ЗАДАЧИ ПО КУРСУ
“МАТЕМАТИКА И КОМПЬЮТЕР”
ВЫПУСК 1
АРИФМЕТИКА И ТЕОРИЯ ЧИСЕЛ

Николай Вавилов, Владимир Халин

Все знание находится повсюду.

Идрис Шах, *Знание как знать*

I believe that mathematical reality lies outside us, and that our function is to discover or observe it, and that the theorems which we prove, and which we describe grandiloquently as our 'creations' are simply our notes of our observations.

Godfrey Harold Hardy

Mathematics is an experimental science, and definitions do not come first, but later on.

Oliver Heavyside

In theory there is no difference between theory and practice. In practice there is.

Yogi Berra

Q: How many mathematicians does it take to screw in a light bulb?
A: None. It's left to the reader as an exercise.

But those who do not care about fanciful things have no reason to read about Ireland.

Lord Dunsany, *My Ireland*

“Dwarf-doors are not made to be seen when shut,” said Gimli. “They are invisible, and their own masters cannot find them or open them, if their secret is forgotten.”

J.R.R Tolkien, *The Lord of the Rings*

“Have you guessed the riddle yet?” the Hatter said, turning to Alice again.
“No, I give it up,” Alice replied. “What's the answer?”
“I haven't the slightest idea,” said the Hatter.

Lewis Carroll, *Alice's adventures in Wonderland*

Think, speak, cast, write, sing, number.

William Shakespeare, *Antony and Cleopatra*

По прихоти своей скитаться здесь и там,
Дивясь божественным природы красотам,
И пред созданьями искусств и вдохновенья
Трепеща радостно в восторгах умиленья.
— Вот счастье! вот права ...

Александр Пушкин, *Из Пиндемонти*

АРИФМЕТИКА И ТЕОРИЯ ЧИСЕЛ

The genealogical trees at the end of the Red Book of Westmarch are a small book in themselves, and all but Hobbits would find them exceedingly dull. Hobbits delighted in such things, if they were accurate: they liked to have books filled with things that they already knew, set out fair and square and with no contradictions.

J.R.R.Tolkien, *The Lord of the Rings*

Дальнейшие выпуски:

Выпуск 2. Комбинаторика и дискретная математика

Выпуск 3. Алгебра, линейная алгебра, основы анализа

After three days without programming, life becomes meaningless.
The Tao of Real Programming

Введение

Скажи мне, чертежник пустыни,
Сыпучих песков геометр,
Ужели безудержность линий
Сильнее, чем дующий ветр?
— меня не касается трепет
Его иудейских забот —
Он опыт из лепета лепит
И лепет из опыта пьет.

Осип Мандельштам

— Что ты сочиняешь, Бубантес? — спросил я тихо.
— Ничего, — сказал он, продолжая свое дело, — курс любви теоретической и практической.
— Практической?
— Да!.. Или опытной. Это все равно. Я вам изложил прежде теорию любви, а вот теперь начинаются опыты.

Осип Сенковский, *Из записок барона Брамбеуса*

Я, например, занялся изящной словесностью по одной простой причине — она сообщает тебе чрезвычайное ускорение. Когда сочиняешь стишок, тебе в голову приходят такие вещи, которые в принципе приходить не должны были.

Иосиф Бродский

Для начала нам потребуется голова. Ее основная задача — думать. Самый простой способ заставить голову думать — предложить ей задачу. Постановка задачи наполовину решает саму задачу. Остается взять да и сделать.
Гораздо сложнее придумать задачу самому. Но так как для дизайнера не должно быть ничего невозможного, придумывание задачи нужно просто сделать задачей. И проблема исчезнет.

Артемий Лебедев, *Ководство*, § 95

— Что вы хренью маетесь? — не выдержал Сенька. — Делать-то чего будем?
— Не “хренью маетесь”, а “занимаетесь ерундой”. Это раз. — Эраст Петрович склонил голову, любуясь своими каракулями. — Я не занимаюсь ерундой, а концентрирую мысль при помощи каллиграфии. Это два. Безупречно написанный иероглиф “справедливость” помог мне перейти от дедукции к п-проекции. Это три.

Борис Акунин, *Любовник смерти*

> Шмудилы, как правило, знакомы не только с назначением своих шмудаков, но и с азами их использования, в то время как нормальным человеком это постигается с трудом.
>
> Владимир Шинкарев, *Митьки*

> Альбом "Митьковские песни" предназначен для прослушивания матросами, старшинами, мичманами, офицерами и адмиралами Военно-морского флота на боевых кораблях, судах дальнего и каботажного плавания и в домашних условиях. Он вызывает у моряков чувство любви к Родине, верность присяге, ненависть к врагам мира, чувства товарищества, братства, достоинства и чести. Митьковские песни помогают увидеть красоту нашей жизни, нашей страны, побуждают моряка к образцовому несению службы, строгому соблюдению морских традиций и ритуалов, любовному отношению к своему кораблю, флагу и морской форме одежды.
>
> Дмитрий Шагин, *Митьковские песни*

> К своему путешествию я готовился заранее, и это неспроста. Если Вы не бывали у нас, то, скорее всего, даже не подозреваете о том, что весной в наших краях поезда ходят, как им заблагорассудится. Железнодорожная колея уже в апреле, а в теплый год — и в марте, начинает непредсказуемо вилять. То и дело она выходит из предписанных ей берегов. Почему? Быть может, она норовит сбежать из-под неусыпного надзора Министерства путей сообщения, чтобы вволю порезвиться на наших еще не просохших ингерманландских полях? Словно далекая желтая река, которая в дни вешнего паводка каждый раз заново созидает свое русло, врезаясь в мягкий, податливый лесс. Особенно для молодых горожан поездка за город по весне представляет собой целое искусство.
>
> Григорий Злотин, *Варшавский вокзал*

Предлагаемый задачник основан на курсе "Математика и компьютер", который мы вели несколько последних лет на экономическом факультете СПбГУ. Сюда включено большинство задач, обсуждавшихся на занятиях, а также задачи, предлагавшиеся в качестве домашних заданий, на зачетах и экзаменах.

Задачи, которые рассматривались в рамках этого курса, относились преимущественно к теории чисел (различные способы и форматы представления чисел, простые числа, основы мультипликативной теории чисел, аддитивные задачи), дискретной математике (множества, наборы, списки, функции, отношения), комбинаторике (перестановки, простейшие задачи перечисления, биномиальные коэффициенты, числа Стирлинга и т.д.) и алгебре (многочлены и рациональные дроби, системы алгебраических уравнений, представление матриц и действия над ними, численные инварианты и канонические формы матриц, простейшие алгебраические системы). В качестве обоснования в пользу такого выбора мы можем сказать следующее.

• В этих областях имеется много легко формулируемых задач, решение которых не требует (почти) никаких специфических знаний.

• Эти темы традиционно являются центральными для самой математики, а в последние десятилетия играют огромную и постоянно возрастающую роль в ее приложениях, связанных с компьютерами и информационными технологиями.

• В то же время все эти важнейшие темы практически не представлены в действующих программах по математике для нематематиков!!!

Кроме того, обсуждалось совсем небольшое количество задач, связанных с основами анализа (графики функций, пределы, суммы и произведения, производные и интегралы), элементарной геометрией и элементарной теорией вероятностей (конечные вероятности, генерация случайных объектов).

Во вводных параграфах мы сделаем несколько замечаний о субстрате, содержании и основных установках курса, и предполагаемом уровне подготовки.

§ 1. Соединить идеи с вычислениями

> Slick's Third Law of the Universe: There are two types of dirt: the dark kind, which is attracted to light objects, and the light kind, which is attracted to dark objects.

Последние три века развитие математики происходило под действием следующей антитезы:

• слогана Лейбница заменить идеи вычислениями,

• слогана Дирихле заменить вычисления идеями.

Однако, как известно, прогрессивное человечество без труда обалдевает даже простейшими идеями[1].

К сожалению, в конце XIX начале XX в преподавании математики под громкие крики о сохранении традиций произошел *полный* разрыв с традицией. Поэтому все преподавание математики извратило тезис Лейбница настолько успешно, что не только школьный курс математики, но и традиционные вузовские курсы **высшей математики**, **математического анализа**, **аналитической геометрии**, **линейной алгебры** превратились в бессвязные наборы бессмысленных и бессодержательных калькулятивных экзерсисов.

С другой стороны, XX веке многие математики извратили тезис Дирихле настолько успешно, что значительная часть математических исследований полностью утратила связь не только с вычислениями, не только с математическим естествознанием, но вообще с чем бы то ни было, включая саму математику, и полностью превратилась в артефакты для артефактов.

Понимая и *полностью* разделяя пафос обоих этих высказываний, мы все же считаем, что в истории математики все разумные люди, — включая, конечно самих Лейбница и Дирихле!! — всегда пытались соединить идеи с вычислениями.

[1] humankind lives under perverted ideas.

В нашем курсе мы хотели показать, что настоящая математика — как и настоящее программирование — основаны на игре и равновесии идей и вычислений. Мы хотели продемонстрировать, что *любую* идею можно превратить в вычисление и что правильно организованное вычисление может привести не только к результату, но и к пониманию. Мы хотели проиллюстрировать, как можно использовать вычисления для того, чтобы проверить — или опровергнуть! — математические утверждения. И, наоборот, как простые и естественные идеи можно использовать для того, чтобы резко упростить или даже вовсе элиминировать вычисления.

Как математики, мы полностью, безоговорочно, *без всяких резерваций* верим в могущество простых, но могущественных идей. Но мы верим также, что, в той мере, в которой математика является наукой, она, как и все остальные науки, основана на эксперименте. Прежде, чем доказывать результат, нужно убедиться в его справедливости. Закон распределения простых или теорема Дирихле о простых в арифметических прогрессиях являются *прежде всего* экспериментальными фактами. Доказательства сообщают этим фактам дополнительную глубину и объем, но мало меняют нашу уверенность в их истинности.

Так как основной задачей курса является именно *полное* согласование математической и алгоритмической точек зрения, то характер изложения и предлагаемых задач радикально отличаются от подавляющего большинства как математических, так и компьютерных курсов. Наше изложение в гораздо большей степени основано на эксперименте, чем принято в математике, и, с другой стороны, в гораздо большей степени основано на чистом умозрении, чем обычно в Computer Science.

Задачи, которые мы предлагали на занятиях, на зачетах и экзаменах, являются *слишком* простыми как для опытного математика (который может решить почти все из них вообще не пользуясь компьютером), так и для опытного вычислителя (который для большинства из них может написать программу, не зная почти ничего об их математической сути). Однако, как нам кажется, для студентов первого курса, еще не страдающих профессиональной гипертрофией соответствующих отделов мозга, решение некоторых из этих задач может потребовать напряжения воображения и должно способствовать улучшению мозгового кровообращения и выработке полезных навыков.

С другой стороны, в дальнейшем мы постараемся объяснить свою точку зрения, состоящую в том, что на начальном этапе изучения программирования борьба за эффективность `is not an issue`. Именно поэтому мы практически не обсуждали в нашем курсе серьезные профессиональные алгоритмы сортировки и поиска, где недостаточно просто полиномиальности, а происходит борьба за степень и мультипликативные константы. Как нам кажется, это отвечает принципиально другому уровню алгоритмического мышления и возможно *только* после того, как студент усвоил основы алгоритмики и научился непринужденно писать коды, выражающие важнейшие математические операции и конструкции.

§ 2. Субстрат

Мы имеем здесь дело с математикой, а не теологией. Пусть другие математики думают, что им доступно проникновение в мысли Бога об их любимом предмете; мне это всегда казалось пустым и бессмысленным занятием.

Андре Вейль[2]

Если что-нибудь может подорвать незаслуженный авторитет одних и упрочить справедливую славу других, так это только возможно большее распространение в массах теоретических сведений о музыке.

Петр Чайковский

То, что настоящий курс преподавался экономистам, накладывало заметные ограничения на характер и направленность изложения.

В частности, излагая арифметику мы ограничивались обсуждением исключительно следующих числовых систем, которые реализованы в качестве числовых доменов[3] *в ядре* системы `Mathematica`.

`Booleans`	{`True`,`False`}	значения истинности
`Integers`	$\mathbb{Z}$	целые числа
`Primes`	$\mathbb{P}$	простые числа
`Rationals`	$\mathbb{Q}$	рациональные числа
`Algebraics`	$\overline{\mathbb{Q}}$	алгебраические числа
`Reals`	$\mathbb{R}$	вещественные числа
`Complexes`	$\mathbb{C}$	комплексные числа

В отличие от `Axiom` и `MuPAD` в `Mathematica` кольца классов вычетов $\mathbb{Z}/m\mathbb{Z}$ не оформлены внутренним образом как домены. Тем не менее, модулярная арифметика легко реализуется при помощи функций `Mod` и `PowerMod`, а также установки опции `Modulus->m`, поддерживаемой большинством алгебраических команд. Дефолтная установка этой опции `Modulus->0` соответствует обычной арифметике целых чисел.

Если бы мы читали аналогичный курс математикам или физикам, то, несомненно, рассматривали бы в нем наравне с перечисленными выше традиционными числовыми системами также по крайней мере следующие конечные, неархимедовы, некоммутативные и неассоциативные числовые си-

[2] А.Вейль, Основы теории чисел. — М,, Мир, 1972, 408с.

[3] В документации к системе домен $\overline{\mathbb{Q}}$ алгебраических чисел обозначается через $\mathbb{A}$, но профессиональные математики чаще обозначают через $\mathbb{A}$ множество *целых* алгебраических чисел.

стемы.

`GaloisField[q]`	$\mathbb{F}_q$	конечное поле из q элементов
`PadicIntegers[p]`	$\mathbb{Z}_p$	целые p-адические числа
`Padics[p]`	$\mathbb{Q}_p$	p-адические числа
`Quaternions`	$\mathbb{H}$	кватернионы
`Octonions`	$\mathbb{O}$	октавы Кэли
`AlbertNumbers`	$\mathbb{J}$	числа Алберта

В действительности, почти все эти числовые структуры реализованы — именно под такими названиями — в стандартных пакетах, входящих в поставку системы. Эти новые типы чисел играют огромную — и все время растущую! — роль не только в самой математике, но и в ее приложениях, как во всех приложениях в Computer Science и передаче информации, так и в современных физических теориях.

• Конечные поля не только естественно возникают в теории чисел, алгебре, комбинаторике и геометрии, но и являются естественным контекстом для большинства результатов криптографии и кодирования.

• Что касается p-адических чисел, то, кроме их центральной роли в теории чисел, алгебре и анализе, они находят все более широкие приложения в математической физике. Кроме того, как оказалось, p-адические числа с конечным p гораздо лучше, чем вещественные, приспособлены для безошибочных вычислений. Вещественные числа $\mathbb{R} = \mathbb{Q}_\infty$ при этом возникают просто как p-адические числа в *бесконечном* простом $p = \infty$.

• Кватернионы и октавы отвечают за существование всех исключительных объектов в математике и, являются адекватным инструментом для описания геометрии физического мира.

К сожалению, школьные и университетские курсы математики *на тысячелетия* отстают от потребностей математики и ее приложений, так что рассмотреть эти системы в рамках отведенного нам времени и предполагавшейся математической подготовки студентов не было никакой возможности. Вот еще два серьезных упущения.

• Ничего не говорится о системе ${}^*\mathbb{R}$ гипервещественных чисел и других неархимедовых системах, в которых вещественные числа расширяются посредством добавления актуально бесконечно малых и/или актуально бесконечно больших. Введение актуально бесконечно малых позволяет дать значительно более внятную и вычислительно эффективную трактовку *всех* вопросов математического анализа.

• Ничего не говорится об арифметике числовых полей, даже простейших квадратичных полей, таких как поле гауссовых чисел $\mathbb{Q}(i)$, поле эйзенштейновых чисел $\mathbb{Q}(\omega)$, пифагорово поле $\mathbb{Q}(\sqrt{2})$ и поле золотого сечения $\mathbb{Q}(\sqrt{5})$. Единственной причиной этого снова была чисто физическая невозможность хотя бы просто упомянуть алгебраическую теорию чисел при ограничениях, накладываемых отведенным временем и подготовкой студентов.

Однако, любой, кто серьезно овладел нашим курсом и знает необходимую математику, сможет легко проводить вычисления в этих и любых других числовых системах.

§ 3. Стиль программирования

> Ведь я, например, нисколько не удивлюсь, если ни с того ни с сего среди всеобщего будущего благоразумия возникнет какой-нибудь джентльмен с неблагородной или, лучше сказать, с ретроградной и насмешливою физиономией, упрет руки в боки и скажет нам всем: а что, господа, не столкнуть ли нам все это благоразумие с одного разу, ногой, прахом, единственно с той целью, чтоб все эти логарифмы отправились к черту и чтоб нам опять по своей глупой воле пожить! Это бы еще ничего, но обидно то, что ведь непременно последователей найдет.
>
> Федор Достоевский, *Записки из подполья*

> Efficiency is only an issue if the code fails to produce an answer. As long as the memory requirements of the code does not exceed the memory of the computer used and as long as the time required does not exceed the user's patience, the code is efficient enough.
>
> Joseph Sloan

Мы считаем, что требование в элементарных курсах программирования пользоваться исключительно “эффективными” алгоритмами является столь же лицемерным и абсурдным, как требование сопровождать все сообщаемые в элементарных математических курсах результаты “полными и подробными” доказательствами.

• Понятие доказательства не является абсолютным, а носит прагматический и психологический характер. Доказательство — это такое рассуждение, которое убеждает нас настолько, что мы готовы убеждать других.

• Точно так же и понятие эффективности алгоритма носит исключительно прагматический и психологический характер. Эффективный алгоритм — это такой алгоритм, который дает ответ на поставленную задачу на имеющемся оборудовании за приемлемое для пользователя время.

• Важнейшей задачей любого курса программирования на начальном этапе является развитие алгоритмического мышления. Для большинства начинающих серьезной трудностью является понимание алгоритма как такового, а также уточнение и перевод математических понятий на язык понятный компьютеру. С этой точки зрения чрезвычайно полезно сравнивать разные алгоритмы, в том числе даже *заведомо* плохие.

• Для небольших значений параметров — скажем, порядка нескольких сотен или нескольких тысяч — время вычисления пренебрежимо мало по сравнению со временем обмена с памятью и вывода на экран и борьба за эффективность алгоритма вообще не имеет никакого смысла.

Здесь часто можно использовать экспоненциальные алгоритмы, которые во многих случаях гораздо проще реализовать. В большинстве реально возникающих задач время, необходимое для реализации более эффективного алгоритма, никак не компенсируется выигрышем в скорости вычисления.

• Более того, для многих задач при таких значениях параметров, при которых они еще решаются в реальное время на бытовом компьютере, асимптотически более эффективные алгоритмы *проигрывают* в скорости менее эффективным за счет больших аддитивных и/или мультипликативных констант, необходимости предварительной обработки данных и других подобных обстоятельств.

• Выбор профессионального алгоритма для решения задачи, находящейся на пределе сегодняшних вычислительных возможностей, является чрезвычайно тонким делом. Например, в большинстве задач линейной алгебры выбор параметров алгоритма определяется абсолютно конкретными деталями используемой системы и зависит от организации конвейера, объема кэша и других подобных обстоятельств. Обсуждение подобных деталей на начальном этапе обучения программированию представляется совершенно абсурдным.

В настоящем пособии параметры подобраны так, чтобы на бытовом компьютере вычисление производилось за несколько секунд — или, в исключительных случаях, за 2–3 минуты. Другим ограничением являлась величина вывода. Обычно параметры выбраны так, чтобы получающийся ответ полностью помещался на один экран. Продолжающееся несколько минут вычисление или непомерно длинный вывод в большинстве случаев следует рассматривать как явное указание на ошибку в программе.

§ 4. Пререквизиты

> "You know, it's at times like this when I'm trapped in a Vogon airlock with a man from Betelgeuse and about to die of asphyxiation in deep space that I really wish I'd listened to what my mother told me when I was young!"
> "Why, what did she tell you?"
> "I don't know, I didn't listen!"
>
> Douglas Adams, *Hitchhiker's Guide to the Galaxy*

Предполагается, что читатель знаком с простейшими понятиями языка `Mathematica` и несколькими десятками основных внутренних функций, примерно в объеме приводимого ниже силлабуса курса "Математика и компьютер", читаемого на первом курсе экономического факультета СПбГУ.

Все эти сведения с исчерпывающей полнотой освещаются в руководстве Стивена Вольфрама [Wo]. Однако, для решения подавляющего большинства задач требуется лишь то, что изложено в нашем учебнике [VH1], [VH2]. Более того, в действительности достаточно лишь знакомства с основной частью главы 3 выпуска 1 (в первую очередь §§1–3, 5, 7–9, 11) и основным

содержанием выпуска 2. Все остальные команды и конструкции языка Mathematica, а также все математические определения и факты, выходящие за рамки школьного курса математики и излагаемых на 1-м курсе анализа и линейной алгебры, нами напоминаются.

Силлабус

> J’ai oublié l’orthographie aussi, et la moitié des mots. Cela n’a pas d’importance, paraît-il.[4]
>
> Samuel Becket, *Molloy*

> Забудьте, если знали, и никогда не вспоминайте, и даже не пытайтесь узнать, что означают слова: гаспаччо, буйабез, вишисуаз, минестроне, авголемоно. Не спрашивайте, из каких продуктов сделаны эти блюда, острые они или пресные, холодные или горячие. Вам этого знать не нужно. Да чего там гаспаччо: забудьте разницу между щами и борщом.
>
> Татьяна Толстая, *Сердца горестные заметы*

1. Основы синтаксиса.

• Объекты, атомарные объекты, тип объекта, имя объекта, объекты типов `Integer`, `Rational`, `Real`, `Complex`, `Symbol`, `String`.

• Разделение аргументов функции или компонент списка `Comma ,` и разделение команд или частей аргумента `Semicolon ;`.

• Выражения, правильно составленные выражения. Заголовок, длина, часть, уровень: `FullForm`, `Head`, `Length`, `Part`, `Level`.

• Группировка и скобки: `Parenthesis ()`, `Braces {}`, `Brackets []`, `DoubleBrackets [[]]`.

• Спецификация уровня: `Level`, `Depth`, `n`, `-n`, {`n`}, {`m,n`}.

• Значения истинности `True`, `False` и простейшие тесты `TrueQ`, `SameQ ===`, `UnsameQ =!=`, `IntegerQ`, `PrimeQ`, `EvenQ`, `OddQ`, `NumberQ`, `NumericQ`.

• Проверка принадлежности списку или домену `MemberQ`, `Element`, `FreeQ`.

• Основные булевы операции `Not !`, `And &&`, `Or ||`, `Xor`, `Implies` и кванторы `ForAll`, `Exists`.

• Важнейшие отношения: `Equal ==`, `Unequal !=`, `Greater >`, `GreaterEqual >=`, `Less <`, `LessEqual <=`, `Order`.

• Переменные и их значения. Текущее значение. Немедленное и отложенное присваивание: `Set ==` и `SetDelayed :=`. Чистка: `ClearAll`.

• Немедленная и отложенная подстановка: `Rule ->` и `RuleDelayed :>`.

• Замены: `Replace /.`, `Replace All //.`, `ReplaceRepeated`.

• Внутренние итеративные конструкции: `Do`, `Sum`, `Product`, `Table`. Формы итератора {`n`}, {`i,n`}, {`i,m,n`}, {`i,m,n,d`}.

[4] Я также забыл орфографию и половину слов. Но это, скорее всего, не имеет никакого значения.

• Понятие о приоритете, таблица приоритетов.

2. Числа.

• Домен `Booleans` и числовые домены `Integers`, `Primes`, `Rationals`, `Algebraics`, `Reals`, `Complexes`,

• Задание целых и рациональных чисел. Команды работы с цифрами: `IntegerDigits`, `FromDigits`, `DigitCount`.

• Арифметические операции `Plus +`, `Minus -`, `Subtract -`, `Times *`, `Divide /`, `Power ^`, `Sqrt`.

• Делимость целых чисел, наибольший общий делитель `GCD` и наименьшее общее кратное `LCM`. Факториал `Factorial !`.

• Деление целых чисел с остатком: `Quotient`, `Mod`, `PowerMod`.

• Простые числа: `Prime`, `PrimeQ`, `PrimePi`. Основная теорема арифметики: `FactorInteger`, `IntegerExponent`.

• Задание вещественных чисел и основные числовые форматы. Приближенное значение `N` и `RealDigits`. Основные константы: `E`, `Pi`, `Infinity`.

• Основные численные функции: `Abs`, `Sign`, `Max`, `Min`, `Round`, `Floor`, `Ceiling`.

• Задание комплексных чисел: `I`, `Re`, `Im`, `Abs`, `Arg`, `Conjugate`, `ComplexExpand`.

• Генерация (псевдо)случайных чисел при помощи функции `Random`, параметры функции `Random`, затравка генератора случайных чисел `SeedRandom`.

3. Элементарные функции.

• Функция, имя функции, аргумент, значение. Функциональная и операторная запись функции.

• Задание многочленов и их структура: `Variables`, `Coefficient`, `CoefficientList`, `Exponent`.

• Основные структурные манипуляции с многочленами: `Expand`, `Factor`, `Collect`, `Decompose`.

• Арифметика многочленов: `PolynomialQuotient`, `PolynomialRemainder`, `PolynomialMod`, `PolynomialGCD`, `PolynomialLCM`, `PolynomialReduce`.

• Рациональные дроби и структурные манипуляции с ними: `Numerator`, `Denominator`, `Together`, `Apart`, `ExpandAll`.

• Решение (алгебраических) уравнений: `Solve`, `Roots`, `Root`, `LinearSolve`, `Reduce`, `Eliminate`.

• Основные элементарные функции: `Exp`, `Log`, `Cos`, `Sin`, `Tan`, `Cot`.

• Символьное дифференцирование и интегрирование: `Limit`, `Series`, `D`, `Derivative`, `Integrate`, `DSolve`.

• Упрощение выражений: `Simplify`, `FullSimplify`, `Refine`, `Function Expand`. Упрощение с предположениями.

4. Функциональное программирование.

• Основные классы функций языка `Mathematica`: арифметические операции, символьные вычисления, структурные манипуляции, булевы функции, числовые функции, приближенные функции, функции работы со списками, команды и директивы процедурного и функционального программирования.

• Функции нескольких аргументов. Функциональная запись `f[x,y]`, префиксная операторная запись `f @@ {x,y}`, инфиксная операторная запись `x~f~y` и постфиксная операторная запись `Sequence[x,y] // f`

• Формат аргументов, список и последовательность аргументов, отличие `f[x][y]`, `f[x,y]` и `f[{x,y}]`.

• Типы аргументов. Явные аргументы (собственно аргументы) и неявные аргументы (параметры, опции, атрибуты). Дефолтные значения. Настройки опций `Option->Choice`.

• Фиктивные переменные, бланки `Blank _` и последоваетльности бланков `BlankSequence __`, `BlankNullSequence ___`.

• Задание функций с помощью конструкции `f[x_]:=rhs`. Опоределение с условием `Condition /;`.

• Индукция и рекурсия. База индукции и шаг индукции. Начальные условия и рекуррентные соотношения.

• Контроль времени `Timing`. Сравнение эффективности алгоритмов. Экспоненциальный, полиномиальный и логарифмический рост.

• Сравнение различных определений степени, факториала, биномиальных коэффицентов, чисел Фибоначчи и чисел Стирлинга с точки зрения вычислительной эффективности.

• Конструкция `Remember`, запоминающая таблицу значений функции

```
f[x_]:=f[x]=rhs].
```

• Чистая и анонимная функция `Function &`, слоты и последовательности слотов: `Slot #`, `SlotSequence ##`.

• Итерации функций: `Nest`, `NestList`, `NestWhile`, `FixedPoint`, `Fold`, `FoldList`.

• Применение функций к спискам: `Apply`, `Map`, `MapAt`, `MapAll`, `MapThread`, `MapIndexed`, `Scan`.

• Распределение и протаскивание действия функций: `Inner`, `Outer`, `Distribute`, `Thread`, `Through`, `Operate`.

5. Списочное программирование.

• Списки и последовательности: `List` и `Sequence`. Выделение частей списка: `Part`, `Extract`.

• Формирование списков: `Table`, `Array`, `Range`, `CharacterRange`.

• Вычеркивания: `Take`, `Drop`, `First`, `Last`, `Most`, `Rest`, `Delete`.

• Вставки и замены: `Insert`, `ReplacePart`, `Append`, `AppendTo`, `Prepend`, `PrependTo`.

• Выборки: `Select`, `Cases`, `DeleteCases`, `Count`, `Position`.

• Простейшие структурные манипуляции: `Join`, `Reverse`, `RotateLeft`, `RotateRight`, `PadLeft`, `PadRight`.

• Вложенные списки и изменение уровней вложенности: `Flatten`, `FlattenAt`, `Partition`, `Split`, `Transpose`.

• Сортировка списков: `Sort`, `Ordering`, `Permutations`, и т.д.

• Основные теоретико-множественные операции: `Union`, `Intersection`, `Complement`.

• Операции над матрицами. Матричное умножение `Dot .`, обратная матрица `Inverse`, функции от матриц `MatrixPower`, `MatrixExp`.

• Системы линейных уравнений: `LinearSolve`, `NullSpace`, `RowReduce`.

• Инварианты матриц: `Det`, `Tr`, `Minors`, `MatrixRank`.

• Собственные числа и векторы: `Eigenvectors`, `Eigenvalues`, `Eigensystem`

6. Процедурное программирование.

• Основной цикл: `In`, `Out`.

• Организация циклов `Do`, `For`, `While`, `Increment ++`, `Decrement --`.

• Условные операторы: `If`, `Which`, `Switch`.

• Передача управления: `Goto`, `Label`.

• Конструкции локализации переменных `Block`, `Module`.

• Удерживание в вбрасывание: `Hold`, `Return`, `Evaluate`.

• Контексты: `Begin`, `End`. Чтение файлов `Get <<`.

Рекомендованная литература

> Read, read, read, read, my unlearned reader! read, — or, by the knowledge of the great saint Paraleipomenon — I tell you beforehand, you had better throw down this book at once; for without *much reading*, by which your Reverence knows I mean *much knowledge*, you will not be able to penetrate the moral of the next marbled page (motley emblem of my work!)
>
> Laurence Sterne, *Tristram Shandy*

[VH1] Н.А.Вавилов, В.Г.Халин, `Mathematica` *5.* для нематематика. Вып.* 1. *Первое знакомство*, ОЦЭиМ, СПб, 2005, pp. 1–180.

[VH2] Н.А.Вавилов, В.Г.Халин, Mathematica *5.* для нематематика. Вып.* 2. *Основы синтаксиса*, ОЦЭиМ, СПб, 2005, pp. 1–136.
[GKP] Р.Грехем, Д.Кнут, О.Паташник, *Конкретная математика. Основание информатики*, Мир, М., 1998, pp. 1–703.
[Iv] О.А.Иванов, *Избранные главы элементарной математики*, Изд-во СПбГУ, СПб, 1995, pp. 1–223.
[Li] В.Липский, *Комбинаторика для программистов*, Мир, М, 1988, pp. 1–213.
[Li] А.Шень, *Программирование: теоремы и задачи*, МЦНМО, М, 2004, pp. 1–294.
[Ma] R.E.Maeder, *Programming in* Mathematica*, 3rd ed.*, Addison-Wesley, 1996.
[Wo] S.Wagon, Mathematica *in Action*, Springer-Verlag, 1999.
[Wo] S.Wolfram, *The* Mathematica *book. 5th ed.*, Wolfram Media, 2003, pp. 1–1464.

Дополнительная литература

Важнейшая функция чтения состоит в том, чтобы избавить от необходимости думать.

Поль Валери, *Тетради*

[AGKS] С.А.Абрамов, Г.Г.Гнездилова, Е.Н.Капустина, М.Н.Селин, *Задачи по программированию*, Наука, М., 1988, pp. 1–224.
[Ak] А.Акритас, *Основы компьютерной алгебры с приложениями*, Мир, М., 1994.
[An] Дж.Андерсон, *Дискретная математика и комбинаторика*, Вильямс, М. – СПб – Киев, 2003, pp. 1–957.
[AHU1] А.Ахо, Дж.Хопкрофт, Дж.Ульман, *Построение и анализ вычислительных алгоритмов*, Мир, М., 1979, pp. 1–536.
[AHU2] А.Ахо, Дж.Хопкрофт, Дж.Ульман, *Структуры данных и алгоритмы*, Вильямс, М. – СПб – Киев, 2000, pp. 1–382.
[BB] Г.Биркгоф, Т.Барти, *Современная прикладная алгебра*, Мир, М., 1976.
[BK] А.Л.Брудно, Л.И,Каплан, *Московские олимпиады по программированию*, Наука, М., 1990, pp. 1–208.
[Wi] Н.Вирт, *Систематическое программирование: введение*, Мир, М., 1977.
[Ga] Д.Гасфилд, *Строки, деревья и последовательности в алгоритмах*, Невский диалект, СПб, 2003.
[Gr] Д.Грис, *Наука программирования*, Мир, М., 1984.
[Dij] Э.Дейкстра, *Дисциплина программирования*, Мир, М., 1978.
[K1] Д.Кнут, *Искусство программирования.* I. *Основные алгоритмы*, Вильямс, М. – СПб – Киев, 2000, pp. 1–712.
[K2] Д.Кнут, *Искусство программирования.* II. *Получисленные алгоритмы*, Вильямс, М. – СПб – Киев, 2000, pp. 1–827.
[K2] Д.Кнут, *Искусство программирования.* III. *Сортировка и поиск*, Вильямс, М. – СПб – Киев, 2000, pp. 1–822.
[CLO] Д.Кокс, Дж.Литтл, Д.О’Ши, *Идеалы, многообразия и алгоритмы*, Мир, М., 2000, pp. 1–687.
[CLR] Т.Кормен, Ч.Лейзерсон, Р.Ривест, *Алгоритмы: построение и анализ*, МНЦМО, М., 2000.
[Li] А.Купиллари, *Трудности доказательства: как преодолеть страх перед математикой*, Техносфера, М., 2002, pp. 1–233.
[KL] А.Г.Кушниренко, Г.В.Лебедев, *Программирование для математиков*, Наука, М., 1988.
[LP] Р.Лидл, Г.Пильц, *Прикладная абстрактная алгебра*, Изд-во Уральского Унив., Екатиренбург, 1996.
[Mac] Дж.Макконелл, *Анализ алгоритмов: вводный курс*, Техносфера, М., 2002, pp. 1–302.

[Me] Ф.В.Меньшиков, *Олимпиадные задачи по программиованию*, Питер, СПб et al., 2006, pp. 1–314.
[NK] П.Ноден, К.Китте, *Алгебраическая алгоритмика*, Мир, М., 1999.
[RND] Э.Рейнгольд, Ю.Нивергельт, Н.Део, *Комбинаторные алгоритмы: теория и практика*, Мир, М., 1980.

Книги по арифметике и теории чисел

> Привычка грамотного человека к чтению есть форма мазохизма.
>
> Михаил Веллер, *Ножик Сережи Довлатова*

Мы приводим только книги по вычислительной, элементарной и рекреативной теории чисел, преимущественно книги на русском языке, доступные восприятию студента, независимо от специальности. Кроме того, мы цитируем некоторые книги, на которые мы постоянно ссылаемся в тексте как на источник исторических и фактических сведений — в первую очередь это книги Диксона, Наркевича, Серпинского, Рибенбойма, Коха и Пипера, Эдвардса. По очевидным причинам мы не пытаемся давать систематическую библиографию, посвященную алгебраической теории чисел, предполагающим подготовку в области алгебры и алгебраической геометрии, и более специальным вопросам аналитической теории чисел, требующим хорошего понимания вещественного и комплексного анализа.

[AR] К.Айрленд, М.Роузен, *Классическое введение в современную теорию чисел*, М., Мир, 1987, pp. 1–415.
[BSh] З.И.Боревич, И.Р.Шафаревич, *Теория чисел*, Наука, М., 1985.
[BZRKJ] В.Боро, Д.Цагир, Ю.Рольфс, Ч.Крафт, Е.Янцен, *Живые числа: пять экскурсий*, Мир, М., 1985, pp. 1–128.
[Bu] А.А.Бухштаб, *Теория чисел*, Учпедгиз, М., 1960.
[Vas] О.Н.Василенко, *Теоретико-числовые алгоритмы в криптографии*, МЦНМО, М., 2003, pp. 1–325.
[VANT] *Вычисления в алгебре и теории чисел*, Мир, М., 1976, pp. 1–305.
[GNSh] А.И.Галочкин, Ю.В.Нестеренко, А.В.Шидловский, *Введение в теорию чисел*, Изд-во МГУ, М., 1995.
[GCh] С.Б.Гашков, В.Н.Чубариков, *Арифметика, алгоритмы, сложность вычислений, 2-е изд.*, Высшая школа, М., 2000, pp. 1–320.
[Ge] А.О.Гельфонд, *Трансцендентные и алгебраические числа*, Гостехиздат, М., 1952.
[GL] А.О.Гельфонд, Ю.В.Линник, *Элементарные методы в аналитической теории чисел*, ГИФМЛ, М., 1962.
[GK] Р.Грегори, Е.Кришнамурти, *Безошибочные вычисления: методы и приложения*, Мир, М., 1998, pp. 1–208.
[DU] Е.Б.Дынкин, В.А.Успенский, *Математические беседы*, Наука, М., 2004, pp. 1–240.
[Da1] Г.Дэвенпорт, *Высшая арифметика*, Наука, М., 1965, pp. 1–175.
[Da2] Г.Дэвенпорт, *Мультипликативная теория чисел*, Наука, М., 1971, pp. 1–199.
[In] А.Э.Ингам, *Распределение простых чисел*, УРСС, М., 2004, pp. 1–160.
[Ca1] Дж.Касселс, *Введение в теорию диофантовых приближений*, ИИЛ, М., 1961, pp. 1–212.
[Ca2] Дж.Касселс, *Введение в геометрию чисел*, Мир, М., 1965, pp. 1–421.
[Ca3] Дж.Касселс, *Рациональные квадратичные формы*, Мир, М., 1982, pp. 1–436.
[Ka] С.Б.Каток, *p-адический анализ в сравнении с вещественным*, МЦНМО, М., 2004, pp. 1–107.
[Ko1] Н.Коблиц, *p-адические числа, p-адический анализ и дзета-функции*, Мир, М., 1982, pp. 1–192.

[Ko2] Н.Коблиц, *Курс теории чисел и криптографии*, Изд-во ТВП, М., 2001.
[Ko] Х.Кох, *Алгебраическая теория чисел*, ВИНИТИ, М., 1990, pp. 1–308.
[La] С.Ленг, *Введение в теорию диофантовых приближений*, Мир, М., 1970, pp. 1–103.
[MP] Ю.И.Манин, А.А.Панчишкин, *Введение в теорию чисел*, ВИНИТИ, М., 1990, pp. 1–348.
[Mi] Ш.Х.Михелович, *Теория чисел*, Высшая школа, М., 1967, pp. 1–336.
[Ni] А.Нивен, *Числа рациональные и иррациональные*, Мир, М., 1966, pp. 1–198.
[Po1] М.М.Постников, *Теорема Ферма: введение в теорию алгебраических чисел*, Наука, М., 1978, pp. 1–128.
[Po2] М.М.Постников, *Введение в теорию алгебраических чисел*, Наука, М., 1982, pp. 1–237.
[Pr] К.Прахар, *Распределение простых чисел*, Мир, М., 1967, pp. 1–511.
[RT] Г.Радемахер, О.Теплиц, *Числа и фигуры*, RCD, Ижевск, 2000, pp. 1–258.
[Ri1] П.Рибенбойм, *Рекорды в исследованиях простых чисел*, НИИХ СПбГУ, СПб, 2002, pp. 1–282.
[Ri2] П.Рибенбойм, *Последняя теорема Ферма*, Мир, М., 2003, pp. 1–429.
[Ro] К.А.Родосский, *Алгоритм Эвклида*, Наука, М., 1988, pp. 1–240.
[Sa] А.Саломаа, *Криптография с открытым ключом*, Мир, М., 1996.
[Se] Ж.-П.Серр, *Курс арифметики*, Мир, М., 1972, pp. 1–184.
[Si] С.Сингх, *Великая теорема Ферма*, МЦНМО, М., 2000, pp. 1–288.
[Tr] Э.Трост, *Простые числа*, ГИФМЛ, М., 1959.
[Fe] Н.И.Фельдман, *Седьмая проблема Гильберта*, Изд-во МГУ, М., 1982, pp. 1–311.
[Ha] Г.Хассе, *Лекции по теории чисел*, ИЛ, М., 1953.
[Hi] А.Я.Хинчин, *Цепные дроби, 3-е изд.*, ГИФМЛ, М., 1961.
[Cha] Л.Чандрасекхаран, *Введение в аналитическую теорию чисел*, Мир, М., 1974, pp. 1–187.
[Ch] А.В.Черемушкин, *Лекции по арифметическим алгоритмам в криптографии*, МЦНМО, М., 2002, pp. 1–103.
[Sch] В.Шмидт, *Диофантовы приближения*, Мир, М., 1983, pp. 1–228.
[Ed] Г.Эдвардс, *Последняя теорема Ферма: генетическое введение в алгебраическую теорию чисел*, Мир, М., 1980, pp. 1–484.
[Ap] T.M.Apostol, *Introduction to analytic number theory*, Springer-Verlag, Berlin et al., 1997.
[BSh] E.Bach, J.Shallit, *Algorithmic number theory,* I, MIT Press, Boston, 1996.
[Co] H.Cohen, *A course in computational number theory*, Springer-Verlag, Berlin et al., 1993.
[CP] R.Crandall, C.Pommerance, *Prime numbers: a computational prospective*, Springer-Verlag, Berlin et al., 2001.
[Gu] R.K.Guy, *Unsolved problems in number theory*, Springer-Verlag, Berlin, 1994.
[Ko] H.Koch, *Einführung in die klassische Mathematik. I. Vom quadratischen Reziprozitätsgesetz bis zum Uniformisierungssatz*, Akademie-Verlag, Berlin, 1986, pp. 1–326.
[KP] H.Koch, H.Pieper, *Zahlentheorie*, Deutscher Verlag der Wiss., Berlin, 1976, pp. 1–232.
[Na1] W.Narkiewicz, *Elementary and analytic theory of algebraic numbers*, PWN, Warszawa, 1974, pp. 1–630.
[Na2] W.Narkiewicz, *Teoria liczb*, PWN, Warszawa, 1977, pp. 1–355.
[Na3] W.Narkiewicz, *Classical problems in number theory*, PWN, Warszawa, 1986, pp. 1–363.
[Na4] W.Narkiewicz, *The development of prime number theory*, Springer-Verlag, Berlin, 2000.
[KP] H.Pieper, *Variationen über ein zahlentheoretisches Thema von Gauß*, Deutscher Verlag der Wiss., Berlin, 1978, pp. 1–183.

[PZ] M.Pohst, H.Zassenhaus, *Algorithmic algebraic number theory*, Cambridge Univ. Press, 1989.

[Ri] P.Ribenboim, *The new book of prime number records*, Springer-Verlag, Berlin et al., 1996.

[Rie] H.Riesel, *Prime numbers and computer methods for factorisation*, Birkhäuser, Basel et al., 1985.

[Sie] W.Sierpiński, *Elementary theory of numbers*, PWN, Warszawa, 1964, pp. 1–480.

[Shp] I.E.Shparlinski, *Number theoretic methods in cryptography: complexity lower bounds*, Birkhäuser, Basel et al., 1999.

Довольно этой канцелярщины, Генрих. Напишите там, “Брак считается совершившимся” и давайте кушать. Ужасно кушать хочется.

Евгений Шварц, *Дракон*

Глава 1. ЦЕЛЫЕ ЧИСЛА

Why are numbers beautiful? It's like asking why is Beethoven's Ninth Symphony beautiful. If you don't see why, someone can't tell you. I *know* numbers are beautiful. If they aren't beautiful, nothing is.

Paul Erdös

В настоящей главе мы обсуждаем основы вычислений с целыми числами, в особенности их позиционную запись. Вычисления с целыми числами являются основой всех обычных вычислений. Рациональное число интерпретируется как пара целых чисел, числителя и знаменателя. Приближенное вещественное число интерпретируется как пара целых чисел, называемых мантиссой[5] и показателем.

§ 1. Арифметика целых чисел

Seid umschlungen, Millionen![6]

Friedrich Schiller

A billion here, a couple of billion there — first thing you know it adds up to be real money.

Bill Gates

В настоящем параграфе мы коротко напоминаем основные команды, связанные с целыми числами. Следующие тесты выясняют, имеет ли число x *формат* целого числа и, в этом случае, является ли оно четным или нечетным.

`IntegerQ[x]`	целочисленность x
`EvenQ[x]`	четность x
`OddQ[x]`	нечетность x

Стоит еще раз подчеркнуть, `IntegerQ[x]` возвращает `True`, только в том случае, когда число x имеет *заголовок* `Integer`. Во всех остальных случаях, даже если x задекларировано как целое, например, посредством указания паттерна, тест `IntegerQ[x]` вернет значение `False`. Тем самым, `IntegerQ[2]` возвращает значение `True`, в то время как `IntegerQ[2.]` — `False`. Дело в том, что с точки зрения системы 2. является вовсе не целым, а приближенным вещественным числом.

[5] То, что традиционно мантисса *записывается* как правильная дробь, роли не играет, вычисления с мантиссами производятся как с целыми числами.

[6] Обнимитесь, миллионы!

Все арифметические операции в Mathematica имеют обычные названия и подчиняются стандартным соглашениям.

`x+y+z`	`Plus[x,y,z]`	сложение
`-x`	`Minus[x]`	переход к противоположному
`x-y`	`Subtract[x,y]`	вычитание
`x*y*z`	`Times[x,y,z]`	умножение
`x/y`	`Divide[x,y]`	деление

Единственное, на что следует обращать внимание, это вопросы приоритета. Например, выражение `a/b*c` будет интерпретировано как ac/b. Поэтому во всех сомнительных случаях следует использовать скобки. Выражение ab/cd предпочтительно вводить в виде `(a*b)/(c*d)`.

1.1. Сколько среди выражений `a/b/c/d`, `a/(b/c)/d`, `a/b/(c/d)`, `a/(b/c/d)`, `a/(b/(c/d))` различных?

Суммы и произведения вычисляются при помощи внутренних команд `Sum` и `Product`, с итераторами обычного формата.

`Sum[a,{i,m,n}]`	сумма $\sum_{i=m}^{n} a_i$
`Product[a,{i,m,n}]`	произведение $\prod_{i=m}^{n} a_i$

1.2. Найдите сумму всех n-значных чисел

Решение. Породив таблицу первых 10 таких сумм

```
Table[Sum[i,{i,10^(n-1),10^n-1}],{n,1,10}]
```

мы увидим следующий ответ

45, 4905, 494550, 49495500, 4949955000, 494999550000, 49499995500000, 4949999955000000, 494999999550000000, 49499999995500000000,

где количество идущих подряд девяток на единицу меньше, чем количество нулей. *После* того как этот ответ написан, он становится очевидным. Если Вы предпочитаете видеть не всю таблицу сразу, а отдельные суммы, которые возникают на экране по мере их вычисления, можно напечатать, например, `For[n=1,n<=10,n++,Print[Sum[i,{i,10^(n-1),10^n-1}]]]`

Максимум и минимум конечной последовательности или конечного списка чисел ищутся при помощи команд `Max` и `Min`.

`Max[m,n]`	максимум m и n
`Min[m,n]`	минимум m и n

Пусть $2s$ натуральных чисел расположены в порядке возрастания:

$$0 < n_1 < n_2 \ldots < n_{2s-1} < n_{2s}.$$

1.3. Разбейте эти числа на пары так, чтобы сумма произведений пар была максимальна.

1.4. Разбейте эти числа на пары так, чтобы сумма произведений пар была минимальна.

1.5. Разбейте эти числа на пары так, чтобы произведение сумм пар было максимально.

1.6. Разбейте эти числа на пары так, чтобы произведение сумм пар было минимально.

Указание. Проведите небольшой компьютерный эксперимент. После того, как ответ известен, он легко доказывается индукцией.

1.7. Найдите все наборы из m натуральных чисел $\leq n$, для которых их сумма равна изх произведению.

§ 2. Вычисление степеней

> O ye POWERS! (for powers ye are, and great ones too)
>
> Laurence Sterne, *Tristram Shandy*

> If all computers of earth were to focus simply on writing down as many digits as possible for the next century, they would write down far less than 2^{2^7} digits.
>
> Andrew Granville[7]

Следующая функция служит для образования степеней.

`x^y` `Power[x,y]` потенцирование

Обратите внимание, что потенцирование использует правую группировку!

2.1. Проверьте, как истолковывается выражение `l^m^n`. В связи с эпиграфом к следующему параграфу вычислите `2^2^7` и `(2^2)^7`.

2.2. Дайте рекуррентное определение функции степени x^n.

Решение. Напрашивающееся определение для того, кто делал вид, что учился математике, но при этом никогда не интересовался программированием, и ничего не вычислял сам, такое:

```
power[x_,0]:=1;
power[x_,n_]:=power[x,n-1]*x
```

Будучи правильным с логической точки зрения в вычислительном смысле это определение *абсолютно чудовищно*. Почему? Подобное определение становится совершенно бесполезным при вычислении чего-нибудь столь крошечного, как 2^{100000}. Для вычислении x^n по этому определению требуется около $n-1$ умножений, в то время как еще древние египтяне знали,

[7] A.Granville, It is easy to determine whether a given integer is prime. — Bull. Amer. Math. Soc., 2004, vol.42, N.1, p.3–38.

что при вычислении x^n можно обойтись количеством умножений порядка $\log_2(n)$. Следующий способ вычисления степени называется **бинарным** или **египетским алгоритмом**:

$$x^n = \begin{cases} (x^{n/2})^2, & \text{если } n \text{ четно,} \\ x^{n-1}x, & \text{если } n \text{ нечетно.} \end{cases}$$

Непродолжительное размышление убедит Вас в том, что 100000 умножений это чуть меньше, чем 2^{100000} умножений. Так, выполняя 10^9 умножений в секунду, мы произведем 100000 умножений за время, меньшее зернистости контроля времени системой `Windows` (1 миллисекунда). Тогда как $2^{100000} \approx 10^{30103}$ умножений не может быть выполнено *никогда*.

2.3. А теперь определите функцию x^n еще раз, при помощи египетского алгоритма.

Решение. Ну, *хотя бы*, так:

```
power[x_,0]:=1;
power[x_,n_]:=If[EvenQ[n],power[x,n/2]^2,power[x,n-1]*x]
```

Хотя, вероятно, в естественных условиях мы воспользовались бы не условным оператором, а определением с условием:

```
power[x_,0]:=x;
power[x_,n_]:=power[x,n/2]^2 /; EvenQ[n]
power[x_,n_]:=power[x,n-1]*x /; OddQ[n]
```

Конечно, по факту эти определения работают за одинаковое время (но все еще раз в пять медленнее, чем встроенная функция `Power` использующая профессиональные алгоритмы).

