспечивает понимание того, что можно назвать "типичным" поведением.

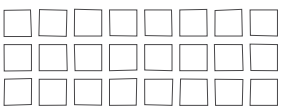
c) *Численный анализ.* Для решения сложных задач численного анализа была разработана остроумная техника, использующая случайные числа. Об этом написано несколько книг.

d) *Компьютерное программирование.* Случайные величины являются хорошим источником данных для тестирования эффективности компьютерных алгоритмов.

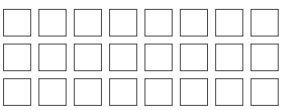
Более важно то, что они играют решающую роль при использовании *рандомизированных алгоритмов*, которые часто намного превосходят своих детерминированных двойников. В этой серии книг нас, в первую очередь, интересует именно такое использование случайных чисел. Этим объясняется то, что случайные числа рассматриваются уже здесь, в главе 3, прежде чем появится большинство других компьютерных алгоритмов.

e) *Принятие решений.* Говорят, что многие администраторы принимают решения, бросая монету, игральную кость либо каким нибудь другим подобным способом. Сплетничают, что некоторые профессора в колледжах ставят оценки, используя тот же метод. Иногда важно принять полностью "беспристрастное" решение. Случайность является также важной частью оптимальных стратегий в теории матричных игр.

f) *Эстетика.* Небольшая добавка случайности оживляет музыку и компьютерную графику. Например, рисунок



в определенном смысле выглядит привлекательнее, чем



[См. D. E. Knuth, *Bull. Amer. Math. Soc.* **1** (1979), 369.]

g) *Развлечения.* Многие считают, что они замечательно проводят время, бросая игральные кости, тасуя колоду карт, вращая колесо рулетки и т. п. Такие традиционные способы использования случайных чисел получили название *метод Монте–Карло*. Это общее название всех алгоритмов, использующих случайные числа.

Те же, кто интересуются этой темой, постоянно вовлекаются в философские дискуссии о значении слова "случайный". Возникает вопрос "А что является неслучайным числом?" Например, будет ли число 2 случайным? Охотнее говорят о *последовательности независимых случайных чисел* с заданным *распределением*, и это означает, если говорить не строго, что каждое число было получено случайно, не имея ничего общего с другими числами в последовательности, и что каждое число имеет заданную вероятность появления в любой заданной области значений.

Равномерным распределением на конечном множестве чисел (в дальнейшем — просто *равномерным распределением*) называется такое распределение, при котором любое из возможных чисел имеет одинаковую вероятность появления. Если не задано определенное распределение на конечном множестве чисел, то принято считать его равномерным.

Каждая из десяти цифр от 0 до 9 будет появляться примерно один раз из 10 в равномерной последовательности случайных цифр. Каждой паре двух последовательных цифр следует появиться один раз из ста и т. д. Однако если взять конкретную случайную последовательность длиной в миллион цифр, то она не всегда будет содержать 100 000 нулей, 100 000 единиц и т. д. Действительно, возможность появления такой последовательности незначительна; на самом деле, если рассматривать достаточно большую совокупность таких *последовательностей*, в среднем будет появляться 100 000 нулей, 100 000 единиц и т. д.

Любая конкретная последовательность, содержащая миллион цифр, так же вероятна, как и любая другая. Если мы выберем миллион цифр наудачу и если

окажется, что первые 999 999 из них — нули, то вероятность того, что последняя цифра в этой последовательности — также нуль, все еще остается точно равной одной десятой в истинно случайной ситуации. Это утверждение большинству кажется парадоксальным, однако оно не противоречит реальности.

Существует несколько приличных возможностей дать абстрактное определение случайности, и мы вернемся к этой интересной теме в разделе 3.5; пока что достаточно интуитивного понимания данной концепции.

В течение многих лет те, кому случайные числа были необходимы для научной работы, вынуждены были таскать шары из урны, предварительно хорошо перемешав их, либо бросать игральные кости, либо раскладывать карты. Таблица, содержащая более 40 000 взятых наудачу из отчетов о переписи случайных цифр, была опубликована в 1927 году Л. Х. К. Типпеттом (L. H. C. Tippett).

С тех пор были построены механические генераторы случайных чисел. Первая такая машина была использована в 1939 году М. Ж. Кендаллом (M. G. Kendall) и Б. Бабингтон-Смитом (B. Babington-Smith) для построения таблицы, содержащей 100 000 случайных цифр. Компьютер Ferranti Mark I, впервые запущенный в 1951 году, имел встроенную программу, использующую резисторный генератор шума, которая поставляла 20 случайных битов на сумматор. Этот метод был предложен А. М. Тьюрингом (A. M. Turing). В 1955 году RAND Corporation опубликовала широко используемые таблицы, в которых содержался миллион случайных цифр, полученных с помощью других специальных устройств. Известный генератор случайных чисел ERNIE применялся на протяжении многих лет для определения выигрышных номеров британской лотереи. [См. статьи Kendall and Babington-Smith *J. Royal Stat. Soc.* **A101** (1938), 147–166; **B6** (1939), 51–61, а также дискуссию S. H. Lavington's с Mark I в *CACM* **21** (1978), 4–12; обозрение в *Math. Comp.* **10** (1956), 39–43; дискуссию об ERNIE W. E. Thomson, *J. Royal Stat. Soc.* **A122** (1959), 301–333.]

Короче говоря, после изобретения компьютеров начались исследования эффективного способа получения случайных чисел, встроенных программно в компьютеры. Можно было применять таблицы, но пользы от этого метода было мало из-за ограниченной памяти компьютера и требуемого времени ввода, поэтому таблицы могли быть лишь слишком короткими; кроме того, было не особенно приятно составлять таблицы и пользоваться ими. Генератор ERNIE мог быть встроен в компьютер, как это было в Ferranti Mark I, но это оказалось неудобно, поскольку невозможно точно воспроизвести вычисления даже сразу по окончании работы программы; более того, такие генераторы часто давали сбои, что было крайне трудно обнаружить. Технологический прогресс позволил вернуться к использованию таблиц в 1990-е годы, так как миллиарды протестированных случайных байтов можно было разместить на компакт-дисках. Дж. Марсалья (George Marsaglia) помог оживить табличный метод в 1995 году, подготовив демонстрационный диск с 650 Мбайт случайных чисел, при генерировании которых запись шума диодной цепи сочеталась с определенным образом скомпонованной музыкой в стиле "рэп". (Он назвал это "белым и черным шумом".)

Несовершенство первых механических методов вначале пробудило интерес к получению случайных чисел с помощью обычных арифметических операций, заложенных в компьютер. Джон фон Нейман (John von Neumann) первым предложил

такой подход около 1946 года; его идея заключалась в том, чтобы возвести в квадрат предыдущее случайное число и выделить средние цифры. Например, мы хотим получить 10-значное число и предыдущее число равнялось 5772156649. Возводим его в квадрат и получаем

$$33317792380594909201;$$

значит, следующим числом будет 7923805949.

Совершенно очевидны претензии, предъявляемые к этому методу: как может быть случайной последовательность, генерируемая таким образом, если каждое число полностью определяется предыдущим? (См. эпиграф фон Неймана к этой главе.) Ответ состоит в том, что эта последовательность *не* случайна, но *кажется* такой. В типичных приложениях фактически существующая связь между двумя числами, следующими одно за другим, на самом деле не имеет значения; поэтому нельзя утверждать, что неслучайный характер последовательности нежелателен. Интуитивно ясно, что метод средин квадратов должен быть достаточно хорошим перемешиванием и заменой цифр предыдущего числа.

В "заумной" технической литературе последовательности, генерируемые детерминистическим путем, таким как этот, называются *псевдослучайными* или *квазислучайными*, однако в данной книге мы в основном просто будем называть их случайными последовательностями, понимая, что они только *кажутся* случайными. "Кажущаяся случайность" — это, возможно, все, что так или иначе может быть сказано о любой случайной последовательности. Случайные числа, генерируемые детерминистическими методами на компьютере, почти всегда работали достаточно хорошо при условии, что метод был выбран удачно. Конечно, детерминистическая последовательность не всегда применима; ею, безусловно, не следует заменять ERNIE в лотерее.

Метод средин квадратов фон Неймана, как было показано, фактически является сравнительно бедным источником случайных чисел. Опасность состоит в том, что последовательность стремится войти в привычную колею, т. е. короткий цикл повторяющихся элементов. Например, каждое появление нуля как числа последовательности приведет к тому, что все последующие числа также будут нулями.

Некоторые ученые экспериментировали с методом средин квадратов в начале 1950-х годов. Работая с четырехзначными числами вместо десятизначных, Дж. Э. Форсайт (G. E. Forsythe) испытал 16 различных начальных значений и обнаружил, что 12 из них приводят к циклическим последовательностям, заканчивающимся циклом 6100, 2100, 4100, 8100, 6100, ..., в то время как две из них приводят к последовательностям, вырождающимся в 0. Более интенсивные исследования, главным образом в двоичной системе счисления, провел Н. К. Метрополис (N. C. Metropolis). Он показал, что если использовать 20-разрядное число, то последовательность случайных чисел, полученная методом средин квадратов, вырождается в 13 различных циклов, причем длина самого большого периода равна 142.

Достаточно легко запустить метод средин квадратов с новым исходным значением, если обнаружить число "нуль", однако избавиться от длинных циклов довольно трудно. В упр. 6 и 7 обсуждается несколько интересных вариантов определения циклов периодических последовательностей, использующих достаточно малый объем памяти.

Теоретические недостатки метода средин квадратов приведены в упр. 9 и 10. С другой стороны, используя 38-разрядные числа, Метрополис получил невырожденную последовательность, содержащую около 750 000 чисел (прежде чем произошло вырождение), и полученные $750\,000 \times 38$ бит удовлетворительно прошли статистический тест на случайность. [*Symp. on Monte Carlo Methods* (Wiley, 1956), 29–36.] Эти опыты показали, что метод средин квадратов *может* давать удовлетворительные результаты, но ему опасно доверять, пока не выполнены тщательные вычисления.

Когда автор работал над первым изданием этой книги, многие генераторы случайных чисел (в литературе на русском языке параллельно употребляется термин "датчик случайных чисел". — *Примеч. ред.*), которые тогда использовались, были недостаточно хороши. Программисты традиционно не интересовались информацией о таких подпрограммах; старые методы, сравнительно неудовлетворительные, слепо переходили от одного программиста к другому, поскольку пользователи не понимали ограничений, при которых можно применять эти методы. Мы увидим здесь, что наиболее важные сведения о генераторах случайных чисел нетрудно изучить, несмотря на то что необходимо быть осторожным, чтобы избежать обычных ловушек.

Так нелегко придумать понятный всем и каждому датчик случайных чисел! В этом автор убедился в 1959 году, когда попытался создать фантастически хороший генератор случайных чисел, используя следующий необычный подход.

Алгоритм К (*Супергенератор случайных чисел*). Пусть задано 10-значное десятичное число X и этот алгоритм использует замену X другим числом так, чтобы получить случайную последовательность. Несмотря на то что от алгоритма можно ожидать на выходе всецело случайную последовательность, соображения, приведенные ниже, показывают, что это, к сожалению, не всегда так. (Читатель не обязан изучать этот алгоритм во всех деталях, но рекомендуется обратить внимание на его сложность; отметим, в частности, шаги К1 и К2.)

К1. [Выбрать число итераций.] Присвоить $Y \leftarrow \lfloor X/10^9 \rfloor$ наибольшую значащую цифру X. (Мы выполним шаги К2–К13 точно $Y + 1$ раз; т. е. применим рандомизированные преобразования *случайное* число раз.)

К2. [Выбрать случайный шаг.] Присвоить $Z \leftarrow \lfloor X/10^8 \rfloor \bmod 10$ следующую наибольшую значащую цифру X. Переходим к шагу К$(3 + Z)$, т. е. к *случайно* выбранному шагу в программе.

К3. [Обеспечить $\geq 5 \times 10^9$.] Если $X < 5000000000$, присвоить $X \leftarrow X + 5000000000$.

К4. [Средина квадрата.] Заменить X числом $\lfloor X^2/10^5 \rfloor \bmod 10^{10}$, т. е. серединой квадрата X.

К5. [Умножить.] Заменить X числом $(1001001001\,X) \bmod 10^{10}$.

К6. [Псевдодополнение.] Если $X < 100000000$, то присвоить $X \leftarrow X + 9814055677$; иначе присвоить $X \leftarrow 10^{10} - X$.

К7. [Переставить половины.] Поменять местами пять младших по порядку знаков X со старшими по порядку пятью знаками, т. е. присвоить $X \leftarrow 10^5(X \bmod 10^5) + \lfloor X/10^5 \rfloor$; это то же самое, что взять десять средних цифр числа $(10^{10} + 1)X$.

К8. [Умножить.] Выполнить шаг К5.

Таблица 1

КОЛОССАЛЬНОЕ СОВПАДЕНИЕ: АЛГОРИТМ К ПРЕОБРАЗОВАЛ ЧИСЛО 6065038420 САМО В СЕБЯ

Шаг	X (после)	
К1	6065038420	
К3	6065038420	
К4	6910360760	
К5	8031120760	
К6	1968879240	
К7	7924019688	
К8	9631707688	
К9	8520606577	
К10	8520506578	
К11	8520506578	
К12	0323372207	$Y = 6$
К6	9676627793	
К7	2779396766	
К8	4942162766	
К9	3831051655	
К10	3830951656	
К11	3830951656	
К12	1905867781	$Y = 5$
К12	3319967479	$Y = 4$
К6	6680032521	
К7	3252166800	
К8	2218966800	
К9	1107855700	
К10	1107755701	
К11	1107755701	
К12	1226919902	$Y = 3$
К5	0048821902	
К6	9862877579	
К7	7757998628	
К8	2384626628	
К9	1273515517	
К10	1273415518	
К11	1273415518	
К12	5870802097	$Y = 2$
К11	5870802097	
К12	3172562687	$Y = 1$
К4	1540029446	
К5	7015475446	
К6	2984524554	
К7	2455429845	
К8	2730274845	
К9	1620163734	
К10	1620063735	
К11	1620063735	
К12	6065038420	$Y = 0$

К9. [Уменьшить цифры.] Уменьшить каждую не равную нулю цифру десятичного представления числа X на единицу.

К10. [Модифицировать на 99999.] Если $X < 10^5$, присвоить $X \leftarrow X^2 + 99999$; иначе присвоить $X \leftarrow X - 99999$.

К11. [Нормировать.] (На этом шаге X не может быть равным нулю.) Если $X < 10^9$, присвоить $X \leftarrow 10X$ и повторить этот шаг.

К12. [Модификация метода средин квадратов.] Заменить X на $\lfloor X(X-1)/10^5 \rfloor \bmod 10^{10}$, т. е. средними 10 цифрами числа $X(X-1)$.

К13. [Повторить?] Если $Y > 0$, уменьшить Y на 1 и возвратиться к шагу К2. Если $Y = 0$, алгоритм завершен. Значение числа X, полученное на предыдущем шаге, и будет желаемым "случайным" значением. ▮

(Программа, реализующая этот алгоритм, обещала быть настолько сложной, что тот, кто прочел бы ее распечатку, без толковых комментариев не смог бы в ней разобраться.)

После рассмотрения всех преобразований алгоритма К не кажется ли правдоподобным, что можно было бы обеспечить бесконечное снабжение невероятно случайными числами? Нет! На самом деле, когда этот алгоритм впервые был реализован на компьютере, он почти немедленно сошелся к 10-значному числу 6065038420,

которое по невероятному совпадению преобразовалось само в себя по алгоритму (табл. 1). С другим стартовым числом последовательность начала повторяться после 7401 значения с периодом длиной 3178.

Мораль этой истории в том, что *случайные числа не следует генерировать методом, выбранным наудачу*. Не мешало бы использовать немного теории.

В следующих разделах будут рассмотрены генераторы случайных чисел более высокого уровня, чем метод средин квадратов и алгоритм К. Соответствующие последовательности гарантированно обладают желаемыми случайными свойствами и не вырождаются. Мы исследуем некоторые причины такого, похожего на случайное, поведения, а также покажем, как можно обращаться со случайными числами. Например, одно из наших исследований будет посвящено программе, которая тасует смоделированную на компьютере колоду карт.

В разделе 3.6 приводятся итоги к этой главе и некоторые библиографические источники.

УПРАЖНЕНИЯ

▸ **1.** [*20*] Предположим, вы хотите получить случайную десятичную цифру. Какой из следующих методов вы выберете?

a) Откроете телефонный справочник, ткнув пальцем куда-нибудь, выберете первый попавшийся номер телефона и возьмете младшую цифру (*цифру младшего разряда*) этого номера.
b) Поступите, как в (a), но выберете младшую цифру номера *страницы*.
c) Бросите игральную кость в форме икосаэдра, имеющую двадцать граней, которые помечены цифрами $0, 0, 1, 1, \ldots, 9, 9$. Когда кость остановится, выберете верхнюю цифру. (Для бросания игральной кости рекомендуется стол с хорошо натянутым сукном.)
d) На одну минуту выставите счетчик Гейгера у источника радиоактивного излучения (предварительно обезопасив себя) и воспользуетесь младшей цифрой показаний счетчика. Предполагается, что на счетчике Гейгера представлены числа в десятичной системе счисления и вначале на нем был установлен нуль.
e) Бросите быстрый взгляд на свои часы и, если секундная стрелка находится между $6n$ и $6(n+1)$ секундами, выберете цифру n.
f) Попросите приятеля задумать любую цифру и воспользуетесь ею.
g) Попросите врага задумать любую цифру и воспользуетесь ею.
h) Предположите, что 10 лошадей участвуют в забеге и вам о них абсолютно ничего не известно. Присвоите каждой лошади в произвольном порядке номер от 0 до 9, а после забега выберете в качестве случайной цифры номер победителя.

2. [*M22*] Какова вероятность того, что в случайной последовательности из миллиона десятичных цифр каждая возможная цифра встречается ровно 100 000 раз?

3. [*10*] Какое число следует за 1010101010 в методе средин квадратов?

4. [*20*] (a) Почему на шаге К11 алгоритма К значение X не может быть равно нулю? Какая ошибка возникла бы в алгоритме, если бы X мог принимать значение "нуль"? (b) Используя табл. 1, установите, что происходит, когда алгоритм К применяется повторно со стартовым значением $X = 3830951656$.

5. [*15*] Объясните, почему в любом случае, даже если совпадение, приведенное в табл. 1, не произошло, алгоритм К не сможет выдать бесконечную последовательность случайных чисел в том смысле, что любая последовательность, генерируемая алгоритмом К, станет в конце концов периодичной.

▸ **6.** [*M21*] Предположим, что необходимо получить последовательность целых случайных чисел X_0, X_1, X_2, ... на интервале $0 \le X_n < m$. Пусть $f(x)$ — любая функция, такая, что неравенство $0 \le x < m$ влечет $0 \le f(x) < m$. Рассмотрим последовательность $X_{n+1} = f(X_n)$. (Примеры таких последовательностей — метод средин квадратов и алгоритм К.)

a) Покажите, что такая последовательность периодична в том смысле, что существуют числа λ и μ, для которых значения

$$X_0,\ X_1,\ \ldots,\ X_\mu,\ \ldots, X_{\mu+\lambda-1}$$

различны, однако $X_{n+\lambda} = X_n$, когда $n \ge \mu$. Определите возможные максимальное и минимальное значения μ и λ.

b) (Р. В. Флойд (R. W. Floyd).) Покажите, что существует такое $n > 0$, что $X_n = X_{2n}$, и наименьшее такое значение n лежит в интервале $\mu \le n \le \mu + \lambda$. Более того, значение X_n является единственным в том смысле, что если $X_n = X_{2n}$ и $X_r = X_{2r}$, то $X_r = X_n$.

c) Используя идеи (b), составьте алгоритм вычисления μ и λ для любой заданной функции f и любого заданного X_0, используя только $O(\mu+\lambda)$ шагов и только ограниченный объем памяти.

▸ **7.** [*M21*] (Р. П. Брент (R. P. Brent), 1977.) Пусть $\ell(n)$ — наибольшее целое число, такое, что $2^{\ell(n)} \le n$, $\ell(n) = 2^k$, где k — целое число. Так, например, $\ell(15) = 8$ и $\ell(\ell(n)) = \ell(n)$.

a) Покажите, что в терминах обозначений упр. 6 существует такое $n > 0$, что $X_n = X_{\ell(n)-1}$. Найдите формулу для наименьшего такого n в терминах чисел μ и λ, определяющих период.

b) Примените этот результат для составления алгоритма, который может быть использован совместно с любым генератором случайных чисел типа $X_{n+1} = f(X_n)$, чтобы предотвратить циклическую неопределенность. Вашему алгоритму следует вычислять период длиной λ и использовать только небольшой объем памяти — вы просто не должны заполнять всю память вычисленными значениями последовательности!

8. [*23*] Выполните полную проверку метода средин квадратов для случая, когда десятичные числа состоят из двух цифр.

a) Можете начать процесс с любого из 100 допустимых чисел 00, 01, ..., 99. Сколько из этих значений в конечном счете приведут к повторению цикла (зацикливанию) 00, 00, ...? [*Пример.* Начиная с числа 43, получим последовательность 43, 84, 05, 02, 00, 00, 00, ...]

b) Сколько существует финальных циклов? Каков размер самого длинного цикла?

c) Какое начальное значение (или значения) даст наибольшее число различных элементов, прежде чем последовательность повторится?