Вычисление степеней можно реализовывать и иначе. Еще один чрезвычайно популярный алгоритм вычисления степени, известный очень давно, но систематически исследованный только де Жонкьером в самом конце XIX века, это **метод простого множителя**:

$$x^n = \begin{cases} (x^{n-1})x, & n \text{ простое,} \\ (x^p)^m, & n = pm, \text{ где } p \text{ наименьший простой делитель } n. \end{cases}$$

2.4. Определите функцию x^n при помощи метода простого множителя и сравните скорость ее работы со скоростью функции основанной на египетском алгоритме.

2.5. Найдите случаи, когда метод простого множителя требует меньше умножений, чем египетский алгоритм.

Ответ. Первый случай, когда метод простого множителя лучше, чем египетский алгоритм, это вычисление x^{15}, которое требует *шести* умножений по египетскому алгоритму

$$x, x^2, x^3, x^6, x^7, x^{14}, x^{15},$$

и только *пять* умножений по методу простого множителя:

$$x, x^2, x^3, x^6, x^{12}, x^{15}.$$

Впрочем, x^n можно вычислять и многими другими способами, например, вычисление x^{15} при помощи описанного в книге Кнута **дерева степеней** тоже дает *пять* умножений:

$$x, x^2, x^3, x^5, x^{10}, x^{15}.$$

Первый случай, для которого дерево степеней более эффективно, чем метод простого множителя, это вычисление x^{23}. При этом используется только *шесть* умножений:

$$x, x^2, x^3, x^5, x^{10}, x^{13}, x^{23},$$

в то время как метод простого множителя требует *семь* умножений:

$$x, x^2, x^4, x^5, x^{10}, x^{11}, x^{22}, x^{23}.$$

Хармс определил функцию, несколько первых значений которой таковы:

$$1,\ 2^2,\ 3^{3^3},\ 4^{4^{4^4}},\ 5^{5^{5^{5^5}}}, \ldots$$

2.6. Дайте рекуррентное определение функции Хармса. Сколько значений этой функции Вам удастся вычислить?

§ 3. Извлечение цифр

> Учитель сказал, что я совсем не знаю математики и поставил мне в дневник какую-то цифру.
>
> Николай Фоменко

В настоящем параграфе мы начнем отрабатывать технику вычислений с цифрами. Это на непросвещенный взгляд чисто схоластическое занятие имеет по крайней мере два важных приложения. Во-первых, числовые последовательности обычно удобнее всего порождать именно через последовательности цифр. Во-вторых, большинство эффективных современных алгоритмов работы с многозначными числами имитирует вычисления с многочленами и основано на разбиении чисел на блоки цифр.

`IntegerDigits[n]`	список цифр числа n
`FromDigits[{a1,...,an}]`	преобразование списка цифр в число
`DigitCount[n,b,d]`	кратность цифры d в числе n
`IntegerExponent[n]`	количество нулей в конце n

3.1. Как узнать, сколько цифр содержит число n? Чему равна сумма его цифр? Какая цифра стоит у него в старшем/младшем разряде?

Ответ. Проще всего так:

```
Length[IntegerDigits[n]],    Total[IntegerDigits[n]],
First[IntegerDigits[n]],     Last[IntegerDigits[n]],
```

3.2. Задайте число $9\ldots9$, состоящее из n девяток.

3.3. Задайте число $123\ldots123$, где группа цифр 123 повторяется n раз.

Решение. По-хорошему это делается, например, так

```
FromDigits[Flatten[Table[{1,2,3},{n}]]]
```

Функция выравнивания `Flatten` убирает лишний уровень вложенности в получающемся при исполнении `Table` списке `{{1,2,3},...{1,2,3}}`.

Можно, конечно, и без затей

```
Sum[123*1000^i,{i,1,n}]
```

но для больших n это гораздо менее эффективно с вычислительной точки зрения. Кроме того, возможности применения команды `FromDigits` гораздо шире.

3.4. Задайте число $10001\ldots10001$, состоящее из n единиц, разделенных группами по три нуля.

Указание. Используйте `Join`, чтобы добавить в список последнюю единицу.

3.5. Задайте число $1234\ldots9991000$, состоящее из выписанных подряд чисел от 1 до 1000.

Указание. Обратите внимание, что наивное `FromDigits[Range[1000]]` дает неправильный результат! Используйте `IntegerDigits` и `Flatten`.

Еще одной внутренней командой является `DigitCount[n]]`, которая показывает *количество* цифр, входящих в запись числа n, в следующем порядке: 1,2,3,4,5,6,7,8,9,0.

3.6. Определите функцию, которая показывает количество цифр, входящих в разложение числа n, в обычном порядке 0,1,2,3,4,5,6,7,8,9.

Решение. Конечно, можно просто циклически переставить кратности, возвращаемые `DigitCount`:

```
dc1[n_]:=RotateRight[DigitCount[n]]
```

С другой стороны, такую функцию легко задать и обращаясь непосредственно к `IntegerDigits`:

```
dc2[n_]:=Table[Count[IntegerDigits[n],i],{i,0,9}]
```

3.7. Каких чисел среди натуральных чисел $\le 10^n$ больше: тех в десятичной записи которых встречается 1 или тех, в записи которых она не встречается? А теперь определите функцию, вычисляющую количество тех чисел $\le 10^n$, в записи которых не встречается 1, и посчитайте первые 10 ее значений.

3.8. Имеется 9! = 362880 девятизначных чисел, состоящих из попарно различных цифр $1, \ldots, 9$. Сколько из них являются полными квадратами?

Указание. Что легче выбрать, девятизначные числа, являющиеся полными квадратами, или квадраты пятизначных чисел, состоящие из различных цифр?

Ответ. Вот эти числа

139854276	152843769	157326849	215384976	245893761
254817369	326597184	361874529	375468129	382945761
385297641	412739856	523814769	529874361	537219684
549386721	587432169	589324176	597362481	615387249
627953481	653927184	672935481	697435281	714653289
735982641	743816529	842973156	847159236	923187456

Поскольку литература по теории чисел и Computer Science никем не редактируется, большинство многозначных чисел в книгах по этим наукам воспроизведено с опечатками. Таким образом, при ссылке на любые численные данные их приходится проверять заново. В предположении, что опечатка затронула лишь одну цифру, часто приходится генерировать списки чисел, отличающихся от исходного лишь в одной позиции.

3.9. Напишите программу, которая порождает список чисел той же разрядности, отличающихся от исходного в одной цифре.

Решение. Например, при помощи `Replace Part`:

```
replacedigit[n_]:=Map[FromDigits[
      ReplacePart[IntegerDigits[n],#[[2]],#[[1]]]]&,
      Complement[Flatten[Outer[List,
      Range[Length[IntegerDigits[n]]],Range[0,9]],1],{{1,0}}]]
```

3.10. Напишите программу, которая порождает список чисел той же разрядности, отличающихся от исходного в двух цифрах.

3.11. Проверьте, что заменяя в четырехзначном числе не более двух цифр, из него всегда можно получить простое число. Верно ли то же самое для пятизначных чисел?

3.12. Из цифр 1, 2, 3, 4, 5, 6, 7, 8, 9 составлены всевозможные числа, не содержащиеся повторяющихся цифр. Найти их сумму.

§ 4. Манипуляции с цифрами

> Recreational Number Theory is that part of Number Theory that is too difficult to study seriously.
>
> H. W. Lenstra, Jr.

Билет с номером $abcdef$ называется счастливым по петербургски, если

$$a + b + c = d + e + f,$$

т.е. сумма первых трех цифр равна сумме трех последних цифр. Тот же билет называется счастливым по московски, если

$$a + c + e = b + d + f,$$

т.е. сумма цифр, расположенных на нечетных местах равна сумме цифр на четных местах.

4.1. Найдите количество счастливых билетов.

4.2. Найдите количество дважды счастливых билетов (т.е. таких, которые одновременно являются счастливыми по петербургски и по московски).

Ответ. Не пытайтесь вывести на экран *список* номеров счастливых билетов: количество счастливых билетов равно 55252, а дважды счастливых — 6700.

Следующие задачи взяты из классического сборника задач московских математических кружков[8]. Впрочем, его авторы вряд ли предполагали, что эти задачи будут решаться с использованием компьютера!

4.3. Найдите все натуральные числа, которые при зачеркивании последней цифры уменьшаются в целое число раз.

Ответ. Очевидно, что число, заканчивающееся на 0 при отбрасывании последней цифры уменьшается в 10 раз. С другой стороны, если последняя цифра не равна нулю, то при ее отбрасывании число уменьшается больше, чем в 10 раз. Теперь вычисляя

```
Select[Range[10,10000],Last[IntegerDigits[#]]!= 0&&
            Mod[#,FromDigits[Most[IntegerDigits[#]]]]==0&]]
```

мы получим следующие числа 11, 12, 13, 14, 15, 16, 17, 18, 19, 22, 24, 26, 28, 33, 36, 39, 44, 48, 55, 66, 77, 88, 99. Теперь секундное размышление убеждает нас, что каждое число, уменьшающееся при отбрасывании последней цифры в $m > 10$ раз, обязано быть двузначным, так что мы действительно нашли все такие числа.

4.4. Существует ли натуральное число x такое, что его произведения на 2,3,4,5,6 записываются теми же цифрами, что само x, но в другом порядке?

Ответ. Условие, что x и y имеют один и тот же набор цифр выражается как

```
Sort[IntegerDigits[x]]==Sort[IntegerDigits[y]]
```

Теперь по аналогии с предыдущей задачей совсем легко написать код, выбирающий числа, удовлетворяющие этому условию. Вот наименьший такой пример

$$x = 142857, \quad 2x = 285714, \quad 3x = 428571,$$
$$4x = 571428, \quad 5x = 714285, \quad 6x = 857142.$$

[8]Д.О.Шклярский, Н.Н.Ченцов, И.М.Яглом, Избранные задачи и теоремы элементарной математики. Ч.I. Арифметика и алгебра. — М., ГИТТЛ, 1954, с.1–455.

Понятно, что пририсовывая к этому примеру любое количество нулей мы получим число, обладающее таким же свойством. Стоит отметить, что это свойство числа 142857 известно много тысячелетий. Каким образом? Дело в том, что это число представляет собой в точности период десятичной дроби, которой записывается 1/7. Вычисляя `Table[N[i/7],{i,1,6}]` мы получим следующий знакомый ответ:

```
0.142857, 0.285714, 0.428571, 0.571429, 0.714286, 0.857143
```

Вас, конечно, не смущают последние цифры трех последних чисел, получающиеся в результате так называемого **округления**.

Поиск дальнейших примеров является весьма хлопотным делом, занимающим на бытовом компьютере несколько минут. Среди семизначных чисел кроме 1428570 возникает еще ровно один пример, а именно,

$$x = 1429857, \quad 2x = 2859714, \quad 3x = 4289571,$$
$$4x = 5719428, \quad 5x = 7149285, \quad 6x = 8579142.$$

4.5. Существуют ли числа, которые при перестановке первой цифры в конец увеличиваются в три раза?

Ответ. Два таких числа, а именно, 142857 и 285714 были найдены нами в предыдущей задаче. Выполняя следующий код,

```
Select[Range[10^6],
          FromDigits[RotateLeft[IntegerDigits[#]]]==3*#&]
```

мы видим, что это единственные шестизначные числа, обладающие таким свойством. С другой стороны, совершенно ясно, что повторяя ту же группу цифр при помощи команды

```
FromDigits[Flatten[Table[{1,4,2,8,5,7},{n}]]]
```

мы получим дальнейшие числа, обладающие тем же свойством.

4.6. Найдите все числа $n \le 10^6$, которые при вычеркивании одной цифры уменьшаются в 9 раз.

Ответ. Исполнение кода

```
Select[Range[10^6],MemberQ[
          Table[9*FromDigits[Drop[IntegerDigits[#],{i}]],
                    {i,1,Length[IntegerDigits[#]]}],#]&]
```

показывает, что имеется 87 таких чисел 45, 135, 225, 315, 405, 450, 675, 1125, ...

4.7. Найдите все числа $n \le 10^6$, которые при вычеркивании одной цифры уменьшаются в 9 раз и обладают тем дополнительным свойством, что получившееся число снова делится на 9.

Ответ. Нужно лишь внести в предыдущий код дополнительный тест `Mod[Total[IntegerDigits[#]],9]==0&`, выбирающий из списка чисел, получающихся из x вычеркиванием одной цифры, те, сумма которых делится на 9. При этом найдется 18 таких чисел, из которых ровно 7 заканчиваются

на 5: 405, 2025, 6075, 10125, 30375, 50625, 70875, а все остальные получаются из них дорисовыванием нулей в конце.

4.8. Для данного натурального n найдите все пары натуральных чисел таких, что $l + m = n$ и m получается из l вычеркиванием одной цифры.

4.9. Найдите все числа, для которых квадрат заканчивается на само это число — последняя цифра $\neq 0$.

§ 5. Палиндромы

I'm guided by the beauty of our weapons.

Leonard Cohen

В настоящем параграфе мы обсудим операцию **реверсии**, состоящую в переворачивании списка цифр числа. Иными словами, если исходное число x записывается как $a_1 \dots a_n$, то **реверсированное** число $\mathrm{rev}(x) = a_n \dots a_1$ состоит из тех же цифр в обратном порядке. Обратите внимание, что число реверсированное к реверсированному, вообще говоря, не обязано совпадать с исходным. Казалось бы, реверсированные числа представляют собой чистую игру ума, но в действительности в докомпьютерную эпоху они очень широко использовались для вычисления произведения не начиная с младшего разряда, как при обычном школьном "умножении столбиком", а начиная со старшего разряда, как это принято в теории приближенных вычислений[9].

5.1. Определите функцию, которая сопоставляет числу реверсированное к нему число.

Ответ. Ну, конечно, это

```
rev[n_]:=FromDigits[Reverse[IntegerDigits[n]]]
```

5.2. Найдите все числа $\leq 10^7$, которые в 4 раза меньше своего реверсированного.

Ответ. Вычисляя

```
Select[Range[10^7],FromDigits[Reverse[IntegerDigits[#]]]==4*#&]
```

мы видим, что таких чисел ровно 4, а именно 2178, 21978, 219978, 2199978, причем все они получаются врисовыванием девяток внутрь первого из них.

5.3. Найдите все числа $\leq 10^7$, которые в 9 раз меньше своего реверсированного.

Ответ. В данном случае ответ таков: 1089, 10989, 109989, 1099989.

5.4. Может ли число быть в 2, 3, 5, 6, 7 или 8 раз своего реверсированного?

Число n, совпадающее со своим реверсированным, называется **палиндромическим** или, коротко, **палиндромом**. Иными словами, палиндромическое число $a_1 \dots a_n$ читается одинаково слева направо и справа налево.

[9] Я.С.Безикович, Приближенные вычисления. — ГИТТЛ, Л.–М., 1941, 290с.; см. с.60–64.

5.5. Задайте тест, который проверяет, является ли число палиндромом.

Ответ. Функция `rev` уже определена, поэтому:

```
palinQ[n_]:=TrueQ[rev[n]==n]
```

5.6. Ясно, что $11^0 = 1$, $11^1 = 11$, $11^2 = 121$, $11^3 = 1331$, $11^4 = 14641$ (бином Ньютона). Дальше из-за переноса разрядов симметрия нарушается, например, $11^5 = 161051$. Есть ли среди степеней 11 еще палиндромы?

Следующие обобщения этой задачи объясняют, что происходит при решении уравнений $\mathrm{rev}(x^m) = \mathrm{rev}(x)^m$

5.7. Проверьте, что квадрат и куб числа 111 являются палиндромами, а остальные степени не являются.

5.8. Квадраты чисел 1111, 11111, 111111, 1111111, 11111111, 111111111 являются палиндромами, а остальные степени не являются.

5.9. Квадраты чисел $11\ldots11$ состоящих более, чем из 10 единиц, не являются палиндромами.

Ответ. Они заканчиваются цифрами 987654321, а начинаются цифрами 12345679.

Следующая задача предлагалась на вологодской областной олимпиаде 1994 года по программированию[10].

5.10. Найдите палиндромы $\leq 10^5$, квадраты которых тоже являются палиндромами.

Ответ. Кроме 1, 2 и 3, уже изученных нами чисел 11, 111, 1111, 11111, состоящих их одних 1, и еще двух очевидных вариаций на тему $11^2 = 121$, а именно, $22^2 = 484$ и $121^2 = 14641$, имеется еще ровно 12 примеров:

$$\begin{array}{ccc} 101^2 = 10201 & 202^2 = 40804 & 212^2 = 44944 \\ 1001^2 = 1002001 & 2002^2 = 4008004 & 10001^2 = 100020001 \\ 10101^2 = 102030201 & 10201^2 = 104060401 & 11011^2 = 121242121 \\ 11211^2 = 125686521 & 20002^2 = 400080004 & 20102^2 = 404090404 \end{array}$$

Этот ответ носит общий характер. Сумма квадратов цифр такого палиндрома не превосходит 9. Таким образом, палиндром с $m > 1$ разрядами может содержать только 0, 1 и 2.

• При четном m палиндром либо соддержит не более 8 единиц, либо начинается и заканчивается двойкой, а все остальные цифры нули.

• При нечетном m палиндром либо соддержит не более 9 единиц, либо в середине стоит двойка и он содержит не более 4 единиц, либо он начинается и заканчивается двойкой, все остальные цифры, кроме, быть может, средней, нули, а средняя цифра может равняться единице.

---

[10] А.С.Сипин, А,И.Дунаев, Областные олимпиады по информатике. — Изд-во Русь, Вологда, 1995, с.1–95.

5.11. Найдите палиндромы $\leq 10^5$, кубы которых тоже являются палиндромами.

Ответ. Кроме 1, 2, $7^3 = 343$ и уже изученных нами чисел 11 и 111 имеется ровно пять примеров

$$101^3 = 1030301, \quad 1001^3 = 1003003001, \quad 10001^3 = 1000300030001,$$
$$10101^3 = 1030607060301, \quad 11011^3 = 1334996994331$$

Чуть позже мы объясним этот ответ.

5.12. Найдите палиндромические простые $p \leq 10000$.

5.13. Найдите первые двести пар $(p, \mathrm{rev}(p))$, где как число p, так и реверсированное к нему число $\mathrm{rev}(p)$ оба простые, причем $p < \mathrm{rev}(p)$.

5.14. Найдите все простые $< 10^6$ такие, что каждая циклическая перестановка цифр в них снова дает простое.

Ответ. Мы не будем приводить соответствующий код, а ограничимся частью ответа. Имеется четыре группы трехзначных простых с этим свойством:

$$113,\ 131,\ 311, \qquad 197,\ 719,\ 971, \qquad 199,\ 919,\ 991, \qquad 337,\ 373,\ 733,$$

Обратите внимание, что те из них, которые имеют две одинаковых цифры, остаются простыми вообще при любой перестановке цифр! Имеется две группы четырехзначных

$$1193,\ 1931,\ 3119,\ 9311, \qquad 3779,\ 7937,\ 7793,\ 9377,$$

и для группы пятизначных чисел:

$$11939,\ 19391,\ 39119,\ 91193,\ 93911, \qquad 19937,\ 37199,\ 71993,\ 93719,\ 99371.$$

Авторы не хотят лишать читателей удовольствия самостоятельно найти шестизначные числа с этим свойством.

В следующей задаче предлагается решить уравнение $\mathrm{rev}(x^2) = \mathrm{rev}(x)^2$.

5.15. Пусть $(a_1 \dots a_m)^2 = b_1 \dots b_n$. Найдите все числа ≤ 10000, для которых $(a_m \dots a_1)^2 = b_n \dots b_1$.

Указание. Ясно, что дорисовывание нулей в конце числа не меняет этого свойства, поэтому рассматривайте только приведенные решения, последняя цифра которых $\neq 0$.

5.16. Решите уравнение $\mathrm{rev}(x^3) = \mathrm{rev}(x)^3$.

Ответ. Кроме 7 все приведенные решения этого уравнения являются решениями предыдущего и, кроме 2, все состоят их фрагментов вида 1 и 11, разделенных нулями и фрагментов вида 111 отделенных с каждой стороны по крайней мере двумя нулями. Приведем список всех приведенных решений меньших миллиона: 1, 2, 7, 11, 101, 111, 1001, 1011, 1101, 10001, 10011,

10101, 11001, 11011, 100001, 100011, 100101, 100111, 101001, 101011, 101101, 110001, 110011, 110101, 111001.

5.17. Решите уравнение $\mathrm{rev}(x^4) = \mathrm{rev}(x)^4$.

Ответ. Единственными приведенными решениями этого уравнения являются числа 1, 11, 101, 1001, 10001, ... Все остальные решения получаются из этих дописыванием нулей.

5.18. Решите уравнение $\mathrm{rev}(x^5) = \mathrm{rev}(x)^5$.

Ответ. Единственным приведенным решением этого уравнения является 1. Интересно, что множество решений каждого из этих уравнений содержится в множестве решений предыдущего.

Следующая простенькая задача предлагалась в 2000 году в качестве утешительной на уральском четвертьфинале студенческой олимпиады по программированию.

5.19. При повороте на π цифры 0, 1 и 8 не меняются, цифры 6 и 9 переходят друг в друга, а все остальные цифры становятся бессмысленными. Среди двузначных чисел при вращении на π не меняются 11, 69, 88 и 96. Найдите все n-значные числа, которые не меняются при таком вращении.

5.20. Существуют ли простые числа, кроме состоящих из одних единиц, которые не меняются при вращении на π?

§ 6. Закон старшего разряда

Но, желаньем подстрекаем
Их сюрпризом удивить,
Не давай, подлец, быка им
В виде опыта доить.

Алексей Константинович Толстой, *Мудрость жизни*

Сейчас произойдет нечто совершенно удивительное. Казалось бы, у случайного числа все должно быть случайно, в том числе и первая цифра.

6.1. Исследуйте поведение первой цифры чисел 2^n, $n \in \mathbb{N}$.

Решение. Следующий код вычисляет, сколько раз каждая из цифр 1,2, ... ,9 встречается как первая цифра среди первых n степеней двойки:

```
firstdigit[n_]:=Table[Count[
            Table[First[IntegerDigits[2^m]],{m,1,n}],
                                                i],{i,1,9}]
```

Ответ потрясает воображение. То, что среди первых цифр первых 10 степеней двойки встречается три единицы, кажется случайностью. Однако, взгляд на первые цифры первых 100 степеней

30, 17, 13, 10, 7, 7, 6, 5, 5

делает закономерности в распределении цифр очевидными. При рассмотрении первых цифр первой 1000 степеней

301, 176, 125, 97, 79, 69, 56, 52, 45

подозрения переходят в уверенность. Оказывается, дальше распределение первых цифр практически не меняется. Вот как расположены первые цифры в нескольких последовательных отрезках по 10000 степеней:

$1-10000$	3010	1761	1249	970	791	670	579	512	458
$10001-20000$	3010	1762	1249	969	793	669	579	512	457
$20001-30000$	3010	1760	1250	969	791	670	581	511	458
$30001-40000$	3011	1761	1249	969	792	668	581	511	458
$40001-50000$	3010	1760	1251	969	791	670	580	512	457
$50001-60000$	3010	1762	1248	969	793	670	579	512	457
$60001-70000$	3011	1760	1250	969	791	670	580	511	458
$70001-80000$	3010	1763	1248	969	793	668	580	512	457
$80001-90000$	3010	1760	1250	969	792	671	579	512	457
$90001-100000$	3010	1762	1248	970	792	669	579	511	459

Не пытайтесь повторить этот смертельный номер. Вычисление этой таблицы на бытовом компьютере легко может занять больше часа.

Если Вы думаете, что это характерно именно для степеней 2, Вас ждет следующий сюрприз.

6.2. Исследуйте поведение первой цифры чисел 3^n, $n \in \mathbb{N}$.

Ответ. Среди первых цифр первых 10 степеней тройки цифра 2 встречается три раза, $3^3 = 27$, $3^5 = 243$, $3^7 = 2187$, а цифра 1 — всего один раз, $3^9 = 19683$. Однако, дальше все постепенно становится на свои места. Вот распределение первых цифр среди первых 100 степеней, первой 1000 степеней и первых 10000 степеней:

28, 19, 12, 8, 9, 7, 7, 5, 5;

300, 177, 123, 98, 79, 66, 59, 52, 46;

3007, 1764, 1247, 968, 792, 669, 582, 513, 458.

Обратите внимание, что из-за аномального поведения в первом десятке цифра 1 все еще встречается чуть реже, а 2 — чуть чаще, чем должны.

6.3. Исследуйте поведение первой цифры чисел m^n, где $n \in \mathbb{N}$, а m не является степенью 10.

Теперь Вы, конечно, больше не думаете, что этот закон применим только к степеням?

6.4. Исследуйте поведение первой цифры чисел Фибоначчи F_n, где $n \in \mathbb{N}$.

Ответ. Да в точности то же самое. Вот распределение первых 100, 1000 и 10000 чисел Фибоначчи по первой цифре:

30, 18, 13, 9, 8, 6, 5, 7, 4

301, 177, 125, 96, 80, 67, 56, 53, 45

3011, 1762, 1250, 968, 792, 668, 580, 513, 456

6.5. Исследуйте поведение первой цифры факториала $n!$, где $n \in \mathbb{N}$.

Конечно, такое точное совпадение ответов для разных классов чисел не может быть случайным. И действительно, **закон Ньюкомба**, известный также как **закон старшего разряда**, утверждает, что цифра d встречается в качестве ведущей с вероятностью $\log_{10}(1 + 1/d)$. Вот значения логарифма с точностью до 6 значащих цифр:

$$0.301030, \quad 0.176091, \quad 0.124939, \quad 0.0969100, \quad 0.0791812,$$
$$0.0669468, \quad 0.0579919, \quad 0.0511525, \quad 0.0457575$$

Как отмечает Арнольд, это объясняет многие реально наблюдаемые явления, связанные с экспоненциальным ростом, например, почему численность населения примерно одной страны из трех начинается с 1. То же относится к основным физическим постоянным — в любой системе единиц!

> I do not know as much as God, but I know as much as God did at my age.
>
> Milton Shulman

Глава 2. РАЦИОНАЛЬНЫЕ ЧИСЛА

Уровень научного образования во всех странах мира неуклонно снижается, а Россия в этом общемировом процессе, как и в других, отстает. Например, некоторые наши школьники до сих пор свободно складывают дроби, тогда как все американские *студенты* и 80% школьных учителей давно уже думают, что $1/2+1/3 = 2/5$.

В.И.Арнольд, *Что такое математика*?

В настоящей главе мы обсуждаем основы вычислений с рациональными числами. Поле $\mathbb{Q}$ рациональных чисел является полем частных кольца $\mathbb{Z}$ целых чисел и, таким образом, с одной стороны вычисления в нем сводятся к вычислениям с целыми числами, а, с другой стороны, интерпретация в терминах дробей позволяет гораздо проще проводить многие вычисления с целыми числами.

§ 1. Рациональные числа

Can you do Division? Divide a loaf by a knife — what's the answer to that?

Lewis Carroll, *Through the looking glass*

Рациональное число x представляется как частное двух целых чисел m/n, $n \neq 0$, при этом автоматически производятся все сокращения, которые система в состоянии найти.

`Numerator[x]`	числитель x
`Denominator[x]`	знаменатель x

Таким образом, `Numerator[m/n]` и `Denominator[m/n]` возвращают не m и n, а $m/\gcd(m,n)$ и $n/\gcd(m,n)$, причем знак относится к числителю.

1.1. Как известно, многие школьники сокращают правильные дроби, зачеркивая одинаковую цифру в числителе и знаменателе. Например, зачеркивая 6 в $26/65$ мы получаем $2/5$. Найдите все правильные дроби со знаменателями < 1000, которые можно сокращать таким образом.

Ответ. Вот все такие дроби с двузначными числителями и знаменателями:

$$\frac{16}{64} = \frac{1}{4}, \qquad \frac{19}{95} = \frac{1}{5}, \qquad \frac{26}{65} = \frac{2}{5}, \qquad \frac{49}{98} = \frac{4}{8}.$$

Вот такие дроби с двузначным числителем и трехзначным знаменателем:

$$\frac{13}{325}=\frac{1}{25}\quad \frac{27}{756}=\frac{2}{56}\quad \frac{34}{136}=\frac{4}{16}\quad \frac{34}{238}=\frac{4}{28}\quad \frac{39}{195}=\frac{3}{15}$$
$$\frac{39}{975}=\frac{3}{75}\quad \frac{49}{196}=\frac{4}{16}\quad \frac{49}{294}=\frac{4}{14}\quad \frac{49}{392}=\frac{4}{32}\quad \frac{59}{295}=\frac{5}{25}$$
$$\frac{67}{268}=\frac{7}{28}\quad \frac{67}{469}=\frac{7}{49}\quad \frac{79}{395}=\frac{7}{35}\quad \frac{83}{332}=\frac{8}{32}\quad \frac{96}{192}=\frac{6}{12}$$
$$\frac{97}{194}=\frac{7}{14}\quad \frac{97}{291}=\frac{7}{21}\quad \frac{98}{196}=\frac{8}{16}\quad \frac{98}{294}=\frac{8}{24}\quad \frac{98}{392}=\frac{8}{32}$$

Что касается дробей с трехзначными числителями и знаменателями, то их уже довольно много. Кроме тривиальных примеров 101/202, 101/303, 102/204, 103/206, встречается и несколько более интересные примеры, скажем, $133/931 = 13/91$.

Как известно, многие школьники складывают дроби складывая их числители и знаменатели. Однако, *хороший* школьник знает, что так можно складывать только *несократимые* дроби. А именно, дробь

$$\frac{a}{b}\dot{+}\frac{c}{d}=\frac{a+c}{b+d}$$

называется **медиантой** дробей a/b и c/d.

1.2. Задайте медианту двух дробей.

Решение. Если ровно двух, то проще всего так:

```
medianta[x_,y_]:=(Numerator[x]+Numerator[y])/
                        (Denominator[x]+Denominator[y])
```

Однако, следует иметь в виду, что из-за возможных сокращений медианта не ассоциативна. Поэтому в природных условиях мы бы, скорее всего, определили медианту так:

```
medianta[x__]:=Total[Numerator[{x}]]/Total[Denominator[{x}]]
```

В 1816 году английский геолог Джон Фарей расположил все (несократимые) дроби $0 \le l/m \le 1$ со знаменателем $m \le n$ в порядке возрастания и высказал предположение, что каждый член этой последовательности является медиантой двух соседних. Получающаяся так последовательность дробей называется **последовательностью Фарея** порядка n.

1.3. Напишите команду, порождающую последовательность Фарея и убедитесь в справедливости гипотезы Фарея.

Решение. Ну, например, так:

```
farey[n_]:=Union[Flatten[Table[i/j,{j,1,n},{i,0,j}],1]]
```

Теперь вычисление

```
Do[Print[farey[n]],{n,1,6}]
```

Вернет нам ответ

```
{0,1}
{0,1/2,1}
{0,1/3,1/2,2/3,1}
{0,1/4,1/3,1/2,2/3,3/4,1}
{0,1/5,1/4,1/3,2/5,1/2,3/5,2/3,3/4,4/5,1}
{0,1/6,1/5,1/4,1/3,2/5,1/2,3/5,2/3,3/4,4/5,5/6,1}
```

Разумеется, 0 истолковывается как $0/1$.

§ 2. Гармонические числа

Из курса анализа хорошо известен гармонический ряд:

$$1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\frac{1}{5}+\dots$$

Его частичные суммы называются **гармоническими числами**:

$$H_n = 1+\frac{1}{2}+\frac{1}{3}+\dots+\frac{1}{n}.$$

Вот несколько первых гармонических чисел:

$$H_1=1,\ H_2=\frac{3}{2},\ H_3=\frac{11}{6},\ H_4=\frac{25}{12},\ H_5=\frac{137}{60},\ H_6=\frac{49}{20},$$
$$H_7=\frac{363}{140},\ H_8=\frac{761}{280},\ H_9=\frac{7129}{2520},\ H_{10}=\frac{7381}{2520}.$$

По самому определению n-е гармоническое число рационально. Являясь дискретным аналогом логарифма, гармонические числа настолько часто возникают в комбинаторных задачах и при анализе алгоритмов, что в систему встроена специальная функция `HarmonicNumber` для их быстрого вычисления.

`HarmonicNumber[n]`	n-е гармоническое число
`EulerGamma`	константа Эйлера

Следующее упражнение основано на том, что знаменатель H_n является делителем $\operatorname{lcm}(2,\dots,n)$ — и, тем более, $n!$. Таким образом, все числа $\operatorname{lcm}(2,\dots,n)H_n$ — и, тем более, $n!H_n$ — целые.

2.1. Убедитесь, что при $n>1$ число H_n не может быть целым.

Указание. Посмотрите на числа $\operatorname{lcm}(2,\dots,n)H_n$, все они нечетны. Это значит, что входящая в знаменатель H_n степень 2 никогда не сокращается.

2.2. Вычислите таблицу первых 100 гармонических чисел. Найдите те простые, на которые в них происходит сокращение.

Посмотрим теперь на числители гармонических чисел. Числитель H_2 делится на 3, но не на 3^2. Зато числитель H_4 делится на 5^2, числитель H_6 — на 7^2, а числитель H_{10} — на 11^2. Вообще, **теорема Вольстенхольма** утверждает, что для любого простого $p > 3$ числитель H_{p-1} делится на p^2.

2.3. Проверьте теорему Вольстенхольма для первой тысячи простых.

Решение. Можно так.

```
Apply[And,Table[TrueQ[
   Mod[Numerator[HarmonicNumber[Prime[i]-1]],Prime[i]^2]==0],
                                              {i,3,1002}]]
```

Если мы интересуемся только целыми частями, n-е гармоническое число довольно точно аппроксимирует $\ln(n)$ — или, в зависимости от точки зрения, аппроксимируется $\ln(n)$.

2.4. Оцените разницу $\ln(n)$ и H_n для $n \le 1000$. Убедитесь, что всегда

$$\ln(n) < H_n < \ln(n) + 1.$$

В действительности существует предел $H_n - \ln(n)$ при $n \longrightarrow \infty$. Этот предел обозначается γ и называется **константой Эйлера**. Вычисление `N[EulerGamma,50]` дает первые 50 знаков константы Эйлера:

$$0.57721\,56649\,01532\,86060\,65120\,90082\,40243\,10421\,59335\,93994.$$

Однако, если нас интересуют знаки после запятой, то для небольших значений n (порядка миллионов или миллиардов) даже $\ln(n)+\gamma$ все еще является не очень хорошим приближением к H_n. Следующая формула для H_n

$$H_n = \ln(n) + \gamma + \frac{1}{2n} - \frac{1}{12n^2} + \frac{\varepsilon}{120n^4}$$

где $0 \le \varepsilon \le n$, позволяет получить гораздо лучшее приближение.

Как доказал Эйлер, ряд

$$\sum_{i=1}^{\infty} \frac{1}{p_i},$$

где p_i обозначает i-е простое число, расходится.

2.5. Сравните два следующих кода:

```
N[Sum[1/Prime[i],{i,1,PrimePi[10^n]}]]
Sum[N[1/Prime[i]],{i,1,PrimePi[10^n]}]
```

В чем их отличие? Какой из них дает более надежный результат? Какой дольше вычисляется? А теперь проведите эксперимент для небольших значений n, скажем, $n = 5, 6, 7$.

Если гармонический ряд расходится логарифмически, то этот ряд расходится еще гораздо медленнее, примерно как $\ln(\ln(n))$! Иными словами, никто никогда не видел — и в течение ближайшего столетия скорее всего и не увидит — никаких его значений, больших чем 7.

2.6. Пронаблюдайте динамику частичных сумм ряда $\displaystyle\sum_{i=1}^{\infty} \frac{1}{p}$ при небольших n, скажем $n = m10^4$, $1 \le m \le 10$. При каком n эта сумма превзойдет 3?

§ 3. Десятичные дроби

> In the Middle Age, in Germany, if you wanted to learn addition and multiplication, you could go to any university. But if you wanted to learn division, you could only do it in one place, Heidelberg.
>
> Israel Gelfand

В настоящем параграфе мы изучим запись рациональных чисел так называемыми `бесконечными десятичными дробями`. Как хорошо известно, никаких бесконечных десятичных дробей не существует. То есть, конечно, они существуют как последовательности цифр, но вот только ни складывать, ни умножать их никто не умеет, поэтому вся излагаемая в школьном курсе математики "теория" вещественных чисел, основанная на использовании бесконечных десятичных дробей, абсолютно бессмысленна.

Что, однако, существует, это *конечные* десятичные дроби и *периодические* десятичные дроби, которыми выражаются рациональные числа. Операции над конечными дестяичными дробями сразу сводятся к операциям над целыми числами. В настоящем параграфе мы постараемся придать смысл операциям над периодическими дробями. Дело это называется **теорией сравнений** и обычно изучается не в школьном курсе арифметики, а в университетском курсе теории чисел. Например, в школьном курсе никогда не доказывается, что рациональное число записывается *периодической* дробью. В действительности этот факт составляет содержание **теоремы Эйлера**, которая, кроме того, утверждает, что период несократимой дроби со знаменателем n не превосходит $\varphi(n)$, где φ некоторая арифметическая функция, называемая **функцией Эйлера**.

В следующий главе мы обсудим, как работают команды `RealDigits` и `FromDigits` для приближенных вещественных чисел. Однако для рациональных чисел их использование имеет некоторую специфику.

`RealDigits[x]`	предпериод и период рационального числа x
`FromDigits[{list,m}]`	восстановление числа по списку цифр

Говорят, что число x изображается **периодической** десятичной дробью, если найдется такое натуральное l, что $10^l x - x$ выражается конечной десятичной дробью. Это значит, что начиная с некоторого места эта дробь состоит из бесконечно повторяющегося блока цифр длины l. Наименьшее такое l называется длиной периода, а сам повторяющийся фрагмент длины l — **периодом** этой дроби. Однако десятичная запись числа не обязана начинаться с периода, ему может предшествовать **предпериод**. Дробь, начинающаяся сразу с периода, называется **чисто периодической**.

Например, дробь $5/12 = 0.416666666\ldots$ имеет предпериод длины 2, и период — длины 1. Как мы уже знаем из предыдущей главы, дробь $1/7 = 0.1428571428571428571\ldots$ является чисто периодической и имеет период 142857 длины 6.

Разложение рационального числа x в конечную/периодическую десятичную дробь получается применением к нему команды `RealDigits[x]`. Ответ имеет следующий формат.

• Для числа $x = m/n$, в знаменатель которого входят только степени 2 и 5, ответ имеет вид {{`x1, ... ,xk`},`r`}, где $x_1, \dots, x_k$ предпериод (кроме начальных нулей), а r — десятичная экспонента. Иными словами, $10^r \cdot 0.x_1 \dots x_k$.

• Для любого другого рационального числа $x = m/n$ ответ имеет вид {{`x1, ... ,xk,{y1, ... ,yl}`},`r`}, где $x_1, \dots, x_k$ и r имеют тот же смысл, что и раньше, а $y_1, \dots, y_l$ период десятичной записи x.

3.1. Убедитесь, что длина периода несократимой дроби m/n равна наименьшему l, для которого найдется такое натуральное k, что знаменатель n этой дроби делит $10^k(10^l - 1)$.

3.2. В условиях предыдущей задачи убедитесь, что длина предпериода m/n равна наименьшему k такому, что n делит $10^k(10^l - 1)$.

3.3. Убедитесь, что длина предпериода и периода правильной дроби m/n не превосходит $\varphi(n)$. Когда здесь достигается равенство?

Командра `FromDigits` примененная к объекту формата {`list,n`} или формата возвращает конечную FromDigits[1, 1, 2, 3, 0]

§ 4. Египетские дроби

> Человек не хуже муравья может переносить тяжести в 20 раз больше собственного веса. Но за большее количество раз, и страшно матерясь.
>
> Николай Фоменко

Древние египтяне представляли рациональное число как сумму целого и нескольких *различных* дробей с числителем равным 1. Такие выражения принято называть **египетскими дробями**. Вот первые интересные примеры египетского разложения правильных дробей:

$$\frac{3}{7} = \frac{1}{3} + \frac{1}{11} + \frac{1}{231}, \qquad \frac{5}{7} = \frac{1}{2} + \frac{1}{5} + \frac{1}{70}, \qquad \frac{6}{7} = \frac{1}{2} + \frac{1}{3} + \frac{1}{42}.$$

Следующая задача содержится в знаменитом папирусе Райнда[11], хранящемся в Британском музее.

4.1. Найдите натуральное n такое, что

$$\frac{2}{73} = \frac{1}{60} + \frac{1}{219} + \frac{1}{292} + \frac{1}{n}.$$

Каждое рациональное число может быть бесконечным числом способов записано как египетская дробь, но для любого m среди этих записей лишь

[11] A.B.Chace, H.P.Manning, R.C.Archibald, The Rhind mathematical papyrus. vol.I. — Berlin, 1929.

конечное число состоит ровно из m слагаемых[12]. Сейчас мы попробуем описать несколько алгоритмов разложения рационального числа в египетскую дробь.

Проще всего реализовать **жадный алгоритм**, [Wa]. Так называется алгоритм, выбирающий на каждом шаге *наибольшее* слагаемое, которое может войти в египетскую дробь.

Для реализации этого алгоритма нам понадобится важная вспомогательная функция, сопоставляющая списку его **список разностей**.

4.2. Определите функцию, которая по списку длины n порождает список длины $n-1$, состоящий из попарных разностей соседних членов исходного списка.

Решение. Вот совсем топорное решение:

```
differences[x_]:=Table[x[[i]]-x[[i+1]],{i,1,Length[x]-1}]
```

Зная, что арифметические операции распределяются по спискам, можно дать гораздо более изящную конструкцию:

```
differences[x_]:=Most[x]-Rest[x]
```

Из общих соображений кажется, что этот код улучшить невозможно, но в действительности следующее определение еще лучше и исполняется чуть быстрее:

```
differences[x_]:=Apply[Subtract,Partition[x,2,1],{1}]
```

4.3. Определите функцию, которая сопоставляет рациональному числу x его разность с наибольшим египетским слагаемым.

Решение. Если число целое или его числитель равен 1, то остаток равен 0, если оно рациональное вне отрезка $[0,1]$, то нужно вычесть целую часть. Остается понять, что в случае, когда $x \in (0,1)$ итз него нужно вычесть $1/\lceil 1/x \rceil$. Все это можно резюмировать так:

```
greedypart[x_]:=Which[IntegerQ[x],0,Numerator[x]==1,0,
            x<0||x>1,x-Floor[x],True,x-1/Ceiling[1/x]]
```

При предыдущих условиях `Ceiling[1/x]` совпадает с `1+Floor[1/x]` или, что то же самое с `1+Quotient[1,x]`

4.4. Закончите программу разложения рационального числа в египетскую дробь по жадному алгоритму.

Решение. Ну, например, так:

```
greedylist[x_]:=Most[FixedPointList[greedypart,x]]
```

```
greedyegypt[x_]:=differences[greedylist[x]]
```

К сожалению, даже в случае небольших числителей и знаменателей жадный алгоритм часто дает *очень* плохие разложения. Так, уже

$$\frac{17}{19} = \frac{1}{2} + \frac{1}{3} + \frac{1}{17} + \frac{1}{388} + \frac{1}{375972}$$

[12]I.Stewart. The riddle of the vanishing camel. Scientific American, June 1992, p.122–124.

выглядит подозрительно, а при разложении 31/311 получаются знаменатели, содержащие больше 500 цифр! Хватать все, что подвернется под руку, не всегда является лучшей стратегией. Имеется много других алгоритмов, которые дают более короткие разложения с гораздо меньшими знаменателями.

Часто выгоднее предполагать, что знаменатели с самого начала быстро растут. Вот два классических алгоритма такого рода, гарантирующие, кроме того, единственность представления (по поводу доказательства см. [GCh]).

4.5. Алгоритм Фибоначчи: любое рациональное число $0 < x < 1$ единственным образом представляется в виде

$$x = \frac{1}{n_1} + \frac{1}{n_2} + \ldots + \frac{1}{n_s},$$

где $n_1 \geq 2$ и $n_{i+1} > n_i^2 - n_i$ для всех i. Реализуйте алгоритм Фибоначчи.

4.6. Алгоритм Остроградского: любое рациональное число $0 < x < 1$ единственным образом представляется в виде

$$x = \frac{1}{n_1} + \frac{1}{n_1 n_2} + \ldots + \frac{1}{n_1 n_2 \ldots n_s},$$

где $n_1 \geq 2$ и $n_{i+1} \geq n_i$ для всех i. Реализуйте алгоритм Остроградского.

4.7. Всегда ли алгоритм Фибоначчи и алгоритм Остроградского дают один и тот же результат?

4.8. Найдите сумму s египетских дробей

$$x = \frac{1}{n_1} + \frac{1}{n_2} + \ldots + \frac{1}{n_s} < 1$$

такую, что между x и 1 нет никаких других сумм s египетских дробей.

Ответ. Это сумма первых s членов ряда

$$\frac{1}{2} + \frac{1}{3} + \frac{1}{7} + \frac{1}{43} + \frac{1}{1807} + \ldots,$$

в котором знаменатель каждого члена равен произведению знаменателей всех предыдущих членов $+1$.

4.9. Ясно, что

$$\frac{1}{n} = \frac{1}{2n} + \frac{1}{2n},$$

Утверждается, что при $n \geq 2$ египетская дробь $1/n$ представляется как сумма двух *различных* египетских дробей:

$$\frac{1}{n} = \frac{1}{x} + \frac{1}{y}, \qquad x < y.$$

Например,

$$\frac{1}{2}=\frac{1}{3}+\frac{1}{6},\qquad \frac{1}{3}=\frac{1}{4}+\frac{1}{12},\qquad \frac{1}{4}=\frac{1}{5}+\frac{1}{20}=\frac{1}{6}+\frac{1}{12},\qquad \frac{1}{5}=\frac{1}{6}+\frac{1}{30},$$

и так далее. Ясно, что почти все эти разложения являются частными случаями тождества

$$\frac{1}{n}=\frac{1}{n+1}+\frac{1}{n(n+1)},$$

но вот второе разложение для $1/4$ так не получается. Найдите все такие разложения.

Следующая задача, предложенная Эрдешем и Штраусом, обсуждается в классической книге Морделла[13].

4.10. Верно ли, что для любого $n>1$ уравнение

$$\frac{4}{n}=\frac{1}{x}+\frac{1}{y}+\frac{1}{z},$$

имеет решение в натуральных числах?

Ответ. Это верно, по крайней мере для всех $n\le 10^7$. Кроме того, в цитированной книге Морделла доказано, что это вообще всегда так, за исключением, возможно, случая, когда n простое число сравнимое с 1, 121, 169, 289, 361 или 529 по модулю 840.

4.11. Проверьте справедливость утверждения предыдущей задачи для первых нескольких тысяч простых, удовлетворяющих этим сравнениям.

Указание. Составляя список при помощи команды `Select` следует ли выбирать числа, удовлетворяющие сравнениям, из списка простых или простые из списка чисел, удовлетворяющих сравнениям?

4.12. Верно ли, что для любого $n>1$ уравнение

$$\frac{5}{n}=\frac{1}{x}+\frac{1}{y}+\frac{1}{z},$$

имеет решение в натуральных числах?

4.13. Верно ли, что для любого m существует такое $n(m)$, что для любого $n>n(m)$ уравнение

$$\frac{m}{n}=\frac{1}{x}+\frac{1}{y}+\frac{1}{z},$$

имеет решение в натуральных числах?

4.14. Можно ли в предыдущей задаче утверждать, что всегда $n(m)\le m$?

[13] L.J.Mordell, Diophantine equations. — Acad. Press, London et al., 1969.

§ 5. Числа Бернулли

Числа Бернулли B_n естественно возникают во многих теоретико-числовых, алгебраических и комбинаторных результатах, а также в классическом анализе. Вот одно из их простейших определений:

$$\frac{t}{e^t-1}=\sum_{n=0}^{\infty}B_n\frac{t^n}{n!}$$

Числа B_n были независимо введены Яковом Бернулли и Такакадзу Секи Кова для вычисления сумм $1^m+2^m+\ldots+n^m$. Сегодня большинство начинающих впервые видят числа Бернулли при вычислении коэффициентов рядов Тэйлора тригонометрических и гиперболических функций. Кроме того, многие алгоритмы используют численные значения B_n. Поэтому в системе встроена функция `BernoulliB`, возвращающая числа Бернулли.

`BernoulliB[n]`	число Бернулли B_n

5.1. Составьте таблицу первых 31 чисел Бернулли. Что Вам сразу бросается в глаза?

Решение. Бесхитростное вычисление

```
Table[BernoulliB[n],{n,0,10}]
```

дает

$$B_0=1,\quad B_1=-\frac{1}{2},\quad B_2=\frac{1}{6},\quad B_3=0,\quad B_4=-\frac{1}{30},\quad B_5=0$$
$$B_6=\frac{1}{42},\quad B_7=0,\quad B_8=-\frac{1}{30},\quad B_9=0,\quad B_{10}=\frac{5}{66}$$

Все дальнейшие числа Бернулли B_{2n+1} с нечетными номерами тоже равны 0, поэтому ограничимся значениями чисел B_{2n} с четными номерами.

$$B_{12}=-\frac{691}{2730},\quad B_{14}=\frac{7}{6},\quad B_{16}=-\frac{3617}{510},\quad B_{18}=\frac{43867}{798},$$
$$B_{20}=-\frac{174611}{330},\quad B_{22}=\frac{854513}{138},\quad B_{24}=-\frac{236364091}{2730},$$
$$B_{26}=\frac{8553103}{6},\quad B_{28}=-\frac{23749461029}{870},\quad B_{30}=\frac{8615841276005}{14322}$$

Трудно не заметить, что знаки чисел Бернулли B_{2n} чередуются в зависимости от четности n: числа Бернулли B_{4n} отрицательны, а B_{4n+2} — положительны.

Знаменитая **теорема фон Штаудта** утверждает, что дробная часть чисел $(-1)^nB_{2n}$ чрезвычайно естественно выражается в терминах суммы египетских дробей, зависящих только от n. В частности, числа Бернулли рациональны.

5.2. Убедитесь, что число B_{2n} является разностью некоторого целого числа и суммы всех египетских дробей вида $1/p$, где p пробегает те простые числа, которые на 1 больше какого-то делителя числа $2n$.