9. [*M14*] Докажите, что метод средин квадратов, использующий $2n$-значные числа в b-ичной системе счисления, имеет следующие недостатки: если последовательность включает любое число, в котором n старших значащих цифр — нули, то последующие числа становятся все меньше и меньше, пока не превратятся в нули.

10. [*M16*] Пусть выполняются предположения предыдущего упражнения. Что можно сказать о последовательности чисел, следующих за X, если *младшие* значащие n цифр числа X равны нулю? Что если $n+1$ младших значащих цифр равны нулю?

▸ **11.** [*M26*] Рассмотрим последовательности генераторов случайных чисел, имеющих вид, описанный в упр. 6. Если выбрать $f(x)$ и X_0 наудачу (другими словами, если предположить, что каждая из m^m возможных функций $f(x)$ равновероятна и каждое из m возможных значений X_0 равновероятно), то какова вероятность того, что последовательность в конечном счете выродится в цикл длиной $\lambda = 1$? [*Замечание.* Предположения этой задачи дают повод задуматься о "случайности" генераторов случайных чисел такого типа. Можно

ожидать, что метод, подобный алгоритму K, отчасти ведет себя так же, как рассмотренный здесь генератор; ответ на эту задачу дает колоссальное число совпадений в табл. 1.]

▸ **12.** [*M31*] Какова средняя длина финального цикла, если выполняются предположения предыдущего упражнения? Какова средняя длина последовательности до вхождения в цикл? (В обозначениях упр. 6 необходимо определить средние значения λ и $\mu + \lambda$.)

13. [*M42*] Если $f(x)$ выбрана наудачу, как в упр. 11, какова средняя длина самого *длинного* цикла, полученного путем варьирования начального значения X_0? [*Замечание.* Мы уже рассмотрели аналогичную проблему для случая, когда $f(x)$ — это случайные перестановки; см. упр. 1.3.3–23.]

14. [*M38*] Если $f(x)$ выбрано наудачу, как в упр. 11, каково среднее число различных финальных циклов, полученных в результате варьирования начальных значений? [См. упр. 8, (b).]

15. [*M15*] Если $f(x)$ выбрано наудачу, как и в упр. 11, чему равна вероятность, что ни один из финальных циклов не имеет длину, равную 1, невзирая на выбор X_0?

16. [*15*] Последовательность, генерируемая, как в упр. 6, должна повторяться после того, как было сгенерировано не более m значений. Предположим, что мы обобщим метод таким образом, что X_{n+1} будет зависеть от X_{n-1} так же, как от X_n; формально пусть $f(x, y)$ — такая функция, для которой $0 \le x, y < m$ влечет неравенства $0 \le f(x, y) < m$. Последовательность строится так: сначала произвольно выбирают X_0 и X_1, а затем полагают, что

$$X_{n+1} = f(X_n, X_{n-1}), \qquad \text{где } n > 0.$$

Чему предположительно равен максимальный период в этом случае?

17. [*10*] Обобщите ситуацию из предыдущего упражнения так, чтобы X_{n+1} зависело от предыдущих k значений последовательности.

18. [*M20*] Придумайте метод, аналогичный методу из упр. 7, для определения цикла генератора случайных чисел, описанного в упр. 17, в общем виде.

19. [*M48*] Выполните упр. 11, используя упр. 15, в более общем случае, когда X_{n+1} зависят от k предыдущих значений последовательности; каждая из m^{m^k} функций $f(x_1, \ldots, x_k)$ считается равновероятной. [*Замечание.* Число функций, которые дают *максимальный* период, анализируется в упр. 2.3.4.2–23.]

20. [*30*] Найдите все неотрицательные числа $X < 10^{10}$, которые при использовании алгоритма K в конечном счете приводят к самовоспроизводящимся числам из табл. 1.

21. [*42*] Докажите или опровергните следующее утверждение: отображение $X \mapsto f(X)$, определенное алгоритмом K, имеет ровно пять циклов длиной 3178, 1606, 1024, 943 и 1.

22. [*21*] (Г. Роллетшек (H. Rolletschek).) Хороша ли идея генерирования случайных чисел с помощью последовательности $f(0)$, $f(1)$, $f(2)$, ..., где f — случайная функция, вместо того, чтобы использовать x_0, $f(x_0)$, $f(f(x_0))$ и т. д.?

▸ **23.** [*M26*] (Д. Фоата (D. Foata) и А. Фучс (A. Fuchs), 1970.) Покажите, что каждая из m^m функций $f(x)$, рассмотренных в упр. 6, может быть представлена как последовательность $(x_0, x_1, \ldots, x_{m-1})$, имеющая такие свойства.

i) $(x_0, x_1, \ldots, x_{m-1})$ — это перестановки последовательности $\big(f(0), f(1), \ldots, f(m-1)\big)$.

ii) $\big(f(0), \ldots, f(m-1)\big)$ может быть единственным образом восстановлена из последовательности $(x_0, x_1, \ldots, x_{m-1})$.

iii) Элементы, которые появляются в циклах из f, имеют вид $\{x_0, x_1, \ldots, x_{k-1}\}$, где k — самый большой индекс, такой, что эти k элементов различны.

iv) $x_j \notin \{x_0, x_1, \ldots, x_{j-1}\}$ влечет $x_{j-1} = f(x_j)$, если x_j не является наименьшим элементом в цикле из f.

v) $(f(0), f(1), \ldots, f(m-1))$ — это перестановка последовательности $(0, 1, \ldots, m-1)$ тогда и только тогда, когда $(x_0, x_1, \ldots, x_{m-1})$ представляет собой *обратную* перестановку к той перестановке, которая в разделе 1.3.3 названа необычным соответствием.

vi) $x_0 = x_1$ тогда и только тогда, когда $(x_1, \ldots, x_{m-1})$ представляет собой ориентированное дерево, построенное в упр. 2.3.4.4–18, с $f(x)$, порождающим x.

3.2. ГЕНЕРИРОВАНИЕ РАВНОМЕРНО РАСПРЕДЕЛЕННЫХ СЛУЧАЙНЫХ ЧИСЕЛ

В ЭТОМ РАЗДЕЛЕ будут рассмотрены методы генерирования последовательности случайных дробей, т. е. случайных *действительных чисел U_n, равномерно распределенных между нулем и единицей (на интервале $[0,1]$)*. Так как компьютер может представлять действительные числа только с определенной точностью, мы будем генерировать целое число X_n между нулем и некоторым числом m: дробь

$$U_n = X_n/m$$

будет, следовательно, лежать между нулем и единицей. Обычно m выбирают равным размеру слова в компьютере. (В этой книге размером слова (*word size*) автор называет число b^e, где b — основание системы счисления, используемой в компьютере, а e — число разрядов машины. — *Примеч. ред.*) Поэтому X_n можно по традиции рассматривать как целое число, занимающее все компьютерное слово, с точкой, которая отделяет целую часть числа от дробной, стоящей в правом конце слова, а U_n, если хотите, — как содержание того же слова с разделяющей точкой, стоящей в левом конце слова.

3.2.1. Линейный конгруэнтный метод

В настоящее время наиболее популярными генераторами случайных чисел являются генераторы, в которых используется следующая схема, предложенная Д. Г. Лемером (D. H. Lehmer) в 1949 году [см. Proc. 2nd Symp. on Large–Scale Digital Calculating Machinary (Cambridge, Mass.: Harvard University Press, 1951, 141–146)]. Выберем четыре "волшебных числа":

$$\begin{array}{rll} m, & \text{модуль;} & 0 < m; \\ a, & \text{множитель;} & 0 \le a < m; \\ c, & \text{приращение;} & 0 \le c < m; \\ X_0, & \text{начальное значение;} & 0 \le X_0 < m. \end{array} \tag{1}$$

Затем получим желаемую последовательность случайных чисел $\langle X_n \rangle$, полагая

$$X_{n+1} = (aX_n + c) \bmod m, \qquad n \ge 0. \tag{2}$$

Эта последовательность называется *линейной конгруэнтной последовательностью*. Получение остатков по модулю m отчасти напоминает предопределенность, когда шарик попадает в ячейку крутящегося колеса рулетки. Например, для $m = 10$ и $X_0 = a = c = 7$ получим последовательность

$$7,\ 6,\ 9,\ 0,\ 7,\ 6,\ 9,\ 0,\ \ldots\ . \tag{3}$$

Как показывает этот пример, такая последовательность не может быть "случайной" при некоторых наборах чисел m, a, c и X_0. Принципы выбора подходящих волшебных чисел будут подробно исследованы в следующих разделах этой главы.

В примере (3) иллюстрируется тот факт, что конгруэнтная последовательность всегда образует петли, т. е. обязательно существует цикл, повторяющийся бесконечное число раз. Это свойство является общим для всех последовательностей вида $X_{n+1} = f(X_n)$, где f преобразует конечное множество само в себя (см. упр. 3.1–6).

Повторяющиеся циклы называются *периодами*; длина периода последовательности (3) равна 4. Безусловно, последовательности, которые мы будем использовать, имеют относительно длинный период.

Заслуживает внимания случай $c = 0$, так как генерируемые числа будут иметь меньший период, чем при $c \neq 0$. Мы убедимся в дальнейшем, что ограничение $c = 0$ уменьшает длину периода последовательности, хотя при этом все еще возможно сделать период достаточно длинным. В оригинальном методе, предложенном Д. Г. Лемером, c выбиралось равным нулю, хотя он и допускал случай, когда $c \neq 0$, как один из возможных. Тот факт, что условие $c \neq 0$ может приводить к появлению более длинных периодов, был установлен В. Е. Томсоном (W. E. Thomson) [Comp. J. 1 p. 83, 86] и независимо от него А. Ротенбергом (A. Rotenberg) [JACM 7 (1960), 75–77]. Многие авторы называют линейную конгруэнтную последовательность при $c = 0$ *мультипликативным конгруэнтным методом*, а при $c \neq 0$— *смешанным конгруэнтным методом*. Буквы m, a, c и X_0 будут использованы в этой главе в том смысле, в каком они вводились раньше. То же самое относится и к константе

$$b = a - 1, \tag{4}$$

которая вводится для упрощения многих наших формул.

Можно сразу отбросить случай, когда $a = 1$, при котором последовательность X_n представима в виде $X_n = (X_0 + nc) \bmod m$ и ведет себя явно не как случайная последовательность. Случай, когда $a = 0$, даже хуже предыдущего. Следовательно, для практических целей предполагаем, что

$$a \geq 2, \qquad b \geq 1. \tag{5}$$

Сейчас можно обобщить формулу (2)

$$X_{n+k} = \big(a^k X_n + (a^k - 1)c/b\big) \bmod m, \qquad k \geq 0, \quad n \geq 0, \tag{6}$$

где $(n + k)$-й член выражается непосредственно через n-й. (Случай, когда $n = 0$, в этом уравнении также достоин внимания.) Из (4) следует, что подпоследовательность, содержащая каждый k-й член последовательности $\langle X_n \rangle$, является также линейной конгруэнтной последовательностью, множитель которой равен $a^k \bmod m$ и приращение равно $\big((a^k - 1)c/b\big) \bmod m$. Важным следствием из (6) является то, что общая последовательность, определенная с помощью a, c и X_0, может быть очень просто выражена в терминах специального случая, когда $c = 1$ и $X_0 = 0$. Пусть

$$Y_0 = 0, \qquad Y_{n+1} = (aY_n + 1) \bmod m. \tag{7}$$

В соответствии с (6) получим $Y_k \equiv (a^k - 1)/b$ (по модулю m). Значит, последовательность, определенная в (2), будет иметь вид

$$X_n = (AY_n + X_0) \bmod m, \qquad \text{где } A = (X_0 b + c) \bmod m. \tag{8}$$

УПРАЖНЕНИЯ

1. [*10*] В примере (3) показана ситуация, когда $X_4 = X_0$, так что последовательность начинается сначала. Приведите пример линейной конгруэнтной последовательности при $m = 10$, для которой число X_0 никогда снова не появится.

▸ **2.** [*M20*] Покажите, что если a и m взаимно простые, то X_0 всегда появляется в периоде.

3. [*M10*] Объясните, почему последовательность имеет определенные недостатки и, вероятно, не очень случайна, если a и m — не взаимно простые числа. Поэтому следует выбирать a и m так, чтобы они были взаимно простыми.

4. [*11*] Докажите формулу (6).

5. [*M20*] Соотношение (6) справедливо при $k \geq 0$. Если это возможно, получите формулы для X_{n+k} в терминах X_n для *отрицательных* значений k.

3.2.1.1. Выбор модуля. Первая задача, которую мы рассмотрим, — нахождение хороших значений параметров, определяющих линейную конгруэнтную последовательность. Сначала выясним, как правильно выбрать число m. Необходимо, чтобы m было довольно большим, так как период не может иметь больше m элементов. (Даже если мы намерены генерировать только случайные нули и единицы, *не* следует брать $m = 2$, ибо тогда последовательность в лучшем случае будет иметь вид $\ldots, 0, 1, 0, 1, 0, 1, \ldots$! Методы получения случайных нулей и единиц из линейной конгруэнтной последовательности обсуждаются в разделе 3.4.)

Другой фактор, который оказывает влияние на выбор m, — скорость генерирования: нужно подобрать значение m так, чтобы $(aX_n + c) \bmod m$ вычислялось быстро.

В качестве примера рассмотрим компьютер MIX. Можно вычислить $y \bmod m$, помещая y в регистры A и X и выполняя деление на m. Если y и m положительны, то $y \bmod m$ появится в регистре X. Но деление — сравнительно медленно протекающая операция, и этот недостаток можно компенсировать, если выбрать значение m таким, что особенно удобно, как *длина слова* нашего компьютера.

Пусть w будет длиной компьютерного слова, а именно — 2^e на e-разрядном двоичном компьютере или 10^e на e-цифровой десятичной вычислительной машине. (В настоящей книге мы часто будем употреблять букву e для обозначения произвольной целой степени. Несмотря на то что эта буква часто используется для обозначения основания натурального логарифма, мы надеемся, что читателю из контекста будет понятно, что она обозначает. Физики сталкиваются с подобными проблемами, когда используют e для обозначения заряда электрона.) Результат операции суммирования обычно дается по модулю w (но не на машинах, использующих процедуру единичного дополнения); умножение по модулю w также очень простое, поскольку затрагиваются только младшие разряды произведения. Таким образом, следующая программа эффективно вычисляет величину $(aX + c) \bmod w$.

```
LDA  A     rA ← a.
MUL  X     rAX ← (rA) · X.
SLAX 5     rA ← rAX mod w.
ADD  C     rA ← (rA + c) mod w.  ∎
```
(1)

Результат появляется в регистре A. В конце программы возможно переполнение; если это нежелательно, то следует, допустим, команда "`JOV *+1`" — "выключить".

"Умная" техника, обычно менее известная, может использовать представление вычисления по модулю $w + 1$. По причинам, объясненным ниже, как правило, требуется, чтобы $c = 0$, когда $m = w + 1$; тогда мы просто должны вычислить

$(aX) \bmod (w+1)$. Делает это следующая программа.

```
01  LDAN X        rA ← −X.
02  MUL  A        rAX ← (rA) · a.
03  STX  TEMP
04  SUB  TEMP     rA ← rA − rX.                    (2)
05  JANN *+3      Выход, если rA ≥ 0.
06  INCA 2        rA ← rA + 2.
07  ADD  =w−1=    rA ← rA + w − 1.   ∎
```

В регистре A сейчас содержится значение $(aX) \bmod (w+1)$. Конечно, оно может лежать где-нибудь между 0 и w включительно, так что читатель может законно удивиться, как можно представить так много значений в регистре A! (Обычно регистр не может хранить число, большее, чем $w-1$.) Ответом является то, что переполнение в программе (2) происходит тогда и только тогда, когда результат равен w (если предположить, что переполнение убрано в исходном положении). Можно отобразить w в виде нуля, так как программу (2) обычно нельзя использовать, когда $X = 0$; но более удобно просто отбросить значение w, если оно появляется в конгруэнтной последовательности по модулю $w+1$. Затем также можно избежать переполнения, просто заменив строки 05 и 06 в (2) строками "`JANN *+4; INCA 2; JAP *-5`".

Для доказательства того, что программа (2) действительно вычисляет $(aX) \bmod (w+1)$, заметим, что в строке 04 младшие разряды произведения вычитаются из старших разрядов. Переполнение не может произойти на этом шаге, и, если $aX = qw + r$ при $0 \le r < w$, получим значение $r - q$ в регистре A после строки 04. Сейчас

$$aX = q(w+1) + (r-q)$$

и мы имеем $-w < r - q < w$, так как $q < w$; следовательно, $(aX) \bmod (w+1)$ равно одному из двух значений ($r-q$ или $r-q+(w+1)$) в зависимости от того $r - q \ge 0$ или $r - q < 0$.

Подобная техника может быть использована для получения произведения двух чисел по модулю $(w-1)$; см. упр. 8.

Для освоения следующих разделов требуется знать простые множители m, чтобы правильно выбрать a. В табл. 1 впервые дается полный список разложений на простые множители $w \pm 1$ почти для каждой известной длины компьютерного слова; при желании методы из раздела 4.5.4 можно использовать для расширения таблицы.

Читатель может поинтересоваться, почему здесь обсуждается использование $m = w \pm 1$, когда выбор $m = w$ так явно удобен. Причина в том, что, *когда $m = w$, цифры правой части X_n гораздо менее случайны, чем цифры левой части.* Если d является делителем m и если

$$Y_n = X_n \bmod d, \tag{3}$$

можно легко показать, что

$$Y_{n+1} = (aY_n + c) \bmod d. \tag{4}$$

(Пусть $X_{n+1} = aX_n + c - qm$, где q — некоторое целое число. Если обе части равенства взять по модулю d, можно потерять qm, когда d — множитель m.)

Для иллюстрации важности выражения (4) предположим, например, что имеется двоичный компьютер. Если $m = w = 2^e$, младшие четыре разряда X_n являются

Таблица 1

РАЗЛОЖЕНИЕ НА ПРОСТЫЕ МНОЖИТЕЛИ $w \pm 1$

$2^e - 1$	e	$2^e + 1$
$7 \cdot 31 \cdot 151$	15	$3^2 \cdot 11 \cdot 331$
$3 \cdot 5 \cdot 17 \cdot 257$	16	65537
131071	17	$3 \cdot 43691$
$3^3 \cdot 7 \cdot 19 \cdot 73$	18	$5 \cdot 13 \cdot 37 \cdot 109$
524287	19	$3 \cdot 174763$
$3 \cdot 5^2 \cdot 11 \cdot 31 \cdot 41$	20	$17 \cdot 61681$
$7^2 \cdot 127 \cdot 337$	21	$3^2 \cdot 43 \cdot 5419$
$3 \cdot 23 \cdot 89 \cdot 683$	22	$5 \cdot 397 \cdot 2113$
$47 \cdot 178481$	23	$3 \cdot 2796203$
$3^2 \cdot 5 \cdot 7 \cdot 13 \cdot 17 \cdot 241$	24	$97 \cdot 257 \cdot 673$
$31 \cdot 601 \cdot 1801$	25	$3 \cdot 11 \cdot 251 \cdot 4051$
$3 \cdot 2731 \cdot 8191$	26	$5 \cdot 53 \cdot 157 \cdot 1613$
$7 \cdot 73 \cdot 262657$	27	$3^4 \cdot 19 \cdot 87211$
$3 \cdot 5 \cdot 29 \cdot 43 \cdot 113 \cdot 127$	28	$17 \cdot 15790321$
$233 \cdot 1103 \cdot 2089$	29	$3 \cdot 59 \cdot 3033169$
$3^2 \cdot 7 \cdot 11 \cdot 31 \cdot 151 \cdot 331$	30	$5^2 \cdot 13 \cdot 41 \cdot 61 \cdot 1321$
2147483647	31	$3 \cdot 715827883$
$3 \cdot 5 \cdot 17 \cdot 257 \cdot 65537$	32	$641 \cdot 6700417$
$7 \cdot 23 \cdot 89 \cdot 599479$	33	$3^2 \cdot 67 \cdot 683 \cdot 20857$
$3 \cdot 43691 \cdot 131071$	34	$5 \cdot 137 \cdot 953 \cdot 26317$
$31 \cdot 71 \cdot 127 \cdot 122921$	35	$3 \cdot 11 \cdot 43 \cdot 281 \cdot 86171$
$3^3 \cdot 5 \cdot 7 \cdot 13 \cdot 19 \cdot 37 \cdot 73 \cdot 109$	36	$17 \cdot 241 \cdot 433 \cdot 38737$
$223 \cdot 616318177$	37	$3 \cdot 1777 \cdot 25781083$
$3 \cdot 174763 \cdot 524287$	38	$5 \cdot 229 \cdot 457 \cdot 525313$
$7 \cdot 79 \cdot 8191 \cdot 121369$	39	$3^2 \cdot 2731 \cdot 22366891$
$3 \cdot 5^2 \cdot 11 \cdot 17 \cdot 31 \cdot 41 \cdot 61681$	40	$257 \cdot 4278255361$
$13367 \cdot 164511353$	41	$3 \cdot 83 \cdot 8831418697$
$3^2 \cdot 7^2 \cdot 43 \cdot 127 \cdot 337 \cdot 5419$	42	$5 \cdot 13 \cdot 29 \cdot 113 \cdot 1429 \cdot 14449$
$431 \cdot 9719 \cdot 2099863$	43	$3 \cdot 2932031007403$
$3 \cdot 5 \cdot 23 \cdot 89 \cdot 397 \cdot 683 \cdot 2113$	44	$17 \cdot 353 \cdot 2931542417$
$7 \cdot 31 \cdot 73 \cdot 151 \cdot 631 \cdot 23311$	45	$3^3 \cdot 11 \cdot 19 \cdot 331 \cdot 18837001$
$3 \cdot 47 \cdot 178481 \cdot 2796203$	46	$5 \cdot 277 \cdot 1013 \cdot 1657 \cdot 30269$
$2351 \cdot 4513 \cdot 13264529$	47	$3 \cdot 283 \cdot 165768537521$
$3^2 \cdot 5 \cdot 7 \cdot 13 \cdot 17 \cdot 97 \cdot 241 \cdot 257 \cdot 673$	48	$193 \cdot 65537 \cdot 22253377$
$179951 \cdot 3203431780337$	59	$3 \cdot 2833 \cdot 37171 \cdot 1824726041$
$3^2 \cdot 5^2 \cdot 7 \cdot 11 \cdot 13 \cdot 31 \cdot 41 \cdot 61 \cdot 151 \cdot 331 \cdot 1321$	60	$17 \cdot 241 \cdot 61681 \cdot 4562284561$
$7^2 \cdot 73 \cdot 127 \cdot 337 \cdot 92737 \cdot 649657$	63	$3^3 \cdot 19 \cdot 43 \cdot 5419 \cdot 77158673929$
$3 \cdot 5 \cdot 17 \cdot 257 \cdot 641 \cdot 65537 \cdot 6700417$	64	$274177 \cdot 67280421310721$