Ответ. Ограничимся несколькими небольшими примерами[14]:

$$B_2 = \frac{1}{6} = 1 - \frac{1}{2} - \frac{1}{3}$$

$$B_4 = -\frac{1}{30} = 1 - \frac{1}{2} - \frac{1}{3} - \frac{1}{5}$$

$$B_6 = \frac{1}{42} = 1 - \frac{1}{2} - \frac{1}{3} - \frac{1}{7}$$

$$B_8 = -\frac{1}{30} = 1 - \frac{1}{2} - \frac{1}{3} - \frac{1}{5}$$

$$B_{10} = \frac{5}{66} = 1 - \frac{1}{2} - \frac{1}{3} - \frac{1}{11}$$

$$B_{12} = -\frac{691}{2730} = 1 - \frac{1}{2} - \frac{1}{3} - \frac{1}{5} - \frac{1}{7} - \frac{1}{13}$$

$$B_{14} = \frac{7}{6} = 2 - \frac{1}{2} - \frac{1}{3}$$

$$B_{16} = -\frac{3617}{510} = -6 - \frac{1}{2} - \frac{1}{3} - \frac{1}{5} - \frac{1}{17}$$

$$B_{18} = \frac{43867}{798} = 56 - \frac{1}{2} - \frac{1}{3} - \frac{1}{7} - \frac{1}{19}$$

$$B_{20} = -\frac{174611}{330} = -528 - \frac{1}{2} - \frac{1}{3} - \frac{1}{5} - \frac{1}{11}$$

$$B_{22} = \frac{854513}{138} = 6193 - \frac{1}{2} - \frac{1}{3} - \frac{1}{23}$$

$$B_{24} = -\frac{236364091}{2730} = -86579 - \frac{1}{2} - \frac{1}{3} - \frac{1}{5} - \frac{1}{7} - \frac{1}{11}$$

$$B_{26} = \frac{8553103}{6} = 1425518 - \frac{1}{2} - \frac{1}{3}$$

$$B_{28} = -\frac{23749461029}{870} = -27298230 - \frac{1}{2} - \frac{1}{3} - \frac{1}{5} - \frac{1}{19}$$

$$B_{30} = \frac{8615841276005}{14322} = 601580875 - \frac{1}{2} - \frac{1}{3} - \frac{1}{7} - \frac{1}{11} - \frac{1}{31}$$

O dear Ophelia, I am ill at these numbers.

William Shakespeare, *Hamlet*

[14]Смотри по этому поводу Г.Полиа, Г.Сеге, Задачи и теоремы из анализа, том II. — М., Наука, 1978, с.161.

Глава 3. ВЕЩЕСТВЕННЫЕ ЧИСЛА

Допущение, что над вещественными числами можно производить все операции согласно обычным формальным законам арифметических действий, считалось само собой разумеющимся вплоть до второй половины XIX века. Поэтому мы будем рассматривать тот факт, что обычные правила вычислений приложимы к вещественным числам, как аксиому.

Рихард Курант[15]

В школе математическими понятиями пользуются, не сомневаясь в их законности и очень часто не задаваясь даже вопросом, что же они, собственно, означают. Не зная, что такое вещественные числа, мы, тем не менее умеем с ними обращаться, т.е. складывать их, умножать и сравнивать по величине.

Виктор Петрович Хавин[16]

Что уж говорить о мучениях посредством $\varepsilon/2$ и $\sqrt{\delta}$, которым преподаватели математического анализа подвергают студентов первого курса из чистого садизма, без всякой необходимости, могущей сыграть роль смягчающего обстоятельства.

Анри Лебег[17]

Из несчетности множества самих вещественных чисел следует даже, что не существует языка, в котором каждое вещественное число имело бы имя. Такая вещь, как, например, бесконечное десятичное разложение, не может, конечно, рассматриваться как *имя* соответствующего вещественного числа, поскольку бесконечное десятичное разложение не может даже быть полностью выписано или включено как часть в какое-нибудь фактически выписанное или произнесенное суждение.

Алонзо Черч[18]

Когда во время **партсобрания** за окном трижды каркает ворона и **члены бюро** незаметно сплевывают через левое плечо или крестят под столом животы, это не проявление суеверия, временно омрачающего высшую форму человеческой деятельности, а искаженное переплетение древних психических феноменов, из которых самым поздним является крестное знамение.

Виктор Пелевин, *Зомбификация*

---

[15]Р.Курант, Курс дифференциального и интегрального исчисления. — М., Наука, 1967, 704с.

[16]В.П.Хавин, Дифференциальное и интегральное исчисление функций одной вещественной переменной. — Лань, СПб, 1998, 446с.

[17]А.Лебег, Об измерении величин, Изд.2. — Учпедгиз, М., 1960, с.1–204; стр.41.

[18]А.Черч, Введение в математическую логику. т.I. – ИИЛ, М.,1960, с.1–484, стр.354.

В настоящей главе мы обсуждаем основы вычислений с вещественными числами. Среди прочих тем мы рассматриваем алгебраические и трансцендентные числа, десятичное представление вещественного числа, непрерывные дроби, основные константы и элементарные функции. Основной пафос систем компьютерной алгебры состоит именно в том, что при помощи них можно производить вычисления бесконечной точности. Однако, в некоторых приложениях, а также при дискретизации вывода (графики или звука), приходится пользоваться приближенными вычислениями, так что мы затрагиваем основные связанные с ними понятия.

§ 1. Точные вещественные числа

> Вредно распространение между людьми мыслей о том, что наша жизнь есть произведение вещественных сил и находится в зависимости от этих сил. Но когда такие ложные мысли называются науками и выдаются за святую мудрость человечества, то вред, производимый таким учением, ужасен.
>
> Лев Толстой, *Путь жизни*

В настоящей главе мы не будем пытаться объяснять, что такое вещественное число. Тем более, что как математики мы знаем, что никакого другого определения вещественного числа, кроме как элемента поля $\mathbb{R}$ вещественных чисел, не существует. *Определить* $\mathbb{R}$ совсем просто, это (единственное с точностью до изоморфизма) полное архимедово линейно упорядоченное поле — *построить* $\mathbb{R}$ несколько сложнее. Любое *индивидуальное* трансцендентное число является манифестацией актуальной бесконечности и — в строгом техническом смысле — столь же бесконечно, как множество *всех* рациональных чисел. Никакая математически последовательная трактовка вещественных чисел без признания этого основополагающего факта невозможна.

Излагаемая в школе "конструкция" $\mathbb{R}$ при помощи `бесконечных десятичных дробей` является, в действительности, чистой **профанацией**, так как она не позволяет осмысленным образом определить алгебраические операции над вещественными числами. Попробуйте, например, при помощи десятичных дробей *доказать*, что $2/7 + 5/7 = 1$ — не говоря уже про гораздо более сложное равенство $\sqrt{2}\sqrt{3} = \sqrt{6}$. Тот факт, что *невозможно* вычислить десятичное разложение суммы двух чисел по десятичным разложениям слагаемых, был замечен еще Дедекиндом, который считал доказательство равенства $\sqrt{2}\sqrt{3} = \sqrt{6}$ *как вещественных чисел* одним из своих главных математических достижений.

Любая попытка определить сумму двух вещественных чисел x и y по их десятичным разложениям сводится к тому, что эти разложения обрезаются до какого-то порядка, берутся суммы получившихся рациональных приближений, и, наконец, $x + y$ определяется как верхняя грань этих сумм. Ясно, что ЭТА ПРОЦЕДУРА НЕ ЯВЛЯЕТСЯ АЛГОРИТМОМ и ничем не отличается от определения $x + y$ по Дедекинду, как верхней грани множества

$$\{a + b \mid a, b \in \mathbb{Q}, a \le x, b \le y\}.$$

Тьюринг обратил внимание на то, что при этом, вообще говоря, нельзя найти *ни одной* десятичной цифры суммы/произведения, не зная *всех* цифр слагаемых/сомножителей. Иными словами, невозможность использовать бесконечные десятичные дроби для реальных вычислений состоит не в том, что нахождение *всех* цифр результата требует бесконечного количества операций, а в том, что уже нахождение *первой* цифры результата может потребовать бесконечного количества операций! В § 3 мы приводим пример, показывающий, что увеличение точности приближенного вычисления на *один* разряд может изменить *любое* количество предшествующих цифр.

На самом деле бесконечными десятичными дробями можно описать $\mathbb{R}$ лишь как упорядоченное множество, но не как поле. Внушаемая школьным курсом уверенность, что вещественное число *является* бесконечной десятичной дробью, представляет собой опасную иллюзию. Любая попытка осмысленным образом ввести операции над бесконечными десятичными дробями приводит к какой-то актуально бесконечной конструкции (Вейерштрасса, Кантора, Дедекинда), в которой иррациональные числа истолковываются как бесконечные множества рациональных чисел, классы таких множеств, или что-то в таком духе. Можно, конечно, пользоваться записью $\pi = 3,1415\ldots$, если понимать, что многоточие здесь не означает *ничего* сверх того, что $3,1415 < \pi < 3,1416$. Запись $\pi \approx 3,1416$ не означает даже этого.

Тем не менее, здесь мы встанем на наивную точку зрения существования каких-то вещественных чисел имеющих какие-то приближения, какие-то **десятичные цифры** — и даже будем производить над этими **десятичными цифрами** какие-то арифметические операции. Разумеется, фактически это значит, что в таких случаях мы будем производить вычисления в **кольце десятичных дробей**

$$\mathbb{Z} < \mathbb{Z}\Big[\frac{1}{10}\Big] = \mathbb{Z}\Big[\frac{1}{2}, \frac{1}{5}\Big] < \mathbb{Q}$$

— иными словами, вычисления с рациональными числами — но **вычисления неограниченной точности**. Конечно, вместо десятичных дробей, мы могли бы с тем же успехом пользоваться двоичными дробями $\mathbb{Z}\Big[\frac{1}{2}\Big]$, как большинство программистов, троичными дробями $\mathbb{Z}\Big[\frac{1}{3}\Big]$, как некоторые математики, или шестидесятиричными дробями $\mathbb{Z}\Big[\frac{1}{60}\Big] = \mathbb{Z}\Big[\frac{1}{2}, \frac{1}{3}, \frac{1}{5}\Big]$ (градусы, минуты, секунды, терции, ...), как астрономы.

Однако, принципиальным отличием систем компьютерной алгебры от *всех* традиционных математических пакетов является не использование вычислений неограниченной точности, а возможность проводить в них **безошибочные вычисления**, известные также под народным названием **вычислений бесконечной точности**. Так принято называть вычисления, в которых наряду с целыми и рациональными числами используются

точные вещественные или комплексные числа, и при этом не производится *никаких* округлений и приближений. При этом алгебраические числа трактуются как корни многочленов с целыми коэффициентами, трансцендентные числа — как полиномиальные переменные, а при вычислении значений трансцендентных функций используются только точные функциональные соотношения.

Точное вещественное число характеризуется тем, что для него — точно так же, как для целого или рационального числа — как **абсолютная точность** `Accuracy`, так и **относительная точность** `Precision` обе *бесконечны*. Так, вычисление

```
{Accuracy[Pi],Precision[Pi]}
```

даст ответ `{Infinity,Infinity}`.

Например, ввод `Log[3]^Cos[1]` будет интерпретирован как *точное* вещественное число $\ln(2)^{\cos(1)}$. Точно так же `Pi`, `E` и `Sqrt[2]` представляют *точные* вещественные числа π, e и $\sqrt{2}$. По умолчанию *все* вычисления с ними — арифметические, логические, вычисления значений функций, etc. — производятся только с бесконечной точностью.

1.1. Что больше, e^π или π^e? Что больше, $\sqrt{2}^{\sqrt{3}}$ или $\sqrt{3}^{\sqrt{2}}$? Больше ли $\log(2)^{\cos(1)}$ и $\log(3)^{\cos(1)}$, чем 1?

Ответ. Конечно, можно посмотреть на какое-то количество десятичных цифр, как мы это делаем в следующем параграфе, но еще проще спросить прямо. Ну, скажем, `Log[3]^Cos[1]>1`.

§ 2. Приближенные вещественные числа: теория

> Q: How many numerical analysts does it take to replace a lightbulb??
> A: 3.9967: (after six iterations).

В настоящем параграфе мы обсудим основные понятия, связанные с **приближенными значениями** вещественных чисел. Стоит иметь в виду, что в системах компьютерных вычислений многие понятия и соглашения, связанные с приближенными вычислениями, радикально отличаются от традиционного численного анализа. Вещественное число y называется **приближением** вещественного числа x с **абсолютной погрешностью**[19] $\varepsilon > 0$, если $|x - y| \le \varepsilon$. Если y — приближение x с абсолютной погрешностью ε, то $y - \varepsilon \le x \le y + \varepsilon$, таким образом, неопределенность x равна 2ε. Таким образом, приближение тем лучше, чем меньше ε.

Вещественные числа можно приближать любыми другими вещественными числами, но чаще всего рассматривается приближение рациональными

[19] В посконных курсах *методов вычислений* абсолютной погрешностью называлась $|x - y|$, а ε — *предельной* абсолютной погрешностью. Однако, если x и y являются точными числами, то никакой необходимости вводить еще одно название для абсолютной величины их разности нет. Если же они не являются точными числами, то выражение $|x - y|$ само по себе просто не имеет никакого смысла, смысл можно придать только неравенству $|x - y| \le \varepsilon$.

числами. В численном анализе, в отличие от классического анализа и теории чисел, при этом как правило рассматривают не все $\mathbb{Q}$, а лишь кольцо $\mathbb{Z}\left[\frac{1}{10}\right]$ десятичных дробей, которые в этом контексте называются **приближенными вещественными числами**. Каждая такая дробь представляется в виде $x = m \cdot 10^{-n}$, где m и $n \geq 0$ целые числа, причем если дополнительно требовать, чтобы m не делилось на 10, то такое представление единственно. Множество

$$\frac{1}{10^n}\mathbb{Z} = \left\{ \frac{m}{10^n} \mid m \in \mathbb{Z} \right\}$$

называется множеством приближенных чисел **абсолютной точности** (`Accuracy`) n. Традиционно числа абсолютной точности n записываются с n знаками после запятой — которая в Computer Science называется **`десятичной точкой`** — даже если эти знаки нули, однако `Mathematica` использует совершенно другое соглашение.

Однако, в большинстве вычислений важна не абсолютная, а **относительная точность** (`Precision`) числа x, равная абсолютной точности не самого числа x, а его **мантиссы**, т.е. числа $x/10^l$, где l подобрано так, чтобы $1/10 \leq x/10^l < 1$.

Как правило для вещественного числа x существует *единственное* приближенное число y абсолютной точности n такое, что абсолютная погрешность приближения x при помощи y равна $10^{-n}/2$. Отображение $x \mapsto y$ называется **округлением** вещественного числа до n разрядов после запятой. Единственными исключениями являются приближенные числа x абсолютной точности $n+1$, заканчивающиеся на 5. Например, число 0.5 одинаково хорошо приближается как 0, так и 1. В этом случае, чтобы избежать накопления систематической ошибки, используется следующее правило: в качестве округления числа x берется то из чисел точности n, последняя цифра которого четна.

Произведение/частное двух чисел фиксированной *абсолютной* точности не обязательно имеют ту же абсолютную точность. Для чисел фиксированной *относительной* точности то же самое верно уже для суммы/разности. Поэтому вместо обычных арифметических операций для приближенных чисел используются **приближенные арифметические операции**, с математической (но не с вычислительной!!!) точки зрения состоящие в следующем:

- выполняется обычная арифметическая операция,
- получившийся результат округляется до нужной точности.

Обозначим так определенные **приближенное сложение** и **приближенное умножение** через $\oplus$ и $\odot$, соответственно. Строго говоря, следовало бы указывать еще и используемую при округлении абсолютную или относительную точность, но уже введение специальных символов $\oplus$ и $\odot$ для приближенных операций является *огромным* прогрессом по сравнению с

традиционными текстами по *методам вычислений*, где приближенные операции обозначались теми же знаками + и ·, что и обычные операции, так что после нескольких страниц становилось вообще непонятно, о чем идет речь.

Математически трудности приближенных вычислений объясняются тем, что операции приближенного сложения и умножения не обладают *никакими* из обычных свойств сложения и умножения. В книге Кнута подробнейшим обсуждаются связанные с этим неожиданности, например, почему при (приближенном) сложении чисел столбиком сверху вниз и снизу вверх *всегда* получаются разные результаты! Отметим еще несколько необычных явлений.

• Для чисел относительной точности 2 выполяется *точное* равенство $10 \oplus 0.1 = 10$. Таким образом, приближенное сложение не обладает свойством сокращения.

• Для чисел абсолютной точности 1 выполняются следующие соотношения:

$$0.1 \cdot 0.1 = 0.2 \cdot 0.2 = 0, \qquad 0.3 \cdot 0.3 = 0.1, \qquad 0.4 \cdot 0.4 = 0.5 \cdot 0.5 = 0.2$$

В частности, произведение двух ненулевых чисел может равняться нулю. Более того, относительно операции приближенного умножения все числа $-1 < x < 1$ фиксированной абсолютной точности нильпотентны, т.е. дают в некоторой степени 0.

• Ассоциативность сложения нарушается — что еще хуже, она может нарушаться сколь угодно сильно!

Подгрузка пакета `ComputerArithmetic` позволяет эмулировать арифметику чисел с плавающей запятой с точностью до 8 значащих цифр, обычную для большинства микрокалькуляторов:

```
<<NumericalMath‘ComputerArithmetic‘; SetArithmetic[8]
```

— впрочем, следующую задачу можно решить и на обычном микрокалькуляторе.

2.1. Приведите примеры нарушения обычных свойств арифметических операций:

• нарушения ассоциативности приближенного умножения для чисел абсолютной точности 8;

• нарушения ассоциативности приближенного сложения для чисел относительной точности 8;

• нарушения дистрибутивности приближенного умножения относительно приближенного сложения для чисел точности 8.

В связи с этим еще раз подчеркнем, что использование приближенных вычислений без контроля погрешностей и, в особенности, использование приближенных вычислений в итеративных процедурах, является грубейшей методологической ошибкой. В подавляющем большинстве

случаев использование приближенных вычислений представляет собой атавизм, пагубную привычку, злостный пережиток, закрепившиеся в то время, когда все вычисления производились вручную. При сегодняшнем уровне вычислительных возможностей почти все практически возникающие вычислительные задачи, в которых фигурируют вещественные числа, могут быть решены точно. За исключением задач, связанных с генерацией графика и звука, правильная стратегия состоит в том, чтобы проводить точные вычисления, не округляя никаких промежуточных результатов — и, тем более, не делая этого многократно!!! Округлять — в случае необходимости — можно только окончательный результат.

§ 3. Приближенные вещественные числа: практика

> Finagle's Third Law: In any collection of data, the figure most obviously correct, beyond all need of checking, is the mistake.

Посмотрим теперь, как описанные в предыдущем параграфе понятия выражаются в системе `Mathematica`.

`Accuracy[x]`	абсолютная точность x
`Precision[x]`	относительная точность x

В первом приближении можно считать, что абсолютная точность `Accuracy` возвращает количество десятичных цифр справа от запятой, в то время как относительная точность `Precision` — общее количество явно заданных десятичных цифр. В действительности, конечно, дело обстоит *гораздо* сложнее, так как внутреннее представление чисел в системе двоичное, так что для двух чисел одинаковой (десятичной) разрядности команды `Accuracy` и `Precision` могут возвращать различные результаты, притом не обязательно даже целые!

Абсолютная точность может быть как больше, так и меньше относительной, в зависимости от того, больше или меньше чем 1 абсолютная величина рассматриваемого числа. Перечислим основные соглашения, связанные с точностью приближенных чисел.

• Любое рациональное число, а также *точные* иррациональные числа, такие как e, π, $\ln(2)$, $\cos(1)$ рассматриваются как имеющие *бесконечную* точность, `infinite precision`.

• Любое число, имеющее явную десятичную точку, и любое *численное* выражение, содержащее хотя бы одно приближенное вещественное число, рассматривается как *приближенное* вещественное число и имеет тип `Real`.

• Приближенное вещественное число может быть либо **числом машинной точности** = `machine precision number`, либо **числом произвольной точности** = `arbitrary precision number`, `variable precision number`. Машинное число имеет 6 отображаемых в ответе десятичных разрядов и еще

около 10 разрядов дополнительной точности, используемых в промежуточных вычислениях. Таким образом, количество значащих разрядов машинного числа, `MachinePrecision`, приблизительно равно 16, в то же время разрядность числа произвольной точности может быть любой и ограничена только конечностью физического мира (о чем подробнее в следующем параграфе).

Следующие тесты позволяют узнать, рассматривает ли система x как число, как явно заданное число (рациональное или приближенное вещественное), и, наконец, как машинное число.

`NumericQ[x]`	x является числом
`NumberQ[x]`	x является *явно заданным* числом
`MachineNumberQ[x]`	x является машинным числом

• В силу особенностей архитектуры процессора и шины, а также организации памяти, вычисления с машинными числами часто значительно эффективнее, чем с числами произвольной точности. С другой стороны, получающаяся при этом точность достаточна для большинства обычных приложений, в частности, для генерации графики. Поэтому по умолчанию используемая системой рабочая точность равна машинной точности, `WorkingPrecision->MachinePrecision`.

• Приближенное число с общим количеством знаков, *меньшим* машинной точности `MachinePrecision`, по умолчанию рассматривается как число машинной точности. Иными словами, 0.1 значит то же самое, что 0.10, 0.100, 0.1000 и т.д., вплоть до машинной точности, что находится в кричащем противоречии с практикой классической вычислительной математики! Например, вычисление

```
{Accuracy[10^6.+10^-6],Precision[10^6.+10^-6]}
```

даст ответ `{9.95459, MachinePrecision}`

• Численные команды — такие как `N`, `NSolve`, `NRoots`, `NSum`, `NProduct`, `NIntegrate`, `NDSolve` и т.д. — имеют параметры или опции, явно декларирующие точность результата и/или промежуточных значений. Кроме того, точность всех чисел, входящих в некоторое выражение, может быть явно задекларирована посредством команды `SetPrecision`, установкой значения опции `WorkingPrecision` и т.д.

Перечислим простейшие способы декларирования точности.

`AccuracyGoal`	ожидаемая абсолютная точность результата
`PrecisionGoal`	ожидаемая относительная точность результата
`SetAccuracy`	декларирование абсолютной точности
`SetPrecision`	декларирование относительной точности
`WorkingPrecision`	количество цифр, сохраняемых во всех промежуточных результатах
`$MaxExtraPrecision`	количество дополнительные разрядов, используемых в промежуточных вычислениях, по умолчанию 50

• Точность приближенного числа с количеством знаков, *большим* машинной точности, равна его *декларированной* точности, либо, по умолчанию, количеству явно указанных знаков. При обычных конфигурациях ядра точность приближенного числа произвольной точности ограничена только объемом оперативной памяти используемого компьютера.

Основная команда округления в системе `Mathematica` это `N`.

`N[x]`	приближенное значение x с машинной точностью
`N[x,m]`	приближенное значение x с точностью m цифр

По умолчанию `N` возвращает машинные числа, иными словами, она отображает на экране первые 6 десятичных знаков числа, но при этом все внутренние вычисления производятся примерно с 16 знаками. Скажем, вычисление

```
{N[Sqrt[2]],N[E],N[Pi]}
```

дает

```
{1.41421,2.71828,3.14159}
```

В то же время вычисление `N[Pi,100]` дает первые сто знаков π.

3.1. Сколько времени занимает вычисление первого миллиона знаков π, e, $\sqrt{2}$? Сколько времени занимает вычисление первых ста тысяч знаков $\ln(2)$, $\cos(1)$, e^{π}, π^2, $\sqrt{2}^{\sqrt{3}}$?

Ответ. Как говорится в `Implementation Notes`, π вычисляется по формуле Чудновских, но все равно гораздо медленнее, чем e — ряд Тэйлора для экспоненты хорошо сходится! Но e, в свою очередь, вычисляется медленнее, чем $\sqrt{2}$, вычисление которого использует быстро сходящиеся итерационные алгоритмы. Вычисление значений экспонент, логарифмов и тригонометрических функций основано непосредственно на рядах Тэйлора + функциональные соотношения + итерационные процедуры. Для таких значений аргументов скорость вычисления определяется главным образом скоростью сходимости соответствующего ряда.

Следующий курьезный пример, который мы уже упоминали в нашем курсе, показывает, что в ПРИБЛИЖЕННЫХ ВЫЧИСЛЕНИЯХ *ни одна* ИЗ ЦИФР НИЧЕГО НЕ ЗНАЧИТ! В данном случае, конечно, причина этого явления хорошо известна специалистам и связана с тем, что фигурирующие здесь числа 43, 67 и 163 возникают как часть составленного Гауссом списка одноклассных мнимых квадратичных полей

$$\mathbb{Q}(\sqrt{d}), \qquad d = -1, -2, -3, -7, -11, -19, -43, -67, -163.$$

Просто для маленьких дискриминантов интересующий нас эффект не так заметен.

3.2. Вычислите $e^{\pi\sqrt{43}}$ с точностью до 12 и 13 знаков, $e^{\pi\sqrt{67}}$ с точностью до 17, 18 и 19 знаков, и $e^{\pi\sqrt{163}}$ с точностью до 29, 30 и 31 знака.

Ответ. Ограничимся наиболее эффектной частью ответа. Вычисление

```
NumberForm[N[Exp[Pi*Sqrt[163]],29],ExponentFunction->(Null&)]
```

дает

$$262537412640768744.00000000000$$

Напомним, что опция `ExponentFunction->(Null&)` введена здесь только для того, чтобы система не пыталась выразить это число в научной форме, с разделением мантиссы и порядка. То же вычисление с точностью до 30 знаков дает

$$262537412640768743.999999999999$$

где все еще неясно, является ли это число целым на самом деле, или всего лишь с точностью до 12 знаков после запятой. Вычисление следующего разряда снимает все сомнения:

$$262537412640768743.9999999999993$$

Как показывает следующая задача, все ссылки на использование вещественных чисел в естествознании и на их роль в **измерении величин**, о которой говорит Лебег, являются чистой фикцией.

3.3. Каждый кристаллограф знает[20], что $\pi/6 = 0,52$, $\sqrt{3}\pi/8 = 0,68$ и $\sqrt{2}\pi/6 = 0,74$. Найдите π, $\sqrt{3}$ и $\sqrt{2}$.

Ответ. Из первого равенства находим $\pi = 3.12$, что хорошо согласуется с теоретическим значением. Подставляя это значение во второе равенство, получаем, что в кристаллографии $\sqrt{3} = 1.74359$, а $\sqrt{2} = 1.42308$.

Следующие две задачи (как и эпиграф к главе 5!) взяты из замечательной книги Гуго Штейнгауза[21].

3.4. Выяснить, верно ли, что

$$x = \sqrt{5 + \sqrt{3 + \sqrt{5 + \sqrt{3 + \ldots}}}} < 3?$$

3.5. Выяснить, при любом ли n выполняется неравенство

$$x = \sqrt{1 + \sqrt{2 + \sqrt{3 + \ldots + \sqrt{n}}}} < 2?$$

---

[20]Смотри, например, А.Анималу, Квантовая теория кристаллических твердых тел. — М., Мир, 1981, 574с., где все эти формулы *именно в таком виде* приведены на страницах 15–16 как плотности простой кубической, объемноцентрированной кубической и гранецентрированной кубической упаковок одинаковых шаров!

[21]Г.Штейнгауз, Задачи и размышления. — М., Мир, 1974, 400с.

§ 4. Машинные числа

Sometimes it is useful to know how large your zero is.
The Tao of Real Programming

В настоящем курсе мы почти не упоминаем системные команды. Тем не менее, при некоторых практических вычислениях полезно понимать ограничения, накладываемые используемой операционной системой, а также архитектурой процессора, памятью компьютера и другими подобными внешними по отношению к вычислению обстоятельствами. Перечислим системные команды, которые позволяют узнать параметры приближенных вычислений. Приводимые ниже численные значения системных параметров относятся к системе `Windows` и бытовым компьютерам 2005–2006 годов.

`MachinePrecision`	машинная точность, по умолчанию $53\log_{10}(2)$
`$MachinePrecision`	численное значение машинной точности, 15.9546
`$MachineEpsilon`	машинный эпсилон 2^{-52}
`$MaxMachineNumber`	наибольшее машинное число 2^{1024}
`$MinMachineNumber`	наименьшее положительное машинное число 2^{-1022}

Машинный эпсилон это наименьшее положительное *машинное* число ε такое, что $1+\varepsilon \neq 1.000000000000000$, где равенство понимается как равенство машинных чисел. Поскольку вычисления в системе производятся в двоичной системе, то фактически `$MachineEpsilon` имеет следующее значение:

$$\texttt{\$MachineEpsilon} = 2^{-52} \approx 2.22045 \cdot 10^{-16}.$$

Машинный эпсилон является абсолютной версией применяемого в теоретической Computer Science **ульпа** = `unit in the last place`. А именно, машинный эпсилон ε — это в точности ульп между 1 и 2. Ульп между 2 и 4 равен 2ε, etc. Теоретически приближенные вычисления могут быть организованы так, чтобы ошибка одной операции не превышала половины ульпа, а ошибка при выполнении нескольких операций — одного ульпа. Однако, по причинам упомянутым в § 2, практически при вычислениях с числами разного масштаба уже выполнение *двух* операций может давать ошибку в миллионы ульпов. Более того, в обычной машинной арифметике контроль ошибки не производится!

Наибольшее машинное число `$MaxMachineNumber` равно

$$\begin{aligned} 2^{1024} = {} & 179769313486231590772930519078902473361797697894230657273430081157732675805500963132708477322407536021120113879871393357658789768814416622492847430639\\ & 474124377767893424865485276302219601246094119453082950850057688381506823424628814739131105408272371\\ & 633505106845862982399472459384797163048353563296242241372160 \approx 1.79769 \cdot 10^{308}. \end{aligned}$$

С другой стороны, наименьшее положительное машинное `$MinMachineNumber` число равно

$$2^{-1022} \approx 2.22507 \cdot 10^{-308}.$$

Таким образом, операции над машинными числами такие, как сложение и умножение, не только не удовлетворяют обычным свойствам наподобие ассоциативности и коммутативности, но и вообще не являются всюду определенными алгебраическими операциями!

Фактически, однако, в силу физических ограничений, накладываемых размером оперативной памяти используемого компьютера, среди чисел произвольной точности тоже есть наибольшее и наименьшее число. Узнать, чему именно они равны для Вашей системы, можно при помощи системных команд `$MaxNumber` и `$MinNumber`.

`$MaxNumber`	наибольшее число произвольное точности
`$MinNumber`	наименьшее положительное число произвольное точности

Вот типичные значения наибольшего и наименьшего чисел произвольной точности:

$$\texttt{\$MaxNumber} = 1.920224672692357 \cdot 10^{646456887}$$

$$\texttt{\$MinNumber} = 5.207723940958924 \cdot 10^{-646456888}$$

Это значит, что выполнение алгебраических операций над числами произвольной точности тоже может привести к переполнению. Например, 2^{10^9} можно вычислить точно, в то же время попытка вычислить $2^{2\cdot 10^9}$ приводит к переполнению.

§ 5. Десятичные цифры

> Так, поле “Номер паспорта”, 6 цифр. Это до запятой или после?
>
> Николай Фоменко

Основными функциями для работы с приближенными вещественными числами являются `RealDigits` и обратная к ней функция `FromDigits`. В предыдущей главе мы уже обсуждали, как эти функции работают для рациональных чисел.

`RealDigits[x]`	список цифр вещественного числа x
`FromDigits[{list,m}]`	восстановление числа по списку цифр

Для приближенных вещественных чисел (=десятичных дробей) эти команды в целом действуют без всяких неожиданностей.

Однако, непосредственная попытка вычислить `RealDigits` от *точного* иррационального числа, как алгебраического `RealDigits[Sqrt[2]]`, так и трансцендетного `RealDigits[Pi]`, приведет к сообщению об ошибке:

```
RealDigits::ndig:  The number of digits to return
```

```
cannot be determined.
```

В отличие от команды `N` нельзя и спросить `RealDigits[Pi,20]`, так как это будет истолковано не как пожелание определить 20 десятичных знаков числа π, а как пожелание найти *все* знаки числа π в двадцатиричной системе.

Правильное обращение к этой команде таково: нужно указать число, десятичные цифры которого мы хотим найти, основание системы счисления и затем количество цифр, которые мы хотим найти. Так, вычисление

```
RealDigits[Pi,10,25]
```

даст первые 25 десятичных цифр числа π и позицию запятой в этом числе:

```
{{3,1,4,1,5,9,2,6,5,3,5,8,9,7,9,3,2,3,8,4,6,2,6,4,3},1}
```

Следует, впрочем, иметь в виду, что если мы требуем больше десятичных цифр, чем точность числа x, то недостающие цифры заменяются символом `Indeterminate`. Например, вычисление

```
RealDigits[N[Pi,15],10,18]
```

вернет

```
{{3,1,4,1,5,9,2,6,5,3,5,8,9,7,9,3,Indeterminate,Indeterminate},1}
```

5.1. Найдите, сколько раз каждая из цифр $0, 1, 2, \ldots, 9$ встречается среди первой тысячи десятичных знаков π.

Решение. Например, так

```
Map[Count[First[RealDigits[Pi,10,1000]],#]&,Range[0,9]]
```

5.2. Какая цифра чаще всего встречается среди первых 10000 десятичных знаков e?

5.3. Найдите наименьшее n такое, что среди первых n знаков π встречается каждая из цифр $0, 1, 2, \ldots, 9$.

5.4. Найдите первое вхождение в десятичное разложение π двух цифр 3 подряд, трех цифр 3 подряд, четырех цифр 3 подряд.

5.5. Найдите цифру, встречающуюся среди первых 100000 десятичных знаков π ровно шесть раз подряд.

5.6. Существует ли цифра, которая встречается среди первых 1000000 десятичных знаков π ровно семь раз подряд?

Следующая задача состоит главным образом в том, чтобы понять, что именно в ней спрашивается.

5.7. Если написать десятичные цифры чисел e и π в обратном порядке, какое из получившихся чисел будет больше?

§ 6. Алгебраические числа

На наивном уровне вещественные числа делятся на рациональные и иррациональные, При этом иррациональность $\sqrt{2}$ выдвигается в качестве мотивации для введения вещественных чисел. Это *полная ерунда*, так как

этим мотивируется лишь необходимость введения *алгебраических* чисел. С точки зрения профессионального алгебраиста вычисления с $\sqrt{2}$ ничуть не сложнее, чем с $1/3$. Дело в том, что число $\sqrt{2}$ алгебраическое, даже *целое* алгебраическое, притом степени 2, так что вычисления с ним сводятся к вычислениям с *целочисленными* матрицами степени 2.

Как с принципиальной, так и с вычислительной точки зрения действительно судьбоносное различие проходит не между рациональными и иррациональными числами, а между алгебраическими и трансцендентными. Напомним, что (комплексное) число x называется **алгебраическим**, если оно является корнем алгебраического уравнения

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0 = 0, \qquad a_i \in \mathbb{Z},$$

для некоторого многолена f с целыми коэффициентами. Если существует такое уравнение со старшим коэффициентом 1, то число x называется **целым алгебраическим** — или, как говорят профессионалы, просто **целым** (при этом числа целые в обычном смысле называются *целыми рациональными*). Целочисленный многочлен f наименьшей возможной степени такой, что $f(x) = 0$, называется **минимальным многочленом** x, а его степень — степенью алгебраического числа x. Вычисления с индивидуальным целым алгебраическим числом степени d не сложнее, чем вычисления с d обычными целыми числами.

Множество всех алгебраических чисел образует поле $\overline{\mathbb{Q}}$, а множество *целых* алгебраических чисел — его подкольцо $\mathbb{A}$. Заметим, впрочем, что в документации к системе `Mathematica` через $\mathbb{A}$ обозначается домен *всех* алгебраических чисел. Гаусс заметил, что число, которое одновременно является целым и рациональным, действительно целое рациональное: $\mathbb{A} \cap \mathbb{Q} = \mathbb{Z}$.

Число x, которое не является алгебраическим, называется **трансцендентным**. Трансцендентное число не удовлетворяет вообще никаким нетривиальным алгебраическим уравнениям и, таким образом, может быть принято за независимую полиномиальную переменную. Конечно, фактически системы компьютерной алгебры именно так и поступают, это значит, что вычисления с индивидуальным трансцендентным числом имеют ту же сложность, что вычисления с целочисленными многочленами.

Долгое время не было известно, существуют ли трансцендентные числа, и многие математики верили, что все вещественные числа являются алгебраическими. Лишь в 1851 году Лиувилль построил первые примеры трансцендентных чисел. Ясно, что до этих примеров не было никакой необходимости в формальном введении вещественного числа, и действительно первые конструкции вещественных чисел начали появляться лишь в 1860-е годы — ровно через три века после того, как были введены комплексные числа и через полвека после того, как они были формально построены! Одновременно с этим была доказана трансцендентность двух самых знаменитых констант: в 1873 Эрмит доказал трансцендентность e, а в 1882 году фон Линдеманн доказал трансцендентность π, полностью решив, тем самым,

классическую проблему “квадратуры круга”. С другой стороны, при помощи совершенно других соображений Кантор доказал, что алгебраические числа встречаются крайне редко, *почти все* вещественные числа являются трансцендентными. В то же время даже сегодня доказательство трансцендентности индивидуального числа часто остается чрезвычайно сложной задачей.

Упомянем некоторые определенные в ядре системы команды, позволяющие работать с алгебраическими числами.

`Algebraics`	домен алгебраических чисел
`Root[f,m]`	m-й корень алгебраического уравнения $f(x)=0$
`RootReduce[x]`	минимальный многочлен x
`ToRadicals[x]`	преобразование x к радикалам

Имя домена `Algebraics` используется обычным образом, вопрос в формате `Element[x,Algebraics]` дает ответ `True`, если число x алгебраическое и `False` в противном случае. Вычисление

```
Map[Element[#,Algebraics]&,{Sqrt[2],2^Sqrt[2],Sqrt[2]^Sqrt[3]}]
```

показывает, что $\sqrt{2}$ алгебраическое число, в то время как $2^{\sqrt{2}}$ и $\sqrt{2}^{sqrt3}$ — нет. Система *знает*, что e, π и e^π не являются алгебраическими, но вот вопрос о том, будут ли $e+\pi$ и $e\pi$ алгебраическими, ставит ее в тупик.

Ясно, что любое число, скомбинированное из рациональных чисел при помощи арифметических операций и извлечения корней, является алгебраическим. Однако, далеко не любое алгебраическое число имеет такой вид. Обычно, самым простым описанием алгебраического числа является его описание в терминах минимального многочлена. В ядре `Mathematica` для этого используются объекты типа `Root`, дополнительные возможности описаны в пакетах расширений.

В простейших случаях команда `RootReduce` пытается преобразовать выражение к явной комбинации радикалов. Если это невозможно, либо если ответ в терминах радикалов слишком сложен, вычисление `RootReduce[x]` преобразует x к единственному объекту формата `Root` — иными словами, ищет минимальный многочлен x. Например, вычисление

```
RootReduce[Sqrt[2]+Sqrt[3]]
```

дает

```
Root[1-10#1^2+#1^4&,4]
```

иными словами, минимальный многочлен $\sqrt{2}+\sqrt{3}$ равен x^4-10x^2+1. Интересно, что решение уравнения `Solve[x^4-10x^2+1==0,x]` возвращает ответ в форме

$$-\sqrt{5-2\sqrt{6}},\quad \sqrt{5-2\sqrt{6}},\quad -\sqrt{5+2\sqrt{6}},\quad \sqrt{5+2\sqrt{6}},$$

и только применение `FullSimplify` возвращает эти корни к виду $\pm\sqrt{2}\pm\sqrt{3}$.

6.1. Найдите минимальный многочлен $\sqrt{2}+\sqrt{3}+\sqrt{5}$.

6.2. Найдите минимальный многочлен $\sqrt{1+\sqrt{1+\sqrt{2}}}$.

6.3. Найдите минимальный многочлен $\sqrt{1+\sqrt{1+\sqrt{1+\sqrt{2}}}}$.

6.4. Найдите минимальный многочлен $\sqrt{2+\sqrt{3+\sqrt{5}}}$.

6.5. Найдите минимальный многочлен $\sqrt[3]{2}+\sqrt{3}$.

6.6. Найдите минимальный многочлен $\sqrt[3]{2}+\sqrt[3]{3}$.

Команда `ToRadicals` пытается — в тех случаях, когда это возможно — преобразовать выражение алгебраического числа как комбинации корней алгебраических уравнений к явному выражению в радикалах даже если она считает, что его описание как корня алгебраического уравнения проще. Как мы уже упоминали, в большинстве случаев сама по себе система предпочитает описание в терминах минимальных многочленов. Дело в том, что в общем случае непосредственное преобразование вложенных радикалов и даже проверка того, что два выражения, содержащие вложенные ралдикалы, равны, представляет собой совсем непростую задачу. С другой стороны, совпадение минимальных многочленов проверяется очень легко.

§ 7. Основные константы

> Mathematician: π is the ratio of the circumference of a circle to its diameter.
> Engineer: π is about 22/7.
> Physicist: π is 3.14159 plus or minus 0.000005
> Computer Programmer: π is 3.141592653589 in double precision.
> Nutritionist: π is a healthy and delicious dessert!
>
> The problems for the exam will be similar to the discussed in the class. Of course, the numbers will be different. But not all of them: π will still be 3.14159 ...
>
> Cambridge mathematical quotes

В системе имплементированы основные математические константы. Вот наибоее известные из них.

`E`	e	основание натурального логарифма
`Pi`	π	длина окружности диаметра 1
`Degree`	1^0	$\pi/180$, градус
`GoldenRatio`	$\phi = \dfrac{1+\sqrt{5}}{2}$	золотое сечение
`EulerGamma`	γ	константа Эйлера
`Catalan`	C	константа Каталана

7.1. В 1674 году Лейбниц открыл следующую формулу:

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \ldots = \frac{\pi}{4}.$$

Проверьте ее.

Ответ. Для этого можно просто беззастенчиво набрать

```
Sum[(-1)^i*1/(2*i+1),{i,0,Infinity}]
```

но обычно при суммировании по арифметической последовательности удобнее не пересчитывать вид слагаемых, а надлежащим образом задавать вид итератора:

```
Sum[(-1)^(i/2-1/2)/i,{i,1,Infinity,2}]
```

7.2. В 1748 году Эйлер открыл следующую формулу:

$$1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \ldots = \frac{\pi^2}{6}$$

и много других подобных формул, в частности,

$$1 - \frac{1}{2^2} + \frac{1}{3^2} - \frac{1}{4^2} + \ldots = \frac{\pi^2}{12}, \qquad 1 + \frac{1}{3^2} + \frac{1}{5^2} + \frac{1}{7^2} + \ldots = \frac{\pi^2}{8}.$$

Проверьте эти формулы и найдите еще несколько таких же.

Указание. Что можно менять в этих формулах? Заметим, что чередование знака в последней формуле даст нам формулу для одной важнейшей константы — константы Каталана:

$$1 - \frac{1}{3^2} + \frac{1}{5^2} - \frac{1}{7^2} + \ldots = -C.$$

7.3. Стали ли бы Вы использовать какую-либо из формул, полученных в двух предыдущих задачах, для фактического вычисления π. Если да, то какую и почему?

В качестве ответа предложим следующее уточнение предыдущей задачи.

7.4. Проверьте, сколько верных знаков π получается при суммировании первых $10000 \cdot n$, $1 \le n \le 10$ членов ряда Лейбница и сколько времени это занимает.

Ответ. Вот π с точностью до 50 знаков *после запятой*, полученное с помощью `N[Pi,51]`:

3.14159 26535 89793 23846 26433 83279 50288 41971 69399 37511

С другой стороны, всего за пару минут выполнение команды

```
Table[N[Sum[4*(-1)^(i-1)/(2i-1),{i,1,10000*n}],51],{n,1,10}]
```

возвращает следующую таблицу:

3.14149 26535 90043 23845 95183 83374 81537 87870 13642 74418
3.14154 26535 89824 48846 25457 27030 24751 30928 52688 19022
3.14155 93202 56469 16438 85564 49123 16786 47638 81274 79068
3.14156 76535 89797 14471 26403 31521 69620 16104 78839 39196
3.14157 26535 89795 23846 26423 83279 50410 41971 66629 37511
3.14157 59869 23127 72920 33837 22142 67194 89566 13878 53420
3.14157 83678 75508 25303 99026 72563 72981 01894 97534 74708
3.14158 01535 89793 72674 38932 87912 07128 90207 11000 30165
3.14158 15424 78682 47028 70603 39959 54829 01599 71987 72722
3.14158 26535 89793 48846 26433 52029 50289 37284 19393 96494

Обратите внимание, что суммирование первых 10000 членов ряда все еще дает ошибку в четвертом знаке после запятой, и даже суммирование первых 100000 членов все еще дает неправильный пятый знак! Ясно, что использовать такую формулу для вычислений невозможно. Что, однако, гораздо интереснее, после ошибки в четвертом знаке сумма дает 6 верных знаков, а последняя — целых 10 верных знаков! Некоторые из остальных сумм дают 7, 8 или 9 верных знаков после одного, двух или трех неверных. Концептуальное объяснение этого удивительного явления предложено в[22].

7.5. Проверьте формулу Валлиса

$$\frac{2}{1}\cdot\frac{2}{3}\cdot\frac{4}{3}\cdot\frac{4}{5}\cdot\frac{6}{5}\cdot\frac{6}{7}\cdot\ldots=\frac{\pi}{2}$$

Решение. Нужно просто понять, как ввести это произведение. Легче всего так:

```
Product[(i/(i+1))^((-1)^i),{i,1,Infinity}]
```

7.6. В 1593 году Франсуа Виет открыл следующую замечательную формулу

$$\frac{2}{\pi}=\frac{\sqrt{2}}{2}\cdot\frac{\sqrt{2+\sqrt{2}}}{2}\cdot\frac{\sqrt{2+\sqrt{2+\sqrt{2}}}}{2}\cdot\ldots$$

Удастся ли Вам проверить ее тем же способом, что предыдущие?

7.7. Чтобы понять, откуда берется формула Виета, вычислите $\cos(\pi/2^n)$.

§ 8. Элементарные функции

Как известно, в `Mathematica` все основные элементарные функции имеют обычные английские имена.

`Exp[x]`	экспонента x
`Log[x]`	логарифм x
`Log[b,x]`	логарифм x по основанию b

[22] J.M.Borwein, P.B.Borwein, K.Dolcher, Pi, Euler numbers, and asymptotic expansions. — Amer. Math. Monthly, 1989, vol.96, p.681–687.

Экспонента и логарифм, также как тригонометрические и гиперболические функции, обратные тригонометрические и гиперболические функции и специальные функции, трактуются как **числовые функции** и мы рекомендуем ознакомиться с соответствующими разделами [Wo] или [VH2].

8.1. (Хэмминг, см. Кнут I-1.2.2.22) Путем численного эксперимента — либо построения графиков — убедитесь, что *для всех практических целей*

$$\log_2(x) = \ln(x) + \log_{10}(x).$$

А теперь объясните это явление.

Также и основные тригонометрические функции имеют обычные в английской математической литературе имена.

`Cos[x]`	косинус x
`Sin[x]`	синус x
`Tan[x]`	тангенс x
`Cot[x]`	котангенс x
`Sec[x]`	секанс x
`Csc[x]`	косеканс x

Мы не будем упражняться в проведенеии тригонометрических преобразований, эта тема упоминается в [VH2] и подробно обсуждается в книге Мадера [Ma] и Выпуске 3 нашей книги “`Mathematica` для нематематика”, посвященном функциональному программированию. Ограничимся лишь несколькими забавными примерами *точных* вычислений со значениями тригонометрических функций. В качестве разминки начнем с совсем простеньких задачек.

8.2. Из школьного курса тригонометрии известно, что $2\cos(\pi/4) = \sqrt{2}$. Вычислите $2\cos(\pi/8)$, $2\cos(\pi/16)$, $2\cos(\pi/32)$ и $2\cos(\pi/64)$.

Решение. Да чего уж там,

```
Map[FunctionExpand,Table[2*Cos[Pi/2^n],{n,3,6}]]
```

Применение `FunctionExpand` необходимо, так как система сама по себе, разумеется не будет преобразовывать точное значение $\cos(\pi/2^n)$ в радикалы. Ответ выглядит примерно так:

$$\sqrt{2+\sqrt{2}},\quad \sqrt{2+\sqrt{2+\sqrt{2}}},\quad \sqrt{2+\sqrt{2+\sqrt{2+\sqrt{2}}}},$$
$$\sqrt{2+\sqrt{2+\sqrt{2+\sqrt{2+\sqrt{2}}}}}.$$

8.3. Упростите $\dfrac{1}{2\sin(10^o)} - 2\sin(70^o)$.

Решение. Это выражение равно 1. Вычисление

```
Simplify[1/(2*Sin[10*Degree])-2*Sin[70*Degree]]
```

дает то же самое, что применение `TrigReduce`, и приводит к выражению $\cos(80^o)/\sin(10^o)$, хотя и чуть более простому, чем исходное выражение, но все еще не окончательному. Но вот вычисление

```
FullSimplify[1/(2*Sin[10*Degree])-2*Sin[70*Degree]]
```

приводит к полному успеху. Того же результата можно добиться сначала переведя выражение в радикалы при помощи `FunctionExpand`, а потом применив `FullSimplify`.

Следующие три задачи взяты из книги[23].

8.4. Вычислите сумму косинусов

$$\cos(5^o) + \cos(77^o) + \cos(149^o) + \cos(221^o) + \cos(293^o).$$

8.5. Вычислите разность

$$\operatorname{tg}(117^o) + \operatorname{tg}(118^o) + \operatorname{tg}(125^o) - \operatorname{tg}(117^o)\operatorname{tg}(118^o)\operatorname{tg}(125^o).$$

8.6. Вычислите произведение косинусов

$$\cos\left(\frac{\pi}{15}\right)\cos\left(\frac{2\pi}{15}\right)\cos\left(\frac{3\pi}{15}\right)\cos\left(\frac{4\pi}{15}\right)\cos\left(\frac{5\pi}{15}\right)\cos\left(\frac{6\pi}{15}\right)\cos\left(\frac{7\pi}{15}\right)$$

§ 9. Арифметическая структура континуума

Q. What's the square root of 69?
A. Eight something.

Cambridge mathematical quotes

'Tis a favorite project of mine
A new value of π to assign.
I would fix it at 3
For it's simpler, you see,
Than 3 point 1 4 1 5 9

Вот несколько наиболее важных функций дискретизации.

`Floor[x]`	$\lfloor x \rfloor$	пол x
`Ceiling[x]`	$\lceil x \rceil = -\lfloor -x \rfloor$	потолок x
`Round[x]`		ближайшее к x целое
`IntegerPart[x]`		целая часть x
`FractionalPart[x]`		дробная часть x

[23] Ч.Тригг, Задачи без изюминки. — М., Мир, 2000, с.1–276.