$10^e - 1$	e	$10^e + 1$
$3^3 \cdot 7 \cdot 11 \cdot 13 \cdot 37$	6	$101 \cdot 9901$
$3^2 \cdot 239 \cdot 4649$	7	$11 \cdot 909091$
$3^2 \cdot 11 \cdot 73 \cdot 101 \cdot 137$	8	$17 \cdot 5882353$
$3^4 \cdot 37 \cdot 333667$	9	$7 \cdot 11 \cdot 13 \cdot 19 \cdot 52579$
$3^2 \cdot 11 \cdot 41 \cdot 271 \cdot 9091$	10	$101 \cdot 3541 \cdot 27961$
$3^2 \cdot 21649 \cdot 513239$	11	$11^2 \cdot 23 \cdot 4093 \cdot 8779$
$3^3 \cdot 7 \cdot 11 \cdot 13 \cdot 37 \cdot 101 \cdot 9901$	12	$73 \cdot 137 \cdot 99990001$
$3^2 \cdot 11 \cdot 17 \cdot 73 \cdot 101 \cdot 137 \cdot 5882353$	16	$353 \cdot 449 \cdot 641 \cdot 1409 \cdot 69857$

числами $Y_n = X_n \bmod 2^4$. Суть выражения (4) состоит в том, что младшие четыре разряда $\langle X_n \rangle$ формируют конгруэнтную последовательность с периодом 16 или меньше. Аналогично пять младших разрядов являются периодичными с периодом не более 32 и наименьший значащий разряд X_n является либо постоянным, либо строго периодичным.

Подобная ситуация не возникает, когда $m = w \pm 1$; в таком случае младшие разряды X_n ведут себя так же случайно, как и старшие. Например, при $w = 2^{35}$ и $m = 2^{35} - 1$ числа последовательности будут не очень случайны, если рассмотреть только их остатки по модулю 31, 71, 127 или 122921 (см. табл. 1); но младшие разряды, которые представляют числа последовательности, взятые по mod 2, будут достаточно случайны.

Альтернатива состоит в том, чтобы в качестве m взять наибольшее простое число, меньшее, чем w. Это простое число можно найти, используя методы из раздела 4.5.4 и таблицы из того же раздела, в которых содержатся подходящие большие простые числа.

В большинстве случаев применения младшие разряды несущественны и выбор $m = w$ является совершенно удовлетворительным при условии, что программист, работающий со случайными числами, делает это сознательно.

Обсуждение до сих пор базировалось на использующих "величины со знаками" компьютерах типа MIX. Подобные идеи применяются в вычислительных машинах с дополнительной системой обозначений, хотя есть несколько полезных разновидностей. Например, компьютер DECsystem 20 имеет 36 бит с двоичным арифметическим дополнением; когда он вычисляет произведение двух неотрицательных чисел, младшие разряды содержат 35 бит со знаком "плюс". На этой вычислительной машине следовало бы полагать, что $w = 2^{35}$, но не 2^{36}. 32-битовое двоичное арифметическое дополнение на компьютерах IBM System/370 другое: младшие разряды операции умножения содержат 32 полных бита. Некоторые программисты считают, что это недостаток, так как младшие разряды могут быть отрицательными, когда исходное число положительно, и досадно корректировать это. На самом деле есть определенные *преимущества* с точки зрения генерирования случайных чисел, так как можно брать $m = 2^{32}$ вместо 2^{31} (см. упр. 4).

УПРАЖНЕНИЯ

1. [*M12*] В упр. 3.2.1–3 сделан вывод о том, что наилучший конгруэнтный генератор будет иметь множитель a, взаимно простой с m. Покажите, что, когда $m = w$, возможно лучшее вычисление $(aX + c) \bmod w$ точно в *трех* операциях MIX, чем в четырех операциях (1), с результатом, появляющимся в регистре X.

2. [*16*] Напишите подпрограмму на MIX, имеющую следующие характеристики.

Вызывающая последовательность:	`JMP RANDM`
Условия на входе:	Адрес ячейки `XRAND` содержит целое X
Условия на выходе:	$X \leftarrow \mathrm{rA} \leftarrow (aX + c) \bmod w$, $\mathrm{rX} \leftarrow 0$, переполнение выключено

(В результате обращения к этой подпрограмме можно получить следующее случайное число линейной конгруэнтной последовательности.)

▸ **3.** [*M25*] Многие компьютеры не имеют возможности делить числа из двух слов на числа из одного слова; они позволяют выполнять только операции над числами из отдельных слов, такие как операция himult — $\text{himult}(x,y) = \lfloor xy/w \rfloor$ и операция lomult — $\text{lomult}(x,y) = xy \bmod w$, когда x и y — неотрицательные целые числа, меньшие, чем компьютерное слово w. Объясните, как вычислить $ax \bmod m$ в терминах himult и lomult, предполагая, что $0 \le a,\ x < m < w$ и $m \perp w$. Можете использовать заранее вычисленные константы, которые зависят от a, m и w.

▸ **4.** [*21*] Исследуйте вычисление линейной конгруэнтной последовательности с $m = 2^{32}$ на машинах с двоичным дополнением, таких, как компьютеры серии System/370.

5. [*20*] Дано, что m меньше, чем длина слова, и что x и y — неотрицательные целые числа, меньшие, чем m. Покажите, что разность $(x - y) \bmod m$ может быть вычислена точно четырьмя операциями без операции деления на машине MIX. Какая программа будет наилучшей для вычисления суммы $(x + y) \bmod m$?

▸ **6.** [*20*] Предыдущее упражнение наводит на мысль, что вычитание по модулю m — более простая операция, чем суммирование по модулю m. Обсудите последовательность, генерируемую по правилу

$$X_{n+1} = (aX_n - c) \bmod m.$$

Будет ли эта последовательность существенно отличаться от линейной конгруэнтной последовательности, определенной ранее? Будет ли она более эффективной при вычислениях?

7. [*M24*] Какие особенности можно заметить в табл. 1?

▸ **8.** [*20*] Напишите программу для вычисления $(aX) \bmod (w-1)$ на компьютере MIX, аналогичную программе (2). Значения 0 и $w-1$ на входе и выходе вашей программы считаются эквивалентными.

▸ **9.** [*M25*] В большинстве языков программирования высокого уровня не предусмотрен хороший способ деления целого числа из двух слов на целое число из одного слова. Не предусматривается это и операцией himult из упр. 3. Цель этого упражнения — найти приемлемый способ преодоления таких ограничений, когда необходимо вычислить $ax \bmod m$ для переменной x и константы $0 < a < m$.

a) Докажите, что если $q = \lfloor m/a \rfloor$, то $a(x - (x \bmod q)) = \lfloor x/q \rfloor (m - (m \bmod a))$.
b) С помощью равенства (a) вычислите $ax \bmod m$, не оперируя числами, которые превосходят m по абсолютной величине, и предполагая, что $a^2 \le m$.

10. [*M26*] Решение упр. 9, (b) иногда применимо, когда $a^2 > m$. Определите точное число множителей a, для которых промежуточные результаты этого метода никогда не превосходят m для всех x между 0 и m?

11. [*M30*] Продолжая упр. 9, покажите, что можно оценить $ax \bmod m$, используя только следующие основные операции:

i) $u \times v$, где $u \ge 0$, $v \ge 0$, и $uv < m$;
ii) $\lfloor u/v \rfloor$, где $0 < v \le u < m$;
iii) $(u - v) \bmod m$, где $0 \le u, v < m$.

Действительно, это всегда возможно, если использовать максимум 12 операций типа (i) и (ii) и ограниченное число операций типа (iii), не считая предварительного вычисления констант, которые зависят от a и m. Например, объясните, как можно выполнить вычисления, когда a равно 62089911 и m равно $2^{31} - 1$. (Эти константы взяты из табл. 3.3.4–1.)

▸ **12.** [*M28*] Рассмотрите вычисления карандашом на бумаге или на счетах.

a) Найдите хороший метод умножения заданного десятичного числа на 10 по модулю 9999998999.
b) Сделайте то же самое, но умножив не на 10, а на 999999900 (по модулю 9999998999).

c) Объясните, как вычислить степень $999999900^n \bmod 9999998999$ для $n = 1,2,3,\ldots$.

d) Выполните такие же вычисления с десятичным разложением числа $1/9999998999$.

e) Покажите, что эти идеи предоставляют возможность реализовать определенные виды линейных конгруэнтных генераторов, имеющих очень большие модули, с помощью лишь несколько операций с генерируемым числом.

13. [*M24*] Повторите предыдущее упражнение, но с модулем 9999999001 и множителями 10 и 8999999101.

14. [*M25*] Обобщите идеи предыдущих двух упражнений для того, чтобы получить большое семейство линейных конгруэнтных генераторов с особенно большими модулями.

3.2.1.2. Выбор множителя. В этом разделе будут рассмотрены методы выбора множителя a для создания *периода максимальной длины*. Длинный период необходим для любой последовательности, используемой в качестве источника случайных чисел. Безусловно, мы ожидаем, что в периоде содержится значительно больше чисел, чем требуется для одноразового использования. Поэтому здесь внимание будет сосредоточено на длине периода. Читателю следовало бы помнить, однако, что длина периода — это только одно из требований к линейным конгруэнтным последовательностям, которые мы хотим использовать, как случайные последовательности. Например, когда $a = c = 1$, последовательность принимает простой вид: $X_{n+1} = (X_n + 1) \bmod m$. Она, очевидно, имеет период длиной m, но несмотря на это в ней нет ничего случайного. Другие соображения, влияющие на выбор множителя, будут приведены ниже в этой главе. Так как возможны только m различных значений, длина периода, несомненно, не может быть больше m. Можно ли достичь максимальной длины периода — m? Пример, приведенный выше, показывает, что это всегда возможно, хотя выбор $a = c = 1$ не обеспечивает желаемых свойств последовательности. Исследуем *все* возможные значения a, c и X_0, которые дают период длиной m. Оказывается, что такие значения параметров могут быть охарактеризованы очень просто; когда m является произведением различных простых чисел, только значение $a = 1$ обеспечивает полный период, но когда m делится на простое число в большой степени, существует значительная свобода в выборе a. Следующая теорема позволяет легко определить, возможно ли достижение периода максимальной длины.

Теорема A. *Линейная конгруэнтная последовательность, определенная числами m, a, c и X_0, имеет период длиной m тогда и только тогда, когда:*

i) *числа c и m взаимно простые;*

ii) *$b = a - 1$ кратно p для каждого простого p, являющегося делителем m;*

iii) *b кратно 4, если m кратно 4.*

Идеи, используемые при доказательстве этой теоремы, впервые возникли по крайней мере сто лет назад. Но первое ее доказательство в этой особой форме было предложено М. Гринбергером (M. Greenberger) для частного случая при $m = 2^e$ [см. *JACM* **8** (1961), 383–389]. Достаточность условий (i)–(iii) в общем случае была доказана Халлом (Hull) и Добеллом (Dobell) [см. *SIAM Review* **4** (1962),

230–254]. Чтобы доказать теорему, мы сначала рассмотрим некоторые вспомогательные теоретико-числовые результаты, представляющие и самостоятельный интерес.

Лемма P. *Пусть p — простое число, а e — положительное целое число, такое, что $p^e > 2$. Если*

$$x \equiv 1 \text{ (по модулю } p^e), \qquad x \not\equiv 1 \text{ (по модулю } p^{e+1}), \tag{1}$$

то

$$x^p \equiv 1 \text{ (по модулю } p^{e+1}), \qquad x^p \not\equiv 1 \text{ (по модулю } p^{e+2}). \tag{2}$$

Доказательство. Если x не кратно p, то оно может быть представлено в виде $x = 1 + qp^e$ для некоторого целого q. По биномиальной формуле получаем

$$\begin{aligned} x^p &= 1 + \binom{p}{1} qp^e + \cdots + \binom{p}{p-1} q^{p-1} p^{(p-1)e} + q^p p^{pe} \\ &= 1 + qp^{e+1} \left(1 + \frac{1}{p}\binom{p}{2} qp^e + \frac{1}{p}\binom{p}{3} q^2 p^{2e} + \cdots + \frac{1}{p}\binom{p}{p} q^{p-1} p^{(p-1)e} \right). \end{aligned}$$

Величины в скобках являются целыми числами, и к тому же каждый член внутри скобок, за исключением первого, кратен p. Для таких k, что $1 < k < p$, биномиальные коэффициенты $\binom{p}{k}$ делятся на p (см. упр. 1.2.6–10); следовательно,

$$\frac{1}{p}\binom{p}{k} q^{k-1} p^{(k-1)e}$$

делится на $p^{(k-1)e}$. Последний член $q^{p-1}p^{(p-1)e-1}$, также делится на p, поскольку $(p-1)e > 1$, когда $p^e > 2$. Итак, $x^p \equiv 1 + qp^{e+1}$ (по модулю p^{e+2}), что и завершает доказательство. (*Замечание.* Обобщение этого результата приведено в упр. 3.2.2–11, (a).) ∎

Лемма Q. *Пусть число m допускает разложение на простые множители в виде*

$$m = p_1^{e_1} \dots p_t^{e_t}. \tag{3}$$

Длина периода λ линейной конгруэнтной последовательности, определенной параметрами (X_0, a, c, m), является наименьшим общим кратным длин λ_j периодов линейных конгруэнтных последовательностей $(X_0 \bmod p_j^{e_j}, a \bmod p_j^{e_j}, c \bmod p_j^{e_j}, p_j^{e_j})$, $1 \le j \le t$.

Доказательство. Если использовать индукцию по t, то достаточно доказать, что если m_1 и m_2 — взаимно простые числа, то длина λ линейной конгруэнтной последовательности, определенной параметрами $(X_0, a, c, m_1 m_2)$, является наименьшим общим кратным длин λ_1 и λ_2 периодов последовательностей, определенных параметрами $(X_0 \bmod m_1, a \bmod m_1, c \bmod m_1, m_1)$ и $(X_0 \bmod m_2, a \bmod m_2, c \bmod m_2, m_2)$. В предыдущем разделе мы заметили (см. (4)), что если элементы этих трех последовательностей соответственно обозначены через X_n, Y_n и Z_n, то справедливо равенство

$$Y_n = X_n \bmod m_1 \qquad \text{и} \qquad Z_n = X_n \bmod m_2 \qquad \text{для всех } n \ge 0.$$

Поэтому по закону D из раздела 1.2.4 находим, что

$$X_n = X_k \qquad \text{тогда и только тогда, когда} \qquad Y_n = Y_k \quad \text{и} \quad Z_n = Z_k. \tag{4}$$

Пусть λ' — наименьшее общее кратное λ_1 и λ_2. Необходимо доказать, что $\lambda' = \lambda$. Так как $X_n = X_{n+\lambda}$ для всех достаточно больших n, $Y_n = Y_{n+\lambda}$ (следовательно, λ кратно λ_1) и $Z_n = Z_{n+\lambda}$ (следовательно, λ кратно λ_2). Таким образом, получим, что $\lambda \geq \lambda'$. Более того, известно, что $Y_n = Y_{n+\lambda'}$ и $Z_n = Z_{n+\lambda'}$ для всех достаточно больших n; поэтому из (4) следует, что $X_n = X_{n+\lambda'}$. Это доказывает, что $\lambda \leq \lambda'$. ∎

Сейчас мы готовы доказать теорему A. Из леммы Q следует, что теорему достаточно доказать для случая, когда m является степенью простого числа, поскольку

$$p_1^{e_1} \dots p_t^{e_t} = \lambda = \operatorname{lcm}(\lambda_1, \dots, \lambda_t) \leq \lambda_1 \dots \lambda_t \leq p_1^{e_1} \dots p_t^{e_t}$$

(lcm — наименьшее общее кратное. — *Примеч. пер.*) выполняется тогда и только тогда, когда $\lambda_j = p_j^{e_j}$ для $1 \leq j \leq t$.

Предположим поэтому, что $m = p^e$, где p — простое число, а e — целое положительное число. Поскольку утверждение теоремы очевидно при $a = 1$, можно считать, что $a > 1$. Период может иметь длину m тогда и только тогда, когда каждое целое число x, такое, что $0 \leq x < m$, встречается в этом периоде. Действительно, никакое значение x в периоде не может встретиться более одного раза. Таким образом, период имеет длину m тогда и только тогда, когда период последовательности с начальным значением $X_0 = 0$ имеет период длиной m. Поэтому достаточно доказать теорему, когда $X_0 = 0$. Из формулы 3.2.1–(6) следует, что

$$X_n = \left(\frac{a^n - 1}{a - 1}\right) c \bmod m. \tag{5}$$

Если c и m — не взаимно простые числа, то значение X_n никогда не может быть равно 1. Следовательно, условие (i) теоремы необходимо. Период имеет длину m тогда и только тогда, когда наименьшее положительное значение n, для которого $X_n = X_0 = 0$, равняется $n = m$. Из (5) и условия (i) следует, что доказательство нашей теоремы сводится к доказательству следующего утверждения.

Лемма R. *Предположим, что* $1 < a < p^e$, *где* p — *простое число. Если* λ — *наименьшее целое положительное число, для которого* $(a^\lambda - 1)/(a - 1) \equiv 0$ (по модулю p^e), *то*

$$\lambda = p^e \quad \textit{тогда и только тогда, когда} \quad \begin{cases} a \equiv 1 \text{ (по модулю } p), & \textit{когда} \quad p > 2, \\ a \equiv 1 \text{ (по модулю } 4), & \textit{когда} \quad p = 2. \end{cases}$$

Доказательство. Предположим, что $\lambda = p^e$. Если $a \not\equiv 1$ (по модулю p), то $(a^n - 1)/(a - 1) \equiv 0$ (по модулю p^e) тогда и только тогда, когда $a^n - 1 \equiv 0$ (по модулю p^e). Значит, условие $a^{p^e} - 1 \equiv 0$ (по модулю p^e) влечет равенство $a^{p^e} \equiv 1$ (по модулю p), но из теоремы 1.2.4F следует, что $a^{p^e} \equiv a$ (по модулю p). Таким образом, предположение, что $a \not\equiv 1$ (по модулю p) приводит к противоречию. Если $p = 2$ и $a \equiv 3$ (по модулю 4), то из упр. 8 следует

$$(a^{2^{e-1}} - 1)/(a - 1) \equiv 0 \text{ (по модулю } 2^e).$$

Эти рассуждения показывают, что в большинстве случаев необходимо, чтобы $a = 1 + qp^f$, где $p^f > 2$ и q не кратны p, всякий раз, когда $\lambda = p^e$.