Использование всех этих функций, *кроме* функции `IntegerPart`, ясно само по себе. Дело в том, что функция `IntegerPart` является целой частью в понимании не математиков, а программистов на ассемблере! Функция `Floor[x]`, которую принято обозначать $\lfloor x \rfloor$, возвращает наибольшее целое, не превосходящее x. Это то, что называется **целой частью** числа x и традиционно обозначалось $[x]$ или $\mathrm{Ent}(x)$ (**антье**). В то же время функция `IntegerPart` в отличие от функции `Floor` игнорирует знак числа x. Это значит, что для отрицательных чисел `IntegerPart[x]` равно `-Floor[-x]`. Таким образом `IntegerPart[x]` совпадает с `Floor[x]` для $x > 0$ и с `Ceiling[x]` для $x < 0$. Еще один нюанс состоит в том, как функция `Round` округляет полуцелые числа. В соответствии с упомянутым в § 2 общим принципом полуцелое число округляется до ближайшего *четного* целого. Таким образом, 0.5 округляется до 0, а 1.5 — до 2.

Следующее равенство известно как **тождество Лиувилля**:

$$\lfloor\sqrt{x}\rfloor + \lfloor\sqrt{x-1^2}\rfloor + \lfloor\sqrt{x-2^2}\rfloor + \ldots = \left\lfloor\frac{x}{1}\right\rfloor - \left\lfloor\frac{x}{3}\right\rfloor + \left\lfloor\frac{x}{5}\right\rfloor - \left\lfloor\frac{x}{7}\right\rfloor + \ldots$$

Суммы в левой и правой частях лишь *формально* бесконечны, в действительности n-е слагаемое слева становится равным 0 при $n > \sqrt{x}$, а n-е слагаемое справа — при $n > (x+1)/2$.

9.1. Проверьте тождество Лиувилля для всех $x < 1000$.

Следующее равенство известно как **тождество Эрмита**:

$$\lfloor x\rfloor + \left\lfloor x + \frac{1}{n}\right\rfloor + \left\lfloor x + \frac{2}{n}\right\rfloor + \ldots + \left\lfloor x + \frac{n-1}{n}\right\rfloor = \lfloor nx\rfloor$$

9.2. Система не упрощает автоматически левую часть тождества Эрмита до правой. Как Вы стали бы проверять тождество Эрмита?

§ 10. Непрерывные дроби

Скоморох-потешник:
Чудес в мире — как мух в сортире.

Леонид Филатов, *Про Федота-стрельца*

Как правило десятичные дроби дают не очень хорошие приближения вещественных чисел. Часто легко найти значительно лучшие рациональные приближения вещественного числа с гораздо меньшими знаменателями. Например, рациональное приближение $22/7$ числа π точнее, чем десятичное приближение $314/100$, а приближение $355/113$ дает 6 правильных десятичных знаков после запятой, и имеет меньшую погрешность, чем десятичная дробь 3.141593 со знаменателем 1000000!

В `Mathematica` подобные рациональные приближения ищутся при помощи функции `Rationalize`. Для построения таких приближений можно использовать также непрерывные дроби. А именно, можно применить к веще-

ственному числу x функцию `ContinuedFraction[x,n]` с большим значением n и потом применить к результату функцию `FromContinuedFraction[y]`.

`Rationalize[x,d]`	хорошее рациональное приближение к x
`ContinuedFraction[x,n]`	разложение x в непрерывную дробь
`FromContinuedFraction[x]`	восстановление x по непрерывной дроби

Функция `Rationalize[x,d]` возвращает рациональное приближение p/q вещественного числа x с наименьшим знаменателем среди приближений абсолютной погрешности d. Кроме того, по умолчанию предполагается, что это приближение достаточно хорошо в следующем абсолютном смысле:

$$\left| x - \frac{p}{q} \right| < \frac{1}{10^4 q^2}.$$

10.1. Найдите рациональные приближения числа e с погрешностью 10^{-n}, где $1 \le n \le 20$.

10.2. Найдите рациональные приближения числа π с погрешностью 10^{-n}, где $1 \le n \le 20$.

10.3. Найдите рациональное приближение π с машинной точностью.

Решение. Это

$$\pi \approx \frac{245850922}{78256779},$$

в чем можно убедиться вычисляя `Rationalize[N[Pi],0]`. Так как `N[Pi]` есть машинное число, то 0 здесь истолковывается как *машинный* ноль. С учетом того, что машинный эпсилон равен 2^{-52}, мы получим тот же результат и вычисляя `Rationalize[N[Pi],2^-53]`.

Функция `ContinuedFraction[x,s]` возвращает список первых s членов в разложении x в **правильную непрерывную дробь**. Список {`n1,...,ns`} истолковывается обычным образом:

$$[n_1, \dots, n_s] = n_1 + \cfrac{1}{n_2 + \cfrac{1}{n_3 + \cfrac{1}{\ddots + \cfrac{1}{n_s}}}}$$

Напомним, что правильность дроби означает, что $n_1 \in \mathbb{Z}$, а $n_i \in \mathbb{N}$ для всех $i \ge 2$. Можно показать, что для *каждой* правильной бесконечной непрерывной дроби

$$[n_1, n_2, n_3, n_4, \dots] = n_1 + \cfrac{1}{n_2 + \cfrac{1}{n_3 + \cfrac{1}{n_4 + \dots}}},$$

последовательность **подходящих дробей** $[n_1]$, $[n_1, n_2]$, $[n_1, n_2, n_3]$, ... имеет предел, называемый **значением** дроби $[n_1, n_2, n_3, \dots]$. При этом значение бесконечной дроби является *иррациональным* числом, причем *каждое* иррациональное число x является значением *единственной* правильной бесконечной непрерывной дроби. Функция `ContinuedFraction[x]`, вызванная с одним аргументом, порождает список всех членов, которые можно получить исходя из заданной точности x. В отличие от десятичных дробей, связанных со случайным выбором базы системы счисления, непрерывные дроби гораздо лучше отражают арифметическую природу вещественных чисел и дают гораздо лучшие приближения.

В предыдущей главе мы упоминали теорему Эйлера, которая утверждает, что рациональные числа и только они раскладываются в периодическую десятичную дробь (конечные дроби являются частным случаем периодических). С глубокой древности известно, что каждое рациональное число раскладывается в *конечную* непрерывную дробь — как мы увидим в главе 5, это в точности алгоритм Эвклида. Естественно возникает вопрос, допускают ли простую характеризацию вещественные числа, раскладывающиеся в *периодическую* непрерывную дробь? Это действительно так, знаменитая теорема Лагранжа утверждает, что числа, раскладывающиеся в периодическую непрерывную дробь, это в точности **квадратичные иррациональности**, иными словами, алгебраические числа степени 2. Для квадратичных иррациональностей вызванная без параметра n функция `ContinuedFraction[x]` возвращает ответ в формате

`{n1,...,ns,{m1,...,mt}}`,

где $n_1, \dots, n_s$ — предпериод, а $m_1, \dots, m_t$ — период.

6.4. Вычислите разложения $\sqrt{n}$ в непрерывную дробь для $n \le 100$. Какие два обстоятельства сразу бросаются в глаза в этом списке?

6.5. Сколько среди первой тысячи простых p тех, для которого разложение $\sqrt{p}$ в непрерывную дробь имеет период 1?

6.6. Найдите среди первой тысячи простых p то, для которого разложение $\sqrt{p}$ в непрерывную дробь имеет самый длинный период.

Ответ. Это $p_{861} = 6679$, для него период имеет длину 172.

В 1767 году Лагранж заметил, что вычислять разложение алгебраического числа в непрерывную дробь довольно легко, если оно является *единственным* вещественным корнем своего минимального уравнения. В частности, это относится к следующему по сложности случаю — **кубическим иррациональностям**, иными словами, корням таких уравнений степени 3 с целыми коэффициентами, которые не имеют рациональных корней. Типичный пример кубических иррациональностей, это кубические корни $\sqrt[3]{n}$ из целых чисел, не являющихся полными кубами.

6.7. Вычислите первые несколько десятков неполных частных разложения $\sqrt[3]{n}$ в непрерывную дробь для $n \le 100$. Какие обстоятельства становятся очевидными в этом списке с ростом n?

Разложения кубических иррациональностей в непрерывные дроби начал изучать еще Якоби. В 1960-е и 1970-е годы их стали систематически вычислять при помощи компьютеров. Вот одно из чудес, которые при этом обнаружили Бриллхарт и Моррисон.

6.8. Найдите первые несколько десятков неполных частных разложения в непрерывную дробь наименьшего положительного корня уравнения $x^3 - 8x - 10 = 0$. Что Вам кажется в высшей степени странным?

Ответ. Вот первые 40 неполных частных:

$$3, 3, 7, 4, 2, 30, 1, 8, 3, 1, 1, 1, 9, 2, 2, 1, 3, 22986, 2, 1, 32, 8,\\ 2, 1, 8, 55, 1, 5, 2, 28, 1, 5, 1, 1501790, 1, 2, 1, 7, 6, 1$$

Видно, что некоторые неполные частные много больше, чем можно было бы ожидать. Лемер заметил, что дискриминант этого уравнения равен $-4\cdot 163$ и Старк[24] доказал, что это действительно то же самое 163, которое мы видели в выражении $e^{\pi\sqrt{163}}$ и аномальность разложения в непрерывную дроби связана с тем, что кольцо целых поля $\mathbb{Q}(\sqrt{-163})$ является кольцом клавных идеалов.

Разложение некоторых трансцендентных чисел в непрерывную дробь также демонстрирует совершенно удивительные закономерности. Следующее разложение является едва ли не самым простым определением числа e:

$$2+\cfrac{1}{1+\cfrac{1}{2+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{4+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{6+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{8+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{10+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{12+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{14+\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{16+\ldots}}}}}}}}}}}}}}}}}}}}}}}$$

Впрочем, в 1731 году Эйлер нашел простое объяснение этого поразительного факта.

6.9. Найдите несколько десятков неполных частных разложения e^{-1}, e^2, e^{-2}, $\sqrt{e}$, $1/\sqrt{e}$ в непрерывную дробь и сформулируйте получившиеся закономерности.

6.10. Удастся ли обнаружить подобные простые закономерности в разложении в непрерывную дробь числа π и его степеней?

Ответ. Для правильных непрерывных дробей очевидных закономерностей нет, но если отказаться от требования, чтобы все числители равнялись 1, то

[24]Ч.М.Старк, Объяснение некоторых экзотических непрерывных дробей, найденных Бриллхартом. — В кн.[VANT], с.155–171.

можно получить много таких представлений. Исторически первым и самым простым из них является полученное в 1655 году разложение Броункера

$$\frac{\pi}{4} = \cfrac{1}{1+\cfrac{1^2}{2+\cfrac{3^2}{2+\cfrac{5^2}{2+\cfrac{7^2}{2+\cfrac{9^2}{2+\cfrac{11^2}{2+\cfrac{13^2}{2+\cfrac{15^2}{2+\ldots}}}}}}}}}$$

§ 7. BBP-формулы для быстрого вычисления цифр

> A value of π to 40 digits would be more than enough to compute the circumference of the Milky Way galaxy to an error less than the size of a proton[25].
>
> D.H.Bailey, J.M.Borwein, P.B.Borwein, S.Plouffe

В настоящем параграфе, который носит чисто дескриптивный характер, мы приводим несколько совершенно поразительных формул, которые позволяют вычислять десятичные знаки π и других трансцендентных чисел не вычисляя предыдущих знаков. Вот самая знаменитая из таких формул, открытие которой[26,27] вне всякого сомнения является одним из самых замечательных событий в многотысячелетней истории вычисления π:

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k}\left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6}\right)$$

Как замечают Адамчик и Вэгон[28] тот факт, что формулы такого типа для иррациональных чисел существуют, *по существу* был известен и ранее. Вот два классических примера, которые показывают, что мы можем вычислить любую (двоичную) цифру $\log(2)$ и $\log(3)$ не вычисляя предыдущих цифр. Следующая формула получается из ряда Тэйлора для $\log(1+x)$ в $x = 1/2$:

$$\log(2) = \sum_{k=0}^{\infty} \frac{1}{2^k}\,\frac{1}{k}$$

Аналогичная формула для $\log(3)$ получается из ряда Грегори — иными словами, ряда Тэйлора для $f((1+x)/(1-x))$ — в $x = 1/2$:

$$\log(3) = \sum_{k=0}^{\infty} \frac{1}{4^k}\,\frac{1}{2k+1}$$

---

[25] D.H.Bailey, J.M.Borwein, P.B.Borwein, S.Plouffe, The quest for Pi. — Math. Intelligencer, 1997, vol.19, N.1, p.50–57.

[26] D.Bailey, P.Borwein, S.Plouffe, On the rapid computation of various polylogarithmic constants. — Math. Comput.

[27] S.Rabinowitz, S.Wagon, A spigot algorithm for π. — Amer. Math. Monthly 102 (1995) 195-203.

[28] V.Adamchik, S.Wagon, A 2000-Year Search Changes Direction

В `Mathematica` формулы такого рода легко ищутся при помощи текстов наподобие следующего:

```
Normal[Series[Log[(1+x)/(1-x)],{x,0,12}]] /. x->1/2
```

Однако, до работы Бейли, Борвайна и Плуффа никому не приходило в голову использовать эти формулы для фактического вычисления двоичных цифр $\log(2)$ и $\log(3)$. Апостериори Адамчик и Вэгон получили множество подобных формул, в частности, знакопеременную формулу для π.

Ограничимся перечислением несколько простейших формул такого типа. В цитированных статьях можно найти как много дальнейших примеров, так и изложение общих методов поиска таких формул в `Mathematica`:

$$\log(3) = \sum_{k=0}^{\infty} \frac{1}{16^{k+1}} \left(\frac{16}{4k+1} + \frac{4}{4k+3} \right)$$

$$\log(5) = \sum_{k=0}^{\infty} \frac{1}{16^{k+1}} \left(\frac{16}{4k+1} + \frac{16}{4k+2} + \frac{4}{4k+3} \right)$$

$$\pi = \sum_{k=0}^{\infty} \frac{(-1)^k}{4^k} \left(\frac{2}{4k+1} + \frac{2}{4k+2} + \frac{1}{4k+3} \right)$$

$$\pi\sqrt{2} = \sum_{k=0}^{\infty} \frac{(-1)^k}{8^k} \left(\frac{4}{6k+1} + \frac{1}{6k+3} + \frac{1}{6k+5} \right)$$

$$\log(5) = \sum_{k=0}^{\infty} \frac{(-1)^k}{4^k} \left(\frac{2}{4k+1} - \frac{1}{4k+3} \right)$$

$$\log(7) = \sum_{k=0}^{\infty} \frac{1}{8^{k+1}} \left(\frac{12}{3k+1} + \frac{6}{3k+2} \right)$$

— Вы мне только скажите: шерсть выпала? — с последней надеждой спросил ей вслед Швондер.

Михаил Булгаков, *Собачье сердце*

Глава 4. КОМПЛЕКСНЫЕ ЧИСЛА

It is the complex case that is easier to deal with.

Cambridge Mathematical Quotes

I met a man once who told me that far from believing in the square root of minus one, he didn't believe in minus one. This is at any rate a consistent attitude[29].

Edward Titchmarsh

Ни один факт вещественного анализа невозможно понять, оставаясь в области вещественных чисел. В настоящей главе мы обсудим основы вычислений с комплексными числами. Комплексные числа, которые были введены итальянскими алгебраистами в XVI веке как игрушка для математических турниров, без всякой видимой практической цели, в XIX–XX веках превратились в основной инструмент математического естествознания. Комплексные числа играют такую же роль в квантовой механике, как вещественные числа в классической. Вся их история подтверждает мысль Харди о том, что ЧИСТАЯ МАТЕМАТИКА *ощутимо* ПОЛЕЗНЕЕ ПРИКЛАДНОЙ.

§ 1. Комплексные числа

Life is complex. It has real and imaginary components.

Tom Potter

Напомним, что алгебраической формой комплексного числа называется его запись в виде $a + bi$, где $a, b \in \mathbb{R}$ — вещественные числа, а i — мнимая единица, $i^2 = -1$. При этом a обычно называется **вещественной частью** z, и обозначается $\mathrm{re}(z)$, а b — **мнимой частью** z и обозначается $\mathrm{im}(z)$. Числа, у которых $\mathrm{im}(z) = 0$, называются **вещественными**, а те, у которых $\mathrm{im}(z) \neq 0$ — **мнимыми**. Числа, у которых $\mathrm{re}(z) = 0$ называются **чисто мнимыми**. При сложении комплексных чисел отдельно складываются их вещественные и мнимые части,

$$\mathrm{re}(z + w) = \mathrm{re}(z) + \mathrm{re}(w), \qquad \mathrm{im}(z + w) = \mathrm{im}(z) + \mathrm{im}(w),$$

а умножение производится по следующей формуле:

$$(a + bi)(c + di) = (ac - bd) + (ad + bc)i.$$

[29]В русском переводе книге Г.С.М.Коксетера *Введение в геометрию* эта фраза передана следующим образом: “Я недавно встретил человека, который сказал мне, что он не верит даже в существование -1, так как из этого следует существование квадратного корня из нее”. Мы предоставляем читателю самостоятельно оценить модальности и еще раз задуматься над *трудностями перевода*.

Комплексное число $\overline{z} = a - bi$ называется **сопряженным** к $z = a + bi$. При этом $z + \overline{z} = 2a$ и $z\overline{z} = a^2 + b^2$ оба вещественные. Любое ненулевое комплексное число z обратимо, при этом $z^{-1} = \overline{z}/(z\overline{z})$.

Использование функций, связанных с комплексными числами, ясно из названия.

`I`	i	мнимая единица i
`z=x+I*y`	$z = x + iy$	комплексное число z, где $x, y \in \mathbb{R}$
`Re[z]`	$\mathrm{re}(z)$	вещественная часть z
`Im[z]`	$\mathrm{im}(z)$	мнимая часть z
`Conjugate[z]`	$\overline{z} = x - iy$	сопряженное к z
`ComplexExpand[z]`		выделение $\mathrm{re}(z)$ и $\mathrm{im}(z)$

Единственный момент, на который стоит обратить внимание, состоит в том, что автоматически операции над комплексными числами, содержащими символы, не исполняются. Чтобы фактически провести вычисление, нужно применять команду `ComplexExpand`, которая по умолчанию исходит из того, что *все* неспецифицированные переменные вещественные. Зададим два комплексных числа $z = a + bi$ и $w = c + di$ в алгебраической форме `z=a+b*I`, `w=c+d*I`. Вычисление `z*w` не дает ничего интересного, так как скобки автоматически не раскрываются. Но вот вычисление `ComplexExpand[z*w]` дает

```
(a*c-b*d)+I*(b*c+a*d),
```

так что система действительно считает a, b, c, d вещественными. Применение `ComplexExpand` необходимо и в том случае, когда коэффициенты являются *точными* вещественными числами, наподобие e, π или $\cos(\pi/7)$.

В ядре `Mathematica` описан домен `Complexes` комплексных чисел, элементы которого имеют тип `Complex`.

`Complex`		тип комплексного числа
`Complexes`	$\mathbb{C}$	домен комплексных чисел

Так, например, вычисление `Head[1+I]` даст `Complex`, а тест

```
Element[1+I,Complexes]
```

даст результат `True`.

Комплексное число $a + bi$, $a, b \in \mathbb{R}$, можно отождествить с **антициркулянтом** $\begin{pmatrix} a & b \\ -b & a \end{pmatrix}$. Легко проверить, что при этом сложению и умножению комплексных чисел соответствуют обычное сложение и умножение матриц. Для сложения это совсем очевидно, а для умножения получается вот что:

$$\begin{pmatrix} a & b \\ -b & a \end{pmatrix} \begin{pmatrix} c & d \\ -d & c \end{pmatrix} = \begin{pmatrix} ac - bd & ad + bc \\ -ad - bc & ac - bd \end{pmatrix}.$$

1.1. Постройте поле $\mathbb{C}$ комплексных чисел при помощи этой конструкции. Что при этом отвечает сопряженному и обратному числам?

При написании программ полезно понимать, что `Mathematica` упорядочивает комплексные числа следующим несколько необычным образом:

• В первую очередь сравниваются вещественные части.

• При равенстве вещественных частей сравниваются *абсолютные значения* мнимых частей.

• Из пары сопряженных комплексных чисел первым указывается то, мнимая часть которого отрицательна.

1.2. Не включая компьютера расположите числа

$$1, -1, i, -i, 1+i, 1-i, -1+i, -1-i, -1+2i, -1-2i, 1+2i, 1+2i, 0$$

в используемом системой порядке.

В следующей задаче

$$\omega = -\frac{1}{2} + i\frac{\sqrt{3}}{2}, \qquad \omega^2 = \overline{\omega} = -\frac{1}{2} - i\frac{\sqrt{3}}{2}.$$

После чтения § 3 она моментально решается устно, пока же мы предлагаем ппровести непосредственные вычисления.

1.3. Вычислите

• $(a+b)(a+b\omega)(a+b\omega^2)$,

• $(a\omega+b\omega^2)(b\omega+a\omega^2)$,

• $(a+b\omega+c\omega^2)(a+b\omega^2+c\omega)$.

• $(a+b+c)(a+b\omega+c\omega^2)(a+c\omega^2+b\omega)$,

• $(a+b\omega+c\omega^2)^3+(a+b\omega^2+c\omega)^3$.

На самом деле следующая задача предполагает решение в терминах тригонометрической, а не алгебраической формы, но она служит хорошей иллюстрацией того, что многократное альтернированное применение `ComplexExpand` и `Simplify` или `FullSimplify` может несколько раз приводить к новой, все более простой форме.

1.4. Вычислите

$$z = \frac{\sqrt{2-\sqrt{3}} + i\sqrt{2+\sqrt{3}}}{\sqrt{\sqrt{2}+1} + i\sqrt{\sqrt{2}-1}}.$$

Решение. Устное вычисление модуля и аргумента числителя и знаменателя z показывает, что

$$z = \sqrt[4]{2}\left(\cos\left(\frac{7\pi}{24}\right) + i\sin\left(\frac{7\pi}{24}\right)\right).$$

Разумеется, предлагая подобную задачу в контрольной работе мы подразумеваем именно такое решение. Интересно, однако, сколько времени понадобится, чтобы придти к тому же результату в алгебраической форме.

Вычисление `FullSimplify[z]` дает представление z как корня многочлена с рациональными коэффициентами:

```
Root[4-4#1^2+2#1^4-2#1^6+#1^8&,6]
```

Иными словами, утверждается, что z есть корень уравнения $x^8-2x^6+2x^4-4x^2+4=0$.

Попробуем вначале выделить вещественную и мнимую часть z, и только потом упростить, `FullSimplify[ComplexExpand[z]]`. При этом получится уже чуть более явный ответ

$$\sqrt{\frac{1}{2}\left((1+i)-(1-i)\sqrt{3}\right)}$$

Кажется, что не будет никакого вреда, если теперь уже в этом выражении выделить вещественную и мнимую часть, но при этом получается достаточно жуткое выражение, которое мы не решаемся здесь воспроизвести без промежуточного упрощения,

```
Simplify[ComplexExpand[FullSimplify[ComplexExpand[z]]]]
```

дает

$$i2^{1/4}\left(\cos\left(\frac{1}{2}\operatorname{arctg}\left(\frac{1+\sqrt{3}}{1-\sqrt{3}}\right)\right)+i\sin\left(\frac{1}{2}\operatorname{arctg}\left(\frac{1+\sqrt{3}}{1-\sqrt{3}}\right)\right)\right)$$

Но вот применение `FullSimplify` приводит к полному успеху, а именно, вычисляя

```
FullSimplify[ComplexExpand[FullSimplify[ComplexExpand[z]]]]
```

мы получим $(-1)^{7/24}2^{1/4}$, так что теперь еще одно применение `ComplexExpand` как раз и даст ответ, указанный в самом начале!

Теперь еще одно применение `FullSimplify` снова даст нам $(-1)^{7/24}2^{1/4}$, так что кажется, что все возможности исчерпаны. Впрочем, мы можем снова выразить косинус и синус через радикалы при помощи `FunctionExpand` и посмотреть, что получится. Удивительным образом, вычисление

```
FullSimplify[FunctionExpand[ComplexExpand[FullSimplify[
            ComplexExpand[FullSimplify[ComplexExpand[z]]]]]]]
```

снова дает ответ в другой форме $(-1)^{1/8}(i+\sqrt{3})/2^{3/4}$, так что можно начинать все сначала, снова `ComplexExpand` и далее по тексту.

До сих пор мы обсуждали вычисления с *точными* комплексными числами. Однако, как и вещественные числа, комплексные числа бывают точными и приближенными. А именно, вещественная и/или мнимая части комплексного числа могут быть *приближенными* вещественными числами — по умолчанию машинными. Важнейший нюанс, который следует иметь в виду, проводя приближенные вычисления, состоит в том, что точность/приближенность вещественной и мнимой части оцениваются отдельно! Так, например, вычисление `Head[1+0*I]` дает `Integer`, а вычисление

`Head[1.+0*I]` — `Real`. Иными словами, система считает, что мнимая часть этих чисел точно равна 0, так что первое из них целое, а второе — приближенное вещественное число. Чтобы указать, что мнимая часть лишь *приближенно* равна 0, нужно явным образом поставить там десятичную точку. Числа `1+0.*I` и `1.+0.*I` оба имеют заголовок `Complex`

§ 2. Тригонометрическая запись комплексного числа

> If you stick your fingers in the mains, it's not the imaginary component which you will feel.
>
> Cambridge Mathematical Quotes

> — Папа, а как пишется число 8?
> — Как бесконечность, повернутая на угол пи пополам.
>
> Николай Фоменко

В предыдущем параграфе мы изучали комплексное число $z = a + bi$ как точку вещественной плоскости (a, b), проекции которой на вещественную и мнимую оси равны a и b, соответственно. Сложение таким образом истолкованных комплексных чисел совпадает с обычным сложением векторов.

Оказывается, однако, что для умножения комплексных чисел *намного* удобнее пользоваться другой системой координат — полярной. В этой системе положение любой точки на плоскости определяется двумя числами: расстоянием r от начала координат до этой точки и углом ϕ между полярной осью и радиусом-вектором данной точки, отсчитываемым в положительном направлении. Положительное вещественное число $r = \sqrt{a^2 + b^2}$ называется **модулем** комплексного числа $z = a + bi$ и обозначается $|z|$. Полярный угол ϕ точки (a, b), изображающей $z = a+bi$, называется **аргументом** комплексного числа z и обозначается $\arg(z)$. Отметим, что аргумент 0 не определен. Пусть поэтому $|z| \neq 0$. В этом случае $\phi = \arg(z)$ — это такой угол, что

$$\cos(\phi) = \mathrm{re}(z)/|z|, \qquad \sin(\phi) = \mathrm{im}(z)/|z|.$$

Эти равенства определяют угол ϕ неоднозначно, лишь с точностью до целого кратного 2π. Среди всех ϕ удовлетворяющих этим равенствам найдется единственное в промежутке $[0, 2\pi)$. Традиционно это значение ϕ называется **главным значением аргумента** и обозначается $\mathrm{Arg}(z)$.

В системе имплементированы функции, вычисляющие по комплексному числу его модулю и аргумент. Стоит, однако иметь в виде, что возвращаемый функцией `Arg` результат находится между π и $-\pi$.

`Abs[z]`	$\|z\|$	модуль x
`Arg[z]`	$\arg(z)$	аргумент z

2.1. Скажите, не включая компьютера, чему равен аргумент 0? А потом проверьте.

Ответ. Правильно, `Interval[{-Pi,Pi}]`.

2.2. Напишите команду, возвращающую главное значение аргумента в промежутке $[0, 2\pi)$.

Решение. Ну, хотя бы, так:

```
arg[x_]:=If[Arg[x]>=0,Arg[x],Arg[x]+2*Pi]
```

если, конечно, Вы не боитесь сообщения `Possible spelling error`.

Если $r = |z|$ — модуль комплексного числа z, а $\phi = \arg(z)$ — его аргумент, то z можно записать в виде:

$$z = a + bi = r \cdot \cos(\phi) + ir \cdot \sin(\phi) = r(\cos(\phi) + i \sin(\phi)),$$

называемом **тригонометрической формой** z. Тригонометрическая форма *идеально* согласована с умножением комплексных чисел, а именно, при умножении комплексных чисел их модули перемножаются, а аргументы складываются, $|zw| = |z||w|$, $\arg(zw) = \arg(z) + \arg(w)$. Таким образом, для

$$z = r(\cos(\phi) + i\ \sin(\phi)), \qquad w = s(\cos(\psi) + i\ \sin(\psi)),$$

имеем

$$zw = r(\cos(\phi) + i\ \sin(\phi)) \cdot s(\cos(\psi) + i\ \sin(\psi)) = rs(\cos(\phi + \psi) + i\ \sin(\phi + \psi)),$$

Эта формула, известная под названием **формулы де Муавра**, является одной из самых полезных формул в математике и, вместе с теоремой Пифагора, содержит **всю** школьную тригонометрию. Особенно часто используется такое ее следствие

$$z^n = r^n(\cos(n\phi) + i\ \sin(n\phi)).$$

2.3. Докажите формулу де Муавра.

Решение. Ну, конечно, просто

```
Simplify[(Cos[x]+I*Sin[x])*(Cos[y]+I*Sin[y])]
```

2.4. Докажите, что

$$\left(\frac{1 + i\ \mathrm{tg}(\phi)}{1 - i\ \mathrm{tg}(\phi)}\right)^n = \frac{1 + i\ \mathrm{tg}(n\phi)}{1 - i\ \mathrm{tg}(n\phi)}.$$

Формула же для сложения комплексных чисел в тригонометрической форме чуть сложнее и известна специалистам по оптике и электричеству под народным названием “суперпозиция синхронных скалярных гармонических колебаний”.

2.5. Найдите формулу для суммы $z = z_1 + z_2$ двух комплексных чисел

$$z_1 = r_1(\cos(\phi_1) + i\ \sin(\phi_1)), \qquad z_2 = r_2(\cos(\phi_2) + i\ \sin(\phi_2))$$

в тригонометрической форме.

Ответ. Модуль r и аргумент ϕ числа z определяются формулами:

$$r^2 = r_1^2 + r_2^2 + 2r_1 r_2 \cos(\phi_2 - \phi_1),$$

$$\operatorname{tg}(\phi) = \frac{r_1 \sin(\phi_1) + r_2 \sin(\phi_2)}{r_1 \cos(\phi_1) + r_2 \cos(\phi_2)}.$$

2.6. Обобщите результат предыдущей задачи на любое количество слагаемых.

§ 3. Корни из 1

> В одном отношении формула для $\cos(2\pi/17)$ не оставляет сомнения. Прийти к ней в рамках традиционных геометрических идей времени Эвклида невозможно.
>
> Семен Григорьевич Гиндикин[30]

Важнейшую роль в строении комплексных чисел играют корни из 1. Напомним, что корнем n-й степени из 1 называется решение уравнения $x^n = 1$. В тригонометрической форме корни n-й степени из 1 выражаются как

$$\varepsilon_k = \cos\left(\frac{2\pi k}{n}\right) + i \sin\left(\frac{2\pi k}{n}\right), \qquad k = 0, \ldots, n-1.$$

Обычно множество всех корней степени n из 1 обозначается μ_n и называется *группой* корней из 1 степени n.

3.1. Напишите команду, которая возвращает список корней из 1 степени n в порядке возрастания аргумента.

Решение. Чтобы можно было реально показать возникающие ответы, проиллюстрируем это на примере $n = 7$. Наивная попытка написать

```
Solve[x^7==1,x]
```

возвращает следующий ответ:

```
{{x->1},{x->-(-1)^1/7},{x->(-1)^2/7},{x->-(-1)^3/7},
          {x->(-1)^4/7},{x->(-1)^5/7},{x->(-1)^6/7}}
```

Иными словами, команда `Solve` возвращает не корни, а правила замены! Ну, с этим справиться легко, нужно просто применить эти правила к x, например, так:

```
ReplaceAll[x,Solve[x^7==1,x]]
```

Теперь вычисление дает нам список корней:

```
{1,-(-1)^1/7},(-1)^2/7,-(-1)^3/7,(-1)^4/7,(-1)^5/7,(-1)^6/7}
```

Однако, эти корни по-прежнему не вычислены! Попробуем выделить в них вещественную и мнимую часть:

---

[30] С.Г.Гиндикин, Рассказы о физиках и математиках. — Наука, М., 1981, с.1–191.

```
ComplexExpand[ReplaceAll[x,Solve[x^7==1,x]]]
```

Получающийся при этом ответ уже больше похож на то, что хотелось:

```
{1,-Cos[Pi/7]-I*Sin[Pi/7],Cos[2*Pi/7]+I*Sin[2*Pi/7],
      -Cos[3*Pi/7]-I*Sin[3*Pi/7],Cos[4*Pi/7]+I*Sin[4*Pi/7],
      -Cos[5*Pi/7]-I*Sin[5*Pi/7],Cos[6*Pi/7]+I*Sin[6*Pi/7]}
```

Однако при этом используется описанный в § 1 внутренний порядок на комплексных числах, в то время как гораздо естественнее упорядочивать корни из 1 по возрастанию аргумента. Это можно сделать в терминах определенной в предыдущем параграфе функции `arg`:

```
Sort[ComplexExpand[ReplaceAll[x,Solve[x^7==1,x]]],
                                    arg[#1]<arg[#2]&]
```

При этом получается следующий ответ:

```
{1,Cos[2*Pi/7]+I*Sin[2*Pi/7],Cos[4*Pi/7]+I*Sin[4*Pi/7],
      Cos[6*Pi/7]+I*Sin[6*Pi/7],-Cos[Pi/7]-I*Sin[Pi/7],
      -Cos[3*Pi/7]-I*Sin[3*Pi/7],-Cos[5*Pi/7]-I*Sin[5*Pi/7]}
```

В принципе это примерно то, что мы хотели. Поэтому не будем продолжать и дальше бороться с системой, чтобы записать аргументы в привычном виде.

3.2. Найдите сумму корней степени $n \geq 2$ из 1.

Ответ. Вычислив эту сумму для нескольких первых случаев, мы видим, что она равна 0. Как только ответ сформулирован, он становится очевидным, так как корни из 1 являются корнями многочлена $x^n - 1$, а формула Виета утверждает, что сумма корней этого многочлена равна коэффициенту при x^{n-1}.

3.3. Найдите сумму попарных произведений корней степени $n \geq 3$ из 1.

Корень из единицы ε называется **первообразным** или, как теперь принято говорить, **примитивным** корнем степени n, если он не является корнем никакой меньшей, чем n степени. Если ε — первообразный корень из 1 степени n, то $\varepsilon^k \neq 1$ для всех $k = 1, \ldots, n-1$ и поэтому $\varepsilon^k \neq \varepsilon^l$ для всех $0 \leq k \neq l \leq n-1$. Таким образом, все числа $\varepsilon^k = 1, \varepsilon, \varepsilon^2, \ldots, \varepsilon^{n-1}$ попарно различны. Это значит, что любой первообразный корень степени n порождает группу μ_n. Корни же степени n, не являющиеся первообразными содержатся в какой-то строго меньшей подгруппе μ_m. Корень ε_k в том и только том случае является первообразным корнем степени n, когда $\gcd(k, n) = 1$.

3.4. Напишите команду, которая возвращает список *первообразных* корней из 1 степени n в порядке возрастания аргумента.

Решение двух следующих задач предполагает знакомство с арифметическими функциями ϕ и μ, которые мы обсуждаем в главе 7.

3.5. Найдите сумму первообразных корней степени $n \geq 2$ из 1.

Ответ. Эта сумма равна значению функции Мебиуса $\mu(n)$.

3.6. Найдите сумму m-х степеней первообразных корней степени $n \geq 2$ из 1.

Ответ. Положим $d = \gcd(m, n)$. Тогда

$$\sum \varepsilon^m = \frac{\phi(n)}{\phi(n/d)}\mu(n/d),$$

Перейдем теперь к явному вычислению первообразных корней небольших степеней. При этом мы получим несколько формул для значений основных тригонометрических функций, которые почему-то не предлагают заучивать в школьном курсе тригонометрии.

3.7. Явно найдите все (первообразные) корни из 1 небольших степеней, скажем, $n \leq 30$. В каких случаях эти корни можно найти решая квадратные уравнения?

Решение. Прежде всего, чтобы превратить $\cos(2\pi/n)$ и $\sin(2\pi/n)$ в комбинацию радикалов или что-то в таком духе, нужно применить `FunctionExpand`. Однако, получающийся при этом ответ совершенно неудобочитаем, поэтому обычно поверх `FunctionExpand` мы применяем `Simplify`. Следует быть чрезвычайно осторожным с применением `FullSimplify`, так как упрощение чего-нибудь совсем невинного — как $\cos(\pi/7)$ или $\cos(\pi/13)$ — длится *минутами* и либо приводит к столь же нечитаемому ответу, либо возвращает исходное выражение.

Вот те степени, для которых такое вычисление дает явные формулы, использующие только квадратные радикалы:

$$1,\ 2,\ 3,\ 4,\ 5,\ 6,\ 8,\ 10,\ 12,\ 15,\ 16,\ 17,\ 20,\ 24,\ 30.$$

Так как все эти степени являются делителями последних пяти из них, то достаточно привести ответ для 16, 17, 20, 24, 30.

Вот корни степени 16:

$$\begin{array}{lllll}
1 & \dfrac{\sqrt{2+\sqrt{2}}+i\sqrt{2-\sqrt{2}}}{2} & \dfrac{1+i}{\sqrt{2}} & \dfrac{\sqrt{2-\sqrt{2}}+i\sqrt{2+\sqrt{2}}}{2} \\
i & \dfrac{-\sqrt{2-\sqrt{2}}+i\sqrt{2+\sqrt{2}}}{2} & \dfrac{-1+i}{\sqrt{2}} & \dfrac{-\sqrt{2+\sqrt{2}}+i\sqrt{2-\sqrt{2}}}{2} \\
-1 & \dfrac{-\sqrt{2+\sqrt{2}}-i\sqrt{2-\sqrt{2}}}{2} & \dfrac{-1-i}{\sqrt{2}} & \dfrac{-\sqrt{2-\sqrt{2}}-i\sqrt{2+\sqrt{2}}}{2} \\
-i & \dfrac{\sqrt{2-\sqrt{2}}-i\sqrt{2+\sqrt{2}}}{2} & \dfrac{1-i}{\sqrt{2}} & \dfrac{\sqrt{2+\sqrt{2}}-i\sqrt{2-\sqrt{2}}}{2}
\end{array}$$

Чтобы получить корни степени 8, нужно взять каждый второй из них, начиная, естественно, с 1, которая является *нулевым*, а не первым по порядку корнем!

Вот корни степени 20:

$$\begin{array}{cll} 1 & \dfrac{\sqrt{10+2\sqrt{5}}+i(-1+\sqrt{5})}{4} & \dfrac{1+\sqrt{5}+i\sqrt{10-2\sqrt{5}}}{4} \\ & \dfrac{\sqrt{10-2\sqrt{5}}+i(1+\sqrt{5})}{4} & \dfrac{-1+\sqrt{5}+i\sqrt{10+2\sqrt{5}}}{4} \\ i & \dfrac{1-\sqrt{5}+i\sqrt{10+2\sqrt{5}}}{4} & \dfrac{-\sqrt{10+2\sqrt{5}}+i(1+\sqrt{5})}{4} \\ & \dfrac{-1-\sqrt{5}+i\sqrt{10+2\sqrt{5}}}{4} & \dfrac{-\sqrt{10+2\sqrt{5}}+i(-1+\sqrt{5})}{4} \\ -1 & \dfrac{-\sqrt{10+2\sqrt{5}}+i(1-\sqrt{5})}{4} & \dfrac{-1+\sqrt{5}-i\sqrt{10+2\sqrt{5}}}{4} \\ & \dfrac{-\sqrt{10+2\sqrt{5}}+i(-1-\sqrt{5})}{4} & \dfrac{-1+\sqrt{5}-i\sqrt{10+2\sqrt{5}}}{4} \\ -i & \dfrac{-1+\sqrt{5}-i\sqrt{10+2\sqrt{5}}}{4} & \dfrac{\sqrt{10+2\sqrt{5}}+i(-1-\sqrt{5})}{4} \\ & \dfrac{1+\sqrt{5}-i\sqrt{10+2\sqrt{5}}}{4} & \dfrac{\sqrt{10+2\sqrt{5}}+i(1-\sqrt{5})}{4} \end{array}$$

Чтобы получить корни степени 10 или 5, нужно просто взять каждый второй или, соответственно, каждый четвертый из них, снова, естественно, начиная с 1.

Вот, наконец, корни степени 24:

$$\begin{array}{cll} 1 & \dfrac{1+\sqrt{3}+i(-1+\sqrt{3})}{2\sqrt{2}} & \dfrac{\sqrt{3}+i}{2} \\ \dfrac{1+i}{\sqrt{2}} & \dfrac{1+i\sqrt{3}}{2} & \dfrac{-1+\sqrt{3}+i(1+\sqrt{3})}{2\sqrt{2}} \\ i & \dfrac{1-\sqrt{3}+i(1+\sqrt{3})}{2\sqrt{2}} & \dfrac{-1+i\sqrt{3}}{2} \\ \dfrac{-1+i}{\sqrt{2}} & \dfrac{-\sqrt{3}+i}{2} & \dfrac{-1-\sqrt{3}+i(-1+\sqrt{3})}{2\sqrt{2}} \\ -1 & \dfrac{-1-\sqrt{3}+i(1-\sqrt{3})}{2\sqrt{2}} & \dfrac{-\sqrt{3}-i}{2} \\ \dfrac{-1-i}{\sqrt{2}} & \dfrac{-1-i\sqrt{3}}{2} & \dfrac{1-\sqrt{3}+i(-1-\sqrt{3})}{2\sqrt{2}} \end{array}$$

$$-i \qquad \frac{-1+\sqrt{3}+i(-1-\sqrt{3})}{2\sqrt{2}} \qquad \frac{+1-i\sqrt{3}}{2}$$

$$\frac{1-i}{\sqrt{2}} \qquad \frac{\sqrt{3}-i}{2} \qquad \frac{1+\sqrt{3}+i(1-\sqrt{3})}{2\sqrt{2}}$$

Чтобы получить корни степени 12, 6 или 3 нужно просто взять каждый второй, соответственно, каждый четвертый или каждый восьмой из них.

Мы не будем приводить ответы для $n = 17$ и $n = 30$ ввиду их громоздкости. Однако, для ценителей конкретности приведем явную формулу на основе которой легко восстановить корни степени 17. А именно, применяя `FunctionExpand` и `Simplify` к $\cos(2\pi/17)$, мы получим следующее выражение:

$$\cos\Big(\frac{2\pi}{17}\Big) = \frac{1}{16}\Big(\sqrt{17}-1+\sqrt{34-2\sqrt{17}}+ \\ 2\sqrt{17+3\sqrt{17}-\sqrt{170+38\sqrt{17}}}\,\Big),$$

Заметим, что это не очень сложное вычисление может быть проведено и непосредственно. В действительности, именно это вычисление и было проведено семнадцатилетним Гауссом при построении правильного 17-угольника! Отсюда уже совсем легко вывести, что

$$\cos\Big(\frac{\pi}{17}\Big) = \frac{1}{16}\sqrt{15+\sqrt{17}+\sqrt{34-2\sqrt{17}}\;+ \\ \sqrt{2\left(34+6\sqrt{17}+\sqrt{578-34\sqrt{17}}-\sqrt{34-\sqrt{17}}-8\sqrt{2\left(17+\sqrt{17}\right)}\right)}},$$

детали приведены в цитированной книге С.Г.Гиндикина.

3.8. В каких случаях первообразные корни из 1 степеней $n \le 30$ можно найти решая кубические уравнения?

Ответ. Кроме степеней, приведенных в ответе к предыдущей задаче, это степени 7, 9, 13, 18, 19, 27. Однако, мы воздержимся от того, чтобы приводить явные формулы, ввиду их громоздкости.

Глава 5. МОДУЛЯРНАЯ АРИФМЕТИКА

> Люди, думающие, что духовные, эзотерические, высшие движения в мире возникают ни с того, ни с сего, должны уяснить, что нет ничего более далекого от истины, чем это предположение. Любому подлинному знанию всегда предшествуют чрезвычайно сложные планирование и подготовка.
>
> Идрис Шах, *Знание как знать*

> The only thing that he did as Deputy Mayor was to reduce the Shirriffs to their proper functions and numbers.
>
> J.R.R Tolkien, *The Lord of the Rings*

Настоящая глава, в которой мы обсуждаем арифметические структуры, связанные с делением целых чисел с остатком, является подготовительной для трех следующих. С математической точки зрения речь здесь идет о вычислениях в кольце классов вычетов $\mathbb{Z}/m\mathbb{Z}$. Это один из самых древних разделов математики, восходящий к *И Цзин*, египетским писцам и секте пифагорейцев. Некоторые из излагаемых в настоящей главе алгоритмов известны *по крайней мере* около 2500 лет и являются самыми старыми алгоритмами, полностью сохранившими свою актуальность. В последние десятилетия модулярная арифметика широчайшим образом используется в безошибочных вычислениях.

§ 1. Делимость целых чисел

> Breast size multiplied by IQ always equals 69.
>
> Cambridge Mathematical Quotes

> Now here's a start I made the other day: Dante wrote a poem, about a place called H—. H-dash, because I don't want any trouble with the censors.
>
> Henry Miller, *Black Spring*

Говорят, что целое число $x \in \mathbb{Z}$ **делит** $y \in \mathbb{Z}$ или, что то же самое, что y **делится** на x, если существует такое $z \in \mathbb{Z}$, что $y = xz$. Это обозначается одним из двух следующих образов:

$$x|y \text{ — } x \text{ делит } y, \qquad \text{либо} \qquad y\dot{:}x \text{ — } y \text{ делится на } x.$$

При этом x называется **делителем** y, а y — **кратным** x.

Перечислим основные свойства делимости.

• **Рефлексивность**: $x|x$.

• **Транзитивность**: если $x|y$ и $y|z$, то $x|z$.

• Если $x|y, z$, то $x|(y+z)$.

• Если $x|y$, то $x|yz$ для любого z.

Эти свойства допускают следующее совместное обобщение.

• Если $x|y_1, \ldots, y_s$, то для любых $z_1, \ldots, z_s$ имеем $x|y_1z_1 + \ldots + y_sz_s$ т.е., иными словами, x делит любую линейную комбинацию элементов $y_1, \ldots, y_s$.

Выяснять, является ли число x делителем числа y лучше всего при помощи определенной в следующем параграфе функции `Mod`. А именно, x в том и только том случае делит y, когда `Mod[y,x]` равно 0. Еще один способ состоит в том, чтобы убедиться, что `Floor[y/x]` совпадает с y/x. Это вычисление чуть менее эффективно, чем вычисление с помощью функции `Mod` — к слову, совсем не потому, что функция `Mod` целочисленная, тем более, что аргументы `Mod` не обязаны быть целыми числами. Однако, в данном случае разница в производительности становится по настоящему заметной только для чисел с несколькими миллионами цифр. Наконец, наименее эффективный способ состоит в том, чтобы явно проверить, содержится ли x в списке делителей y посредством `MemberQ[Divisors[y],x]`. Так как этот способ требует полной факторизации y, и поиска в списке, то он применим только к числам, содержащим не более несколько десятков разрядов.

Общим делителем x и y называется такое число $d \in \mathbb{Z}$, что $d|x$, $d|y$. **Наибольшим общим делителем** этих чисел называется такой их общий делитель d, который делится на любой другой их общий делитель, иными словами, если для любого целого числа z из того, что $z|x$ и $z|y$ следует, что $z|d$. Наибольший общий делитель элементов x и y обычно обозначается НОД(x, y) или $\gcd(x, y)$, от английского `greatest common divisor`. Если дополнительно предполагать, что $d \geq 0$, то он определен однозначно и обладает следующими свойствами.

• **Поглощение**: $\gcd(x, y) = x \iff x|y$. В частности, $\gcd(x, x) = x$ и $\gcd(x, 0) = x$.

• **Коммутативность**: $\gcd(x, y) = \gcd(y, x)$.

• **Ассоциативность**: $\gcd(\gcd(x, y), z) = \gcd(x, \gcd(y, z))$.

• Умножение **дистрибутивно** относительно операции взятия наибольшего общего делителя: для любых x, y, z выполнено равенство $\gcd(zx, zy) = z\gcd(x, y)$.

Наибольший общий делитель допускает **линейное представление**: если $d = \gcd(x, y)$, то найдутся такие a, b, что $d = ax + by$. Обратно, наибольший общий делитель чисел x и y может быть определен как такой их общий делитель, который допускает линейное представление.

Целые числа x и y называются **взаимно простыми**, если их наибольший общий делитель равен 1. Взаимно простые числа **комаксимальны**, иными словами, для них найдутся такие a, b, что $ax+by = 1$. Следуя Кнуту мы обозначаем взаимноую простоту чисел x и y посредством $x \perp y$.

Двойственным образом к понятию наибольшего общего делителя, иными словами, заменой всюду **делит** на **делится**) вводится понятие наименьшего общего кратного. А именно, **общим кратным** чисел x и y называется такое число $m \in \mathbb{Z}$, что $m \vdots x$, $m \vdots y$. **Наименьшим общим кратным** этих чисел называется такое их общее кратное m, которое делит любое другое их общее кратное, иными словами, из того, что $z \vdots x$ и $z \vdots y$ следует $z \vdots m$. Наименьшее общее кратное элементов x и y обычно обозначается $\mathrm{НОК}(x, y)$ или $\mathrm{lcm}(x, y)$ (`least common multiple`).

Если дополнительно предполагать, что $d \geq m$, то оно определено однозначно и удовлетворяет аналогам тождеств, только что перечисленных для наибольшего общего делителя. Кроме того, для натуральных x, y оно связано с наибольшим общим делителем соотношением $\gcd(x, y)\,\mathrm{lcm}(x, y) = xy$.

Перечислим некоторые команды языка `Mathematica`, связанные с только что введенными понятиями.

`Divisors[n]`	список делителей n
`GCD[m,n]`	наибольший общий делитель m и n
`ExtendedGCD[m,n]`	линейное представление GCD
`LCM[m,n]`	наименьшее общее кратное m и n

Функции `GCD` и `LCM` имеют атрибуты `Flat` и `Orderless` и, поэтому, могут вызываться с любым количеством аргументов. Функция `ExtendedCGD[x,y]` возвращает ответ в формате `{d,{a,b}}`, где d — наибольший общий делитель x и y, а a и b — коэффициенты его линейного представления, $d = ax + by$.

1.1. Верно ли, что операции взятия наибольшего общего делителя и наименьшего общего кратного дистрибутивны друг относительно друга? Иными словами, всегда ли выполняются равенства

$$\gcd(\mathrm{lcm}(x, y), z) = \mathrm{lcm}(\gcd(x, z)\gcd(y, z),$$

$$\mathrm{lcm}(\gcd(x, y), z) = \gcd(\mathrm{lcm}(x, z)\,\mathrm{lcm}(y, z)?$$

1.2. Убедитесь, что если m нечетно, то $2^m - 1$ и $2^n + 1$ взаимно просты.