Остается показать, что это условие *достаточно* для того, чтобы $\lambda = p^e$. Применив лемму P, находим, что для всех $g \geq 0$

$$a^{p^g} \equiv 1 \pmod{p^{f+g}}, \qquad a^{p^g} \not\equiv 1 \pmod{p^{f+g+1}};$$

следовательно,

$$\begin{aligned} (a^{p^g} - 1)/(a-1) &\equiv 0 \pmod{p^g}, \\ (a^{p^g} - 1)/(a-1) &\not\equiv 0 \pmod{p^{g+1}}. \end{aligned} \tag{6}$$

В частности, $(a^{p^e} - 1)/(a-1) \equiv 0 \pmod{p^e}$. Сейчас для конгруэнтной последовательности, определяемой параметрами $(0, a, 1, p^e)$ X_n, справедливо $X_n = (a^n - 1)/(a-1) \bmod p^e$. Значит, ее период равен λ, т. е. $X_n = 0$ тогда и только тогда, когда n кратно λ. Следовательно, p^e кратно λ. Это может случиться, только если $\lambda = p^g$ для некоторых g и соотношения (6) означают, что $\lambda = p^e$. ▮

Итак, теорема A доказана. ▮

В завершение этого раздела рассмотрим специальный случай использования исключительно мультипликативных генераторов, когда $c = 0$. Несмотря на то что процесс генерирования случайных чисел является немного более быстрым в случае, теорема A показывает, что максимальный период не может быть достигнут. Действительно, это совершенно очевидно, так как последовательность удовлетворяет соотношению

$$X_{n+1} = aX_n \bmod m, \tag{7}$$

и значение $X_n = 0$ может появиться, только если последовательность вырождается в нуль. Вообще, если d — любой делитель m и если X_n кратно d, все последующие элементы мультипликативной последовательности X_{n+1}, X_{n+2}, ... будут кратны d. Так что, когда $c = 0$, необходимо, чтобы X_n и m были взаимно простыми числами для всех n, что и ограничивает длину периода максимум до $\varphi(m)$ — числа целых взаимно простых чисел с m, лежащих между 0 и m.

Приемлемой длины периода можно достичь, даже если оговорить, что $c = 0$. Давайте сейчас попытаемся найти такие условия, которым удовлетворяет множитель, чтобы в этом специальном случае период стал настолько длинным, насколько это возможно.

Согласно лемме Q период последовательности зависит исключительно от периодов последовательностей при $m = p^e$. Рассмотрим эту ситуацию. Итак, $X_n = a^n X_0 \bmod p^e$ и ясно, что период будет иметь длину 1 (здесь можно только сказать, что длина периода не больше, чем e. — *Примеч. ред.*), если a кратно p. Поэтому будем считать, что a и p взаимно простые. Тогда период будет наименьшим целым числом λ, таким, что $X_0 = a^\lambda X_0 \bmod p^e$. Если наибольшим общим делителем X_0 и p^e является p^f, то это условие эквивалентно условию

$$a^\lambda \equiv 1 \pmod{p^{e-f}}. \tag{8}$$

По теореме Эйлера (упр. 1.2.4–28) $a^{\varphi(p^{e-f})} \equiv 1 \pmod{p^{e-f}}$; следовательно, λ является делителем

$$\varphi(p^{e-f}) = p^{e-f-1}(p-1).$$

Когда a и m — взаимно простые числа, наименьшее число λ, для которого $a^\lambda \equiv 1$ (по модулю m), принято называть *порядком a по модулю m*. Любое такое значение a, которое имеет *максимальный* возможный порядок по модулю m, называют *первообразным элементом* по модулю m.

Обозначим через $\lambda(m)$ порядок первообразного элемента, а именно — максимальный возможный порядок по модулю m. Из замечаний следует, что $\lambda(p^e)$ является делителем $p^{e-1}(p-1)$; достаточно легко (см. упр. 11–16, приведенные ниже) можно определить значения $\lambda(m)$ во всех следующих случаях:

$$\begin{gathered}\lambda(2)=1, \qquad \lambda(4)=2, \qquad \lambda(2^e)=2^{e-2}, \quad \text{если} \quad e\ge 3;\\ \lambda(p^e)=p^{e-1}(p-1), \quad \text{если} \quad p>2;\\ \lambda(p_1^{e_1}\dots p_t^{e_t})=\operatorname{lcm}\bigl(\lambda(p_1^{e_1}),\dots,\lambda(p_t^{e_t})\bigr).\end{gathered} \tag{9}$$

Все сказанное можно подытожить в следующей теореме.

Теорема B. [C. F. Gauss, *Disquisitiones Arithmeticæ* (1801), §90–92.] *Максимальным периодом, возможным, когда $c=0$, является $\lambda(m)$, где $\lambda(m)$ определено в* (9). *Этот период достигается, если*

i) *X_0 и m — взаимно простые числа;*

ii) *a является первообразным элементом по модулю m.* ▮

Заметим, что можно получить период длиной $m-1$, если m — простое число; т. е. это всего на единицу меньше, чем максимальная длина периода. Так что для практических целей такой период может быть настолько длинным, насколько это необходимо.

Теперь возникает вопрос, как найти первообразные элементы по модулю m? В упражнениях, данных в конце раздела, приводится совершенно очевидный ответ на вопрос, когда m является простым числом или степенью простого числа. Результаты сформулированы в следующей теореме.

Теорема C. *Число a является первообразным элементом по модулю p^e тогда и только тогда, когда выполняется одно из следующих условий:*

i) *$p=2$, $e=1$ и a — нечетное число;*

ii) *$p=2$, $e=2$ и $a \bmod 4 = 3$;*

iii) *$p=2$, $e=3$ и $a \bmod 8 = 3$, 5 или 7;*

iv) *$p=2$, $e\ge 4$ и $a \bmod 8 = 3$ или 5;*

v) *p — нечетное число, $e=1$, $a\not\equiv 0$ (по модулю p) и $a^{(p-1)/q}\not\equiv 1$ (по модулю p) для любого простого делителя q числа $p-1$;*

vi) *p — нечетное число, $e>1$, a удовлетворяют условию* (v) *и $a^{p-1}\not\equiv 1$ (по модулю p^2).* ▮

Условия (v) и (vi) теоремы легко проверяются на компьютере для больших p. Эффективные методы оценки степени, когда известны множители числа $p-1$, обсуждаются в разделе 4.6.3.

Теорема C применима только к степеням простых чисел. Но если заданы значения a_j, являющиеся первообразными элементами по модулю $p_j^{e_j}$, то можно найти

единственное значение a, такое, что $a \equiv a_j$ (по модулю $p_j^{e_j}$) при $1 \le j \le t$, используя китайский алгоритм (алгоритм, построенный на основании китайской теоремы об остатках. — *Примеч. пер.*), рассматриваемый в разделе 4.3.2. Число a будет первообразным элементом по модулю $p_1^{e_1} \ldots p_t^{e_t}$. Таким образом, существует приемлемый эффективный путь построения множителей, удовлетворяющих условию теоремы B, для любых модулей m умеренной размерности, хотя вычисления в общем случае могут быть весьма длинными.

В распространенном случае, когда $m = 2^e$, где $e \ge 4$, изложенные выше условия приводят к единственному требованию: $a \equiv 3$ или 5 (по модулю 8). В этой ситуации четвертая часть всех возможных множителей даст длину периода, равную $m/4$, а $m/4$ будет максимальной длиной периода, когда $c = 0$.

Существует второй, еще более распространенный случай, когда $m = 10^e$. Используя леммы P и Q, нетрудно получить необходимые и достаточные условия достижения максимального периода для десятичного компьютера (см. упр. 18).

Теорема D. *Если* $m = 10^e$, $e \ge 5$, $c = 0$ *и* X_0 *не кратно* 2 *или* 5, *то период линейной конгруэнтной последовательности равен* $5 \times 10^{e-2}$ *тогда и только тогда, когда* $a \bmod 200$ *равно одному из следующих* 32 *чисел:*

$$\begin{array}{l} 3,\ 11,\ 13,\ 19,\ 21,\ 27,\ 29,\ 37,\ 53,\ 59,\ 61,\ 67,\ 69,\ 77,\ 83,\ 91,\ 109,\ 117,\\ 123,\ 131,\ 133,\ 139,\ 141,\ 147,\ 163,\ 171,\ 173,\ 179,\ 181,\ 187,\ 189,\ 197. \end{array} \qquad (10)$$

УПРАЖНЕНИЯ

1. [*10*] Чему равна длина периода линейной конгруэнтной последовательности с параметрами $X_0 = 5772156648$, $a = 3141592621$, $c = 2718281829$ и $m = 10000000000$?

2. [*10*] Будут ли следующие два условия гарантировать максимальную длину периода, когда m является степенью 2? (i) c — нечетное число; (ii) $a \bmod 4 = 1$.

3. [*13*] Предположим, что $m = 10^e$, где $e \ge 2$, и пусть c — нечетное число, не кратное 5. Покажите, что линейная конгруэнтная последовательность будет иметь период максимальной длины тогда и только тогда, когда $a \bmod 20 = 1$.

4. [*M20*] Предположим, что $m = 2^e$ и $X_0 = 0$. Если числа a и c удовлетворяют условиям теоремы A, чему равно $X_{2^{e-1}}$?

5. [*14*] Найдите все множители a, удовлетворяющие условиям теоремы A, когда $m = 2^{35} + 1$. (Простые множители m можно найти в табл. 3.2.1.1–1.)

▸ **6.** [*20*] Найдите все множители a, удовлетворяющие условиям теоремы A, когда $m = 10^6 - 1$ (см. табл. 3.2.1.1–1.)

▸ **7.** [*M23*] Период конгруэнтной последовательности не должен начинаться с X_0, но всегда можно найти индексы $\mu \ge 0$ и $\lambda > 0$, такие, что $X_{n+\lambda} = X_n$ при всех $n \ge \mu$, и для которых μ и λ являются наименьшими возможными значениями с этими свойствами (см. упр. 3.1–6 и 3.2.1–1). Если μ_j и λ_j — индексы, соответствующие последовательностям

$$(X_0 \bmod p_j^{e_j},\ a \bmod p_j^{e_j},\ c \bmod p_j^{e_j},\ p_j^{e_j}),$$

и если μ и λ соответствуют составной последовательности $(X_0, a, c, p_1^{e_1} \ldots p_t^{e_t})$, то согласно формулировке леммы Q λ является наименьшим общим кратным $\lambda_1, \ldots, \lambda_t$. Чему равно значение μ в терминах значений $\mu_1, \ldots, \mu_t$? Чему равно максимально возможное значение μ, получаемое путем варьирования X_0, a и c, когда $m = p_1^{e_1} \ldots p_t^{e_t}$ фиксировано?

8. [*M20*] Покажите, что если $a \bmod 4 = 3$, то $(a^{2^{e-1}} - 1)/(a-1) \equiv 0$ (по модулю 2^e), когда $e > 1$. (Воспользуйтесь леммой P.)