1.3. Найдите наибольший общий делитель всех чисел, получающихся из ланного числа всевозможными перестановками его цифр.

1.4. Для каждого n найдите наименьшее m такое, что n делит m^m.

1.5. Сколько существует натуральных чисел ≤ 10000, для которых разность $2^x - x^2$ не делится на 7?

1.6. Многие числа вида $2^n + 1$ делятся на $2^2 - 1 = 3$. Может ли $2^n + 1$ делиться на $2^m - 1$ при $m \geq 3$?

1.7. Пусть n произвольное натуральное число. Существует ли делящееся на n число, в десятичную запись которого входят только нули и единицы?

§ 2. Деление с остатком

There is division betwixt the Dukes, and a worse matter than that.

William Shakespeare, *King Lear*

Основное известное с глубокой древности свойство целых чисел состоит в том, что если $m, n \in \mathbb{Z}$, причем $n > 0$, то существуют *единственные* $q, r \in \mathbb{Z}$ такие, что $m = qn + r$, где $0 \leq r < n$. Число q называется **неполным частным** (`quotient`) при делении m на n, а число r — **остатком** (`remainder`). Операция, вычисляющая q и r по m и n называется **делением с остатком**. Внутренние функции `Quotient` и `Mod` как раз и ищут по числам m и n их неполное частное и остаток. Заметим, что неполное частное двух целых чисел это в точности *целая часть* их частного, так что `Quotient[m,n]` всегда дает тот же результат, что `Floor[m/n]`.

`Quotient[m,n]`	неполное частное при делении m на n
`Mod[m,n]`	остаток от деления m на n

Предостережение. По этому поводу стоит подчеркнуть, что подавляющее большинство русскоязычных книг по программированию абсолютно невозможно использовать ровно потому, что переводчики не улавливают разницы между словами `ratio` — частное и `quotient` — неполное частное. Полная утрата смысла при переводе с одного языка на другой явление весьма обычное. Например, большинство словарей тракуют слова `accuracy` и `precision`, `speed` и `velocity` как синонимы. Между тем, эти слова не имеют между собой ничего общего: `accuracy` имеет размерность измеряемой величины, а `precision` — величина безразмерная; `velocity` обозначает вектор (скорость), а `speed` — скаляр (модуль скорости). Переводчики компьютерной литературы всегда характеризовались счастливым сочетанием полного незнания английского языка, полного незнания русского языка и полного непонимания смысла происходящего[31]. Именно отсюда возникают всякие анекдотические `остаточные классы` вместо правильных `классов вычетов` и тому подобная галиматья.

2.1. Дайте рекурсивные определения функций `Quotient` и `Mod`.

Решение. Например, так

```
q[0,m_]:=0; r[0,m_]=0;
q[n_,m_]:=q[n-1,m]+If[r[n-1,m]+1==m,1,0];
q[n_,m_]:=(r[n-1,m]+1)*If[r[n-1,m]+1<m,1,0];
```

2.2. Убедитесь, что произведение пяти последовательных целых чисел делится на 120.

2.3. Убедитесь, что $m!n!$ делит $(m+n)!$.

2.4. Какой остаток дает число $2^{2006} - 1$ при делении на $2^{20} - 1$?

[31] Впрочем, в последние годы уточнение `компьютерной литературы` стало излишним.

2.5. Какой остаток дает число 20062006...2006 (100 раз) при делении на 133?

2.6. Убедитесь, что остаток от деления простого числа на 30 равен 1 или простому числу. Какое свойство числа 30 при этом используется?

По умолчанию остаток при делении m на n, возвращаемый функцией `Mod`, лежит между 0 и $n-1$. Однако, во многих теоретико-числовых и комбинаторных алгоритмах полезно использовать **деление с отступом** d, когда m представляется в виде $m = qn + r$, где $d \le r < d + n$.

`Quotient[m,n,d]`	неполное частное при делении m на n с отступом d
`Mod[m,n,d]`	остаток от деления m на n с отступом d

Таким образом, по определению `Mod[m,n,d]=m-Quotient[m,n,d]*n`. В комбинаторных алгоритмах особенно часто используется отступ 1, а в теоретико-числовых — отступ $-n/2$ ("симметричная система вычетов").

2.7. Дайте рекурсивные определения `Quotient[m,n,d]` и `Mod[m,n,d]`.

§ 3. Модулярная арифметика

A mathematician called Ben
Could only count modulo ten
He said 'When I go
Past my last little toe
I have to start over again.'

Рассмотрим некоторое натуральное число m. Говорят, что x, y **сравнимы по модулю** m, и пишут $x \equiv y \pmod m$, если их разность делится на m. Это означает в точности, что x и y дают одинаковые остатки при делении на m.

Например, два числа сравнимы по модулю 2 в том и только том случае, когда они оба одновременно четны или оба одновременно нечетны. Два числа сравнимы по модулю 3 в том и только том случае, когда они оба одновременно делятся на 3, оба одновременно дают остаток 1 или оба одновременно дают остаток 2 при делении на 3.

Отношение сравнимости по модулю m представляет собой отношение эквивалентности на $\mathbb{Z}$. Иными словами, оно обладает следующими свойствами.

- **Рефлексивность**: $x \equiv x \pmod m$.
- **Симметричность**: $x \equiv y \pmod m \iff y \equiv x \pmod m$.
- **Транзитивность**:

$$x \equiv y \pmod m \quad \text{и} \quad y \equiv z \pmod m \qquad \Longrightarrow \qquad x \equiv z \pmod m.$$

Классы этой эквивалентности имеют вид $x = x \mod m = x + m\mathbb{Z}$, они состоят из всех чисел, дающих при делении на m тот же остаток, что x. Эти классы называются **классами вычетов** по модулю m.

Так как остаток при делении любого целого числа на $m > 0$ может принимать лишь значения $0, 1, 2, \ldots, m-1$, то имеется ровно m классов вычетов по модулю m, а именно, классы $0, 1, 2, \ldots, m-1$. Множество классов вычетов по модулю m обозначается обычно $\mathbb{Z}/m\mathbb{Z}$ и называется **кольцом классов вычетов** по модулю m.

Любое множество представителей этих классов называется **полной системой вычетов** по модулю m. Очевидным примером полной системы вычетов по модулю m является набор $0, 1, 2, \ldots, m-1$, но в некоторых вопросах удобнее брать другие системы вычетов, например, для упрощения вычислений в качестве системы вычетов по модулю $m = 2l + 1$ обычно удобнее брать $-l, \ldots, -1, 0, 1, \ldots, l$.

В действительности, отношения сравнения по фиксированному модулю m является не просто отношением эквивалентности на $\mathbb{Z}$, а **конгруэнцией**. Иными словамит, оно согласовано с основными арифметическими операциями.

- Если $x \equiv y \pmod m$ и $z \equiv w \pmod m$, то $x + z \equiv y + w \pmod m$.
- Если $x \equiv y \pmod m$ и $z \equiv w \pmod m$, то $xz \equiv yw \pmod m$.

Эти свойства утверждают, что формулы сложения и умножения по модулю m

$$\overline{x} + \overline{y} = \overline{x + y}, \qquad \overline{x} \cdot \overline{y} = \overline{xy},$$

служат **корректными** (не зависящими от выбора представителей) определениями операций на множестве классов вычетов $\mathbb{Z}/m\mathbb{Z}$. Легко проверить, что эти операции задают на $\mathbb{Z}/m\mathbb{Z}$ структуру коммутативного кольца с 1.

Класс $\overline{x}$ элемента x по модулю m в том и только том случае обратим в $\mathbb{Z}/m\mathbb{Z}$, когда x взаимно прост с m. Кольцо классов вычетов $\mathbb{Z}/m\mathbb{Z}$ по модулю m в том и только том случае является полем, когда $m = p$ — простое число. Построенное нами поле $\mathbb{Z}/p\mathbb{Z}$ из p элементов обозначается обычно $\mathbb{F}_p$ и называется полем из p элементов или простым полем характеристики p. Иногда оно обозначается $\mathbb{F}_(p)$ и называется полем Галуа из p элементов (при этом `GF` является сокращением от `Galois Field`: *In Galois Fields full of flowers primitive elements dance for hours*. Мы детальнее рассмотрим эту тему в выпуске 3, посвященном алгебре.

Приведем для примера таблицы операций по модулям 2, 3 и 4. Иными словами, мы строим кольца $\mathbb{Z}/2\mathbb{Z} = \mathbb{F}_2 = \{0, 1\}$, $\mathbb{Z}/4\mathbb{Z} = \mathbb{F}_3 = \{0, 1, 2\}$ и $\mathbb{Z}/4\mathbb{Z} = \{0, 1, 2, 3\}$ из двух, трех и четырех элементов. При этом для краткости мы опускаем черту над классом и пишем просто x вместо $\overline{x}$.

$+$	0	1
0	0	1
1	1	0

$\times$	0	1
0	0	0
1	0	1

$+$	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

$\times$	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

+	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

×	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

3.1. Реализуйте операции сложения и умножения по модулю m.

3.2. Постройте таблицы сложения и умножения по модулям 5, 6 и 7.

§ 4. Алгоритм Эвклида

> The Euclidean algorithm is the granddaddy of all algoriths, because it is the oldest nontrivial algorithm that has survived to the present day.
>
> Donald Knuth

> Trespassers will be shot, survivors will be shot again!

Из основной теоремы арифметики вытекает, что наибольший общий делитель двух чисел моментально находится, если известно их разложение на простые множители. Однако, практическое разложение на простые множители является совершенно нетривиальной задачей. Тем более замечательно, что с глубокой древности известны быстрые алгоритмы нахождения наибольшего общего делителя и его линейного представления, не требующие разложения на простые множители! Эти алгоритмы основаны на наблюдении, что $\gcd(x, y) = \gcd(x - y, y) = \gcd(y, x - y)$.

Традиционно эти алгоритмы известны под собирательным именем **алгоритма Эвклида**. Версия этого алгоритма была известна в Греции примерно за два века до Эвклида, а его — более эффективные! — варианты с незапамятных времен использовались в древнем Египте, Китае и Индии. Алгоритм Эвклида и его модификации и сегодня остаются одним из основных инструментов модулярной арифметики. Кроме того, он теснейшим образом связан со многими классическими вопросами математики, в частности, с непрерывными дробями.

Описанный в *Элементах* Эвклида алгоритм по сути состоит в следующем[32]. Пусть $x, y \in \mathbb{Z}$, заменяя x и y на их абсолютные величины и пользуясь тем, что $\gcd(x, y) = \gcd(y, x)$, можно с самого начала считать, что $0 \leq y \leq x$. Если $y = 0$, то $\gcd(x, y) = x$. Если же $y \neq 0$, то заменяя (x, y) на $(y, x - y)$, мы получим пару чисел с тем же наибольшим общим делителем.

[32]Разумеется, Эвклид не рассматривал ни отрицательных чисел, ни нуля, да и единица была числом лишь с большой натяжкой, так что его формулировки назойливо репетитивны и тяжеловесны, но *по существу* в предложениях 1 и 2 Книги VII говорится именно это.

При этом большее из чисел y, $x-y$ *меньше*, чем x. Повторяя эту процедуру, мы будем получать все меньшие и меньшие пары натуральных чисел. Ясно, что этот процесс должен оборваться на конечном шаге, а оборваться он может только на паре вида $(d,0)$. Так как наибольший общий делитель пары при этом не изменился, то $\gcd(x,y)=\gcd(d,0)=d$.

4.1. Напишите программу, реализующую певроначальный алгоритм Эвклида.

Решение. Да чего там, нужно просто перечислить все использованные свойства наибольшего общего делителя, а дальше система сама решит, что делать:

```
gcd[x_,y_]:=gcd[Abs[x],Abs[y]] /; x<0||y<0
gcd[x_,0]:=x
gcd[x_,y_]:=gcd[y,x] /; x<y
gcd[x_,y_]:=gcd[y,x-y] /; x>=y
```

Напомним, что `/; = Condition` вызываемое в формате `lhs:=rhs /; test` задает **присваивание с условием**. При этом `lhs` полагается равным `rhs` только если `test` дает значение `True`.

Сегодня вместо вычитания в алгоритме Эвклида обычно используется деление с остатком. А именно, если $x=qy+r$, то $\gcd(x,y)=\gcd(y,r)$, поэтому на шагом алгоритма Эвклида может быть замена пары (x,y) на пару (y,r). Именно в таком варианте этот алгоритм обычно излагается в учебниках алгебры.

4.2. Напишите программу, реализующую версию алгоритма Эвклида, использующую деление с остатком.

Уже в самом примитивном виде алгоритм Эвклида является *несравненно* более эффективным способом нахождения наибольшего делителя двух чисел, чем разложение этих чисел на множители.

4.3. Сравните время работы реализованного Вами алгоритма Эвклида на парах случайных целых чисел с несколькими десятками разрядов и время работы встроенной функции `FactorInteger`.

Эффективность работы алгоритма Эвклида в массовом случае основана на том, что с очень большой вероятностью наибольший общий делитель двух *случайных* чисел очень мал.

4.4. Убедитесь, что с вероятностью $>99\%$ наибольший общий делитель двух случайных 100-разрядных чисел меньше 1000.

В действительности, знаменитая теорема Дирихле 1849 года утверждает, что с вероятностью $6/\pi^2\approx 0.607927$ два случайных целых числа взаимно просты.

4.5. Проверьте утверждение этой теоремы для случайных 100-разрядных чисел.

4.6. Убедитесь, что для любого натурального n существуют натуральные числа x и y такие, что в алгоритме Эвклида требуется ровно n делений.

4.7. Убедитесь, что наименьшие натуральные числа, для которых в алгоритме Эвклида требуется ровно n делений, это числа Фибоначчи $x = F_{n+2}$, $y = F_{n+1}$.

Однако, алгоритм Эвклида далеко не оптимален и его легко улучшить. Во-первых, алгоритм сходится несколько быстрее, если на каждом шаге брать в нем не неотрицательный, а наименьший по модулю остаток — вот где понадобится деление с отступом!

4.8. Напишите программу, реализующую алгоритм Эвклида с выбором на каждом шаге наименьшего по модулю остатка.

Указание. Как и в предыдущей задаче используйте функцию `Mod`, но теперь не от двух, а от трех аргументов.

Еще более существенная экономия получается, если применять деление с остатком только к нечетным числам, а для четных чисел использовать деление пополам. А именно,

- если оба числа x, y четны, то $\gcd(x, y) = 2\gcd(x/2, y/2)$;
- если четно ровно одно из них, скажем y, то $\gcd(x, y) = \gcd(x, y/2)$.

Таким образом, вычисление наибольшего общего делителя двух целых чисел сводится к делению пополам и вычислению наибольшего общего делителя двух *нечетных* чисел. В классическом китайском труде *Математика в девяти книгах* содержится алгоритм вычисления $\gcd(x, y)$, основанный на этой идее. Разумеется, фактически и там не производилось никакого деления с остатком, а меньшее из чисел x или y просто *вычиталось* из большего. В отличие от классического алгоритма Эвклида в данном случае вычитание может быть даже лучше, чем деление с остатком, так как разность двух нечетных чисел четна и, значит, алгоритм сходится очень быстро. Этот алгоритм называется **бинарным алгоритмом**.

4.9. Напишите программу, реализующую классический китайский бинарный алгоритм. То же для варианта бинарного алгоритма, использующего деление с остатком вместо вычитания.

В учебниках `высшей алгебры` обычно говорится, что проделав **обратный ход** в алгоритме Эвклида можно найти линейное представление $d = ax + by$ наибольшего общего делителя $d = \gcd(x, y)$ чисел x и y. Однако, классически известно, что совсем небольшая модификация алгоритма Эвклида позволяет искать наибольший общий делитель d *одновременно* с его линейным представлением. Эта модификация, известная как **расширенный алгоритм Эвклида** = `extended Euclid algorithm`[33], была развита индийскими математиками V–VI века в связи с китайской теоремой об остатках и в явном виде описана Бхаскара I. *Единственное* изменение, которое при этом необходимо внести в обычный алгоритм Эвклида, состоит в том, что вместо целых чисел рассматриваются строки длины 3 с целыми компонентами, последняя из которых отвечает исходному числу, а первые две —

[33]Откуда `ExtendedGCD`. Иногда, в том числе и в русском переводе книги Кнута, *ошибочно* называется `обобщенным` алгоритмом Эвклида.

коэффициентам его представления как линейной комбинации x и y. Таким образом, алгоритм начинается со строк $(1, 0, x)$ и $(0, 1, y)$. Шаг алгоритма состоит в том, что строки (u, v, x) и (w, z, y) заменяются на

• строки (w, z, y) и $(u - w, v - z, x - y)$ в варианте, использующем вычитание;

• строки (w, z, y) и $(u - qw, v - qz, r)$, где $x = qy + r$, в варианте, использующем деление с остатком.

Так как при этом третья координата ведет себя так же, как остатки в обычном алгоритме Эвклида, то через конечное число шагов мы неминуемо придем к строкам вида (a, b, d) и $(u, v, 0)$, в этот момент алгоритм останавливается и мы можем заключить, что $d = \gcd(x, y)$ и $d = ax + by$.

4.10. Напишите программы, реализующие расширенный алгоритм Эвклида в обоих вариантах.

Отметим две замечательные особенности этого алгоритма. Во-первых, он является самопроверяющимся, так как наибольший делитель, это в точности общий делитель, допускающий линейное представление. Во-вторых, он дает полезный субпродукт (= `by-product`). А именно, тот факт, что вторая строка, получающаяся на момент остановки алгоритма, равна $(u, v, 0)$, означает, что $ux + vy = 0$.

4.11. Что можно сказать об u и v, получающихся на момент остановки расширенного алгоритма Эвклида?

Указание. Проведите эксперимент!

Все описанные выше алгоритмы моментально обобщаются на случай любого количества чисел. Конечно, можно воспользоваться ассоциативностью операции gcd и искать наиброльший общий делитель нескольких чисел по индукции, например, для трех чисел так $\gcd(x, y, z) = \gcd(\gcd(x, y), z)$.

4.12. Реализуйте рекурсивный алгоритм для вычисления наибольшего общего делителя нескольких чисел и его линейного представления.

Для небольших примеров рекурсивный алгоритм будет работать, но легко предложить гораздо более эффективные алгоритмы!

4.13. Реализуйте несколько различных вариантов алгоритма Эвклида для нескольких чисел и сравните скорость их работы с рекурсивным алгоритмом.

4.14. Реализуйте расширенный алгоритма Эвклида для нескольких чисел.

Указание. Для s чисел этот алгоритм оперирует с s целочисленными строками длины $s + 1$, но имеются различные варианты организации шага алгоритма.

К счастью, для всех практических целей эти потуги ускорить работу алгоритма Эвклида излишни, так как мы можем с успехом использовать внутренние функции `GCD` и `ExtendedGCD`. Имплементация этих функций включает не только эффективную комбинацию алгоритма Эвклида и бинарного алгоритма, но и быстрые алгоритмы точной арифметики, развитые в последние 15–20 лет.

§ 5. Китайская теорема об остатках

Я не слыхал рассказов Оссиана,
Не пробовал старинного вина —

Осип Мандельштам

Сейчас мы установим фундаментальный результат, который позволяет

- сводить изучение сравнений к примарному случаю,
- сводить вычисления по большому модулю к вычислениям по нескольким маленьким модулям.

Этот результат является основным инструментом при проведении быстрых вычислений с очень большими числами.

Для случая произвольного количества *взаимно простых* модулей этот результат был известен Сунь Цзу в первом веке нашей эры и использовался для логистических и астрономических вычислений. Для случая двух модулей излагаемый ниже алгоритм в явном виде содержится в трактате Ариабхаты 499 года. В 1247 году Чин Чжу-Шао обобщил теорему на случай произвольного количества не обязательно взаимно простых модулей.

Начнем с ключевого случая двух взаимно простых модулей, который позволяет провести индукцию. Пусть $m \perp n$ — два взаимно простых модуля. Тогда **китайская теорема об остатках** утверждает, что для любых a, b существует такое x, что

$$\begin{cases} x \equiv a \pmod{m} \\ x \equiv b \pmod{n} \end{cases}$$

причем это x единственно по модулю mn. Иными словами, если дополнительно предполагать, что $0 \leq x < mn$, то x единственно.

Начнем с простого но важного замечания, что

$$x \equiv y \pmod{m} \quad \text{и} \quad x \equiv y \pmod{n} \qquad \Longrightarrow \qquad x \equiv y \pmod{mn}.$$

В самом деле, $x - y$ делится как на m так и на n, а, значит, делится на $\operatorname{lcm}(m, n)$. Но m и n взаимно просты, так что $\operatorname{lcm}(m, n) = mn$. Тем самым, единственность x очевидна и существование следует теперь из принципа Дирихле! Это значит, что если мы просто переберем *все* числа от 0, до $mn - 1$, то мы обязательно наткнемся на решение этой системы.

5.1. Реализуйте полный перебор, находящий x в китайской теореме для случая двух модулей.

Впрочем, описанный метод решения трудно назвать *эффективным* алгоритмом. Хотелось бы явно предъявить *какое-то* решение этой системы. Так как $m \perp n$, то найдутся такие c и d, что $cm + dn = 1$. Их можно найти, например, при помощи расширенного алгоритма Эвклида. Теперь в качестве x можно взять $x = dna + cmb$. В самом деле,

$$x \equiv dna + cma \equiv a \pmod{m}, \qquad x \equiv dnb + cmb \equiv b \pmod{n}.$$

Однако, указанное решение может быть $\geq mn$. Чтобы получить решение, удовлетворяющее неравенствам $0 \leq x < mn$, нужно еще взять остаток $dna + cmb$ по модулю mn.

5.2. Реализуйте алгоритм, вычисляющий x в китайской теореме для случая двух модулей.

В действительности, такой алгоритм реализован в пакете

`NumberTheory‘NumberTheoryFunctions‘`

под именем `ChineseRemainder`. Для случая двух модулей эта функция вызывается в формате `ChineseRemainder[{a,b},{m,n}]`.

С математической точки зрения эта теорема утверждает, что $\mathbb{Z}/mn\mathbb{Z} \cong \mathbb{Z}/m\mathbb{Z} \oplus \mathbb{Z}/n\mathbb{Z}$, иными словами, вычисления по модулю mn *полностью* сводятся к вычислениям отдельно по модулю m и по модулю n.

Китайская теорема об остатках непосредственно обобщается на случай любого количества модулей. А именно, пусть $m = m_1 \dots m_s$, где m_i попарно взаимно просты, а $x_1, \dots, x_s$ — произвольные целые числа. Тогда существует такое x, что

$$\begin{cases} x \equiv x_1 \pmod{m_1} \\ \dots \\ x \equiv x_s \pmod{m_s} \end{cases}$$

причем это x единственно по модулю m. Иными словами, утверждается, что если m_i попарно взаимно просты, то имеется изоморфизм колец

$$\mathbb{Z}/m_1 \dots m_s\mathbb{Z} \cong \mathbb{Z}/m_1\mathbb{Z} \oplus \dots \oplus \mathbb{Z}/m_s\mathbb{Z},$$

так что вычисления по модулю $m = m_1 \dots m_s$ полностью сводятся к вычислениям отдельно по модулю каждого из m_i.

Проще всего доказать китайскую теорему по индукции. Она же, естественно, дает и алгоритм для нахождения x. А именно, предположим, что мы уже умеем решать аналогичную систему для $s-1$ взаимно простого модуля. Пусть y удовлетворяет первым $s-1$ сравнению по модулям $m_1, \dots, m_{s-1}$. Тогда x можно искать как решение системы

$$\begin{cases} x \equiv y \pmod{m_1 \dots m_{s-1}} \\ x \equiv x_s \pmod{m_s} \end{cases}$$

5.3. Реализуйте рекурсивный алгоритм, находящий x в китайской теореме для любого количества модулей.

Впрочем, можно не использовать индукции, а сразу указать решение. Для этого положим $n_i = m_1 \dots \widehat{m}_i \dots m_s$, и заметим, что числа n_i взаимно просты *в совокупности*. Тем самым, существуют a_i такие, что $a_1n_1 + \dots + a_sn_s = 1$, их можно найти, например, при помощи обобщенного алгоритма Эвклида. Остается положить

$$x = a_1n_1x_1 + \dots + a_sn_sx_s,$$

и, если мы хотим получить каноническое решение, поделить это x с остатком на $m_1 \dots m_s$

5.4. Реализуйте алгоритм, непосредственно вычисляющий x в китайской теореме для любого количества модулей.

После подгрузки упомянутого выше пакета x можно искать вызывая функцию `ChineseRemainder` в формате

`ChineseRemainder[{x1,...,xs},{m1,...,ms}]`.

5.5. Пусть $x_s \equiv i \pmod{p_i}$ для первых s простых p_i, причем $0 \le x < p_1 \dots p_s$. Верно ли, что при любом $s \ge 2$ это x_s является простым?

Ответ. Первые шесть из них действительно простые:

$$x_2 = 5,\ x_3 = 23,\ x_4 = 53,\ x_5 = 1523,\ x_6 = 29243,\ x_7 = 299513,$$

но вот $x_8 = 4383593 = 23 \cdot 190591$ простым не является. Следующее $x_9 = 188677703$ снова простое, но вот $x_{10} = 5765999453 = 41 \cdot 131 \cdot 809 \cdot 1327$ имеи четыре простых делителя. Дальше простые встречаются, но довольно редко, x_{18}, x_{63}, x_{105} и т.д.

5.6. Пусть $x_s \equiv (-1)^{i-1} \pmod{p_i}$ для первых s простых p_i, причем $0 \le x < p_1 \dots p_s$. Верно ли, что при любом $s \ge 2$ это x_s является простым?

Ответ. Первые пять из них действительно простые:

$$x_2 = 5,\ x_3 = 11,\ x_4 = 41,\ x_5 = 881,\ x_6 = 14741,$$

но $x_7 = 74801 = 131 \cdot 571$ простым не является.

Для практически встречающихся задач (несколько сотен относительно небольших модулей) описанный выше классический алгоритм вполне эффективен. Однако, он требует нахождения линейного представления 1 и вычисления остатка по модулю $m = m_1 \dots m_s$. Для *действительно* больших чисел обычно используется более эффективный **алгоритм Гарнера**[34], который не требует деления на m. На русском языке детали можно найти в [K2] или [Vas].

> The reader will be content to wait for a full explanation of these matters till the next year, — when a series of things will be laid open which he little expects.
>
> Laurence Sterne, *Tristram Shandy*

[34] H.Garner, The residue number system. — IRE Trans. Electronic Computers, 1959, vol.8, p.140–147.

Глава 6. ПРОСТЫЕ ЧИСЛА

Die Mathematik ist die Königin der Wissenschaften, und die Zahlentheorie ist die Königin der Mathematik[35].

Карл Фридрих Гаусс

Mathematician: 3 is a prime, 5 is a prime, 7 is a prime,
Physicist: 3 is a prime, 5 is a prime, 7 is a prime, 9 is an experimental error, 11 is a prime,...
Engineer: 3 is a prime, 5 is a prime, 7 is a prime, 9 is a prime, 11 is a prime,...
Programmer: 3 is a prime, 5 is a prime, 7 is a prime, 7 is a prime, 7 is a prime,...
Salesperson: 3 is a prime, 5 is a prime, 7 is a prime, 9 – we'll do for you the best we can,...
Computer Software Salesperson: 3 is a prime, 5 is a prime, 7 is a prime, 9 will be prime in the next release,...
Biologist: 3 is a prime, 5 is a prime, 7 is a prime, 9 – results have not arrived yet,...
Advertiser: 3 is a prime, 5 is a prime, 7 is a prime, 11 is a prime,...
Lawyer: 3 is a prime, 5 is a prime, 7 is a prime, 9 — there is not enough evidence to prove that it is not a prime,...
Accountant: 3 is a prime, 5 is a prime, 7 is a prime, 9 is a prime, deducing 10% tax and 5% other obligations.
Statistician: Let's try several randomly chosen numbers: 17 is a prime, 23 is a prime, 11 is a prime...
Professor: 3 is a prime, 5 is a prime, 7 is a prime, and the rest are left as an exercise for the student.
Computational linguist: 3 is an odd prime, 5 is an odd prime, 7 is an odd prime, 9 is a very odd prime,...
Psychologist: 3 is a prime, 5 is a prime, 7 is a prime, 9 is a prime but tries to suppress it,...

Pure mathematics, on the other hand, seems to me a rock on which all idealism founders: 317 is a prime, not because we think so, or because our minds are shaped in one way rather than another, but *because it is so*, because mathematical reality is built that way.

Godfrey Harold Hardy[36]

[35]Математика королева наук, а теория чисел королева математики.

[36]Обычно мы не приводим переводов с английского. В данном случае такой перевод необходим. “С другой стороны, чистая математика представляется мне скалой, о которую разбивается всякий идеализм: число 317 простое не потому, что нам так кажется, и не потому, что так а не иначе создан наш разум, а *потому, что это так*, потому что так устроена математическая реальность.” — Г.Г.Харди, Апология математика. — RCD, Ижевск, 2000, с.1–102. Интересно, что в русском переводе смысл фразы заменен на *прямо противоположный*: “скалой, на которой *зиждется* всякий идеализм”. Мало того, что переводчик не отличает `founders` от `is founded`, он к тому же не понимает смысл и пафос всего окружающего текста!

Содержание этой главы основано на книге Рибенбойма (имеются также слегка переработанный португальский текст и более короткая популярная версия этой книги[37,38], а также цитированный в предисловии сокращенный русский перевод предварительного варианта книги). Многие исторические детали взяты из книг Наркевича и Серпиньского, а вычислительные аспекты всех рассматриваемых здесь проблем в общих чертах обсуждаются в книгах Ризеля и Крандалла—Померанса. Однако, конкретные детали меняются так быстро, что за ними можно следить только по `Internet` сайтам.

§ 1. Простые числа

Enter any 11-digit prime number to continue...

Натуральное число $p \neq 1$ называется **простым**, если у него нет собственных делителей, т.е. делителей, отличных от самого p и 1. В системе `Mathematica` имеется несколько важных и удобных функций для работы с простыми числами.

`Prime[n]`	n-е простое число
`PrimePi[n]`	количество простых $\leq n$
`PrimeQ[n]`	тест простоты n
`ProvablePrimeQ[n]`	детерминистический тест простоты n
`Primes`	домен простых чисел

Использование этих функций ясно само по себе. Стоит, однако, подчеркнуть, что имплементация теста `PrimesQ` в системе вероятностная. *Известно*, что этот тест дает *достоверный* ответ для всех простых $\leq 10^{16}$ — и, тем самым во всех рассматриваемых в настоящей главе примеров! Для получения достоверного, а не просто в высшей степени правдоподобного ответа в общем случае необходимо использовать функцию `ProvablePrimeQ` из пакета `NumberTheory‘PrimeQ‘`.

1.1. Породите список первых n простых.

Ответ. Проще всего так: `Table[Prime[i],{i,1,n}]`. Конечно, для фактического вывода такого списка нужно подставить вместо n конкретное значение.

1.2. Определите номер простого числа p.

1.3. Породите список всех простых, не превосходящих n.

Ответ. Можно, например, так: `Table[Prime[i],{i,1,PrimePi[n]}]`.

1.4. Найдите наибольшее простое строго меньшее, чем n.

Ответ. Например, так: `Prime[PrimePi[n]-1]`.

[37] P.Ribenboim, Números primos: mistérios e records. Ass. Inst. Nac. Math. Pura e Appl. Rio de Janeiro, 2001.

[38] P.Ribenboim, The little book of big primes, Springer, N.Y. et al, 1991.

1.5. Найдите m наибольших простых строго меньших, чем n.

Решение. Можно, конечно, взять m последних элементов из списка всех простых, меньших, чем n, но это плохая идея. Конечно, скорость работы следующей программы тоже уменьшается с ростом n, но совсем не так быстро.

```
prevprimes[n_,m_]:=
NestList[Prime[PrimePi[#-1]]&,Prime[PrimePi[n-1]],m-1]
```

Для чисел порядка 10^9 команда `prevprimes` для небольших значений m работает сотые доли секунды, в то время как порождение всего списка простых меньщих m требует нескольких минут.

1.6. Найдите наименьшее простое строго большее, чем n.

Решение. Вот простая процедурная программа из книги Стена Вагона [Wa], решающая эту задачу:

```
PrimeAfter[1]:=2
PrimeAfter[n_]:=Module[{p=n+1+Mod[n,2]},
                       While[Not[PrimeQ[p]],p+=2];p]
```

Поскольку сами мы редко организуем циклы, поясним, что именно здесь происходит. Так как первая строка в комментарии не нуждается, объясним вторую. Команда `Module` используется для локализации переменной p, которой придается первое *нечетное* значение большее n. После этого проверяется, будет ли это p простым и, если нет, его значение инкрементируется на два (`p+=2` это просто программистское сокращение для `p=p+2`), пока получившееся p не станет простым. После чего возвращается получившееся значение p.

1.7. Найдите первые четыре простых числа вида $1\ldots1$.

Предостережение. У начинающего возникнет искушение задать число, состоящее из n единиц, посредством

```
ones[n_]:=Sum[10^i,{i,0,n-1}]
```

Следует, однако, иметь в виду, что это один из многочисленных случаев, когда гораздо естественнее воспользоваться внутренними командами работы со списками, в данном случае списком цифр

```
ones[n_]:=FromDigits[Table[1,{n}]
```

Конечно, для совсем небольших чисел, имеющих лишь несколько сотен разрядов, разница во времени работы этих конструкций малозаметна. Однако, с ростом n время выполнения первой конструкции растет как n^3, а второй как $2n$. Поэтому для чисел с несколькими тысячами разрядов конструкции, использующие списки, работают в сотни раз быстрее, чем явные арифметические вычисления. С другой стороны, для чисел с сотнями миллионов разрядов реальным ограничением второй констукции становится используемая память. Поэтому если Вы хотите работать с *по-настоящему* большими числами, правильнее всего задать число, состоящее из n единиц, как

`(10^n-1)/9`. К сожалению, раскладывать числа такого порядка на множители мы все равно не умеем.

1.8.

Найдите первые 100 простых чисел, состоящих из

- единиц и троек.
- единиц и пятерок.
- единиц и семерок.
- единиц и девяток.

Какие из них встречаются реже всего и почему?

Вот еще одна классическая задача. Простые числа p и $p+2$, разность которых равна 2, называются **близнецами**.

1.9. Найдите первые 1000 пар близнецов. Как меняется отношение количества близнецов $\le n$ к общему количеству простых чисел $\le n$?

Ответ. Например, при помощи следующей конструкции:

```
Select[Table[{Prime[i],Prime[i]+2},{i,1,n}],PrimeQ[#[[2]]]&]
```

Варварское, но верное решение состоит теперь в том, чтобы взять такое n, для которого этот список заведомо содержит больше 100 пар, например, $n = 1000$, и выбрать из него первые 100 посредством `Take[list,100]`.

За исключением случая 3, 5, 7 числа вида $p, p+2, p+4$ не могут быть все три простыми (кстати, почему?) — но вот числа $p, p+2, p+6$ и $p, p+4, p+6$ могут. Еще интереснее искать **четверки** простых чисел $p, p+2, p+6, p+8$. Первая такая четверка — это 5, 7, 11, 13, следующая — 11, 13, 17, 19.

1.10. Найдите первые 100 четверок. Есть ли среди них еще хотя бы одна такая, для которой последние цифры не есть 1, 3, 7, 9?

Ответ. Перечислим лишь те четверки, которые меньше 10000. Кроме двух уже упомянутых, их еще всего 10 штук:

101	103	107	109	191	193	197	199
821	823	827	829	1481	1483	1487	1489
1871	1873	1877	1879	2081	2083	2087	2089
3251	3253	3257	3259	3461	3463	3467	3469
5651	5653	5657	5659	9431	9433	9437	9439

1.11. Убедитесь, что четверки встречаются очень часто.

Указание. Искать большие четверки лучше, конечно, не посредством `Select`, а организуя цикл начиная с некоторого места.

Ответ. Набрав что-нибудь в духе

```
Timing[Block[{i=10000000000000000001},While[
      Not[PrimeQ[i]&&PrimeQ[i+2]&&PrimeQ[i+6]&&PrimeQ[i+8]],
            i=i+10];{i,i+2,i+6,i+8}]]
```

мы в течение секунд увидим что-нибудь в духе

$$100000000000000081721 \quad 100000000000000081723$$
$$100000000000000081727 \quad 100000000000000081729$$

Для чисел небольшого порядка обычно это требует не больше миллиона итераций. Это значит, что четверок очень много.

1.12. Найдите первые 100 простых вида n^2+1.

1.13. Найдите первые 100 простых вида n^2+n+1.

§ 2. Теорема Эвклида

> I was interviewed in the Israeli Radio for five minutes and I said that more than 2000 years ago, Euclid proved that there are infinitely many primes. Immediately the host interuppted me and asked: “Are there still infinitely many primes?”
>
> Noga Alon

В десятой книге “Элементов” Эвклида содержится следующее утверждение, известное как **теорема Эвклида**: множество рациональных простых чисел бесконечно. Основная идея доказательства теоремы Эвклида состоит в том, чтобы построить бесконечную последовательность попарно взаимно простых чисел $\neq 1$.

Особенно выпукло использованный для этого прием можно выразить следующим образом. Пусть $n_1, \dots, n_s$ — любые натуральные числа. Тогда число $n_1 \dots n_s + 1$ взаимно просто с каждым из $n_1, \dots, n_s$. Определим теперь рекуррентно последовательность чисел E_i, $i \in \mathbb{N}$, полагая $E_1 = 2$ и $E_i = E_1 \dots E_{i-1} + 1$ для всех $i \geq 2$. В силу только что сказанного все E_i представляют собой попарно взаимно простые числа > 1. Пусть теперь q_i — наименьший простой делитель числа E_i. Тогда все q_i, $i \in \mathbb{N}$, попарно различные простые числа.

Кнут предложил называть E_i **числами Эвклида**. Таким образом, они имеют бесконечно много различных простых делителей. С другой стороны, Одони[39] доказал, что имеется бесконечно много простых чисел, которые *не* делят ни одно из чисел Эвклида.

Ясно, что

$$E_2 = 2+1 = 3, \qquad E_3 = 2\cdot 3 + 1 = 7, \qquad E_4 = 2\cdot 3\cdot 7 + 1 = 43.$$

Обратите внимание, что все это простые числа!

2.1. Напишите рекуррентную программу для вычисления чисел Эвклида. Верно ли что все они простые?

Ответ. Нет, уже следующее число Эвклида $E_5 = 2\cdot 3\cdot 7\cdot 43 + 1 = 1807$ не простое: $1807 = 13\cdot 139$. Число $E_6 = 3263443$ снова простое, но, насколько

[39] R.W.K.Odoni, On the prime divisors of the sequence $w_{n+1} = 1 + w_1 w_2 \dots w_n$. — J. London Math. Soc. 1985, vol.32, p.1–11.

нам известно, среди E_n, $n \geq 7$, не удалось найти ни одного простого числа. Большинство специалистов верят, что все они составные.

2.2. Найдите разложения нескольких следующих чисел Эвклида E_n на простые. В тех случаях, когда это невозможно, найдите какие-то их простые делители.

Указание. Разложить числа Эвклида начиная с E_{10} на простые множители при помощи стандартных внутренних функций `Mathematica` нет никаких шансов. Попробуйте использовать функцию `FactorIntegerECM`, определенную в пакете `NumberTheory‘FactorIntegerECM‘`.

Ответ. Так как при переходе от числа Эвклида E_n к числу Эвклида E_{n+1} количество цифр почти удваивается, то в E_{30} уже больше 109 миллионов цифр. Поскольку числа Эвклида — и их наименьшие простые делители! — настолько быстро растут, уже при $n \geq 11$ поиск их явных разложений на простые множители на бытовом компьютере при помощи элементарных методов чрезвычайно затруднителен или прямо невозможен. Вот разложения нескольких первых из них:

$$E_7 = 10650056950807 = 547 \cdot 607 \cdot 1033 \cdot 31051,$$
$$E_8 = 113423713055421844361000443 = 29881 \cdot 67003 \cdot 9119521 \cdot 6212157481,$$
$$E_9 = 128649386832786717405371459983609615466532594851958 07 =$$
$$5295435634831 \cdot 31401519357481261 \cdot 77366930214021991992277,$$

Уже число E_{10} содержит 105 знаков, поэтому ограничимся указанием тех простых делителей нескольких следующих чисел, которые ищутся в течение 1–2 минут:

E_{10}	181	1987	112374829138729
E_{11}	2287		
E_{12}	73		
E_{13}	2589377038614498251653		
E_{14}	52387	5020387	5783021473
E_{15}	13999	74203	9638659

Обратите внимание, что двадцати двух разрядное число, указанное в качестве простого множителя числа Эвклида E_{13} действительно является простым! Оно найдено при помощи команды `FactorIntegerECM`. Команда `FactorInteger` надежно ищет лишь простые множители содержащие не более *двенадцати* разрядов.

Однако сам Эвклид использовал для доказательства бесконечности множества простых не числа Эвклида, а приориалы. Пусть q простое число. Произведение $q\# = \prod p$ всех простых $p \leq q$ не превосходящих q называется **приориалом** числа q. Обозначение $q\#$ было предложено в 1987 году Дабнером[40] и в настоящее время стало общепринятым. Так как $q\#$ делится на

[40]H.Dubner, Factorial and primorial primes. — J. Recr. Math., 1987, vol.19, p.197–203.

все простые $\le q$, то ни $q\# + 1$ ни $q\# - 1$ не делятся ни на одно из них и, значит, содержат новые простые делители. Довольно часто эти числа часто содержат очень большие простые множители, а в некоторых случаях сами являются большими простыми (`primorial primes`). В следующих задачах простой множитель q числа n называется большим, если $q > n/q$, и, кроме того, в случае, когда $n/q = p$ само является простым, $q > p^2$.

2.3. Сколько среди чисел вида $n\# + 1$, $1 \le n \le 50$, простых?

Ответ. Никаких других простых, кроме очевидных $2\#+1 = 3$, $3\#+1 = 7$, $5\# + 1 = 31$, $7\# + 1 = 211$ и $11\# + 1 = 2311$ не просматривается.

2.4. Вычислите разложение на множители чисел вида $n\# + 1$, $n \le 50$. Какое наибольшее количество различных простых множителей при этом встречается? В каких случаях встречаются большие простые множители?

2.5. Сколько среди чисел вида $n\# - 1$, $1 \le n \le 50$, простых?

Ответ. А вот здесь, кроме очевидных простых $3\# - 1 = 5$, $5\# - 1 = 29$, $11\# - 1 = 2309$ и $13\# - 1 = 30029$ есть еще одно, уже совсем не очевидное:

$$89\# - 1 = 237687418963455507706505376013583 09.$$

2.6. Вычислите разложение на множители чисел вида $n\# - 1$, $1 \le n \le 50$. Какое наибольшее количество различных простых множителей при этом встречается? В каких случаях встречаются большие простые множители?

Вместо приориалов для доказательства теоремы Эвклида можно было бы использовать и факториалы. Снова $n!$ делится на все простые $\le n$ и, значит, $n! + 1$ и $n! - 1$ содержат новые простые делители. Опять же, очень часто эти числа содержат громадные простые множители, а в некоторых случаях и сами являются простыми (**factorial primes**).

2.7. Сколько среди чисел вида $n! + 1$, $2 \le n \le 50$ простых? Квадратов простых?

Указание. На последних из этих чисел `FactorInteger` работает очень медленно, пользуйтесь `FactorIntegerECM`.

Ответ. Кроме очевидных случаев $1!+1 = 2$, $2!+1 = 3$, $3!+1 = 7$ простыми оказываются только

$$11! + 1 = 39916801,$$
$$27! + 1 = 10888869450418352160768000001,$$
$$37! + 1 = 13763753091226345046315979581580902400000001,$$
$$41! + 1 = 33452526613163807108170062053440751665152000000001.$$

Единственные квадраты простых встречаются в самом начаале: $4! + 1 = 25 = 5^2$, $5! + 1 = 121 = 11^2$ и $7! + 1 = 5041 = 71^2$.

2.8. Вычислите разложение на множители чисел вида $n! + 1$, $e \le n \le 50$. Какое наибольшее количество различных простых множителей при этом встречается? В каких случаях встречаются большие простые множители?

2.9. Сколько среди чисел вида $n! - 1$, $2 \leq n \leq 50$, простых?

Ответ. Кроме очевидных случаев $3! - 1 = 5$, $4! - 1 = 23$, $6! - 1 = 719$ и $7! - 1 = 5039$ простыми оказываются только

$$12! - 1 = 479001599,$$
$$30! - 1 = 265252859812191058636308479999999,$$
$$32! - 1 = 263130836933693530167218012159999999,$$
$$33! - 1 = 8683317618811886495518194401279999999,$$
$$38! - 1 = 52302261746660111176000722410007429119999999 9.$$

2.10. Вычислите разложение на множители чисел вида $n! + 1$, $2 \leq n \leq 50$. Какое наибольшее количество различных простых множителей при этом встречается? В каких случаях встречаются большие простые множители?

§ 3. Теорема Банга—Жигмонди

I can do without essentials but I must have my luxuries.

Oscar Wilde

В действительности, чтобы найти бесконечное количество простых чисел, не обязательно даже, чтобы члены последовательности были попарно взаимно просты, достаточно, чтобы (начиная с некоторого места) каждый следующий член имел *примитивный* простой делитель, не являющийся делителем ни одного из предыдущих членов этой последовательности.

Оказывается, каждая пара *взаимно простых* натуральных чисел $x > y$ порождает такую последовательность, n-м членом которой является $x^n - y^n$. Банг (1886 год, в частном случае $y = 1$) и Жигмонди[41] (1892 год, в общем случае) обнаружили следующий исключительно важный факт, очень часто используемый в теории чисел, комбинаторике и теории групп: $x^n - y^n$ *почти всегда* имеет **примитивный** простой делитель, который не делит никакую из разностей $x^m - y^m$ при $m < n$. В дальнейшем теорема Банга—Жигмонди многократно переоткрывалась и известна под разными названиями. В старых учебниках теории чисел она чаще всего называется теоремой Биркгофа—Вандивера[42].

Одним из забавных следствий этой теоремы является утверждение, что для любого s существует простое число p такое, что десятичное разложение $1/p$ имеет период s.

Очевидные исключения получаются здесь при $n = 1$ и $x = y + 1$ и при $n = 2$, $x + y = 2^k$ для некоторого k. Кроме того, имеется еще одно не совсем очевидное исключение.

---

[41] K.Zsigmondy, Zur Theorie der Potenzreste. — Monatshefte Math. Phys., 1892, Bd.3, S.265–284.

[42] G.D.Birkhoff, H.S.Vandiver, On the integral divisors of $a^n - b^n$. — Ann. Math., 1904, vol.5, p.173–180.

3.1. Имеется еще *ровно одна* тройка (x, y, n), для которой $x^n - y^n$ не имеет примитивных простых делителей. Найдите ее.

Ответ. Организовав полный перебор легко обнаружить, что при $x = 2$, $y = 1$, и $n = 6$ число $2^6 - 1 = 63 = 3^2 \cdot 7$ не имеет примитивных простых делителей: $2^2 - 1 = 3$ и $2^3 - 1 = 7$.

3.2. Укажите для каждого $n \le 150$ наименьший примитивный простой делитель $2^n - 1$. Что можно сказать об их величине?

Решение. Обозначим наименьший простой делитель $2^n - 1$ через q_n. Множество *всех* новых простых делителей $2^n - 1$ можно найти, например, так:

```
new[n_]:=Complement[First[Transpose[FactorInteger[2^n-1]]],
          Apply[Union,Table[
               First[Transpose[FactorInteger[2^i-1]]],
                                                {i,2,n-1}]]]
```

Теперь q_n это просто первый элемент `new[n]`. Начало ответа легко посчитать в уме: $q_2 = 3$, $q_3 = 7$, $q_4 = 5$, $q_5 = 31$, $q_7 = 127$, $q_8 = 17$, $q_9 = 73$, $q_{10} = 11$. Вот еще несколько значений:

$q_{11} = 23$	$q_{12} = 13$	$q_{13} = 8191$	$q_{14} = 43$
$q_{15} = 151$	$q_{16} = 257$	$q_{17} = 131071$	$q_{18} = 19$
$q_{19} = 524287$	$q_{20} = 41$	$q_{21} = 337$	$q_{22} = 683$
$q_{23} = 47$	$q_{24} = 241$	$q_{25} = 601$	$q_{26} = 2731$
$q_{27} = 262657$	$q_{28} = 29$	$q_{29} = 233$	$q_{30} = 331$
$q_{31} = 2147483647$	$q_{32} = 65537$	$q_{33} = 599479$	$q_{34} = 43691$
$q_{35} = 71$	$q_{36} = 37$	$q_{37} = 223$	$q_{38} = 174763$
$q_{39} = 79$	$q_{40} = 61681$	$q_{41} = 13367$	$q_{42} = 5419$
$q_{43} = 431$	$q_{44} = 397$	$q_{45} = 631$	$q_{46} = 2796203$
$q_{47} = 2351$	$q_{48} = 97$	$q_{49} = 4432676798593$	$q_{50} = 251$

3.3. Укажите для каждого $n \le 120$ наименьший примитивный простой делитель $3^n - 1$. А теперь выберете те n, для которых $(3^n - 1)/2$ простое.

3.4. Укажите для каждого $n \le 110$ наименьший примитивный простой делитель $3^n - 2^n$. А теперь попробуйте найти его еще для 10 значений n. Как Вы думаете, с чем связан столь резкий рост времени, необходимого для факторизации?

Ответ. Около 5/6 этого времени идет на факторизацию одного числа, а именно,

$$3^{118} - 2^{118} = 5 \cdot 19471 \cdot 145142890373531870546641 \cdot$$
$$14130386091162273752461387579$$

Множители с 24 и 29 цифрами достаточно близки для того, чтобы система могла их найти с помощью квадратичного решета, но недостаточно близки для того, чтобы она могла это сделать за секунду.

3.5. Убедитесь, что если x и y взаимно просты, то каждый примитивный простой делитель p числа $x^n - y^n$ удовлетворяет сравнению $p \equiv 1 \pmod{n}$.

3.6. Что будет, если заменить последовательность $x^n - y^n$ на последовательность $x^n + y^n$? Всегда ли верно, что $x^n + y^n$ имеет примитивный простой делитель, не делящий никакую из сумм $x^m + y^m$ при $m \leq n$, или тут тоже есть исключения?

Вот еще одна забавная иллюстрация теоремы Банга—Жигмонди.

3.7. Исследуйте факторизацию чисел

$$(p_1 \dots p_n + 1)^{2^m} - 1,$$

где $p_1, \dots, p_n$ суть первые n нечетных простых и убедитесь, что это произведение имеет не менее $n + m$ различных простых делителей.

§ 4. Простые Мерсенна

> The trouble with integers is that we have examined only the very small ones. Maybe all the exciting stuff happens at really big numbers, ones we can't even begin to think about in any very definite way. Our brains have evolved to get us out of the rain, find where the berries are, and keep us from getting killed. Our brains did not evolve to help us grasp really large numbers or to look at things in a hundred thousand dimensions.
>
> Ronald L. Graham

В этом и следующем параграфе мы обсудим два важнейших классических класса простых, возникающие во многих вопросах теории чисел, алгебры и комбинаторики.

Если m — собственный делитель n, то $x^n - 1$ делится на $x^m - 1$. Поэтому если $M_n = 2^n - 1$ простое, то n простое. Большинство ранних авторов были уверены, что верно и обратное, т.е. если p простое, то M_p тоже простое. Это заблуждение было развеяно в 1536 году Худальрикусом Региусом, который заметил, что $M_{11} - 1 = 23 \cdot 89$. В 1588 году Пьетро Катальди проверил, что M_{17} и M_{19} простые, при этом он заявил, что M_{23}, M_{29}, M_{31} и M_{37} тоже простые. В 1640 году Пьер Ферма проверил, что в действительности M_{23} и M_{37} составные. Позже Эйлер заметил, что M_{29} составное, а в 1772 году он показал, что M_{31} простое.

В связи с проблемой четных совершенных чисел Марин Мерсенн в 1644 году утверждал, что числа

$$M_p, \qquad p = 2,\ 3,\ 5,\ 7,\ 13,\ 17,\ 19,\ 31,\ 67,\ 127,\ 257$$

просты, а все остальные числа M_p для $p \leq 257$ составные. Как позже выяснилось, этот список содержал ошибки. Числа вида $M_p = 2^p - 1$, где p

простое, называются **числами Мерсенна**. Почти все самые большие известные простые числа являются числами Мерсенна.

4.1. Найдите первые 15 простых чисел Мерсенна и исправьте все ошибки в его списке.

Ответ. Можно, например, так:

```
Select[Table[2^Prime[n]-1,{n,1,300}],PrimeQ]
```

Топорно, но для таких маленьких чисел это не имеет никакого значения. Напомним, что функция `Select[list,crit]` осуществляет выбор элементов из списка `list`, в данном случае из списка первых 300 чисел вида 2^p-1, где p простое, удовлетворяющих критерию `crit`, в данном случае критерию `PrimeQ`б осуществляющему проверку 2^p-1 на простоту. При этом получится ровно 15 простых, а именно,

$$
\begin{aligned}
M_2 &= 3\\
M_3 &= 7\\
M_5 &= 31\\
M_7 &= 127\\
M_{13} &= 8191\\
M_{17} &= 131071\\
M_{19} &= 524287\\
M_{31} &= 2147483647\\
M_{61} &= 2305843009213693951\\
M_{89} &= 618970019642690137449562111\\
M_{107} &= 162259276829213363391578010288127\\
M_{127} &= 170141183460469231731687303715884105727
\end{aligned}
$$

и M_{521}, M_{607}, M_{1279}, которые слишком велики, чтобы воспроизводить их здесь.

Заметим, что на протяжении 75 лет M_{127} оставалось самым большим известным простым числом. А именно, используя сформулированный чуть ниже критерий Люка—Лемера в 1876 году Люка доказал, что M_{127} простое. Только в 1951 году удалось найти большие простые числа. Некоторые считают, впрочем, что первое безукоризненное доказательство простоты M_{127} было дано только в 1894 году Фокембергом, но даже и в этом случае рекорд простоял 57 лет! Еще дольше, 84 года, продержался рекорд простого числа, *не* являющегося числом Мерсенна, установленный Ландри в 1867 году, а именно, $M_{59}/179951$.

Вот все остальные индексы p, для которых сегодня (весна 2006 года) известно, что число Мерсенна M_p является простым:

2203, 2281, 3217, 4253, 4423, 9689, 9941, 11213, 19937, 21701, 23209, 44497, 86243, 110503, 132049, 216091, 756839, 859433, 1257787, 1398269, 2976221, 3021377, 6972593, 13466917, 20996011, 24036583, 25964951, 30402457

Последние из этих чисел M_p настолько велики, что для десятичной записи каждого из них нужно *несколько* книг объемом 1000 страниц. Например, число $M_{30402257}$ содержит 9152052 десятичные цифры.

4.2. Породите список первых 15 пар вида (p, M_p), где M_p простое Мерсенна.

Ответ. Можно, например, так

```
Select[Table[{Prime[n],2^Prime[n]-1},{n,1,300}],
                                        PrimeQ[#[[2]]]&]
```

Обратите внимание на использование критерия в формате анонимной функции! Дело в том, что в данном случае производится не проверка простоты элемента списка, а проверка простоты второй части этого элемента. Не забывайте, что вызов анонимной функции должен заканчиваться амперсандом.

Проверять простоту числа M_p **значительно** проще, чем простоту других чисел того же порядка. Это связано с тем, что для них имеется следующий критерий простоты, открытый в 1876 году Люком (alias Лукас, Lucas) и упрощенный в 1930 году Лемером. Чтобы сформулировать этот критерий, определим прежде всего **числа Люка** L_n. Положим $L_1 = 4$ и зададим следующие числа рекуррентно посредством $L_{n+1} = L_n^2 - 2$.

4.3. Напишите программу для вычисления чисел Люка.

Ответ. Поскольку рекуррентная программа очевидна, ограничимся перечислением нескольких первых L_n:

$$L_2 = 14, \quad L_3 = 194, \quad L_4 = 37634, \quad L_5 = 1416317954,$$
$$L_6 = 2005956546822746114,$$
$$L_7 = 4023861667741036022825635656102100994.$$

Числа Люка растут довольно быстро растут, уже у L_{100} больше, чем 10^{27} цифр.

Так вот, **критерий Люка—Лемера** утверждает, что для того, чтобы выяснить, является ли число Мерсенна M_p простым, необходимо выполнить всего одно деление, а именно, M_p в том и только том случае простое, когда оно делит L_{p-1}.

4.4. Напишите тест простоты чисео Мерсенна, основанный на критерии Люка—Лемера и сравните скорость его работы с `PrimeQ`.

Обратимся теперь к разложению на простые множители тех чисел Мерсенна, которые не являются простыми. Заметим, что поиск простых делителей резко упрощается следующим **критерием Ферма—Эйлера**. Пусть p и q — нечетные простые. Тогда если $p|M_q$, то

$$p \equiv 1 \pmod{q}, \qquad p \equiv \pm 1 \pmod{8}.$$

4.5. Разложите на множители все остальные числа Мерсенна до M_{127}

Ответ. Поскольку все эти числа имеют небольшие делители, можно обойтись внутренней функцией `FactorInteger`.

$M_{11} = 2047 = 23 \cdot 89$,

$M_{23} = 8388607 = 47 \cdot 178481$,

$M_{29} = 536870911 = 233 \cdot 1103 \cdot 2089$,

$M_{37} = 137438953471 = 223 \cdot 616318177$,

$M_{41} = 2199023255551 = 13367 \cdot 164511353$,

$M_{43} = 8796093022207 = 431 \cdot 9719 \cdot 2099863$,

$M_{47} = 140737488355327 = 2351 \cdot 4513 \cdot 13264529$,

$M_{53} = 9007199254740991 = 6361 \cdot 69431 \cdot 20394401$,

$M_{59} = 576460752303423487 = 179951 \cdot 3203431780337$,

$M_{67} = 147573952589676412927 = 193707721 \cdot 761838257287$,

$M_{71} = 228479 \cdot 48544121 \cdot 212885833$,

$M_{73} = 439 \cdot 2298041 \cdot 9361973132609$,

$M_{79} = 2687 \cdot 202029703 \cdot 1113491139767$,

$M_{83} = 167 \cdot 57912614113275649087721$,

$M_{97} = 11447 \cdot 13842607235828485645766393$,

$M_{101} = 7432339208719 \cdot 341117531003194129$,

$M_{103} = 2550183799 \cdot 3976656429941438590393$,

$M_{109} = 745988807 \cdot 870035986098720987332873$,

$M_{113} = 3391 \cdot 23279 \cdot 65993 \cdot 1868569 \cdot 1066818132868207$,

$M_{131} = 263 \cdot 1035079443105516238671861923746823456 9$,

$M_{137} = 32032215596496435569 \cdot 5439042183600204290159$,

$M_{139} = 5625767248687 \cdot 123876132205208335762278423601$,

$M_{149} = 86656268566282183151 \cdot 8235109336690846723986161$,

$M_{151} = 18121 \cdot 55871 \cdot 165799 \cdot 2332951 \cdot 7289088383388253664437433$,

$M_{157} = 852133201 \cdot 60726444167 \cdot 1654058017289 \cdot 2134387368610417$,

$M_{163} = 150287 \cdot 704161 \cdot 110211473 \cdot 27669118297 \cdot 36230454570129675721$,

$M_{167} = 2349023 \cdot 79638304766856507377778616296087448490695649$,

$M_{173} = 730753 \cdot 1505447 \cdot 70084436712553223 \cdot 155285743288572277679887$,

4.6. А теперь напишите программу поиска делителей M_p, использующую критерий Ферма—Эйлера, которая работает быстрее, чем `FactorInteger`.

Бросается в глаза наличие у некоторых M_p совсем маленьких простых делителей, скажем $23|M_{11}$, $47|M_{23}$ и $167|M_{83}$. Оказывается, это не случайность. А именно, **критерий Эйлера—Лагранжа** утверждает, что если $p \equiv 3 \pmod 4$, то $q = 2p + 1$ в том и только том случае является простым, когда $q|M_p$.

4.7. Найдите все $p < 1000$ такие, что $q = 2p + 1$ делит M_p.

§ 5. Простые Ферма

Мы всегда готовы говорить правду. Но как мы ее узнаем?

Леонид Шебаршин, *Заметки бывшего начальника разведки*

Начнем со следующего незамысловатого наблюдения

5.1. Проверьте (или докажите!), что если $2^m + 1$, где $m \in \mathbb{N}$, простое, то $m = 2^n$, $n \in \mathbb{N}_0$.

Число вида $F_n = 2^{2^n} + 1$, где $n \in \mathbb{N}_0$, называется **числом Ферма**. Числа Ферма возникают в самых различных вопросах теории чисел, комбинаторики, алгебры и геометрии. Интересно заметить, что сам Ферма достаточно определенно утверждал, что **все** числа Ферма F_n простые, но смог проверить лишь, что

$$F_0 = 3, \qquad F_1 = 5, \qquad F_2 = 17, \qquad F_3 = 257, \qquad F_4 = 65537$$

просты.

5.2. Подтвердите или опровергните утверждение Ферма.

Ответ. Приведенные выше пять чисел являются *единственными* известными сегодня простыми числами Ферма! В 1732 году Эйлер нашел разложение на множители следующего числа Ферма

$$F_5 = 4294967297 = 641 \cdot 6700417 = (2^7 5 + 1)(2^7 52347 + 1).$$

Число F_6 тоже легко раскладывается на множители:

$$F_6 = 18446744073709551617 = 274177 \cdot 67280421310721 = (2^8 1071 + 1)(2^8 262814145745 + 1).$$

В это большинстве классических книг по теории чисел утверждается, что это разложение было найдено в 1880 году Ландри и Ле Лассером. Однако в 1964 году К.Бирманн обнаружил, что что Томас Клаузен привел эту факторизацию в письме к Гауссу, датированном 1 января 1855 года, и что он знал, что оба множителя простые!

А вот разложить на множители число F_7 в докомпьютерную эпоху не было никакой возможности. В действительности, следующее разложение было найдено лишь в 1970 году[43]:

$$F_7 = 59649589127497217 \cdot 5704689200685129054721 = (2^9 116503103764643 + 1)(2^9 11141971095088142685 + 1).$$

[43] M.A.Morrison, J.Brillhart, The factorisation of F_7. — Bull. Amer. Math. Soc., 1971, vol.77, p.264.

Число F_8, было факторизовано лишь в 1980 году[44]:

$$F_8 = 1238926361552897\cdot$$
$$93461639715357977769163558199606896584051237541638188580280321 =$$
$$(2^{11}604944512477+1)$$
$$(2^{11}4563556626726463758259939365215180497268126833087802176771 5+1)$$

Приятно, что сегодня это разложение за секунды ищется на бытовом компьютере при помощи функции `FactorIntegerECM`.

Что касается F_9, то в 1903 году Вестерн обнаружил, что оно делится на $2^{16}37+1 = 2424833$. Несмотря на это полная факторизация F_9 на множители была получена лишь в 1990 году[45] При этом оказалось, что два других делителя числа F_9 содержат 46 и 96 цифр, соответственно.

Единственными другими числами Ферма, которые сегодня *полностью* разложены на простые множители, являются F_{10} и F_{11}, смотри[46]. Для всех остальных чисел Ферма известны лишь *какие-то* простые делители, но не полная факторизация. Вот начало факторизации F_{10}:

$$F_{10} = (2^{12}11131+1)(2^{14}395937+1)\dots$$

два пропущенных делителя содержат 40 и 252 цифр, соответственно. Вот начало факторизации F_{11}:

$$F_{11} = (2^{13}39+1)(2^{13}119+1)(2^{14}10253207784531279+1)$$
$$(2^{13}434673084282938711+1)\dots$$

и еще один делитель, содержащий 564 цифр.

Вычислительная сложность этой задачи растет *невероятно* быстро с ростом n. Числа Ферма F_{12}, F_{13}, F_{14} и F_{15} имеют 1234, 2467, 4933, 9865 разрядов, соответственно. Даже установление простоты чисел такого порядка на бытовых компьютерах может оказаться проблематичным, а искать разложение таких чисел на множители сегодня мы просто не умеем. Вот начало факторизации F_{12}:

$$F_{12} = (2^{14}7+1)(2^{16}397+1)(2^{16}973+1)(2^{14}11613415+1)$$
$$(2^{14}76668221077+1)\dots$$

Вот начало факторизации F_{13}:

$$F_{13} = (2^{16}41365885+1)(2^{17}20323554055421+1)(2^{19}6872386635861+1)$$
$$(2^{19}609485665932753836099+1)\dots .$$

[44]R.P.Brent, J.M.Pollard, The factorisation of the eighth Fermat number. — Math. Comput., 1981, vol.36, p.627–630.

[45]A.K.Lenstra, H.W.Lenstra, M.S.Manasse, J.M.Pollard, The factorisation of the ninth Fermat number. — Math. Comput., 1993, vol.61, p.319–149.

[46]R.P.Brent, Factorisation of the tenth Fermat number. — Math. Comput., 1999, vol.68, p.429–451.

Однако, оказывается, что узнать, является ли число Ферма F_n простым, совсем просто. Например, сегодня мы не знаем никаких простых делителей чисел Ферма F_{14} F_{20}, F_{22}, F_{24} и многих других. В то же время известно, что эти числа не являются простыми. Это устанавливается при помощи следующего легко проверяемого теста, известного как **критерий Пепина**. Для того, чтобы число Ферма F_n, $n > 1$, было простым, необходимо и достаточно, чтобы

$$3^{(F_n-1)/2} \equiv -1 \pmod{F_n}.$$

5.3. Проверьте, что числа F_n, $n = 10, \ldots, 15$, не являются простыми.

Указание. Непосредственно возвести 3 в степень такого порядка нет шансов, поэтому используйте функцию `PowerMod`. Посмотрите, какой остаток она возвращает и аккуратно сформулируйте условие!

Обратимся теперь к задаче поиска простых делителей чисел Ферма. Не следует думать, что Эйлер настолько любил считать, чтобы делить вручную 10-значное число на все простые подряд, пока *случайно* не наткнулся на делитель 641. В действительности, ему пришлось для этого выполнить всего одно или два деления, а, скорее всего, ни одного.

Реконструируем рассуждения Эйлера, чтобы читатель мог на этом *игрушечном* примере представить, при помощи каких примерно соображений ищутся простые делители у чисел, содержащих многие сотни или тысячи десятичных знаков, *если известна их структура*. Дело в том, что для того, чтобы разложить число Ферма F_n на множители, достаточно проверять не все простые $p \leq \sqrt{F_n}$, а лишь простые вида $p = 2^{n+1}m + 1$, где $m \in \mathbb{N}$. Это вытекает из следующего легко проверяемого соображения, известного как сравнение Эйлера: любой делитель числа F_n, $n \geq 3$, имеет вид $2^{n+2}m + 1$, для некоторого $m \in \mathbb{N}$.

В силу сравнения Эйлера делителями F_5 могут быть только простые числа вида $p = 128m + 1$. Первые два таких числа, это $p = 257$ и $p = 641$, которые получаются при $m = 2$ и $m = 5$, соответственно. Однако очевидно, что $641 = 2^4+5^4$ делит $a = 2^{32}+2^{28}5^4$. С другой стороны, применяя формулу для разности квадратов, мы видим, что $641 = 2^7 5 + 1$ делит $b = 2^{28}5^4 - 1$. Таким образом, 641 делит и разность этих чисел $F_5 = a - b$. Для того, чтобы проверить, будет ли 6700417 простым, достаточно, *в худшем случае*, произвести еще не более 4 делений, а именно, проверить, что оно не делится на простые числа вида $p = 128m + 1$, $5 \leq m \leq 20$, каковых, очевидно (см. таблицу простых) ровно 4, а именно, 641, 769, 1153, 1409. Однако, зная Эйлера, можно предположить, что он, скорее всего, и здесь обошелся вообще без явных вычислений. Экстраполируя этот пример, мы понимаем, что сказано в эпиграфе к этой главе: один изобретательный математик может с успехом заменить *сотни* вычислителей.

5.4. А теперь напишите программу, раскладывающую числа F_6, F_7 и F_8 на множители быстрее, чем это делает внутренняя команда `FactorInteger`.

Приведем список известных небольших простых делителей нескольких

следующих чисел Ферма:

F_{15}	$2^{21}579+1$, $2^{17}17753925353+1$, $2^{17}1287603889690528658928101555+1$
F_{16}	$2^{19}1575+1$, $2^{20}180227048850079840107+1$
F_{17}	$2^{19}59251857+1$
F_{18}	$2^{20}13+1$
F_{19}	$2^{21}33629+1$, $2^{21}308385+1$
F_{21}	$2^{23}534689+1$
F_{23}	$2^{25}5+1$
F_{25}	$2^{29}48413,29+1$, $2^{27}1522849979+1$, $2^{27}16168301139+1$
F_{26}	$2^{29}143165+1$
F_{27}	$2^{30}141015+1$, $2^{29}430816215+1$
F_{28}	$2^{36}25709319373+1$
F_{29}	$2^{31}1120049+1$
F_{30}	$2^{32}149041+1$, $2^{33}127589+1$

Заметим, что числа Ферма дают еще один подход к доказательству теоремы Эвклида бесконечности числа простых. В самом деле, из следующей задачи (взятой непосредственно из переписки Гольдбаха и Эйлера[47]) вытекает, что числа Ферма попарно взаимно просты.

5.5. Проверьте (или докажите!), что

$$F_0F_1F_2\dots F_n = F_{n+1}-2.$$

Таким образом, если F_m делит F_n, при некотором $n>m$, то F_m делит 2, что невозможно.

Рассматриваются различные вариации на тему чисел Ферма. Вот три наиболее известные из них.

• Числа вида $b^{2^n}+1$ называется **обобщенными числами Ферма**[48,49], обычные числа Ферма получаются здесь при $b=2$.

• Число $C_n = n\cdot 2^n+1$ называется **числом Каллена**.

• Число $W_n = n\cdot 2^n-1$ называется **числом Вудалла**.

[47] Доказательство, основанное на той же идее, но содержащее несколько чрезвычайно удачных ухудшений, воспроизводится в книге “Задачи и теоремы из анализа”, поэтому некоторые авторы ошибочно приписывают его Пойа и Сеге.

[48] H.Dubner, W.Keller, Factors of generalized Fermat numbers. — Math. Comput., 1995, vol.64, p.397–405.

[49] A.Björn, H.Riesel, Factors of generalized Fermat numbers. — Math. Comput., 1998, vol.67, p.441–446.

5.6. Вычислите первые 140 чисел Каллена $C_n = n \cdot 2^n + 1$ и убедитесь, что все они, кроме $C_1 = 3$ составные. Достаточно ли этого, чтобы сформулировать гипотезу, что все числа C_n, $n > 1$, составные?

Ответ. Нет, число C_{141} простое[50]. Кроме того, известны следующие простые Каллена[51,52]:

$$C_{4713},\ C_{5795},\ C_{6611},\ C_{18496},\ C_{32292},\ C_{32469},\ C_{59656},\ C_{90825},\ C_{262419},\ C_{361275},\ C_{481899}.$$

В отличие от простых Каллена, среди простых Вудалла с небольшими индексами достаточно много простых.

5.7. Найдите первые 15 чисел Вудалла $W_n = n \cdot 2^n - 1$.

Ответ. Вот они: $W_2 = 7$, $W_3 = 23$, $W_6 = 383$,

$W_{30} = 32212254719$,

$W_{75} = 283341988972178712821759 9$,

$W_{81} = 195845982777569926302400511$,

$W_{115} = 47769131098520414182480566228824883 19$,

$W_{123} = 13079603478523572189373461473158590627 83$,

и, кроме того, W_{249}, W_{362}, W_{384}, W_{462}, W_{512}, W_{751}, W_{822}. Следующие простые Вудалла непомерно велики:

$$W_{5312},\ W_{7755},\ W_{9531},\ W_{12379},\ W_{15822},\ W_{18885},\ W_{22971},\ W_{23005},\ W_{98726},\ W_{143018},\ W_{151023},\ W_{667071}.$$

§ 6. Распределение простых

Nature is a good approximation of Mathematics

Zvi Artstein

Обозначим через $\pi(n)$ количество натуральных простых не превосходящих $n \in \mathbb{N}$. Ясно, что если $n = p \in \mathbb{P}$ простое, то $\pi(p) = \pi(p-1) + 1$. Напомним, что в соответствии с общим приницпом образования имен на языке `Mathematica` функция π называется `PrimePi`.

6.1. Составьте таблицу значений функций $\pi(n)$ и $n/\pi(n)$ для $n = 10^m$, $1 \le m \le 14$, и сравните вычисленные значения с $\ln(n)$.

---

[50] R.M.Robinson, A report on primes of the form $k \cdot 2^n + 1$ and on factors of Fermat numbers. — Proc. Amer. Math. Soc., 1958, vol.9, p.673–681.

[51] W.Keller, Factors of Fermat numbers and large primes of the form $k \cdot 2^n + 1$. — Math. Comput., 1983, vol.41, p.661–673.

[52] W.Keller, New Cullen primes. — Math. Comput., 1995, vol.64, p.1733–1741.

Ответ. Вот как выглядят эти значения:

n	$\pi(n)$	$n/\pi(n)$	$\ln(n)$
10	4	2.5	2.30259
100	25	4.	4.60517
1000	168	5.95238	6.90776
10000	1229	8.1367	9.21034
1 00000	9592	10.4254	11.5129
10 00000	78498	12.7392	13.8155
100 00000	6 64579	15.0471	16.1181
1000 00000	57 61455	17.3567	18.4207
10000 00000	508 47534	19.6666	20.7233
1 00000 00000	4550 52511	21.9755	23.0259
10 00000 00000	41180 54813	24.2833	25.3284
100 00000 00000	3 76079 12018	26.5901	27.6310
1000 00000 00000	34 60655 36839	28.8963	29.9336
10000 00000 00000	320 49417 50802	31.2018	32.2362

Эта таблица показывает, что маленьких простых чисел довольно много: скажем, среди первых 10000 чисел 1229 простых — больше, чем 12%. В действительности, среди первого миллиарда натуральных чисел все еще больше 5% простых!!!

Глядя — в пятнадцатилетнем возрасте! — на такую, или чуть более короткую, таблицу, Гаусс заметил, что, при переходе от каждой степени 10 к следующей, отношение $n/\pi(n)$ увеличивается примерно на $\ln(10) \cong 2.30259$. Это натолкнуло его на замечательное предположение, что при n стремящемся к ∞ функция $n \mapsto \pi(n)$ растет примерно как $n \mapsto n/\ln(n)$. Утверждение об асимптотической эквивалентности $\pi(n)$ и $n/\ln(n)$ называется **асимптотическим законом** распределения простых чисел или, иногда, просто **теоремой о простых числах**.

Первое выдающееся продвижение в направлении доказательства асимптотического закона получил в 1849 году П.Л.Чебышев. Однако, полностью асимпотический закон распределения простых был доказан лишь в 1896 году независимо Адамаром и де ла Валле-Пуссеном с использованием теории функций комплексной переменной. Запомнить дату 1896 чрезвычайно легко, поскольку Адамар прожил 98 лет, а де ла Валле-Пуссен — 96. Рибенбойм выражает уверенность, что это произошло именно как следствие того, что они доказали столь замечательную теорему. Элементарное доказательство асимптотического закона нашли Сельберг[53] и Эрдеш в 1949 году.

При изучении подгрупп в симметрической группе Жозеф Бертран использовал следующее утверждение, известное как **постулат Бертрана**: при $n > 7$ между $n/2$ и $n-2$ всегда содержится хотя бы одно простое число. Сам Бертран проверил это утверждение для всех $n < 1500000$, но

[53] A.Selberg, An elementary proof of the prime number theorem. — Ann. Math., 1951, vol.85, p.203–362.

его доказательством в общем случае он не владел. Первое доказательство постулата Бертрана придумал в 1852 году П. Л. Чебышев.

6.2. Проверьте постулат Бертрана для всех $n \le 2000000$.

Шинцель предложил следующее усиление постулата Бертрана, известное как **гипотеза Шинцеля**: для каждого $n \ge 117$ между n и $n+\sqrt{n}$ найдется хотя бы одно простое число.

6.3. Проверьте гипотезу Шинцеля для всех $117 \le n \le 100000$. Как Вы думаете, с чем связано ограничение $n \ge 117$?

Верно ли, что для любых $x, y \ge 2$ выполняется неравенство $\pi(x+y) \le \pi(x)+\pi(y)$?

6.4. Проверьте выполнение этого неравенства для всех $x, y \le 1000$.

§ 7. Теорема Дирихле

Какое чудо, если есть
Тот, кто затеплил в нашу честь
Ночное множество созвездий !
А если всё само собой
Устроилось, тогда, друг мой,
Еще чудесней!

Александр Кушнер

В простейшем варианте знаменитая **теорема Дирихле** о простых в арифметических прогрессиях, с которой собственно и начинается современная аналитическая теория чисел, утверждает, что для любых *взаимно простых* a и d в арифметической прогрессии $a+nd$, $n \in \mathbb{N}$, бесконечно много простых. В частности, существует бесконечно много простых с *любыми* наперед заданными m последними цифрами, если только последняя цифра не равна 0, 2, 4, 5, 6, 8.

Однако, как мы сейчас увидим, в действительности теорема Дирихле утверждает *гораздо* больше, чем просто бесконечность простых в арифметической прогрессии.

7.1. Найдите количество простых $\le 10^6$, последняя цифра которых равна 1, 3, 7, 9.

Решение. Вычисляя

```
Map[Length[Select[Range[#,10^6,10],PrimeQ]]&,{1,3,7,9}]
```

мы видим, что количество простых с последней цифрой 1, 3, 7, 9 практически не зависит от этой цифры: Вот как распределяются по последней цифре 78498 простых чисел меньших одного миллиона:

$$19617, \qquad 19665, \qquad 19621, \qquad 19593.$$

Эта закономерность становится еще очевиднее, если продолжить вычисления. Вот как распределяются по последней цифре 664579 простых чисел меньших десяти миллионов:

$$166104, \qquad 166230, \qquad 166211, \qquad 166032,$$

и вот, наконец, 5761455 простых чисел меньших ста миллионов:

$$1440299, \qquad 1440474, \qquad 1440495, \qquad 1440186$$

Мы видим, что каждый раз количество простых с данной последней цифрой почти в точности равно четверти общего количества простых. Не пытайтесь повторить последнее вычисление, на бытовом компьютере это может занять минут десять–пятнадцать.

Взглянем теперь на последние *две* цифры.

7.2. Породите множество чисел, которые могут быть остатками простого числа по модулю 100.

Решение. Как всегда, `Mathematica` дает почти бесконечное разнообразие способов решения этой задачи. Вот первые приходящие в голову:

```
Map[FromDigits[#]&,Flatten[Outer[List,Range[0,9],{1,3,7,9}],1]]
Select[Range[100],MemberQ[{1,3,7,9},Last[IntegerDigits[#]]]&]
Select[Range[100],MemberQ[{1,3,7,9},Mod[#,10]]&]
Select[Range[100],GCD[#,10]==1&]
Apply[Union,Map[Range[#,100,10]&,{1,3,7,9}]]
```

Конечно, при получении списка из 40 чисел вопрос выбора алгоритма является чисто схоластическим. Заметим, впрочем, что из приведенных выше определений последнее заведомо лучше предыдущих: оно лучше первого потому, что гораздо легче обобщается на любое количество цифр и лучше трех других потому, что не требует выбора по критерию из огромного списка. Уже при порожении списка шестизначных чисел оно эффективнее раз в 50.

7.3. Найдите распределение простых $\leq 10^6$ по последним двум цифрам.

Ответ. Вот это распределение, где строки занумерованы цифрами 1, 3, 7, 9, а столбцы отвечают десяткам:

1964	1958	1937	1964	1955	1970	1960	1986	1942	1981
1969	1965	1976	1967	1959	1977	1962	1956	1969	1965
1932	1970	1976	1973	1956	1961	1943	1960	1984	1966
1957	1973	1926	1970	1960	1967	1955	1960	1958	1967

Снова в каждый класс попадает примерно одинаковое количество простых, а именно около $1/40$ от общего количества.

Теперь уже очевидно, что не только в каждой арифметической прогрессии содержится бесконечное количество простых, но и то, что простые распределены равномерно между всеми прогрессиями основание которых взаимно просто с фиксированной разностью d. Именно это, конечно, и утверждает теорема Дирихле!

7.4. Проверьте утверждение теоремы Дирихле еще на нескольких примерах.

Ответ. Вот, скажем, как распределяются 78496 простых меньших одного миллиона по двум классам по модулям 3 и 4:

- по модулю 3: 39231 из них дают остаток 1 и 39266 остаток 2;
- по модулю 4: 39175 из них дают остаток 1 и 39322 остаток 3.

Убедительно?

> All the wonders of our universe can in effect be captured by simple rules, yet there can be no way to know all the consequences of these rules, except in effect just to watch and see how they unfold.
>
> Stephen Wolfram, *A New Kind of Science*

Глава 7. МУЛЬТИПЛИКАТИВНАЯ ТЕОРИЯ ЧИСЕЛ

> В математике несравненно явственнее, чем в других дисциплинах, ощущается, насколько растянуто шествие всего человечества. Среди наших современников есть люди, чьи познания в математике относятся к эпохе более древней, чем египетские пирамиды, и они составляют подавляющее большинство. Математические познания незначительной части людей дошли до эпохи средних веков, а уровня математики XVIII века не достигает и один человек на тысячу. Выяснилось, что, желая превратить первобытного человека в математика, мы не можем рассчитывать на эволюцию. Поэтому расстояние между теми, кто идет в авангарде, и необозримой массой путников все увеличивается.
>
> Гуго Штейнгауз

> Из всех проблем, рассматриваемых в математике, нет таких, которые считались бы в настоящее время более бесплодными и бесполезными, чем проблемы, касающиеся природы чисел и их делителей. В этом отношении нынешние математики сильно отличаются от древних, придававших гораздо большее значение исследованиям такого рода. А именно, они не только считали, что отыскание истины похвально само по себе и достойно человеческого познания, но, кроме того, совершенно справедливо полагали, что при этом замечательным образом развивается изобретательность и перед человеческим разумом раскрываются новые возможности решать сложные задачи. Математика, вероятно, никогда не достигла бы столь высокой степени совершенства, если бы древние не приложили столько усилий для изучения вопросов, которыми сегодня многие пренебрегают из-за их мнимой бесплодности.
>
> Леонард Эйлер, *De numeris amicabilibus*

> Наиболее очевидной особенностью произведения искусства может считаться его бесполезность.
>
> Поль Валери, *Всеобщее определение искусства*

> Pure mathematics, as a whole, is *distinctly* more useful, than applied.
>
> Godfrey Harold Hardy, *A mathematician's apology*

В настоящей главе мы обсуждаем **мультипликативную теорию чисел**, иными словами, свойства целых чисел относительно *умножения* такие, как делимость, разложение числа в произведение простых и поведение остатков степеней числа по фиксированному модулю. На протяжении тысячелетий эти вопросы поражали человеческое воображение, не в последнюю

очередь благодаря своей бесполезности. Харди прямо заявил[54], что теория чисел практически бесполезна, *в том смысле*, что она НЕ МОЖЕТ *прямо* СПОСОБСТВОВАТЬ НИ УСИЛЕНИЮ СУЩЕСТВУЮЩЕГО ОБЩЕСТВЕННОГО НЕРАВЕНСТВА, НИ СОЗДАНИЮ БОЛЕЕ ЭФФЕКТИВНЫХ СПОСОБОВ УБИЙСТВА. Однако, в последние десятилетия эти вопросы вдруг оказались *центральными* для важнейших приложений математики, в первую очередь в задачах кодирования и криптографии. Имеются самые серьезные основания считать, что обе части *этого* высказывания Харди полностью утратили актуальность, как раз потому, что В ЦЕЛОМ ЧИСТАЯ МАТЕМАТИКА *ощутимо* ПОЛЕЗНЕЕ ПРИКЛАДНОЙ.

§ 1. Основная теорема арифметики

Van Roy's Law: An unbreakable toy is useful for breaking other toys.

Основная теорема арифметики: каждое ненулевое целое число $n \in \mathbb{Z}$ может быть однозначно представлено в виде

$$n = \pm p_1^{k_1} \dots p_s^{k_s},$$

где p_i — попарно различные простые числа, расположенные в порядке возрастания, а $k_i > 0$. Записанное в таком виде разложение на простые называется **каноническим разложением**. Степень k, с которой простое число p входит в каноническое разложение числа n называется **p-адическим показателем** n и обозначается $v_p(n)$. В системе `Mathematica` имплементированы функции, находящие по целому числу n его каноническое разложение на простые, а по паре, состоящей из целого числа n и простого числа p, соответствующий p-адический показатель $v_p(n)$.

`FactorInteger[n]`	каноническое разложение n
`IntegerExponent[n,m]`	наибольшая степень m делящая n
`FactorIntegerECM[n]`	какой-то делитель n

Выполнение `FactorInteger[n]` возвращает список $(p, v_p(n))$, в который входят все те простые, для которых $v_p(n) > 0$, расположенные в порядке возрастания. Таким образом, второй элемент каждой такой пары это в точности `IntegerExponent[n,p]`. В тех случаях, когда нам нужно только найти наибольшую степень, в которой *данное* простое число p делит n, нужно пользоваться непосредственно командой `IntegerExponent`, которая не требует разложения n на простые множители и работает много быстрее. Для отрицательного числа n в качестве первого элемента списка фигурирует пара $(-1, 1)$. Таким образом, например, вычисление `FactorInteger[-240]` возвращает список `{{-1,1},{2,4},{3,1},{5,1}}`.

[54]В 1915 году и потом, чуть в другой форме в 1920 году, а вовсе не в написанной через четверть века *Апологии математика*, как почему-то считает Арнольд!

1.1. Убедитесь, что $3^{2^n}-1$ делится на 2^{n+2}, но не делится на 2^{n+3}.

1.2. Убедитесь, что $2^{3^n}+1$ делится на 3^{n+2}, но не делится на 3^{n+2}.

1.3. Как найти список простых делителей n?

Решение. Например, так

```
Last[Transpose[FactorInteger[Abs[n]]]]
```

Зачем мы использовали здесь `Abs`?

1.4. Задайте функцию, которая сопоставляет числу n сумму его *простых* делителей.

1.5. Задайте функцию, сопоставляющую n число различных простых делителей $\omega(n)$. Задайте функцию, которая сопоставляет n число простых делителей с учетом кратности.

Решение. Например, так

```
Length[FactorInteger[n]]
```

```
Total[Last[Transpose[FactorInteger[n]]]]
```

Число n называется бесквадратным, если все простые множители входят в него с показателем 1. Иными словами, бесквадратное число не делится на p^2 ни для одного простого p.

1.6. Задайте функцию, которая определяет, является ли число n бесквадратным.

Если известно разложение

$$m = p_1^{k_1} \dots p_s^{k_s}, \qquad n = p_1^{l_1} \dots p_s^{l_s}$$

двух натуральных чисел на простые множители, то m в том и только том случае делит n, когда $k_i \le l_i$ для всех $i = 1, \dots, s$. В частности, в этом случае наибольший общий делитель m и n можно вычислять следующим образом:

$$\gcd(m,n) = p_1^{\min(k_1,l_1)} \dots p_s^{\min(k_s,l_s)},$$
$$\operatorname{lcm}(m,n) = p_1^{\max(k_1,l_1)} \dots p_s^{\max(k_s,l_s)}.$$

1.7. Напишите программу, вычисляющую $\gcd(m,n)$ и $\operatorname{lcm}(m,n)$ на основе этих формул. Почему такой метод практически не применяется?

В 1808 году Лежандр опубликовал следующую формулу, выражающую p-адический показатель факториала:

$$v_p(n!) = \left\lfloor \frac{n}{p} \right\rfloor + \left\lfloor \frac{n}{p^2} \right\rfloor + \left\lfloor \frac{n}{p^3} \right\rfloor + \dots$$

1.8. Проверьте формулу Лежандра для всех $n, p \le 1000$.

§ 2. Числа с одним или двумя простыми делителями

Lead me not into temptation, I can find it myself.
Oscar Wilde

Число q для которого $\omega(q) = 1$ называется **примарным**. Таким образом, примарное число является степенью простого числа: $q = p^m$ для некоторого простого p и некоторого натурального m. Примарных числа располагаются в ряду натуральных чисел лишь с суть большей частотой, чем сами простые числа.

2.1. Определите функцию $\pi^*(n)$, которая определяет количество примарных чисел, меньших данного числа n.

Решение. Конечно, было бы неправильно выбирать примарные числа, меньшие n. Нужно воспользоваться внутренней функцией `PrimePi`. А именно, примарные числа, не превосходящие n, это, во-первых, простые числа, не превосходящие n. Во-вторых, это квадраты простых чисел, не превосходящие n. Ясно, что таких квадратов столько же, сколько простых чисел, не превосходящих $\sqrt{n}$. В третьих, это кубы простых чисел, не превосходящие n, которых столько же, сколько простых чисел, не превосходящих $\sqrt[3]{n}$. Продолжая действовать таким образом, мы получим следующую формулу для $\pi^*(n)$:

$$\pi^*(n) = \pi(n) + \pi(\sqrt{n}) + \pi(\sqrt[3]{n}) + \dots$$

где количество слагаемых справа равно $\lfloor \log_2(n) \rfloor$. Таким образом, искомую функцию можно задать, например, так:

```
primary[n_]:=Sum[PrimePi[Power[n,1/m]],{m,1,Log[2,n]}]
```

2.2. Найдите количество примарных чисел $< 10^n$, $1 \le n \le 14$, и отношение этого количества к количеству простых чисел $< 10^n$.

Ответ. Вычисление, использующее построенную в предыдущей задаче функцию $\pi^*(n)$ дает следующую таблицу:

n	$\pi^*(n)$	$\pi(n)$	$\pi^*(n)/\pi(n)$
10	7	4	1.750000000
100	35	25	1.400000000
1000	193	168	1.148809524
10000	1280	1229	1.041497152
1 00000	9700	9592	1.011259383
10 00000	78734	78498	1.003006446
100 00000	6 65134	6 64579	1.000835115
1000 00000	57 62859	57 61455	1.000243688
10000 00000	508 51223	508 47534	1.000072550
1 00000 00000	4550 62595	4550 52511	1.000022160
10 00000 00000	41180 82969	41180 54813	1.000006837
100 00000 00000	3 76079 92088	3 76079 12018	1.000002129
1000 00000 00000	34 60657 67406	34 60655 36839	1.000000666
10000 00000 00000	320 49424 20923	320 49417 50802	1.000000209

Философский вывод: с ростом n отношение $\pi^*(n)/\pi(n)$ быстро стремится к 1. Следующий по порядку вклад в $\pi^*(n)$ после $\pi(n)$ дает $\log_2(n)$ — иными словами, никаких примарных чисел, кроме простых чисел и степеней 2 *по существу* не бывает!

2.3. Найдите все тройки последовательных примарных чисел.

Ответ. Три такие тройки, а именно, $(2,3,4)$, $(3,4,5)$ и $(7,8,9)$ видны невооруженным глазом, а других таких троек нет. Дело в том, что второй элемент такой тройки должен быть степенью 2, а первый или третий — степенью 3. Однако, как вытекает из положительного решения проблемы Каталана, никаких других возможностей для этого, кроме перечисленных выше, нет.

2.4. Найдите все пары последовательных примарных чисел ≤ 1000000.

Ответ. Пять таких примеров $(2,3)$, $(3,4)$, $(4,5)$, $(7,8)$ и $(8,9)$ расположены в первом десятке. Вот *все* остальные примеры, которые возникают до одного миллиона:

16	17	31	32	127	128	256	257
8191	8192	65536	65537	131071	131072	524287	524288

Ничего, кроме пяти чисел Мерсенна и трех чисел Ферма. А что еще Вы ожидали увидеть, ведь одно из этих чисел должно быть степенью 2.

Примарных чисел слишком мало, а вот числа, у которых ровно *два различных* простых делителя, расположены в ряду натуральных чисел уже довольно часто.

2.5. Существуют ли восьмерки последовательных натуральных чисел

$$n,\ n+1,\ n+2,\ n+3,\ n+4,\ n+5,\ n+6,\ n+7$$

таких, что каждое из них имеет ровно два различных простых делителя?

Ответ. Нельзя сказать, что такая восьмерка является самым обычным делом, так как ее возникновение требует стечения ряда счастливых обстоятельств, типа простых близнецов, удачного расположения степеней 2 и 3 и т.д. Тем не менее, две такие восьмерки существуют:

$141 = 3 \cdot 47$	$142 = 2 \cdot 71$	$143 = 11 \cdot 13$	$144 = 2^4 \cdot 3^2$
$145 = 5 \cdot 29$	$146 = 2 \cdot 73$	$147 = 3 \cdot 7^2$	$148 = 2^2 \cdot 37$
$212 = 2^2 \cdot 53$	$213 = 3 \cdot 71$	$214 = 2 \cdot 107$	$215 = 5 \cdot 43$
$216 = 2^3 \cdot 3^3$	$217 = 7 \cdot 31$	$218 = 2 \cdot 109$	$219 = 3 \cdot 73$

Ясно, все элементы этих восьмерок, кроме первого/последнего, образуют *семерку* последовательных чисел таких, что каждое из них имеет ровно два различных простых делителя.

2.6. Существуют ли другие такие семерки последовательных натуральных чисел $n, n+1, n+2, n+3, n+4, n+5, n+6$?

Ответ. Да, вот еще две:

$$323 = 17 \cdot 19 \qquad 324 = 2^2 \cdot 3^4 \qquad 325 = 5^2 \cdot 13 \qquad 326 = 2 \cdot 163$$
$$327 = 3 \cdot 109 \qquad 328 = 2^3 \cdot 41 \qquad 329 = 7 \cdot 47$$

$$2302 = 2 \cdot 1151 \qquad 2303 = 7^2 \cdot 47 \qquad 2304 = 2^8 \cdot 3^2 \qquad 2305 = 5 \cdot 461$$
$$2306 = 2 \cdot 1153 \qquad 2307 = 3 \cdot 769 \qquad 2308 = 2^2 \cdot 577$$

Снова совершенно ясно, что все элементы семерок, кроме первого/последнего, образуют *шестерку* последовательных чисел таких, что каждое из них имеет ровно два различных простых делителя.

2.7. Существуют ли другие такие шестерки последовательных натуральных чисел $n, n+1, n+2, n+3, n+4, n+5$?

Ответ. Да, вот самая маленькая:

$$91 = 7 \cdot 13 \qquad 92 = 2^2 \cdot 23 \qquad 93 = 3 \cdot 31 \qquad 94 = 2 \cdot 47$$
$$95 = 5 \cdot 19 \qquad 96 = 2^5 \cdot 3$$

Однако шестерок, семерок и восьмерок последовательных чисел с двумя различными простыми делителями крайне мало. А вот пятерок — видимо-невидимо. Это связано с тем, что при построении такой пятерки не так критична роль простого числа 3, которое среди $n, n+1, n+2, n+3, n+4$ как правило делит только $n+2$.

2.8. Найдите первые 100 пятерок последовательных натуральных чисел $n, n+1, n+2, n+3, n+4$ таких, что каждое из них имеет ровно два различных простых делителя?

2.9. Сколько таких пятерок до ста тысяч? До одного миллиона? До десяти миллионов?

Ответ. Соответственно, 118, 301 и 1141 штук. Вот, например, как выглядит последняя пятерка перед десятью миллионами:

$$9997771 = 7 \cdot 1428253, \quad 9997772 = 2^2 \cdot 2499443, \quad 9997773 = 3 \cdot 3332591,$$
$$9997774 = 2 \cdot 4998887, \quad 9997775 = 5^2 \cdot 399911.$$

А теперь слегка изменим точку зрения и будем считать простые множители *на самом деле*, иными словами, с учетом кратности. Ясно, что здесь нам не удастся построить даже четверку последовательных чисел имеющих ровно два простых множителя (одно из них должно делиться на $4 = 2^2$ и, значит, должно быть равно 4). Посмотрим, существуют ли тройки $(n, n+1, n+2)$ такие, что каждое из чисел $n, n+1, n+2$ имеет ровно два простых множителя. В силу только что сказанного, $n+1$ должно быть четным, одно из чисел n или $n+2$ делится на 3 и тогда с очень большой вероятностью второе делится на 5 или на 7.

2.10. Найдите 100 троек последовательных натуральных чисел $n, n+1, n+2$ таких, что каждое из них имеет ровно два простых делителя, считаемых с учетом кратности.

Ответ. Таких троек зиллионы. Вот первые десять из них:

$33 = 3 \cdot 11$	$34 = 2 \cdot 17$	$35 = 5 \cdot 7$
$85 = 5 \cdot 17$	$86 = 2 \cdot 43$	$87 = 3 \cdot 29$
$93 = 3 \cdot 31$	$94 = 2 \cdot 47$	$95 = 5 \cdot 19$
$141 = 3 \cdot 47$	$142 = 2 \cdot 71$	$143 = 11 \cdot 13$
$201 = 3 \cdot 67$	$202 = 2 \cdot 101$	$203 = 7 \cdot 29$
$213 = 3 \cdot 71$	$214 = 2 \cdot 107$	$215 = 5 \cdot 43$
$217 = 7 \cdot 31$	$218 = 2 \cdot 109$	$219 = 3 \cdot 73$
$301 = 7 \cdot 43$	$302 = 2 \cdot 151$	$303 = 3 \cdot 101$
$393 = 3 \cdot 131$	$394 = 2 \cdot 197$	$395 = 5 \cdot 79$
$445 = 5 \cdot 89$	$446 = 2 \cdot 223$	$447 = 3 \cdot 149$

Более того, и дальше такие тройки встречаются чрезвычайно часто. Вот, скажем, последняя тройка, появляющаяся перед 10^9:

$$999999445 = 5 \cdot 199999889, \quad 999999446 = 2 \cdot 499999723,$$
$$999999447 = 3 \cdot 333333149.$$

§ 3. Квадратичное решето

> Частичный успех похож на успех, но все-таки это — не успех. Частичная неудача похожа на неудачу, но все-таки это — не неудача.
>
> Ле-цзы, Гл.VI, *Сила и судьба*

В отличие от проверки простоты числа поиск разложения составного числа на простые множители считается алгоритмически сложной задачей. Функция `FactorInteger` кроме пробного деления и квадратичного решета использует $(p-1)$-алгоритм Полларда и ρ-алгоритм Полларда. На русском языке описание этих алгоритмов можно найти в книге [Vas]. Эта функция быстро возвращает достоверный ответ в одном из следующих двух случаев.

• Либо все простые делители числа n совсем маленькие, скажем, $\le 10^{12}$ (в этом случае работают пробное деление и $(p-1)$-алгоритм Полларда).

• Либо у n есть два близких по величине простых делителя (в этом случае работает квадратичное решето).

В случае, когда эта функция не дает ответа в течение нескольких минут следует попробовать воспользоваться функцией `FactorIntegerECM`, которую нужно подгрузить посредством

```
<<NumberTheory`FactorIntegerECM`
```

Функция `FactorIntegerECM` также также является вероятностной, но в отличие от `FactorInteger` она основана на алгоритме Ленстры[55] в варианте

[55] H.W.Lenstra, Factoring Integers with Elliptic Curves. — Ann. Math., 1987, vol.126, p.649–673.

Монтгомери[56] и при некотором везении успешно ищет делители числа n, состоящие из нескольких десятков цифр. В отличие от функции `FactorInteger` функция `FactorIntegerECM` возвращает *один* какой-то делитель m числа n, не обязательно даже простой. После этого можно еще раз применить `FactorInteger` или `FactorIntegerECM` к самому этому делителю m и к частному n/m и далее продолжать в таком же духе.

Большинство этих алгоритмов либо предполагают знание серьезной алгебры и алгебраической геометрии, либо являются хотя и элементарными, но весьма специальными. Поэтому мы ограничимся восходящей к Ферма простейшей версией **квадратичного решета** [Ch], [Vas], позволяющей искать *большие* простые делители n, близкие по величине к $\sqrt{n}$.

Метод Ферма основан на следующем соображении. Пусть n *нечетное* число — если число n четно, его факторизация сводится к факторизации $n/2$. Если $n = u^2 - v^2$, для некоторых $u > v > 0$, то то $n = (u+v)(u-v)$. Обратно, если $n = x \cdot y$, для некоторых $x \geq y > 0$, то

$$n = \left(\frac{x+y}{2}\right)^2 - \left(\frac{x-y}{2}\right)^2,$$

так что можно влять $u = (x+y)/2$, $v = (x-y)/2$.

• Первоначальная идея Ферма состояла в том, чтобы перебирать в качестве кандидатов на роль u числа $\lfloor\sqrt{n}\rfloor$, $\lfloor\sqrt{n}\rfloor + 1$, ... и для каждого из них проверять, является ли $(\lfloor\sqrt{n}\rfloor + i)^2 - n$ полным квадратом. При этом (после изучения § 9 и § 10) большинство кандидатов легко отбраковываются тем, что они не являются квадратичными вычетами для какого-то из совсем небольших простых.

3.1. Реализуйте исходный вариант метода Ферма.