▸ **9.** [*M22*] (В. Э. Томсон (W. E. Thomson).) Когда $c = 0$ и $m = 2^e \geq 16$, теоремы B и C утверждают, что период имеет длину 2^{e-2} тогда и только тогда, когда множитель a удовлетворяет равенству $a \bmod 8 = 3$ или $a \bmod 8 = 5$. Покажите, что каждая такая последовательность, по существу, является линейной конгруэнтной последовательностью с $m = 2^{e-2}$, имеющей *полный* период, в следующем смысле:

a) если $X_{n+1} = (4c+1)X_n \bmod 2^e$ и $X_n = 4Y_n + 1$, то

$$Y_{n+1} = \big((4c+1)Y_n + c\big) \bmod 2^{e-2};$$

b) если $X_{n+1} = (4c-1)X_n \bmod 2^e$ и $X_n = \big((-1)^n(4Y_n+1)\big) \bmod 2^e$, то

$$Y_{n+1} = \big((1-4c)Y_n - c\big) \bmod 2^{e-2}.$$

[*Замечание.* В этих формулах c есть нечетное целое число. В литературе содержится достаточно утверждений о том, что последовательности с $c = 0$, удовлетворяющие теореме B, так или иначе являются более случайными, чем последовательности, удовлетворяющие условиям теоремы A, вопреки тому факту, что период — это только четверть длины периода, получаемого в условиях теоремы B. В данном упражнении такие утверждения опровергаются; в сущности, следует удалить два разряда длины слова, чтобы сохранить возможность прибавить c, когда m является степенью 2.]

10. [*M21*] Для каких значений m справедливо $\lambda(m) = \varphi(m)$?

▸ **11.** [*M28*] Пусть x — нечетное целое число, большее, чем 1. (a) Покажите, что существует единственное целое число $f > 1$, такое, что $x \equiv 2^f \pm 1$ (по модулю 2^{f+1}). (b) Дано, что $1 < x < 2^e - 1$ и что f является соответствующим целым числом п. (a). Покажите, что порядок x по модулю 2^e равен 2^{e-f}. (c) В частности, это доказывает утверждения (i)–(iv) теоремы C.

12. [*M26*] Пусть p — простое нечетное число. Если $e > 1$, докажите, что a является первообразным элементом по модулю p^e тогда и только тогда, когда a — первообразный элемент по модулю p и $a^{p-1} \not\equiv 1$ (по модулю p^2). (Предположите, что $\lambda(p^e) = p^{e-1}(p-1)$. Этот факт доказан в упр. 14 и 16 ниже.)

13. [*M22*] Пусть p — простое число. Задано, что a не является первообразным элементом по модулю p. Покажите, что каждое a кратно p или $a^{(p-1)/q} \equiv 1$ (по модулю p) для некоторых простых чисел q, делящих $p-1$.

14. [*M18*] Предположим, что $e > 1$, p — нечетное простое число и a — первообразный элемент по модулю p. Докажите, что либо a, либо $a+p$ является первообразным элементом по модулю p^e. [*Указание.* См. упр. 12.]

15. [*M29*] (a) Пусть a_1, a_2 взаимно просты с m и пусть их порядки по модулю m равны соответственно λ_1 и λ_2. Предположим, что λ является наименьшим общим кратным λ_1 и λ_2. Докажите, что $a_1^{\kappa_1}a_2^{\kappa_2}$ имеют порядок λ по модулю m для соответствующих целых чисел κ_1 и κ_2. [*Указание.* Рассмотрите сначала случай, когда λ_1 и λ_2 — взаимно простые числа.] (b) Пусть $\lambda(m)$ — максимальный порядок любого элемента по модулю m. Докажите, что $\lambda(m)$ кратно порядку каждого элемента по модулю m, т. е. что $a^{\lambda(m)} \equiv 1$ (по модулю m) всегда, когда a и m — взаимно простые числа. (Не используйте теорему B.)

▸ **16.** [*M24*] (*Существование первообразных корней.*) Пусть p — простое число.

a) Рассмотрим полином $f(x) = x^n + c_1x^{n-1} + \cdots + c_n$, где c_i — целые числа. Дано, что a — целое число, для которого $f(a) \equiv 0$ (по модулю p). Покажите, что существует полином

$$q(x) = x^{n-1} + q_1x^{n-2} + \cdots + q_{n-1}$$

с целыми коэффициентами, такой, что $f(x) \equiv (x-a)q(x)$ (по модулю p) для всех целых x.

b) Пусть $f(x)$ — такой же полином, как в (a). Покажите, что $f(x)$ имеет не более n различных "корней" по модулю p, т. е. существует не более n целых чисел a, $0 \le a < p$, таких, что $f(a) \equiv 0$ (по модулю p).

c) Так же, как и в упр. 15, (b), полином $f(x) = x^{\lambda(p)} - 1$ имеет $p - 1$ различных корней; следовательно, существует целое число a с порядком $p - 1$.

17. [*M26*] Не все значения, перечисленные в теореме D, можно получить, используя построения, приведенные в разделе, например 11 — не первообразный элемент по модулю 5^e. Как это возможно, если 11 *является* первообразным элементом по модулю 10^e согласно теореме D? Какие из чисел, перечисленных в теореме D, являются *одновременно* первообразными элементами по модулям 2^e и 5^e?

18. [*M25*] Докажите теорему D (см. предыдущее упражнение).

19. [*40*] Составьте таблицу нескольких подходящих множителей a для каждого из значений m, внесенных в табл. 3.2.1.1–1, предполагая, что $c = 0$.

▸ **20.** [*M24*] (Дж. Марсалья (G. Marsaglia).) Назначение упражнения состоит в изучении длины периода *произвольной* линейной конгруэнтной последовательности. Пусть $Y_n = 1 + a + \cdots + a^{n-1}$, так что $X_n = (AY_n + X_0) \bmod m$ для некоторой константы A из условия 3.2.1–(8).

a) Докажите, что длина периода $\langle X_n \rangle$ равна длине периода $\langle Y_n \bmod m' \rangle$, когда $m' = m/\gcd(A, m)$.

b) Докажите, что длина периода $\langle Y_n \bmod p^e \rangle$ удовлетворяет следующим требованиям, когда p — простое число. (i) Если $a \bmod p = 0$, длина периода равна 1. (ii) Если $a \bmod p = 1$, она равна p^e, за исключением случаев, когда $p = 2$, $e \ge 2$ и $a \bmod 4 = 3$. (iii) Если $p = 2$, $e \ge 2$ и $a \bmod 4 = 3$, она равна удвоенному порядку a по модулю p^e (см. упр. 11), за исключением случая, когда $a \equiv -1$ (по модулю 2^e), при котором она равна 2. (iv) Если $a \bmod p > 1$, длина периода равна порядку a по модулю p^e.

21. [*M25*] Пусть в линейной конгруэнтной последовательности с максимальным периодом $X_0 = 0$ s — наименьшее положительное целое число, такое, что $a^s \equiv 1$ (по модулю m). Докажите, что $\gcd(X_s, m) = s$ (gcd — наибольший общий делитель. — *Примеч. пер.*).

▸ **22.** [*M25*] Обсудите проблему нахождения модулей $m = b^k \pm b^l \pm 1$ таким образом, чтобы генераторы, использующие вычитание с заимствованием и суммирование с переносом (см. упр. 3.2.1.1–14), имели очень длинные периоды.

3.2.1.3. Потенциал. В предыдущем разделе было показано, что максимальный период может быть достигнут, когда $b = a - 1$ кратно каждому простому делителю m, и b должно быть также кратно 4, если m кратно 4. Если z — основание системы счисления машины ($z = 2$ для бинарного компьютера и $z = 10$ для десятичного компьютера), m — длина слова в компьютере z^e, множитель

$$a = z^k + 1, \qquad 2 \le k < e, \tag{1}$$

удовлетворяет этим условиям. По теореме 3.2.1.2A можно брать $c = 1$. Рекуррентное соотношение теперь имеет вид

$$X_{n+1} = \big((z^k + 1)X_n + 1\big) \bmod z^e, \tag{2}$$

и это уравнение означает, что можно избежать умножения; просто достаточно перемещения и суммирования.

Например, пусть $a = B^2 + 1$, где B — размер байта компьютера MIX. Программа

```
LDA X;   SLA 2;   ADD X;   INCA 1
```  (3)

может использоваться вместо программы, приведенной в разделе 3.2.1.1, и время выполнения программы уменьшается от $16u$ до $7u$.

По этой причине множители, имеющие вид (1), широко обсуждались в литературе. Они действительно рекомендованы многими авторами. Однако первые несколько лет экспериментирования с этим методом убедительно показали, что *множителей, имеющих простой вид* (1), *следует избегать*. Сгенерированные числа просто недостаточно случайны.

Ниже в этой главе будет рассмотрена одна довольно сложная теория, связанная с недостатками всех известных плохих линейных конгруэнтных генераторов случайных чисел. Однако некоторые генераторы (такие, как (2)) настолько ужасны, что достаточно сравнительно простой теории, чтобы исключить их из рассмотрения. Эта простая теория связана с понятием "потенциал", которое мы сейчас обсудим.

Потенциал линейной конгруэнтной последовательности с максимальным периодом определяется как наименьшее целое число s, такое, что

$$b^s \equiv 0 \pmod{m}. \tag{4}$$

(Целое число s всегда существует, когда множитель удовлетворяет условиям теоремы 3.2.1.2A, так как b кратно каждому простому делителю m.)

Можно анализировать случайность последовательности, положив $X_0 = 0$, так как 0 встречается в периоде. При этом предположении соотношение 3.2.1–(6) сводится к

$$X_n = \big((a^n - 1)c/b\big) \bmod m;$$

и, если разложить выражение $a^n - 1 = (b+1)^n - 1$ по биномиальной формуле, получится

$$X_n = c\left(n + \binom{n}{2} b + \cdots + \binom{n}{s} b^{s-1}\right) \bmod m. \tag{5}$$

Все члены разложения b^s, b^{s+1} и т. д. можно исключить, так как они кратны m.

Уравнение (5) столь поучительно, что необходимо рассмотреть некоторые специальные случаи. Если $a = 1$, потенциал равен 1 и $X_n \equiv cn \pmod{m}$, как мы уже видели, так что последовательность наверняка не случайна. Если потенциал равен 2, то $X_n \equiv cn + cb\binom{n}{2}$, и снова последовательность не совсем случайна. Действительно, в этом случае

$$X_{n+1} - X_n \equiv c + cbn.$$

Таким образом, разность между последовательно генерируемыми числами выражена простой зависимостью от n. Точка (X_n, X_{n+1}, X_{n+2}) всегда лежит на одной из четырех плоскостей в трехмерном пространстве:

$$x - 2y + z = d + m, \qquad x - 2y + z = d - m,$$
$$x - 2y + z = d, \qquad x - 2y + z = d - 2m,$$

где $d = cb \bmod m$.

Если потенциал равен 3, то последовательность становится более или менее похожей на случайную, но все еще существует высокая степень зависимости между X_n, X_{n+1} и X_{n+2}. Тесты показывают, что последовательности с потенциалом 3 попрежнему недостаточно хороши. Сообщалось, что приемлемые результаты были получены в некоторых случаях при потенциале, равном 4, но это оспаривалось