Указание. Поначалу тот факт, что число m является полным квадратом, можно проверять грубо, например, так $\lfloor\sqrt{m}\rfloor^2 = m$. После знакомства с символами Лежандра можно резко улучшить алгоритм, введя в него в качестве первого шага проверку того, что m будет квадратичным вычетом по модулю первых нескольких десятков простых.

• Еще проще реализовать следующую версию метода Ферма, в которой не нужно проверять, является ли число полным квадратом. Положим $u = \lfloor\sqrt{n}\rfloor$, $v = 0$ и вычислим разность $r = u^2 - v^2 - n$. Если $r = 0$, то $n = u^2 - v^2$ и мы достигли полного счастья. Если же $r \neq 0$, то либо $u^2 - v^2 < n$, либо $u^2 - v^2 > n$. В первом случае увеличим u на единицу, а во втором случае увеличим v на единицу. Легко доказать, что за конечное число шагов мы придем к такой паре (u, v), для которой $r = 0$, причем для *первой* такой пары разность $y = u - v$ является наибольшим натуральным делителем n, не превосходящим $\sqrt{[n]}$.

3.2. Реализуйте улучшеный вариант метода Ферма.

---

[56] P.L.Montgomery, Speeding up the Pollard and Elliptic Curve Methods of Factorization. — Math. Comput., 1987, vol.48, p.243–264.

Современные субэкспоненциальные версии метода квадратичного решета — такие, как алгоритм Диксона, алгоритм Бриллхарта—Моррисона, алгоритм Померанца[57,58], алгоритм Дэвиса—Монтгомери — развивают ту же общую идею, но делают это *гораздо* более эффективно. Некоторые улучшения в этом направлении известны уже довольно давно. Например, еще Лежандр заметил, что для факторизации n не обязательно уметь решать *уравнение* $u^2 - v^2 = n$, достаточно найти какое-то *нетривиальное* решение сравнения $u^2 \equiv v^2 \pmod n$. При этом u и v в описанной выше процедуре за один шаг можно увеличивать не на 1, а так, чтобы изменился знак неравенства.

§ 4. Теорема Ферма

> Christopher Robin knew that it was enchanted because nobody had ever been able to count whether it was sixty-three or sixty-four, not even when he tied a piece of string round each tree after he had counted it.
>
> Alan Alexander Miln, *The house at Pooh Corner*

Во многих современных алгоритмах, в частности в криптографии, необходимо уметь строить большие простые и быстро проверять числа на простоту. В настоящем параграфе мы обсудим важнейшее *необходимое* условие простоты числа.

`PowerMod[a,d,n]`	a^b по модулю n
`MultiplicativeOrder[a,n]`	порядок a по модулю n
`CarmichaelLambda[n]`	функция Кармайкла

Функция `PowerMod[a,d,n]` вычисляет остаток a^d при делении на n. Так как она осуществляет приведение по модулю n *на каждом шаге*, то она намного эффективнее, чем комбинация `Mod` и `Power`.

4.1. Сравните эффективность вычисления 2^d по модулю 13 при помощи `PowerMod` и при помощи `Mod`. В какой момент вычисления появляется радостная надпись `Overflow occurred in computation`?

Наименьшее d такое, что $a^d \equiv 1 \pmod n$ называется **порядком** a по модулю n. Порядок a по модулю n вычисляется при помощи функции

`MultiplicativeOrder[a,n]`.

4.2. Возьмите Найдите порядки

В 1637 году Пьер де Ферма заметил, что

$$a^{p-1} \equiv 1 \pmod p$$

[57] C.Pomerance, The quadratic sieve factoring algorithm. — Lect. Notes Computer Science, 1985, vol.209, p.169–183.

[58] C.Pomerance, Factoring. — Proc Symp. Appl. Math., 1990, vol.42, p. 24–47.

для любого простого p и любого a взаимно простого с p. Иными словами, порядок любого такого a делит $p-1$. Этот факт известен как **теорема Ферма**.

4.3. Делится ли число $2222^{5555}+5555^{2222}$ на 7? а число $3333^{4444}+4444^{3333}$?

Элемент a такой, что все его степени $1, a, a^2, \dots, a^{p-2}$ попарно различны по модулю p, называется **примитивным корнем** по модулю p. В 1769 году Ламберт заметил, что по любому простому модулю существует примитивный корень, позже это было доказано Эйлером и Гауссом. Впрочем, это совсем очевидно. В самом деле, показатель m мультипликативной группы $(\mathbb{Z}/p\mathbb{Z})^*$ делит $p-1$. С другой стороны уравнение $x^m=1$ имеет в поле $\mathbb{Z}/p\mathbb{Z}$ ровно $p-1$ корень, так что $p-1\le m$. Таким образом, $m=p-1$, но это и значит, что группа $(\mathbb{Z}/p\mathbb{Z})^*$ циклическая.

4.4. Составьте таблицу примитивных корней по модулю первых нескольких десятков простых.

4.5. Составьте таблицу наименьших примитивных корней по модулю первой 1000 простых. Что можно сказать об их величине?

4.6. Убедитесь, что 2 является примитивным корнем по модулю всех простых p, удовлетворяющих следующим двум условиям:

- $p\equiv 3 \pmod 8$,
- $(p-1)/2$ тоже простое.

4.7. Убедитесь, что если q нечетное простое, удовлетворяющее следующим двум условиям:

- $p=2q+1$ тоже простое,
- $q\equiv 1 \pmod 4$,

то $1+q$ является примитивным корнем по модулю p.

4.8. Убедитесь, что если q нечетное простое, удовлетворяющее следующим двум условиям:

- $p=2q+1$ тоже простое,
- $q\equiv 3 \pmod 4$,

то q является примитивным корнем по модулю p.

4.9. Для каких p найдется такой примитивный корень по модулю p, который взаимно прост с $p-1$.

Ответ. Это действительно так для всех достаточно больших простых[59]

4.10. Верно ли, что по любому простому модулю $p\ge 5$ сущесвуют два последовательных примитивных корня?

4.11. Верно ли, что для всех достаточно больших простых можно найти сколь угодно длинные серии последовательных примитивных корней?

[59] M.Hausman, Primitive roots satisfying a coprime condition. — Amer. Math. Monthly, 1976, vol.83, p.720–723.

4.12. Для каких простых чисел существует такой примитивный корень a по модулю p, для которого $a^2 \equiv 1 + a \pmod{p}$?

4.13. Ясно, что 2 является примитивным корнем по модулю чисел Ферма $F_0 = 3$ и $F_1 = 5$. Докажите, что 2 не является примитивным корнем по модулю остальных простых чисел Ферма, но 3 является.

До сих пор мы искали примитивные корни для фиксированного простого p. Однако, можно встать на противоположную точку зрения, зафиксировать a и искать те простые p, для которых a является примитивным корнем.

4.14. Для каждого $a \leq 100$ попытайтесь найти наименьшее p по модулю которого a является примитивным корнем.

§ 5. Псевдопростые числа

> Christopher Robin knew that it was enchanted because nobody had ever been able to count whether it was sixty-three or sixty-four, not even when he tied a piece of string round each tree after he had counted it.
>
> Alan Alexander Miln, *The house at Pooh Corner*

Теорема Ферма утверждает, что $a^{p-1} \equiv 1 \pmod{p}$ для всех простых p. *Составное* число n, для которого выполняется сравнение

$$a^{n-1} \equiv 1 \pmod{n}$$

называется **псевдопростым** по основанию a. Совершенно особую роль играют псевдопростые числа по основанию 2, которые часто называются просто псевдопростыми. В класических книгах по теории чисел псевдопростые по основанию 2 часто называются еще **числами Пуле**, в честь Пуле, который в 1938 году опубликовал список псевдопростых $\leq 10^8$ (впрочем, содержавший ошибки).

5.1. Найдите первые 100 псевдопростых по основанию 2.

Ответ. Для дальнейших ссылок приведем часть ответа. Вот псевдопростые по основанию 2 меньшие 10000:

341	561	645	1105	1387	1729	1905	2047	2465	2701	2821
3277	4033	4369	4371	4681	5461	6601	7957	8321	8481	8911

5.2. Найдите количество псевдопростых $\leq 10^6$ по основанию 2 и сравните его с количеством простых $\leq 10^6$.

Указание. Выбирайте псевдопростые не среди всех чисел $\leq 10^6$, а среди *нечетных* чисел, меньших миллиона, т.е. из `Range[1,10^6,2]`.

Ответ. На 78498 простых меньших миллиона приходится только 245 псевдопростых по основанию 2. Это значит, что если для числа n выполняется сравнение $2^{n-1} \equiv 1 \pmod{n}$, то с вероятностью больше, чем 99,6% оно является простым.

Все приведенные выше псевдопростые по основанию 2 нечетны.

5.3. Существует ли четное псевдопростое по основанию 2 число?

Ответ. Да, как доказал в 1950 году Лемер, таких чисел бесконечно много. Наименьшее из них равно $161038 = 2 \cdot 73 \cdot 1103$.

5.4. Существуют ли псевдопростые по основанию 2, являющиеся полными квадратами?

Ответ. Да, по крайней мере два: $1194649 = 1093^2$ и $12327121 = 3511^2$.

Вот легко конструируемое бесконечное множество псевдопростых по основанию 2.

5.5. Убедитесь, что для любого простого $p \geq 5$ число $n = (2^{2p} - 1)/3$ псевдопросто по основанию 2. Найдите несколько первых получающихся по этой формуле псевдопростых.

В следующей задаче строится еще одно множество псевдопростых по основанию 2, которое, скорее всего, бесконечно.

5.6. Убедитесь, что для любого $n \geq 3$ число Ферма $F_n = 2^{2^n} + 1$ является либо простым, либо псевдопростым по основанию 2.

5.7. Для первых 100 простых p найдите наименьшее псевдопростое $\mathrm{sps}(p)$ по основанию p.

Решение. Для разнообразия напишем процедурную программу вычисляющую по a наименьшее псевдопростое по основанию a

```
smallest[a_]:=Block[{n=3},While[
            PrimeQ[n]||PowerMod[a,n-1,n]!=1,n=n+2];n]
```

Осталось применить это к списку первых 100 простых:

```
Map[{#,smallest[#]}&,Prime[Range[20]]]
```

Можно, конечно, и совсем топорно:

```
Table[{Prime[i],smallest[Prime[i]]},{i,1,100}]
```

Вот начало ответа:

2	3	5	7	11	13	17	19	23	29	31	37	41	43	47	53
341	91	217	25	15	21	9	9	33	15	15	9	15	21	65	9

Обращает на себя внимание обилие появлений во второй строке 9 и 15. Оказывается, асимптотически каждое из них появляется ровно в каждом третьем случае, что подчеркивает совершенно особую роль простых 3 и 5.

5.8. Найдите среди первых 10^6 простых количество тех, для которых 9 или 15 являются псевдопростыми.

5.9. Найдите количество псевдопростых $\leq 10^6$ по основаниям 3 и 5 и сравните его с количеством простых $\leq 10^6$.

Результаты этих штудий показывают, что тесты по основаниям 2, 3 и 5 сразу отбраковывают подавляющее большинство непростых чисел.

5.10. Найдите количество $\leq 10^6$ псевдопростых по основанию 2 и по еще какому-то основанию (скажем, 3, 5, 7, 11 и 13) и сравните его с общим количеством простых $\leq 10^6$.

Ответ. На 78498 простых меньших миллиона приходится только от 73 до 94 псевдопростых по основанию 2 и еще по одному из оснований $p = 3, 5, 7, 11, 13$. Например, если нечетное n не делится на 7 и для него выполняются сравнения $2^{n-1} \equiv 1 \pmod{n}$, $7^{n-1} \equiv 1 \pmod{n}$, то с вероятностью больше, чем 99,9% оно является простым.

Кажется, что добавляя сюда дополнительные тесты псевдопростоты, мы сможем перевести наше подозрение в том, что число является простым, в уверенность. Однако, как мы сейчас убедимся, это не так.

Составное число n называется **числом Кармайкла**, если оно является псевдопростым по любому основанию a взаимно простому с n. Ясно, что это достаточно проверять только для $a < n$. Это можно выразить чуть иначе. А именно, обозначим через $\lambda(n)$. наименьшее d такое, что $a^d \equiv 1 \pmod{n}$ для всех a взаимно простых с n. Функция $n \mapsto \lambda(n)$ известна как **функция Кармайкла**. По теореме Ферма $\lambda(p)$ делит $p-1$ для всех простых p. Так вот, число Кармайкла — это такое составное число, для которого тем не менее $\lambda(n)$ делит $n-1$.

5.11. Найдите числа Кармайкла $\leq 10^6$.

Ответ. Проще всего при помоuи функции `CarmichaelLambda`:

```
Select[Range[3,10^6,2],
          Not[PrimeQ[#]]&&Mod[#-1,CarmichaelLambda[#]]==0&]
```

Имеется 43 таких числа: 561, 1105, 1729, 2465, 2821, 6601, 8911, ...

В действительности, количество чисел Кармайкла бесконечно[60], так что ограничиться при проверке простоты числа никаким конечным количеством тестов псевдопростоты не удастся!

5.12. Напишите программу нахождения чисел Кармайкла не использующих функцию `CarmichaelLambda`.

§ 6. Простые Виферuха

> — I've a right to think — said Alice sharply, for she was beginning to feel a little worried.
> — Just about as much right, — said the Duchess, — as pigs have to fly.
>
> Lewis Carroll, *Alice's adventures in Wonderland*

В связи с первым случаем (великой) теоремы Ферма возник вопрос поиска простых чисел p, для которых выполняется сравнение

$$2^{p-1} \equiv 1 \pmod{p^2}.$$

[60] W.R.Alford, A.Granville, C.Pomerance. There are infinitely many Carmichael numbers. — Ann. Math., 1994, vol.140, p.703–722.

Такие числа называются **простыми Вифериха** (по основанию 2).

6.1. Найдите два первых простых числа Вифериха.

Ответ. Первое такое число, а именно 1093, было найдено Майсснером в 1913 году, а второе, 3511 — Беегером в 1922 году. Авторам не известны другие простые числа, удовлетворяющие этому сравнению. До 10^{14} других чисел Вифериха просто больше нет[61,62], а дальше вычисления на домашнем компьютере становятся слишком `time-consuming`.

А теперь продолжим изучения сравнения по модулю p^2 уже не для основания 2, а для других небольших оснований.

6.2. Найдите первые два простых числа p, для которых выполняется сравнение

$$3^{p-1} \equiv 1 \pmod{p^2}.$$

Решение. Поскольку мы собираемся многократно проверять это свойство по разным основаниям, определим функцию, проверяющую, является ли p простым Вифериха по основанию n:

```
wieferich[n_,p_]:=TrueQ[PowerMod[n,p-1,p^2]==1]
```

Теперь легко найти два первых числа с требуемым свойством, это $p = 11, 1006003$.

6.3. Найдите первые два простых числа p, для которых выполняется сравнение

$$5^{p-1} \equiv 1 \pmod{p^2}.$$

6.4. Очевидно, что $7^4 \equiv 1 \pmod{5^2}$. Найдите еще одно простое число p, для которых выполняется сравнение

$$7^{p-1} \equiv 1 \pmod{p^2}.$$

6.5. Найдите первые три простых числа p, для которых выполняется сравнение

$$17^{p-1} \equiv 1 \pmod{p^2}.$$

6.6. Найдите первые пять простых чисел p, для которых выполняется сравнение

$$19^{p-1} \equiv 1 \pmod{p^2}.$$

Простое число p называется **простым Вильсона**, если выполняется сравнение

$$(p-1)! \equiv -1 \pmod{p^2}.$$

6.7. Найдите первые три простых числа Вильсона.

Ответ. Это 5, 13, 563. До 10^8 никаких других простых Вильсона нет.

---

[61] D.H.Lehmer, On Fermat's quotient, base 2. — Math. Comput., 1981, vol.36, p.289–290.

[62] R.Crandall, K.Dilcher, C.Pomerance, A search for Wieferich and Wilson primes. — Math. Comput., 1997, vol.66, p.433–449.

§ 7. Сильно псевдопростые числа

> В каждом ручье, как известно, водится свой небольшой водяной, и его-то уж с виду нипочем не отличить от простой лягушки, разве что сам скажет.
>
> Михаил Успенский, *Там, где нас нет*

На самом деле тест простоты, основанный на теореме Ферма, легко усилить. Дело в том, что для простого p мультипликативная группа кольца классов вычетов $\mathbb{Z}/\mathbb{Z}$ циклическая. Таким образом, для нечетного p в ней существует ровно два корня из 1, а именно 1 и -1. Это значит, что не только $a^{p-1} \equiv 1 \pmod p$, но и

$$a^{(p-1)/2} = \pm 1 \pmod p.$$

Если $a^{(p-1)/2} = 1 \pmod p$, а число $(p-1)/2$ в свою очередь четно, то можно повторить это рассуждение. Таким образом, если 2^s — наибольшая степень 2, делящая $p-1$, то первое число в последовательности

$$a^{(p-1)/2}, a^{(p-1)/4}, \ldots, a^{(p-1)/2^s},$$

отличное от $1 \mod p$, должно равняться $-1 \mod p$.

Это дает основание для следующего определения. Нечетное число n называется **сильно псевдопростым** по основанию a, если оно псевдопростое по основанию a и первое число в последовательности

$$a^{(n-1)/2}, a^{(n-1)/4}, \ldots, a^{(n-1)/2^s},$$

где $n-1 = 2^s m$, m нечетно, отличное от $1 \mod n$, равно $-1 \mod n$. Сейчас мы напишем тест сильной псевдопростоты, при этом нам будет технически удобнее записывать последовательность степеней в обратном порядке, начиная с a^m и заканчивая a^{n-1}.

Прежде всего, нужно отделить нечетную часть числа.

7.1. Задайте функцию, сопоставляющую натуральному числу n его нечетную часть m, т.е. такое нечетное m, что $n = 2^s m$.

Решение. Проще всего, конечно, при помощи внутренней функции `IntegerExponent`:

```
oddpart[n_]:=n/2^IntegerExponent[n,2],
```

но можно и рекурсивно:

```
oddpart[n_]:=If[OddQ[n],n,oddpart[n/2]].
```

7.2. Пусть $n-1 = 2^s m$, где m нечетно. Определите функцию, сопоставляющую основанию a последовательность

$$a^m \mod n, \quad a^{2m} \mod n, \quad \ldots, \quad a^{(n-1)/2} \mod n, \quad a^{n-1} \mod n.$$

Решение. Проще всего вычислить первый член этой последовательности, а потом многократно возводить его в квадрат по модулю n. Это можно сделать при помощи `Nest` и `Table`, но еще удобнее воспользоваться командой `NestList`, специально предназначенной для формирования списков вида $\{x, f(x), f^2(x), \ldots, f^s(x)\}$:

```
powersequence[a_,n_]:=ReplaceAll[NestList[Mod[#^2,n]&,
            PowerMod[a,oddpart[n-1],n],IntegerExponent[n-1,2]],
                                                n-1->-1]
```

Например, вычисление `powersequence[2,341]` дает `{32,1,1}` так что 341 проходит тест псевдопростоты по основанию 2, но не проходит тест сильной псевдопростоты.

Теперь у нас все готово, чтобы написать тест, проверяющий, что число n удовлетворяет сильному условию псевдопростоты по основанию a, в общем случае.

7.3. Задайте тест, которые дает значение `True` если и только если n сильно псевдопростое по основанию a.

Решение. Условие сильной псевдопростоты состоит в том, что состоит в том, число не является простым и либо все члены построенной в предыдущей задаче последовательности равны 1, либо последний из них, который не равен 1, равен -1. Это можно выразить, например, так:

```
strongQ[a_,n_]:=Union[powersequence[a,n]]=={1}||
                          MemberQ[Most[powersequence[a,n]],-1]
```

7.4. Найдите количество сильно псевдопростых $\leq 10^6$ по основанию 2 и сравните его с количеством псевдопростых по основанию 2 и с количеством простых.

Ответ. Всего 46 непростых чисел меньших одного миллиона проходят тест сильной псевдопростоты по основанию 2. Вот те из них, которые меньше 100000:

2047, 3277, 4033, 4681, 8321, 15841, 29341, 42799, 49141, 52633, 65281, 74665, 80581, 85489, 88357, 90751.

Это значит, что сильно псевдопростых по основанию 2 примерно в 5 раз меньше, чем псевдопростых по основанию 2. Таким образом, число, удовлетворяющее тесту псевдопростоты по основанию 2 с вероятностью большей, чем 99.9% является простым! Иными словами, тест сильной псевдопростоты по основанию 2 имеет примерно такую же силу, что тест псевдопростоты по *всем* основаниям!

Это дает нам надежду добавив тесты сильной псевдопростоты по другим основаниям повысить вероятность. Чтобы не напрягать Ваш компьютер, мы не будем предлагать искать сильно псевдопростые по нескольким основаниям. Например, в работе[63] проверено, что первым непростым числом,

[63] C.Pomerance, J.L.Selfridge, S.S.Wagstaff, The pseudoprimes to $25 \cdot 10^9$. — Math. Comput., 1980, vol.36, p.1003–1026.

которое проходит тест сильной псевдопростоты по основаниям 2, 3, 5 и 7 является

$$3215031751 = 151 \cdot 751 \cdot 28351$$

и что никаких других таких чисел меньших 25 миллиардов нет! Еще более замечательный пример привел в 1996 году Даниэль Бляйхенбахер, который обнаружил, что число

$$\begin{gathered}
\begin{aligned}
&68528663395046912244223605902738356719751082784386681071 = \\
&\quad 1821574545258925963 9 \cdot 4337082250616490391 \cdot 867416450123298079
\end{aligned}
\end{gathered}$$

является сильно псевдопростым по всем основаниям $a \leq 100$.

7.5. Проверьте это утверждение и найдите такое основание, по которому это число не является сильно псевдопростым.

§ 8. Теорема Эйлера

> И опять я возрадуюсь радостью Моцарта, которая завидна как сама по себе, так и потому, что радует всех, кто может приобщиться к ней. Мне, по крайней мере, неописуемую радость доставило даже отдаленное понимание Моцарта и представление об испытанной им радости.
>
> Серен Кьеркегор

В настоящем параграфе мы обсудим обобщение теоремы Ферма с простых на все натуральные модули. Это обобщение дается в терминах **функции Эйлера** φ. Функция Эйлера φ имплементирована в системе под именем `EulerPhi`.

`EulerPhi[n]`	функция Эйлера $\varphi(n)$
`MoebiusMu[n]`	функция Мебиуса $\mu(n)$

По определению для натурального n значение функции Эйлера $\varphi(n)$ равно количеству $0 \leq m \leq n$ взаимно простых с n. Если $m = p$ — просто, то $\varphi(p) = p - 1$.

Хорошо известна следующая формула для функции Эйлера. Если $n = p_1^{l_1} \dots p_s^{l_s}$ есть каноническое разложение n на простые, то

$$\varphi(n) = n \left(1 - \frac{1}{p_1}\right) \dots \left(1 - \frac{1}{p_s}\right).$$

Эта формула сразу вытекает из двух следующих наблюдений:

- Если $m = p^s$ — примарно, то

$$\varphi(p^s) = p^s - p^{s-1} = p^s \left(1 - \frac{1}{p}\right).$$

- Функция Эйлера мультипликативна, иными словами,

$$\varphi(mn) = \varphi(m)\varphi(n)$$

для любых взаимно простых m и n.

8.1. Определите функцию Эйлера двумя способами, непосредственно по определению и используя разложение числа n на простые при помощи функции `FactorInteger` и сравните время работы этих программ между собой и со временем вычисления внутренней функции `EulerPhi`.

Решение. Определение функции Эйлера кодируется легко

```
euler1[n_]:=Length[Select[Range[0,n-1],GCD[#,n]==1&]]
```

но вот только использовать такую программу невозможно. Не только из-за контроля времени, а просто физически невозможно — прикиньте, сколько места в памяти занимает список всех чисел до нескольких миллиардов. Любая попытка обратиться к подобной программе для многозначных чисел моментально придет к сообщению

```
No more memory available.
Mathematica kernel has shut down.
```

Поэтому попробуем иначе:

```
euler2[n_]:=n*Apply[Times,Map[(1-1/#)&,
                        First[Transpose[FactorInteger[n]]]]]
```

Обратите внимание, что здесь мы вычисляем `FactorInteger[n]` *один раз*, а потом применяем ко всем его элементам функцию $x \mapsto 1 - 1/x$ и берем произведение получившегося списка. Интересно, что эта функция работает почти ровно с той же скоростью, что внутренняя функция `EulerPhi` — и даже чуть быстрее. Это значит ровно то, что вычисление `EulerPhi[n]` тоже использует разложение n на простые множители.

В действительности сложность вычисления $\varphi(n)$ приблизительно эквивалентна сложности разложения n на простые. Например, если $n = pq$ есть произведение двух различных простых, то $\varphi(n) = (p-1)(q-1)$ так что $p + q = n - \varphi(n) + 1$. Это значит, что зная $\varphi(n)$ мы можем найти p и q просто решив одно квадратное уравнение.

8.2. Составьте таблицу значений $\varphi(n)$ при $n \le 100$.

Ответ. Вот начало этой таблицы:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	1	2	2	4	2	6	4	6	4	10	4	12	6	8	8	16	6	18

8.3. Убедитесь, что для любого составного n выполняется неравенство $\varphi(n) \le n - \sqrt{n}$.

8.4. Убедитесь, что для любого натурального n выполняется равенство $\sum \varphi(d) = n$, где сумма берется по всем $d|n$.

Обычное доказательство этого факта использует **формулу обращения Мебиуса**, в терминах функции Мебиуса μ, значение которой $\mu(n)$ равно 0, если n делится на p^2 и равно $(-1)^s$, если $n = p_1 \dots p_s$ есть произведение s *различных* простых. В системе функция Мебиуса имплементирована под

именем `MoebiusMu`. Мы подробно обсуждаем формулу Мебиуса в выпуске 2 и еще раз возвращаемся к ней в выпуске 3 в связи с круговыми многочленами, поэтому пока отметим лишь формулу, связывающую ϕ и μ:

$$\phi(n) = \sum \mu(d)\frac{n}{d},$$

где сумма берется по всем делителям d числа n.

8.5. Определите функцию Эйлера при помощи этой формулы и сравните время работы этой программы со временем вычисления внутренней функции `EulerPhi`.

8.6. Убедитесь, что арифметическая функция $n \mapsto n\varphi(n)$ инъективна. Иными словами, если $m\varphi(m) = n\varphi(n)$ для некоторых натуральных m и n, то $m = n$.

8.7. Убедитесь, что функция $n \mapsto \varphi(n)$ настолько далека от инъективности, насколько это возможно. А именно, если уравнение $\varphi(x) = l$ разрешимо, то существуют *по крайней мере* два различных натуральных числа m и n такие, что $\varphi(m) = \varphi(n) = l$. Проверьте это утверждение для всех $l < 100000$.

В общем случае это утверждение до сих пор не проверено и составляет содержание знаменитой **гипотезы Кармайкла**[64]. Интересно, что до этого этот факт считался доказанным и даже фигурировал как упражнение в учебнике Кармайкла[65]. Если контрпример к гипотезе Кармайкла существует, то он *велик* — заведомо $\geq 10^{10^4}$ — так что найти его нет никаких шансов.

8.8. Убедитесь, что для нечетного l уравнение $\varphi(x) = 2l$ имеет 0, 2 или 4 решения.

8.9. Сколько решений может иметь уравнение $\varphi(x) = l$ для данного l?

Ответ. Гипотеза Серпиньского утверждает, что сколько угодно, *кроме* ровно одного.

8.10. Существуют ли *составные* числа, меньшие одного миллиона, для которых $\varphi(n)$ делит $n - 1$?

Ответ. Вопрос о том, существуют ли вообще хотя бы одно такое число, составляет содержание **гипотезы Лемера**[66]. До нескольких миллиардов таких чисел нет, а дальше их систематических поиск на домашнем компьютере невозможен.

8.11. Рассмотрим число $9\ldots9^{9\ldots9} + 1$, где количество девяток в основании степени равно m, а количество девяток в показателе степени равно n. Сколькими нулями заканчивается это число?

---

[64] R.D.Carmichael, Note on Euler's φ-function. — Bull. Amer. Math. Soc., 1922, vol.28, p.109–110.

[65] R.D.Carmichael, The theory of numbers. — N.Y., 1914.

[66] D.H.Lehmer, On Euler's totient function. — Bull. Amer. Math. Soc., 1932, vol.38, p.745–751.

8.12. Вычислите

$$7^{7^{\cdots^{7}}} \mod 13, \qquad 11^{11^{\cdots^{11}}} \mod 17, \qquad 9^{9^{\cdots^{9}}} \mod 23,$$

где показатель степени повторен n раз.

Указание. Как Вы думаете, в чем будет состоять проблема? Ясно, что уже четвертое повторение показателя степени приведет к переполнению, `Overflow[]`. Поэтому некоторые — или все! — шаги нужно провести вручную, пользуясь теоремой Эйлера.

§ 9. Квадратичные вычеты

— А, дурацкий анекдот, — сказало чадо с пренебрежением. — "Все прекрасно, но не делится пополам". Этот, что ли?
— Именно! — воскликнул Симонэ и разразился хохотом.
— Делится пополам? — улыбаясь, спросила госпожа Мозес.
— Не делится! — сердито поправило чадо.
— Ах, не делится? — удивилась госпожа Мозес. — А что именно не делится?

Аркадий Стругацкий, Борис Стругацкий, *Дело об убийстве или отель 'У погибшего альпиниста'*

Во многих вопросах теории чисел, комбинаторики, теории кодирования возникает вопрос о решении уравнения

$$x^2 \equiv a \pmod{n}.$$

Элемент a называется **квадратичным вычетом** по модулю n, если это уравнение имеет решение и **квадратичным невычетом** в противном случае. Особое значение имеет случай, когда p является простым числом. При исследовании этого уравнения естественно возникает два вопроса:

- выяснить, разрешимо ли уравнение $x^2 \equiv a \pmod{p}$;
- фактически решить это уравнение.

Оказывается, на вопрос о *существовании* решения легко дать совсем простой ответ, который столь же легко превращается в эффективный алгоритм, в то время как фактический поиск этого решения является чрезвычайно сложной вычислительной задачей.

`JacobiSymbol[n,m]`	символ Якоби $\left(\frac{n}{m}\right)$

В 1798 году Лежандр ввел следующий символ

$$\left(\frac{a}{p}\right) = \begin{cases} +1, & a \text{ квадратичный вычет по модулю } p, \\ -1, & a \text{ квадратичный невычет по модулю } p, \end{cases}$$

Обычно символ Лежандра распространяют на все целые числа дополнительно полагая $\left(\frac{a}{p}\right) = 0$ в случае, когда a делится на p. В системе имплементирована функция `JacobiSymbol`, которая позволяет вычислять символ

Лежандра — в действительности, несколько более общий символ Якоби, смысл которого мы обсудим чуть позже.

9.1. Напишите программу для вычисления символа Лежандра непосредственно по определению. А теперь не проводя никаких вычислений скажите, почему эта программа чудовищно плоха.

Решение. Можно, например, при помощи условного оператора `Which`, который возвращает результат, фигурирующий после первого теста дающего `True`:

```
legendre[a_,p_]:=Which[Mod[a,p]==0,0,
        MemberQ[Map[Mod[#,p]&,Range[p]^2],Mod[a,p]],+1,True,-1]
```

Обратите внимание на *порядок* тестов. Что произойдет, если переставить два первых теста? Однако, эта программа безнадежно плоха — по той же причине, по которой никуда не годилась основанная на определении программа для вычисления функции Эйлера.

Таким образом, нам нужна либо формула для символа Лежандра, либо хороший алгоритм для ее вычисления. Сейчас мы предъявим и то и другое. Следующая формула называется **критерием Эйлера**:

$$\left(\frac{a}{p}\right) \equiv a^{(p-1)/2} \pmod{p}.$$

9.2. Напишите программу для вычисления символа Лежандра, основанную на критерии Эйлера.

Решение. Можно, конечно, так:

```
legendre[a_,p_]:=Mod[a^((p-1)/2),p,-1]
```

Обратите внимание на использование отступа -1, гарантирующее, что получающиеся вычеты равны 1 или -1 — а не 1 и $n-1$ как происходило бы при вызове `Mod` с двумя аргументами. Однако, гораздо лучше так:

```
legendre[a_,p_]:=Mod[PowerMod[a,(p-1)/2,p],p,-1]
```

Как легко убедиться экспериментально, команда `PowerMod`, проводящая редукцию $\bmod p$ на каждом шаге, работает гораздо быстрее, чем `Mod` и `Power`.

Еще одна характеризация символа Лежандра дается **леммой Гаусса**, которая утверждает, что $\left(\frac{a}{p}\right) = (-1)^s$, где s есть количество тех чисел $m = 1, 2, \ldots, (p-1)/2$, для которых остаток произведения am по модулю p строго больше, чем $(p-1)/2$.

9.3. Напишите программу для вычисления символа Лежандра, основанную на лемме Гаусса.

9.4. Верно ли, что для любого простого p между 1 и $p-1$ найдутся *два* последовательных числа, которые оба являются квадратичными вычетами?

Ответ. Да, для любого $p \geq 7$.

9.5. Для каких простых p между 1 и $p-1$ найдутся *три* последовательных числа, которые оба являются квадратичными вычетами?

Ответ. Для всех $p \geq 19$.

9.6. Для каких простых p между 1 и $p-1$ найдутся *четыре* последовательных числа, которые оба являются квадратичными вычетами?

§ 10. Квадратичный закон взаимности

Numero impari deus gaudet.

Вергилий

Falstaff. I hope good luck lies in odd numbers. They say there is divinity in odd numbers.

William Shakespeare, *The merry wives of Windsor*

Why get even, when you can get odd?

Oscar Wilde

This is better, than reading Vergil

Henry Miller, *Black Spring*

Spotkałem kiedyś czlowieka tak nieoczytanego, że musiał cytaty z klasyków wymyślać sam[67].

Stanisław Jerzy Lec.

Обратимся теперь к еще одному эффективному способу вычисления символа Лежандра. Для этого резюмируем основные свойства символа Лежандра, которые полностью его определяют.

- Если $a \equiv b \pmod p$, то $\left(\dfrac{a}{p}\right) = \left(\dfrac{b}{p}\right)$.
- Если a взаимно просто с p, то $\left(\dfrac{a^2 b}{p}\right) = \left(\dfrac{b}{p}\right)$.
- Символ Лежандра **мультипликативен** по первому аргументу:

$$\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right).$$

Две следующие формулы открыты Пьером де Ферма, первую из них доказал Эйлер, а вторую Лагранж.

- **Erster Ergänzunssatz**: $\left(\dfrac{-1}{p}\right) = (-1)^{(p-1)/2}$.

[67] Однажды я встретил человека до такой степени необразованного, что он вынужден был сам выдумывать все цитаты из классиков.

Иными словами, -1 является квадратичным вычетом по модулю простых вида $4m+1$ и невычетом по модулю простых вида $4m+3$. Ответ на аналогичный вопрос для 2 зависит от класса p^2 по модулю 16.

• **Zweiter Ergänzunssatz**: $\left(\frac{2}{p}\right) = (-1)^{(p^2-1)/8}$.

Сформулируем, наконец, самое глубокое из свойств символа Лежандра и единственное, доказательство которого не совсем очевидно.

• **Квадратичный закон взаимности**

$$\left(\frac{p}{q}\right)\left(\frac{q}{p}\right) = (-1)^{(p-1)(q-1)/4}.$$

Это свойство было открыто Эйлером, а потом переоткрыто и доказано Лежандром и Гауссом. В настоящее время известно около 200 различных доказательств этого результата, самые простые из которых (доказательство Золотарева и доказательство Суона) занимают 3–4 строчки.

10.1. Проверьте последние три свойства экспериментально для первых 100 нечетных простых p.

10.2. Напишите программу вычисления символа Лежандра, основанную на этих свойствах.

Указание. Допустим, мы хотим вычислить $\left(\frac{a}{p}\right)$. Прежде всего приведем a по модулю p, взяв при этом вычет от $-(p-1)/2$ до $(p-1)/2$. Если a отрицательно, воспользуемся первым дополнением к закону взаимности. Разложив его теперь на простые множители $a = p_1^{l_1} \dots p_s^{l_s}$ (скажем, при помощи команды `Factor Integer`), выделим бесквадратную часть $q_1 \dots q_t$, где $q_1, \dots, q_t$ — попарно различные простые, входящие в разложение a в нечетной степени. Если $q_1 = 2$, воспользуемся вторым дополнением к закону взаимности. Таким образом, в силу мультипликативности нам остается только вычислить символ Лежандра $\left(\frac{q}{p}\right)$, где $q < p$ нечетное простое. Преобразуем этот символ в $(-1)^{(p-1)(q-1)/4}\left(\frac{p}{q}\right)$ по закону взимности и начнем всю процедуру с самого начала. А теперь превратите это в рабочий код.

Якоби следующим образом обобщил символ Лежандра. Пусть m — любое целое число, а n — нечетное число. Пусть $n = p_1^{k_1} \dots p_s^{k_s}$ — каноническое разложение n на простые. Тогда **символ Якоби** определяется как

$$\left(\frac{m}{n}\right) = \left(\frac{m}{p_1}\right)^{k_1} \dots \left(\frac{m}{p_s}\right)^{k_s}.$$

В случае, когда $n = p$ простое, символ Якоби совпадает с символом Лежандра. Легко проверить, что символ Якоби обладает *всеми* перечисленными в предыдущем пункте формальными свойствами и, кроме того, следующим дополнительным свойством.

• Символ Якоби **мультипликативен** по второму аргументу:

$$\left(\frac{l}{mn}\right) = \left(\frac{l}{m}\right)\left(\frac{l}{n}\right).$$

10.3. Напишите программу вычисления символа Якоби, основанную на этих свойствах.

> "Would you tell me, please, which way I ought to go from here?"
> "That depends a good deal on where you want to get to," said the Cat.
>
> Lewis Carroll, *Alice's adventures in Wonderland*

Глава 8. АДДИТИВНАЯ ТЕОРИЯ ЧИСЕЛ

Простые числа не нужно складывать. Простые числа нужно умножать[68].

Лев Ландау

Грубое суеверие это распространяется преимущественно так называемыми учеными, то есть людьми особенно ограниченными и потерявшими способность самобытного, разумного мышления, вследствие постоянного изучения чужих мыслей и занятия самыми праздными и ненужными вопросами.

Лев Толстой, *Почему христианские народы вообще и в особенности русский находятся теперь в бедственном положении*

Если мы и вправе гордиться, то не тем, чего мы добились, а лишь тем, что мы хотели и чего не могли сделать.

Георгий Адамович, *Комментарии*

Была мысль найти объединяющую идею. Нашлись только разъединяющие.

Виктор Ерофеев, *Энциклопедия русской души*

В настоящей главе мы обсуждаем простейшие факты **аддитивной теории чисел**, иными словами, свойства целых чисел относительно *сложения*. Типичными классическими задачами аддитивной теории чисел являются проблемы о совершенных и дружественных числах, проблема Варинга о представлении натурального числа как суммы m-х степеней и проблема Гольдбаха о представлении натуральных чисел как сумм двух или трех простых. В отличие от мультипликативной теории чисел, которая уже в XIX веке оформилась в огромную самостоятельную науку, занимающую одно из центральных мест в математике, аддитивная теория чисел представляет собой хаотическое нагромождение тонких наблюдений, хитрых трюков и искусственных приемов, и сегодня мы почти столь же далеки от решения основных задач этой теории, как 2500 лет назад.

§ 1. Суммы делителей

All this divided York and Lancaster.

William Shakespeare, *King Richard* III

Многие важные результаты выражаются в терминах делителей числа n. В системе имеются две внутренние функции, `Divisors` и `DivisorSigma`,

[68] “Моню не нужно нюхать, Моню нужно учить.”

первая из которых порождает список всех делителей числа n, а вторая — сумму их m-х степеней.

`Divisors[n]`	натуральные делители n
`DivisorSigma[n,m]`	сумма m-х степеней делителей

Пусть каноническое разложение числа n имеет вид $n = p_1^{k_1} \dots p_s^{k_s}$. Тогда количество делителей

$$d(n) = (k_1 + 1) \dots (k_s + 1),$$

а сумма делителей — формулой

$$\sigma(n) = \sum_{i \le k_1, \dots, i_s \le k_2} p_1^{i_1} \dots p_s^{i_s} = \frac{p_1^{k_1+1} - 1}{p_1 - 1} \dots \frac{p_s^{k_s+1} - 1}{p_s - 1}$$

(сумма геометрической прогрессии).

1.1. Задайте функции $d(n)$ и $\sigma(n)$ на основе внутренней функции `FactorInteger` и сравните скорость их работы с `Length[Divisors[n]]` и `DivisorSigma[1,n]`.

Из формулы для $d(n)$ вытекает, что

• $d(n)$ зависит не от самого n, а от его арифметической структуры, иными словами, от того, с какими степенями в n входят различные простые;

• значение $d(n)$ может быть абсолютно любым.

1.2. Для любого простого числа q найдите наименьшее число, имеющее ровно q делителей.

Ответ. Небольшой компьютерный эксперимент убедит Вас в том, что это 2^{q-1}. Вообще, любое примарное число p^{m-1} имеет ровно m делителей, но в случае, когда m раскладывается на множители, как правило, удается построить много меньшее, чем 2^{m-1} число с тем же количеством делителей.

1.3. Составьте таблицу, в которой для каждого числа $m \le 100$ указано наименьшее число, имеющее ровно m делителей.

Ответ. Вот начало этой таблицы:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2	4	6	16	12	64	24	36	48	1024	60	4096	192	144	120

1.4. Задайте функцию, сопоставляющую каждому натуральному m наименьшее натуральное n имеющее ровно m делителей.

1.5. Убедитесь, что для любого натурального числа $n > 1$ найдется такое m, что $d^m(n) = 2$. Сколь велико может быть это m?

Ответ. Поэкспериментировав с функцией d, легко убедиться, что для любого $n > 2$ имеем $d(n) < n$, поэтому применение d может оборваться только

на значении 2. С другой стороны, мы только что заметили, что $d(2^{n-1}) = n$, поэтому если от n можно дойти до 2 за m шагов, то от 2^{n-1} требуется уже $m+1$ шаг.

Обратимся теперь к функции σ. Ясно, что σ возрастающая функция, $\sigma(n) > n$ для всех $n > 1$, причем $\sigma(n) = n+1$ только в случае, когда $n = p$ простое.

1.6. Убедитесь, что если n составное, то $\sigma(n) > n + \sqrt{n}$.

1.7. Любое ли m может быть значением функции σ?

1.8. Найдите все решения уравнения $\sigma(n) = \sigma(n+1)$ при $n \leq 100000$.

Лео Мозер привел примеры, показывающие, что в отличие от арифметической функции $n \mapsto n\varphi(n)$, функция $n \mapsto n\sigma(n)$ не инъективна, иными словами, равенство

$$m\sigma(m) = n\sigma(n)$$

возможно и при $m \neq n$. А именно, при $m = 12$, $n = 14$ обе части здесь равны 336. Ясно, что умножая обе части этого равенства на любое число k взаимно простое с 2, 3 и 7, мы получим новую тройку чисел $m = 12k$, $n = 14k$ удовлетворяющую этому условию. Поэтому интересно искать примитивные пары, для которых $(m/k, n/k)$ не являются решениями этого уравнения ни при каком $k > 1$.

В действительности пример Мозера является первым из примеров следующего типа: $m = 2^{p-1}M_q$, $n = 2^{q-1}M_p$, где M_p и M_q различные простые числа Мерсенна.

1.9. Укажите еще 902 примитивных решения уравнения $m\sigma(m) = n\sigma(n)$.

1.10. Задайте функцию, сопоставляющую паре (m, n) сумму их общих делителей.

Во многих задачах возникают различные варианты функции σ, например, функция $\widehat{\sigma}$, сопоставляющая n сумму его *собственных* делителей. Следующая функция σ^* естественно возникает в задаче о количестве представлений натурального числа как суммы четырех квадратов.

1.11. Задайте функцию σ^*, которая сопоставляет каждому натуральному числу сумму тех его делителей, которые не делятся на 4.

1.12. Пусть $p_1, \ldots, p_s$ суть все различные простые делители числа n, а

$$m = \frac{n}{p_1 \ldots p_s}$$

Для нескольких десятков n вычислите сумму $\psi(n)$ делителей числа n, являющихся кратными числа m, и угадайте формулу для этой суммы в общем случае.

Ответ. Искомая формула лишь знаком отличается от формулы для функции Эйлера:

$$\psi(n) = n\left(1 + \frac{1}{p_1}\right)\ldots\left(1 + \frac{1}{p_s}\right).$$

§ 2. Совершенные числа

No, I certainly do not believe in this superstition. But you know, they say it brings luck even if you don't believe in it.

Niels Bohr

God not only plays dice. He also somethimes throws the dice where they cannot be seen.

Stephen Hawking

Числа Мерсенна играют абсолютно исключительную роль в одной из *старейших* нерешенных проблем математики, относящейся к *четным* совершенным числам. Число n называется **совершенным**, если оно равно сумме своих собственных делителей. Иными словами, $\sigma(n) = 2n$. В терминах функции $\widehat{\sigma}(n) = \sigma(n) - n$ это условие записывается еще естественнее, $\widehat{\sigma}(n) = n$.

Уже в "Элементах" Эвклида содержалось наблюдение (Книга IX, теорема 36), что если $2^p - 1$ простое, то $2^{p-1}(2^p - 1)$ совершенное.

2.1. Найдите совершенные числа $\leq 10^7$.

Ответ. Можно просто полным перебором с использованием `DivisorSigma`. Вот они:

$$6 = 2 \cdot 3 = 1 + 2 + 3,$$

$$28 = 2^2 \cdot 7 = 1 + 2 + 4 + 7 + 14,$$

$$496 = 2^4 \cdot 31 = 1 + 2 + 4 + 6 + 16 + 31 + 62 + 124 + 248$$

$$8128 = 2^6 \cdot 127 = 1 + 2 + 4 + 8 + 16 + 32 + 64 + 127 + 254 + 508 + 1016 + 2032 + 4064$$

$$\begin{aligned} 33550336 = 2^{12} \cdot 8191 = 1 + 2 + 4 + 8 + 16 + 32 + 64 + 128 + 256 + 512 + \\ 1024 + 2048 + 4096 + 8191 + 16382 + 32764 + \\ 65528 + 131056 + 262112 + 524224 + 1048448 + \\ 2096896 + 4193792 + 8387584 + 16775168 \end{aligned}$$

Три первых были известны уже в VI веке до н.э., а четвертое нашел Никомах из Герасы около 100 года н.э.

Эйлер показал, что *все* четные совершенные числа имеют такой вид. Точнее, имеет место следующая **теорема Эвклида—Эйлера**: множество четных совершенных чисел совпадает с множеством чисел вида $2^{p-1}M_p$, где M_p — простое число Мерсенна.

2.2. Найдите еще 38 совершенных чисел.

Теорема Эвклида—Эйлера сводит вопрос о бесконечности множества четных совершенных чисел к вопросу о бесконечности множества простых чисел Мерсенна. Как мы уже упоминали, ответ на этот вопрос неизвестен. До сих пор неизвестно и то, существуют ли *нечетные* совершенные числа.

Эти две задачи, видимо, являются *самыми* старыми нерешеннными проблемами в математике, так как они были сформулированы еще в начале V века до н.э. в секте пифагорейцев.

Не пытайтесь искать нечетные совершенные числа вручную. Уже в середине 1970-х годов было доказано, что у нечетного совершенного числа по крайней мере 8 различных простых делителей[69]. Тогда же, см., например[70,71], было проверено, что среди чисел $< 10^{200}$ нечетных совершенных чисел нет. Кроме того, известно много других ограничений на нечетные совершенные числа. Например, их наибольший простой делитель должен быть > 300000 и получены аналогичные оценки для следующих делителей[72]. С тех пор все эти оценки многократно улучшались. Очевидно, что с учетом этих ограничений найти нечетное совершенное число на бытовом компьютере нет *никаких* шансов.

2.3. Если у числа n не более трех различных простых делителей, *как правило* из того, что n делит $\sigma(n)$ вытекает, что n совершенно. Найдите исключения.

Ответ. Имеются два таких числа, а именно 120 и 672, для которых $\sigma(n) = 3n$.

Для совершенного числа n выполняется равенство $\sigma(n) = 2n$. Число n называется **сверхсовершенным**, если $\sigma(\sigma(n)) = 2n$.

2.4. Найдите сверхсовершенные числа меньшие одного миллиона и сформулируйте гипотезу о том, как выглядят все сверхсовершенные числа.

Ответ. Таких чисел семь:

$$2 = 2^{2-1}, \quad 4 = 2^{3-1}, \quad 16 = 2^{5-1}, \quad 64 = 2^{7-1}, \quad 4096 = 2^{13-1},$$
$$65536 = 2^{17-1}, \quad 262144 = 2^{19-1}$$

Все они являются степенями двойки, а список показателей уже встречался нам в связи с числами Мерсенна. Как заметил Сурьянараяна[73], эта гипотеза верна: любое четное сверхсовершенное число имеет вид 2^{p-1}, для некоторого *простого* числа Мерсенна M_p.

Число n называется **избыточным**, если $\widehat{\sigma}(n) > n$, и **недостаточным**, если $\widehat{\sigma}(n) < n$.

2.5. Каких чисел среди чисел $< 10^6$ больше, избыточных или недостаточных?

---

[69] P.Hagis, Outline of a proof that every odd perfect number has at least eight prime factors. — Math. Comput., 1980, vol.35, p.1027–1031.

[70] B.Tuckerman, A search procedure and lower bound for odd perfect numbers. — Math. Comput., 1973, vol.27, p.943–949.

[71] P.Hagis, A lower bound for the set of odd perfect numbers. — Math. Comput., 1973, vol.35, p.1027–1031.

[72] P.Hagis, On the largest prime divisor of an odd perfect number. — Math. Comput., 1975, vol.29, p.922–924.

[73] D.Suryanarayana, Super perfect numbers. — Elemente Math., 1969, vol.24, p.16–17.

Число n называется **полусовершенным**, если оно является суммой *каких-то* — не обязательно всех! — своих собственных делителей. Число называется **причудливым**, если оно избыточно, но не полусовершенно.

2.6. Найдите все причудливые числа, меньшие 500.

Ответ. Такое число ровно одно, а именно, 70.

§ 3. Дружественные числа

> God not only plays dice. He also somethimes throws the dice where they cannot be seen.
>
> Stephen Hawking

В связи с совершенными числами невозможно не упомянуть и о другой пифагорейской задаче — задаче о дружественных числах. Числа m и n называются **дружественными**, если сумма собственных делителей числа m равна n, а сумма собственных делителей числа n равна m. Иными словами, одновременно выполняются равенства $\widehat{\sigma}(m) = n$ и $\widehat{\sigma}(n) = m$ или, что то же самое, $\sigma(m) = \sigma(n) = m + n$.

Известный болтун и фантазер Ямвлих из Хальциса приписывает лично товарищу Пифагору с острова Самос открытие первой пары дружественных чисел

$$220 = 2^2 \cdot 5 \cdot 11, \qquad 284 = 2^2 \cdot 71.$$

Впрочем, Леонард Диксон отмечает[74], что уже в относящейся к более ранней дате части Библии в знак примирения Иаков подарил Исаву, брату своему, *ровно* 220 овец и 220 коз[75], а Поль Таннери считал, что магические свойства пары 220, 284 были известны уже в древнем Египте.

В IX веке арабский математик абу-Хасан Сабит ибн-Корра ибн Марван аль-Харрани доказал следующий результат. **Теорема Сабита ибн-Корры**: если все три числа $p = 3 \cdot 2^{n-1} - 1$, $q = 3 \cdot 2^n - 1$ и $r = 3^2 2^{2n-1} - 1$ нечетные простые, то числа $2^n pq$ и $2^n r$ — дружественные.

3.1. Найдите три пары дружественных чисел.

Указание. Как всегда, когда речь идет о небольшом переборе вначале по-простому: `Select`, `PrimeQ` и поверх `Map`. Обратите внимание, что $n \geq 2$, иначе $p = 2$.

Ответ. Пифагорейская пара получается, если взять в теореме Сабита ибн-Корры $n = 2$. С помощью этой теоремы в XIII веке другой арабский ученый, ибн аль-Банна, открыл следующую пару дружественных чисел,

$$17296 = 2^4 \cdot 23 \cdot 47, \qquad 18416 = 2^4 \cdot 1151,$$

отвечающую случаю $n = 4$. Теорема Сабита ибн-Корры была независимо переоткрыта в 1636 году Пьером Ферма и в 1638 Рене Декартом. При

[74] L.E.Dickson, History of the theory of numbers, vol. I. — Chelsea, 1952.

[75] "Двести коз, двадцать козлов, двести овец, двадцать овнов", Книга Бытия, XXXII, 14.

этом Ферма переоткрыл пару, отвечающую случаю $n = 4$, а Декарт нашел следующую пару,

$$9363584 = 2^7 \cdot 191 \cdot 383, \qquad 9437056 = 2^7 \cdot 73727,$$

отвечающую случаю $n = 7$. Сейчас мы можем найти все эти пары за доли секунды.

3.2. Найдите все дружественные числа $\leq 10^6$.

Ответ. В данном случае, конечно, лучше не выбирать их из списка, а организовать цикл, вычисляющий все эти числа за секунды:

220	284	1184	1210	2620	2924
5020	5564	6232	6368	10744	10856
12285	14595	17296	18416	63020	76084
66928	66992	67095	71145	69615	87633
79750	88730	100485	124155	122265	139815
122368	123152	141664	153176	142310	168730
171856	176336	176272	180848	185368	203432
196724	202444	280540	365084	308620	389924
319550	430402	356408	399592	437456	455344
469028	486178	503056	514736	522405	525915
600392	669688	609928	686072	624184	691256
635624	712216	643336	652664	667964	783556
726104	796696	802725	863835	879712	901424
898216	980984				

Кроме того, имеется две пары дружественных чисел, одно из которых меньше миллиона:

947835 1125765 998104 1043096

Обратите внимание, что все эти пары либо четные, либо нечетные. Вопрос о существовании четно-нечетных пар[76] по-прежнему открыт.

Эйлер обнаружил и *пятьдесят девять* новых пар дружественных чисел, как четных, так и *нечетных* С тех пор было обнаружено много сот новых пар, но, тем не менее, вопрос о *бесконечности* множества таких пар открыт так же широко, как во время Пифагора.

Леонард Диксон предложил следующее обобщение понятия дружественных чисел[77,78] — совершенно другое обобщение, предложенное Каталаном,

[76]Вальтер Боро, Дружественные числа, двухтысячелетняя история одной арифметической задачи. — В книге “Живые числа”, М., Мир, 1985, с.11–41.

[77]L.E.Dickson, Amicable number triples. — Amer. Math. Monthly, 1913, vol.20, p.84–91.

[78]Th.E.Mason, On amicable numbers and their generlizations. — Amer. Math. Monthly, 1921, vol.28, p.195–200.

обсуждается в следующем параграфе. А именно, он говорит, что $n_1, \ldots, n_m$ образуют m-ку дружественных чисел, если

$$\sigma(n_1) = \ldots = \sigma(n_m) = n_1 + \ldots + n_m.$$

Существуют ли дружественные m-ки при $m \geq 3$?

3.3. Постройте четыре первых дружественных тройки.

Ответ. Вот самая маленькая[79] из них:

$$1980 = 2^2 \cdot 3^2 \cdot 5 \cdot 11, \qquad 2016 = 2^5 \cdot 3^2 \cdot 7, \qquad 2556 = 2^2 \cdot 3^2 \cdot 71,$$

с суммой 6552. Вот следующая

$$9180 = 2^2 \cdot 3^3 \cdot 5 \cdot 17, \qquad 9504 = 2^5 \cdot 3^3 \cdot 11, \qquad 11556 = 2^2 \cdot 3^3 \cdot 107,$$

с суммой 30240. Еще две совсем маленькие тройки с суммами 70680, 87360 без труда строятся за секунды.

3.4. Постройте две дружественных тройки с одинаковой суммой.

Ответ. Две таких тройки встречаются довольно рано. А именно тройка

$$37380 = 2^2 \cdot 3 \cdot 5 \cdot 7 \cdot 89, \quad 41412 = 2^2 \cdot 3 \cdot 7 \cdot 17 \cdot 29, \quad 42168 = 2^3 \cdot 3 \cdot 7 \cdot 251$$

и тройка

$$38940 = 2^2 \cdot 3 \cdot 5 \cdot 11 \cdot 59, \quad 40608 = 2^5 \cdot 3^3 \cdot 47, \quad 41412 = 2^2 \cdot 3 \cdot 7 \cdot 17 \cdot 29$$

обе имеют сумму 120960.

§ 4. Общительные числа

> God not only plays dice. He also somethimes throws the dice where they cannot be seen.
>
> Stephen Hawking

В действительности, как задача о совершенных числах, так и задача о дружественных числах являются частными случаями вопроса о траекториях функции $\widehat{\sigma} : n \mapsto \sigma(n) - n$. Число n совершенно, если оно является неподвижной точкой этой функции, и является одним из дружественных чисел, если $\widehat{\sigma}^2(n) = n$. Естественно возниккает вопрос, имеет ли эта функция более длинные циклы? Элементы таких циклов называются **общительными числами**. Иными словами, число m общительное, если существует такое $s \geq 1$, что $\widehat{\sigma}^s(n) = n$.

Начинающаяся с числа n последовательность

$$n,\ \widehat{\sigma}(n),\ \widehat{\sigma}^2(n),\ \widehat{\sigma}^3(n), \ldots$$

[79] Мы не рассматриваем тройки с повторяющимися элементами.

называется **аликвотной последовательностью**. Общительное число n это такое число, для которого аликвотная последовательность возвращается в n, иными словами, является **аликвотным циклом**. В пакете

```
NumberTheory‘NumberTheoryFunctions‘
```

реализованы функции `AliquotSequence` и `AliquotCycle`, возвращающие аликвотную последовательность и ее период, хотя, конечно, эти функции за несколько секунд можно напись от руки.

4.1. Существуют ли общительные числа, не являющиеся совершенными или дружественными?

Ответ. Существуют, хотя найти их непросто, так как дополнительным параметром здесь служит длина аликвотного цикла, а коротких циклов (кроме циклов длины 4) среди маленьких чисел весьма мало! Следующий примитивный код

```
Timing[Block[{i=1},While[
          Implies[Nest[divsum,i,5]==i,divsum[i]==i],i++];i]]
```

позволяет за секунду найти цикл длины 5:

$$12496 \qquad 14288 \qquad 15472 \qquad 14536 \qquad 14264.$$

Этот цикл нашел Пуле в 1918 году. За пару минут прямым перебором можно обнаружить и цикл длины 4:

$$1264460 \qquad 1547860 \qquad 1727636 \qquad 1305184.$$

Известно много десятков циклов длины 4. Вот наименьшие элементы в остальных циклах до 10^7:

$$2115324, \qquad 2784580, \qquad 4938136, \qquad 7169104,$$

Кроме того, имеется еще пять циклов в интервале от 10^7 до 10^8, начинающеися с

$$18048976, \qquad 18656380, \qquad 28158165, \qquad 46722700, \qquad 81128632,$$

и четыре цикла в интервале от 10^8 до 10^9, начинающиеся с

$$174277820, \qquad 209524210, \qquad 330003580, \qquad 498215416.$$

Мы не будем приводить остальные известные 4-циклы. Самый далекий известный на апрель 2006 года[80] 4-цикл начинается с

$$627582618769848520570574836939315116811634898281546 12$$

[80] Eric W.Weisstein, `http://mathworld.wolfram.com/packages/Divisors.m`.

Вот еще несколько коротких циклов, состоящих из небольших чисел. Вот два 6-цикла и два 8-цикла

21548919483, 23625285957, 24825443643,
26762383557, 25958284443, 23816997477,

90632826380, 101889891700, 127527369100,
159713440756, 129092518924, 106246338676,

1095447416, 1259477224, 1156962296, 1330251784,
1221976136, 1127671864, 1245926216, 1213138984,

1276254780, 2299401444, 3071310364, 2303482780,
2629903076, 2209210588, 2223459332, 1697298124,

И, наконец, 9-цикл

805984760, 1268997640, 1803863720, 2308845400, 3059220620,
3367978564, 2525983930, 2301481286, 1611969514,

Самый длинный известный нам цикл имеет длину 28:

14316	19116	31704	47616	83328	177792	295488
629072	589786	294896	358336	418904	366556	274924
275444	243760	376736	381028	285778	152990	122410
97946	48976	45946	22976	22744	19916	17716

В различных местах мы видели упоминание о существовании циклов длины 3 для сравнительно небольших чисел (порядка нескольких триллионов). Однако, все приведенные в доступных нам книгах примеры содержат опечатки и ни один из них не привел к циклу длины 3 даже после всевозможных замен двух произвольных цифр. А дождаться результата полного перебора у нас не хватило терпения.

Каталан высказал гипотезу, что у функции σ_0 нет бесконечных траекторий: каждая траектория за конечное число шагов доходит либо до 1, либо до общительного числа.

4.2. Найдите число, которое само не является общительным, но начинающаяся с которого аликвотная последовательность доходит до общительного числа, не являющегося ни совершенным, ни дружественными.

Ответ. В качестве совсем простых примеров можно взять 9464, 12032 или 15476, сумма собственных делителей которых равна 12496, или же $\widehat{\sigma}(16312) = 14288$, $\widehat{\sigma}(29066) = 14536$. Легко строятся и более длинные траектории. Например, $\widehat{\sigma}(18922) = 9464$.

§ 5. Суммы квадратов

They should square between themselves.
William Shakespeare, *Antony and Cleopatra*

Начнем с сумм двух квадратов. Эта тема фактически восходит к “Арифметике” Диофанта. Ответ на вопрос о представимости числа как суммы двух квадратов был сформулирован Ферма и доказан Эйлером.

5.1. Рассмотрев примеры $p \leq 1000$, найдите те простые числа, которые нельзя представить как суммы двух квадратов целых (или, что в данном случае то же самое, натуральных) чисел.

Ответ. Простое число p можно представить как сумму двух квадратов в точности когда $p = 2$ или $p = 4m+1$ для некоторого m. Необходимость этого условия очевидна, достаточность проще всего доказывается с помощью целых гауссовых чисел.

5.2. Убедитесь, что примарные числа вида p^2, где $p = 4m + 3$, можно представить как сумму двух квадратов.

5.3. Найдите все числа $n \leq 1000$, которые можно представить как сумму двух квадратов *целых* чисел. Сформулируйте ответ в общем случае.

Ответ. Такое представление возможно в точности когда все показатели, с которыми в разложение n входят простые числа вида $p = 4m+3$ четны. То, что произведение двух чисел, каждое из которых представимо как сумма двух квадратов, тоже представимо как сумма двух квадратов, вытекает из из следующей формулы

$$(u^2 + v^2)(x^2 + y^2) = (ux - vy)^2 + (uy + vx)^2,$$

выражающей мультипликативность модуля комплексного числа.

5.4. Найдите все числа $n \leq 1000$, которые можно представить как сумму двух квадратов *натуральных* чисел. Постарайтесь сформулировать ответ в общем случае.

Ответ. Здесь угадать ответ уже несколько сложнее. Кроме условия из предыдущей задачи возникает еще одно условие:

- Показатели, с которыми в разложение n входят простые числа вида $p = 4m + 3$, четны.
- Показатель, с которым 2 входит в разложение n, нечетен или n делится хотя бы на одно простое вида $4m + 1$.

5.5. Найдите все числа $n \leq 1000$, которые можно представить как сумму двух квадратов *натуральных* чисел. Сформулируйте ответ в общем случае.

Ответ. Зная ответ на предыдущую задачу это несложно:

- Показатели, с которыми в разложение n входят простые числа вида $p = 4m + 3$, четны.
- Число n делится хотя бы на одно простое вида $4m + 1$.

Прямоугольный треугольник с целыми длинами сторон называется **пифагоровым**. Из только что сказанного вытекает, в частности, что n в том и только том случае является длиной гипотенузы пифагорова треугольника, когда n делится на какое-то простое вида $4m+1$.

5.6. Найдите все числа $n \leq 1000$, которые можно представить как сумму двух квадратов *взаимно простых* натуральных чисел. Сформулируйте ответ в общем случае.

Ответ. Это те числа, которые не делятся на 4 и не имеют простых делителей вида $4m+3$.

5.7. Найдите все числа $n \leq 1000$, которые нельзя представить как сумму трех квадратов *целых* чисел.

Ответ. Посмотрев на получившуюся таблицу нетрудно угадать ответ в общем случае. А именно, натуральное число n в том и только том случае не может быть представлено как сумма трех квадратов, когда n имеет вид $n = 4^k(8m+7)$ для некоторых $k, m \geq 0$. Это было впервые доказано Гауссом.

5.8. Верно ли, что любое число вида $n = 8m+3$, где $m \geq 0$, можно представить как сумму трех квадратов *нечетных* чисел?

В следующей задаче предлагается узнать, чему *не может* равняться длина диагонали прямоугольного параллелепипеда, длины сторон которого целые числа.

5.9. Найдите все числа n, для которых n^2 нельзя представить как сумму трех квадратов *натуральных* чисел.

Ответ. Проведя небольшой эксперимент, легко угадать ответ — это в точности числа вида 2^m и $2^m \cdot 5$ для $m \geq 0$. Этот ответ действительно верен[81].

Еще в "Арифметике" Диофанта высказано предположение, что *каждое* натуральное число является суммой четырех квадратов *целых* чисел $n = x_1^2 + x_2^2 + x_3^2 + x_4^2$. Иными словами, допускается, что некоторые из x_i могут равняться нулю. Важные продвижения в направлении доказательства этого предположения были получены Ферма и Эйлером, а в 1770 году Лагранж нашел полное доказательство утверждения о представимости любого натурального числа как суммы четырех квадратов. Это утверждение обычно называется **теоремой Лагранжа**.

5.10. Проверьте теорему Лагранжа для всех $n \leq 3000$.

В следующей задаче угадать ответ уже слегка затруднительно.

5.11. Найдите все числа n, которые нельзя представить как суммы четырех *различных* квадратов целых чисел.

Указание. Прежде, чем *пытаться* угадывать ответ, составьте хотя бы список таких $n \leq 1000$.

Ответ. Как обнаружил Полл[82], единственными такими числами являются

[81] A.Hurwitz, Sur la décomposition des nombres en cinq carrés. — C. R. Acad. Sci. Paris, 1884, vol.98, p.504–507.

[82] G.Pall, On sums of squares. — Amer. Math. Monthly, 1933, vol.40, p.10–18.

числа вида $4^k m$, где $k \geq 0$, $m = 1, 2, 3, 5, 6, 7, 9, 10, 11, 13, 15, 17, 18, 19, 22, 23, 25, 27, 31, 33, 34, 37, 43, 47, 55, 58, 67, 73, 82, 97, 103$.

Теорема Якоби утверждает, что в действительности количество различных представлений числа в виде суммы четырех квадратов *целых* чисел равно $8\sigma^*(n)$, где $\sigma^*(n)$ есть сумма тех делителей числа n, которые не делятся на 4. Например, число 1 имеет 8 таких представлений, так как в равенстве $1 = x_1^2 + x_2^2 + x_3^2 + x_4^2$ любая из переменных может принимать значение ± 1, а остальные три — значение 0.

5.12. Проверьте теорему Якоби для всех $n \leq 1000$.

5.13. Найдите все числа n, которые нельзя представить как суммы пяти квадратов *натуральных* чисел.

Ответ. Таких чисел совсем мало, вот они все: 1, 2, 3, 4, 6, 7, 9, 10, 12, 15, 18, 33.

§ 6. Суммы степеней: отсутствие единственности

> “It seemed to me” I continued “a rather dull number”. To which Ramanujan replied “No, Hardy! It is a very interesting number. It is the smallest number expressible as a sum of two cubes in two different ways.”
>
> Godfrey Harold Hardy, *Ramanujan*[83]

> Six or seven thousand is their utmost power.
>
> William Shakespeare, *King Richard* III

В теории чисел чрезвычайно популярны вопросы, связанные с представлением натуральных чисел как сумм степеней.

6.1. Найдите наименьшее число, которое представляется как сумма двух кубов *двумя* существенно различными способами.

Ответ. Обычно при первой попытке проще всего написать совсем простую программу и попытаться подобрать параметры вручную. Вот простая рекуррентная процедура, которая порождает по шагам все числа, представимые в виде $x^3 + y^3$, где $1 \leq x \leq y \leq n$, с учетом кратности:

```
twocubes[1]:={2};
twocubes[n_]:=twocubes[n]=
          Sort[Join[twocubes[n-1],Table[i^3+n^3,{i,1,n}]]]
```

Теперь можно вычислить этот список для какого-нибудь совсем небольшого конкретного значения n, скажем, для $n = 100$ и исключить из него повторения, например, так:

```
ReplaceList[twocubes[100],{___,x_,x_,___}->x]
```

[83] Г.Г.Харди, Двенадцать лекций о Рамануджане. — М., Ин-т Компьютерных Иссл., 2002, 335с.

При этом получится 45 чисел, которые представимы как суммы двух кубов двумя различными образами. Наименьшее из них равно, как хорошо известно из книги Харди, $1729 = 1^3 + 12^3 = 9^3 + 10^3$.

6.2. Найдите наименьшее число, которое представляется как сумма двух кубов *тремя* существенно различными способами.

Ответ. Вот это число:

$$87539319 = 167^3 + 436^3 = 228^3 + 423^3 = 255^3 + 414^3.$$

Теоретически в этом можно было бы убедиться применив такую же замену

```
ReplaceList[twocubes[500],{___,x_,x_,x_,___}->x]
```

как и в предыдущей задаче. Однако, фактически в данной ситуации гораздо быстрее пользоваться воспользоваться командой `Split`, которая сравнивает только соседние элементы. Код мог бы выглядеть, например, так:

```
Timing[Map[#[[1]]&,
          DeleteCases[Split[twocubes[500]],{x_}|{x_,x_}]]]
```

Команда `Split` здесь разбивает список на отрезки, состоящие из одинаковых элементов (по умолчанию в качестве теста в ней используется `SameQ`), после чего `DeleteCases` убирает в получившемся списке все элементы вида `{x}` или `{x,x}`. В рассматриваемом интервале найдется еще одно число, представляющееся как сумма двух кубов тремя существенно различными образами

$$119824488 = 11^3 + 493^3 = 90^3 + 492^3 = 346^3 + 428^3.$$

В следующей задаче мы предлагаем отказаться от положительности.

6.3. Найдите наименьшее число, которое представляется как сумма двух кубов *целых* чисел тремя существенно различными способами.

Ответ. Найти такое число, конечно, гораздо легче:

$$4104 = 2^3 + 16^3 = 9^3 + 15^3 = (-12)^3 + 18^3.$$

Заметим, что мы не предлагаем найти наименьшее число, которое представляется как сумма двух кубов *четырьмя* существенно различными способами. В основном тексте своей статьи на эту тему[84] Сильверман указывает

$$26059452841000 = 4170^3 + 29620^3 = 12900^3 + 28810^3 = 14577^3 + 28423^3 = 21930^3 + 24940^3$$

[84] J.H.Silverman, Taxicabs and sums of two cubes. — Amer. Math. Monthly, 1993, vol.100, p.331–340.

как наименьший *известный* пример, но в примечании при корректуре он пишет, что Дардис и Розенстил нашли *наименьший* такой пример:

$$6963472309248 = 2421^3 + 19083^3 = 5436^3 + 18948^3 = \\ 10200^3 + 18072^3 = 13322^3 + 16630^3.$$

Однако, поиск таких примеров полным перебором на домашнем компьютере удовольствие сомнительное. Впрочем, если отказаться от требования положительности слагаемых, можно найти гораздо меньший пример:

$$42549416 = 74^3 + 348^3 = 272^3 + 282^3 = (-475)^3 + 531^3 = (-2662)^3 + 2664^3.$$

6.4. Найдите наименьшее число, которое представляется как сумма двух четвертых степеней двумя существенно различными способами.

Ответ. Это $635318657 = 59^4 + 158^4 = 133^4 + 134^4$.

6.5. Найдите все числа, которые представляются как суммы двух четвертых степеней чисел ≤ 500 двумя существенно различными способами.

Ответ. Кроме полученного в предыдущем упражнении имеется еще ровно пять таких чисел, вот они все:

$$3262811042 = 7^4 + 239^4 = 157^4 + 227^4,$$
$$8657437697 = 193^4 + 292^4 = 256^4 + 257^4,$$
$$10165098512 = 118^4 + 316^4 = 266^4 + 268^4$$
$$51460811217 = 177^4 + 474^4 = 399^4 + 402^4$$
$$52204976672 = 14^4 + 478^4 = 314^4 + 454^4.$$

6.6. Найдите наименьшее число, которое представляется как сумма *трех* кубов двумя существенно различными способами.

Ответ. Это

$$251 = 1^3 + 5^3 + 5^3 = 2^3 + 3^3 + 6^3,$$

а если дополнительно требовать, чтобы все слагаемые были различны, то $1009 = 1^3 + 2^3 + 10^3 = 4^3 + 6^3 + 9^3$.

6.7. Найдите наименьшее число, которое представляется как сумма *трех* кубов *тремя* существенно различными способами.

Ответ. Это

$$5104 = 1^3 + 12^3 + 15^3 = 2^3 + 10^3 + 16^3 = 9^3 + 10^3 + 15^3.$$

6.8. Найдите наименьшее число, которое представляется как сумма *трех* кубов *четырьмя* существенно различными способами.

Ответ. Это

$$13896 = 1^3 + 12^3 + 23^3 = 2^3 + 4^3 + 24^3 = 4^3 + 18^3 + 20^3 = 9^3 + 10^3 + 23^3.$$

6.9. Найдите наименьшее число, которое представляется как сумма *трех* кубов *пятью* существенно различными способами.

Ответ. Это

$$161568 = 2^3 + 16^3 + 54^3 = 9^3 + 15^3 + 54^3 = 17^3 + 39^3 + 46^3 = \\ 18^3 + 19^3 + 53^3 = 26^3 + 36^3 + 46^3.$$

6.10. Найдите первые 10 чисел чисел, которые представляются как сумма *трех* четвертых степеней *двумя* существенно различными способами.

Ответ. Таких чисел очень много, самое маленькое из них совсем крошечное, $6578 = 1^4 + 2^4 + 9^4 = 3^4 + 7^4 + 8^4$. Вот несколько следующих по величине:

$$16562 = 1^4 + 9^4 + 10^4 = 5^4 + 6^4 + 11^4,$$
$$28593 = 2^4 + 2^4 + 13^4 = 6^4 + 9^4 + 12^4,$$
$$35378 = 1^4 + 11^4 + 12^4 = 4^4 + 9^4 + 13^4,$$
$$43218 = 2^4 + 11^4 + 13^4 = 7^4 + 7^4 + 14^4,$$
$$54977 = 4^4 + 8^4 + 15^4 = 9^4 + 10^4 + 14^4,$$
$$94178 = 8^4 + 9^4 + 17^4 = 3^4 + 13^4 + 16^4.$$

6.11. Существуют ли числа, которые представляются в виде суммы трех четвертых степеней *тремя* существенно различными способами?

Ответ. Сколько угодно, самое маленькое из них

$$811538 = 4^4 + 23^4 + 27^4 = 7^4 + 21^4 + 28^4 = 12^4 + 17^4 + 29^4.$$

Вот несколько следующих таких чисел: 1733522, 2798978, 3750578, 4614722, 5978882, 6573938, 7303842. Решить аналогичную задачу для сумм *двух* четвертых степеней в течение нескольких минут нам не удалось, во всяком случае никаких *маленьких* решений у нее нет.

Не удалось нам за несколько минут и найти такие числа, которые двумя различными способами представляются как сумма пятых степеней.

Еще одна вариация на эту тему это поиск чисел, для которых имеет место совпадение сумм различных степеней.

6.12. Найдите наименьшее натуральное решение системы

$$x_1^2 + x_2^2 + x_3^2 + x_4^2 = y_1^2 + y_2^2 + y_3^2 + y_4^2,$$
$$x_1^3 + x_2^3 + x_3^3 + x_4^3 = y_1^3 + y_2^3 + y_3^3 + y_4^3.$$

Ответ. Вот это решение

$$234 = 2^2 + 3^2 + 10^2 + 11^2 = 1^2 + 5^2 + 8^2 + 12^2$$
$$2366 = 2^3 + 3^3 + 10^3 + 11^3 = 1^3 + 5^3 + 8^3 + 12^3$$

Интересно, что, кроме того, $2 + 3 + 10 + 11 = 1 + 5 + 8 + 12$.

6.13. Найдите наименьшее натуральное решение системы

$$x_1^2 + x_2^2 + x_3^2 + x_4^2 = y_1^2 + y_2^2 + y_3^2 + y_4^2,$$
$$x_1^4 + x_2^4 + x_3^4 + x_4^4 = y_1^4 + y_2^4 + y_3^4 + y_4^4.$$

6.14. Найдите наименьшее натуральное решение системы

$$x_1^3 + x_2^3 + x_3^3 + x_4^3 = y_1^3 + y_2^3 + y_3^3 + y_4^3,$$
$$x_1^5 + x_2^5 + x_3^5 + x_4^5 = y_1^5 + y_2^5 + y_3^5 + y_4^5.$$

§ 7. Суммы степеней: проблема Варинга

> Young Octavius and Mark Antony come down upon us with a mighty power.
>
> William Shakespeare, *Julius Caesar*

В том же 1770 году английский математик Эдвард Варинг высказал гитотезу, обобщающую теорему Лагранжа о сумме квадратов, а именно, он предположил, что *любое* натуральное число представимо в виде суммы 9 кубов неотрицательных целых чисел, 19 четвертых степеней и т.д. Впрочем, сам Варинг говорил, что любое натуральное число представляется в виде суммы *не более*, чем 9 кубов натуральных чисел, *не более*, чем 19 четвертых степеней и т.д. Вопрос о справедливости этого утверждения известен как **проблема Варинга**. В 1909 году Гильберт получил принципиальное решение проблемы Варинга, придумав совсем простое доказательство того, что для любого m найдется такое $g(m)$, что *любое* натуральное число n представимо в виде суммы $g(m)$ неотрицательных m-х степеней. К сожалению, доказательство Гильберта дает явно завышенную оценку для $g(m)$.

Впрочем, профессиональных числовиков обычно мало интересует вопрос о представимости *всех* натуральных чисел в таком виде. С их точки зрения гораздо интереснее найти такое наименьшее $G(m)$, что любое *достаточно большое* натуральное число n представимо в виде суммы $G(m)$ неотрицательных m-х степеней. Проведем несколько простых экспериментов. Для этого, прежде всего, определим функцию, возвращающую списки с элементами вида $(x_1, \ldots, x_s, n)$, где $x_1^m + \ldots + x_s^m = n$.

7.1. Напишите программу, вычисляющую множество всех $(x_1, \ldots, x_s, n)$, где

$$x_1^m + \ldots + x_s^m = n, \qquad 0 \le x_i \le l,$$

Ответ. Вот простая рекуррентная программа, которая на бытовом компьютере будет в течение секунд работать до сумм порядка нескольких тысяч:

```
powers[l_,1,m_]:=Table[{i,i^m},{i,0,l}];
powers[1,s_,m_]:=Table[Join[Table[0,{s-i}],
                        Table[1,{i}],{i}],{i,0,s}];
powers[l_,s_,m_]:=powers[l,s,m]=
            Sort[Join[powers[l-1,s,m],Table[Map[
            Join[Most[#],{l,Last[#]+l^m}]&,powers[l,s-1,m]]]]]
```

Теперь множество всех чисел, представимых в виде суммы s штук m-х степеней из данного интервала можно найти, например, как

`Last[Transpose[powers[l,s,m]]]`.

Впрочем, если нас интересует только само это множество, а не решения уравнения $x_1^m + \ldots + x_s^m = n$, это можно сделать гораздо быстрее.

7.2. Напишите программу, возвращающую множество сумм

$$x_1^m + \ldots + x_s^m, \qquad 0 \le x_i \le l.$$

В 1909 году Виферих доказал, что *любое* натуральное число можно представить как сумму *девяти* кубов.

7.3. Проверьте теорему Вифериха для всех $n \le 3000$.

Кстати, это именно то место, до которого Варинг вручную проверил это утверждение в 1770 году!

7.4. Найдите числа ≤ 10000, не представимые в виде суммы восьми кубов.

Ответ. Таких чисел ровно два, а именно 23 и 239. Для первого из них это очевидно и без компьютера, так как единственными кубами, меньшими, чем 23, являются 0,1 и 8. В то же время как сумма девяти кубов 239 представляется, притом даже двумя способами:

$$1^3+1^3+1^3+3^3+3^3+3^3+3^3+4^3+4^3 = 239 = 1^3+2^3+2^3+2^3+2^3+3^3+3^3+3^3+5^3.$$

Если нас интересует представление не всех, а лишь достаточно больших натуральных чисел, т.е. вычисление $G(3)$, а не $g(3)$, то ситуация меняется.

7.5. Докажите, что любое число $240 \le n \le 454$ представимо в виде суммы восьми кубов.

7.6. Докажите, что любое число $455 \le n \le 8042$ представимо в виде суммы семи кубов.

7.7. Докажите, что любое число $n \ge 8043$, до которого Ваш компьютер в состоянии досчитать, представимо в виде суммы шести кубов.

В действительности в том же 1909 году Эдмунд Ландау доказал, что $G(3) \le 8$, иными словами, все натуральные числа *начиная с некоторого места* представимы как суммы восьми кубов. В 1943 году Юрий Владимирович Линник доказал знаменитую теорему о семи кубах, утверждающую,

что $G(3) \leq 7$. В 1851 году Якоби высказывал даже предположение, что $G(3) \leq 5$, но до сих пор это предположение доказано только в том смысле, что "почти все" (в некотором точном смысле) натуральные числа представимы в таком виде — и даже как суммы четырех кубов!

7.8. Число $16^n \cdot 31$ нельзя представить в виде суммы 15 четвертых степеней.

§ 8. Суммы степеней: гипотезы Ферма и Эйлера

> But, true it is, from France there comes a power, have secret feet in some of our best ports and are at point to show their open banner.
>
> William Shakespeare, *King Lear*

Еще одна классическая задача о суммах степеней, которая вызывала огромный интерес как профессионалов, так и любителей и породила громадную литературу, это проблема Ферма. А именно, Ферма утверждал, что при $m \geq 3$ уравнение

$$x^m + y^m = z^m$$

не имеет решений в натуральных числах. В течение 300 лет эта гипотеза оставалась, вероятно, самой знаменитой нерешенной проблемой математики. На полях своей копии *Арифметики* Диофанта Ферма написал, что нашел *поистине замечательное* доказательство этого факта. Однако, за исключением случая $n = 4$, где он действительно дал безукоризненное доказательство основанное на методе математической индукции ("бесконечного спуска"), в его бумагах не было обнаружено никаких следов этого *замечательного доказательства*. Можно с полной уверенностью утверждать, что "доказательство" Ферма было ошибочным. Более того, профессиональные математики *знают*, в чем именно состояла ошибка Ферма: он считал, что кольца целых круговых полей являются кольцами главных идеалов. На протяжении двух веков это неявное предположение ускользало от внимания даже лучших математиков. Необходимость доказательства этого факта не заметил Эйлер в своем (неполном) доказательстве теоремы Ферма при $n = 3$. Спустя почти два века это же сомнительное предположение было использовано Ламе и Коши в еще одном (ошибочном) доказательстве теоремы Ферма. Вскрытие этой ошибки Дирихле и Куммером привело к созданию алгебраической теории чисел и теории колец. На протяжении примерно 100 лет, после проникновения отрывочных сведений о проблеме Ферма в широкие неумытые массы, эти массы терроризировали математические факультеты и институты своими безграмотными сочинениями на эту тему. К счастью, в 1994 году проблема Ферма была наконец положительно решена Эндрю Уайлсом[85]. На русском языке драматическую историю проблемы Ферма можно найти в книгах Эдвардса [Ed], Рибенбойма [Ri2] и Сингха [Si].

[85] A.Wiles, Modular elliptic curves and Fermat's last theorem. — Ann. Math., 1995, vol.141, p.443–551.

Здесь мы обсудим обобщение гипотезы Ферма, предложенное Эйлером. А именно, Эйлер утверждал, что при $m \geq 4$ уравнение

$$x^m + y^m + z^m = u^m$$

не имеет решений в натуральных числах, при $m \geq 5$ уравнение

$$x^m + y^m + u^m + v^m = z^m$$

не имеет решений в натуральных числах, и так далее.

8.1. Напишите программу, вычисляющую суммы

$$x_1^m + \ldots + x_s^m, \qquad 0 \leq x_i \leq n,$$

s штук m-х степеней натуральных чисел $\leq n$.

Ответ. Фактически ничем, кроме начальных условий, не отличается от соответствующей рекуррентной процедуры, описанной в предыдущем параграфе:

```
nowers[n_,1,m_]:=Table[{i,i^m},{i,1,n}];
nowers[1,s_,m_]:={Join[Table[1,{s}],{s}]};
nowers[n_,s_,m_]:=nowers[n,s,m]=
            Sort[Join[nowers[n-1,s,m],Table[Map[
            Join[Most[#],{n,Last[#]+n^m}]&,nowers[n,s-1,l]]]]]
```

В случае, когда общее число слагаемых больше степени, это уравнение как правило имеет решения, часто совсем небольшие.

8.2. Найдите все решения уравнения $x_1^3 + x_2^3 + x_3^3 = x_4^3 + x_5^3$ в натуральных числах ≤ 100.

Ответ. В рассматриваемой области имеется 863 значения, представимых как в виде суммы трех, так и в виде суммы двух кубов. Имеется несколько совсем крошечных решений. Вот все такие решения с суммой до 5000:

$1^3 + 7^3 = 344 = 4^3 + 4^3 + 6^3$ $\quad 7^3 + 8^3 = 855 = 1^3 + 5^3 + 9^3$
$7^3 + 9^3 = 1072 = 2^3 + 4^3 + 10^3$ $\quad 7^3 + 11^3 = 1674 = 6^3 + 9^3 + 9^3$
$2^3 + 14^3 = 2752 = 8^3 + 8^3 + 12^3$ $\quad 3^3 + 15^3 = 3402 = 7^3 + 11^3 + 12^3$
$5^3 + 15^3 = 3500 = 3^3 + 9^3 + 14^3$ $\quad 10^3 + 14^3 = 3744 = 6^3 + 11^3 + 13^3$
$7^3 + 16^3 = 4439 = 4^3 + 10^3 + 15^3$ $\quad 13^3 + 14^3 = 4941 = 1^3 + 3^3 + 17^3$

8.3. Найдите все решения уравнения $x_1^4 + x_2^4 + x_3^4 = x_4^4 + x_5^4$ в натуральных числах ≤ 100.

Ответ. В рассматриваемой области имеется 52 значения, представимых как в виде суммы трех, так и в виде суммы двух четвертых степеней. Вот совсем крошечное решение:

$$3^4 + 5^4 + 8^4 = 4802 = 7^4 + 7^4.$$

В случае, когда $s = m$ найти решения — скажем, найти контрпример к гипотезе Эйлера для сумм четвертых степеней — *значительно* сложнее. Конечно, уже очень давно было известно, что четвертую степень можно представить как сумму четырех, пяти или шести четвертых степеней. Вот совсем маленькие примеры:

$$353^4 = 30^4 + 120^4 + 272^4 + 315^4,$$
$$15^4 = 4^4 + 6^4 + 8^4 + 9^4 + 14^4,$$
$$91^4 = 14^4 + 24^4 + 34^4 + 49^4 + 58^4 + 84^4.$$

Обратите внимание, что во втором и третьем примерах все основания степени ≤ 100

8.4. Найдите все решения уравнения

$$x_1^4 + x_2^4 + x_2^4 + x_4^4 = x_5^4$$

в натуральных числах ≤ 200.

8.5. Найдите все решения уравнения

$$x_1^4 + x_2^4 + x_2^4 + x_4^4 + x_5^4 = x_6^4$$

в натуральных числах ≤ 100.

8.6. Найдите все решения уравнения

$$x_1^4 + x_2^4 + x_2^4 + x_4^4 + x_5^4 + x_6^4 = x_7^4$$

в натуральных числах ≤ 100.

А втот опровергнуть *подобными методами* гипотезу Эйлера для $n = 4$ вряд ли удастся. Поиск найденного в 1988 году Ноамом Элкисом[86] решения

$$2682440^4 + 15365639^4 + 18796760^4 =$$
$$= 18063007729216928108884849904 1 = 20615673^4$$

уравнения $x_1^4 + x_2^4 + x_3^4 = x_4^4$ на бытовом компьютере методом прямого перебора просто невозможенен. После этого Роджер Фрай провел систематический поиск решений этого уравнения на суперкомпьютере и выяснил, что *единственным* его решением в числах $\leq 10^6$ является

$$95800^4 + 217519^4 + 414560^4 = 3185874984000794592032 1 = 422481^4.$$

Поиграем теперь немного с суммами пятых степеней.

8.7. Проверьте, что уравнение $x_1^5 + x_2^5 + x_2^5 + x_4^5 = x_5^5$ не имеет решений в натуральных числах ≤ 50.

В действительности это уравнение имеет совсем небольшое решение

$$27^5 + 84^5 + 110^5 + 133^5 = 61917364224 = 144^5,$$

но так как прямой поиск этого решения требует сравнения сотен миллиардов чисел порядка сотен миллиардов, время, необходимое для поиска этого решения на бытовом компьютере теми методами, которые мы обсуждали в этом параграфе, измеряется уже не секундами, а десятками минут.

[86] N.D.Elkies, On $A^4 + B^4 + C^4 = D^4$. — Math. Comput., 1988, vol.51, p.825–835.

§ 9. Гипотеза Гольдбаха

We show in a certain precise sense that the Goldbach conjecture is true with probablity larger than 0.99999 and that its complete truth could be determined with a budget of $10 billion[87].

Doron Zeilberger

В адресованном Эйлеру письме 1742 года Гольдбах высказал следующее предположение, которое стало знаменитым как **гипотеза Гольдбаха**: любое целое $n > 5$ есть сумма трех простых чисел.

На это Эйлер тут же ответил, что в действительности это предположение эквивалентно следующему утверждению, которое сегодня называется **четной гипотезой Гольдбаха**: любое четное целое $n > 2$ есть сумма двух простых чисел.

Однако, так как четная гипотеза Гольдбаха оказалась чрезвычайно трудной, в дальнейшем стали рассматривать такое ее ослабление. **Нечетная гипотеза Гольдбаха**: любое нечетное целое $n > 5$ есть сумма трех простых чисел.

Нечетная гипотеза Гольдбаха доказана в асимптотическом смысле. А именно, **теорема Харди—Литтлвуда—Виноградова** утверждает, что любое натуральное число *начиная с некоторого места* является суммой трех простых. Иными словами, существует такое натуральное число n_0, что каждое $n > n_0$ является суммой трех простых чисел.

Четная гипотеза Гольдбаха не доказана даже в асимптотическом смысле, Однако, **теорема ван дер Корпута—Чудакова—Эстермана** утверждает, что *почти все* четные n являются суммами двух простых чисел. Обратите внимание, что в то время как в предыдущей теореме **почти все** означает **все, кроме конечного числа**, в этой теореме **почти все** используется в гораздо более слабом смысле — **все, кроме множества плотности** 0.

Наилучшим приближением к четной гипотезе Гольдбаха в смысле **все, кроме конечного числа**, до сих пор остается следующая **теорема Чена**[88]: существует такое натуральное число n_0, что каждое четное $n > n_0$ есть сумма простого числа и числа, являющегося произведением не более, чем двух простых. С другой стороны, самый точный из результатов, полученных на пути Шнирельмана[89], утверждает, что *каждое* число есть сумма не более 6 простых — всего в два раза хуже, чем гипотеза Гольдбаха!

Однако, для обеих гипотез имеется громадный объем свидетельских показаний (body of evidence).

---

[87] D.Zeilberger, Theorems for a price: tomorrow's semirigorous mathematical culture. — Notices Amer. Math. Soc., 1993, vol. 40, N.8, p.978–981; reprinted in Math. Intelligencer, 1994, vol.16, N.4, p.11–14.

[88] J.R.Chen, On the representation of a large even integer as a sum of a prime and a product of at most two primes. I, II. — Sci. Sinica., 1973, vol.16, p.157–176; 1978, vol.21, p.421–430.

[89] O.Ramaré, On Shnirel'man's constant. — Ann. Scuola Norm. Super. Pisa, 1995, vol.22, p.645–706.

• В 1998 году Саутер[90] объявил, что нечетная гипотеза Гольдбаха проверена для всех $n < 10^{20}$.

• В 2001 году Рихштайн[91] объявил, что четная гипотеза Гольдбаха проверена для всех $n < 4 \cdot 10^{14}$.

Конечно, повторить результаты Саутера и Рихштайна на бытовом компьютере Вам не удастся, тем не менее, просто полным перебором Вы сможете легко сделать следующее.

9.1. Проверьте четную гипотезу Гольдбаха для всех $n \le 10^5$.

9.2. Проверьте нечетную гипотезу Гольдбаха для всех $n \le 10^4$.

Проблема Гольдбаха фигурирует в качестве одной из `Millenium Problems` и за ее полное решение предлагается 1000000 долларов. Впрочем, как отмечается в эпиграфе к настоящему параграфу, этот бюджет занижен по крайней мере в 10000 раз!

Следующая симпатичная задача — подразумевавшееся жюри решение[92] опирается на справедливость гипотезы Гольдбаха! — предложена С.Г.Волченковым.

9.3. Разбейте натуральные числа от 1 до n на минимальное количество групп, сумма каждой из которых является простым числом.

§ 10. Другие аддитивные задачи

Царь: Вызывает антирес
И такой ишо разрез:
Как у вас там ходют бабы —
В панталонах али без?
Посол: Йес!

Леонид Филатов, *Про Федота-стрельца*

Знаменитая **гипотеза Харди—Литтлвуда** утверждает, что любое *достаточно большое* натуральное число n, *не являющееся квадратом*, представимо в виде m^2+p для некоторого натурального m и некоторого простого p. Более простое предположение, что каждое достаточно большое натуральное число представимо в виде $l^2 + m^2 + p$ доказал в 1959 году Линник.

10.1. Найдите те числа ≤ 100, которые не представляются в виде $m^2 + p$. Повторите этот эксперимент для ≤ 1000. Появляются ли при этом новые числа не являющиеся квадратами?

10.2. Найдите те числа ≤ 100, которые не представляются в виде $l^2 + m^2 + p$. Повторите этот эксперимент для ≤ 1000, а потом для $n \le 10000$. Появляются ли при этом новые числа?

---

[90] Y.Saouter, Checking the odd Goldbach conjecture up to 10^{20}. — Math. Comput., 1998, vol.67, p.863–866.

[91] J.Richstein, Verifying the Goldbach conjecture up to $4 \cdot 10^{14}$. — Math. Comput., 2001, vol.70, p.1745–1749.

[92] В.Н.Пинаев, Четвертьфинальные соревнования студенческого командного первенства мира по программированию. Центральный регион России. — РГАТА, Рыбинск, 1999, с.1–30.

Ответ. Нет, 1, 2, 3, 6, 14 являются единственными такими числами.

В 1849 году де Польньяк сформулировал весьма сомнительное утверждение, что *любое* нечетное число $n > 1$ представимо в виде $2^k + p$, для некоторой степени 2 и числа p, которое либо просто, либо равно 1. Романов[93] доказал, что числа представимые в таком виде имеют положительную плотность и уточнил гипотезу де Польньяка следующим образом: верно ли, что *начиная с некоторого места* каждое нечетное натуральное число представляется в виде $2^k + p$ для некоторой степени 2 и некоторого простого p.

10.3. Каких чисел среди нечетных $n \leq 10^6$ больше: тех, которые представляются в виде $2^k + p$ или тех, которые в таком виде не представляются?

Как заметил в 1950 году Эрдеш, существует бесконечно много нечетных чисел, которые не представляются в таком виде. А в 1960 году Крокер построил следующий простой контрпример к гипотезе Романова.

10.4. Убедитесь, что ни одно из чисел вида $2^{2^n} - 5$, при $n \geq 3$, не представимо в виде $2^k + p$.

Любые два различных подмножества множества $\{2^i \mid 0 \leq i \leq n\}$ имеют различные суммы. Эрдеш[94] поставил проблему нахождения максимального числа m для которого найдутся такие

$$0 < x_1 < x_2 < \ldots < x_m \leq 2^n,$$

что любые два различных подмножества множества $\{x_1, \ldots, x_m\}$ имеют различные суммы. Как мы только что видели, $m \geq n + 1$. В конце 1960-х годов Эрдеш оценивал задачу нахождения точного значения m в 250 долларов.

10.5. Верно ли, что для достаточно больших n можно утверждать, что $m \geq n + 2$?

> "I shan't call it the end, till we've cleared up the mess," said Sam gloomily. "And that'll take a lot of time and work."
>
> J.R.R Tolkien, *The Lord of the Rings*

[93] N.P.Romanoff, Über einige Sätze der additiven Zahlentheorie. — Math. Ann., 1934, Bd.109, S.668–678.

[94] P.Erdös, Problems and results in additive number theory. — Coll. Théorie des Nombres, Bruxelles, 1956, p.127–137.

Простые p_n, $1 \le n \le 400$

2	3	5	7	11	13	17	19	23	29
31	37	41	43	47	53	59	61	67	71
73	79	83	89	97	101	103	107	109	113
127	131	137	139	149	151	157	163	167	173
179	181	191	193	197	199	211	223	227	229
233	239	241	251	257	263	269	271	277	281
283	293	307	311	313	317	331	337	347	349
353	359	367	373	379	383	389	397	401	409
419	421	431	433	439	443	449	457	461	463
467	479	487	491	499	503	509	521	523	541
547	557	563	569	571	577	587	593	599	601
607	613	617	619	631	641	643	647	653	659
661	673	677	683	691	701	709	719	727	733
739	743	751	757	761	769	773	787	797	809
811	821	823	827	829	839	853	857	859	863
877	881	883	887	907	911	919	929	937	941
947	953	967	971	977	983	991	997	1009	1013
1019	1021	1031	1033	1039	1049	1051	1061	1063	1069
1087	1091	1093	1097	1103	1109	1117	1123	1129	1151
1153	1163	1171	1181	1187	1193	1201	1213	1217	1223
1229	1231	1237	1249	1259	1277	1279	1283	1289	1291
1297	1301	1303	1307	1319	1321	1327	1361	1367	1373
1381	1399	1409	1423	1427	1429	1433	1439	1447	1451
1453	1459	1471	1481	1483	1487	1489	1493	1499	1511
1523	1531	1543	1549	1553	1559	1567	1571	1579	1583
1597	1601	1607	1609	1613	1619	1621	1627	1637	1657
1663	1667	1669	1693	1697	1699	1709	1721	1723	1733
1741	1747	1753	1759	1777	1783	1787	1789	1801	1811
1823	1831	1847	1861	1867	1871	1873	1877	1879	1889
1901	1907	1913	1931	1933	1949	1951	1973	1979	1987
1993	1997	1999	2003	2011	2017	2027	2029	2039	2053
2063	2069	2081	2083	2087	2089	2099	2111	2113	2129
2131	2137	2141	2143	2153	2161	2179	2203	2207	2213
2221	2237	2239	2243	2251	2267	2269	2273	2281	2287
2293	2297	2309	2311	2333	2339	2341	2347	2351	2357
2371	2377	2381	2383	2389	2393	2399	2411	2417	2423
2437	2441	2447	2459	2467	2473	2477	2503	2521	2531
2539	2543	2549	2551	2557	2579	2591	2593	2609	2617
2621	2633	2647	2657	2659	2663	2671	2677	2683	2687
2689	2693	2699	2707	2711	2713	2719	2729	2731	2741

Простые p_n, $401 \le n \le 800$

2749	2753	2767	2777	2789	2791	2797	2801	2803	2819
2833	2837	2843	2851	2857	2861	2879	2887	2897	2903
2909	2917	2927	2939	2953	2957	2963	2969	2971	2999
3001	3011	3019	3023	3037	3041	3049	3061	3067	3079
3083	3089	3109	3119	3121	3137	3163	3167	3169	3181
3187	3191	3203	3209	3217	3221	3229	3251	3253	3257
3259	3271	3299	3301	3307	3313	3319	3323	3329	3331
3343	3347	3359	3361	3371	3373	3389	3391	3407	3413
3433	3449	3457	3461	3463	3467	3469	3491	3499	3511
3517	3527	3529	3533	3539	3541	3547	3557	3559	3571
3581	3583	3593	3607	3613	3617	3623	3631	3637	3643
3659	3671	3673	3677	3691	3697	3701	3709	3719	3727
3733	3739	3761	3767	3769	3779	3793	3797	3803	3821
3823	3833	3847	3851	3853	3863	3877	3881	3889	3907
3911	3917	3919	3923	3929	3931	3943	3947	3967	3989
4001	4003	4007	4013	4019	4021	4027	4049	4051	4057
4073	4079	4091	4093	4099	4111	4127	4129	4133	4139
4153	4157	4159	4177	4201	4211	4217	4219	4229	4231
4241	4243	4253	4259	4261	4271	4273	4283	4289	4297
4327	4337	4339	4349	4357	4363	4373	4391	4397	4409
4421	4423	4441	4447	4451	4457	4463	4481	4483	4493
4507	4513	4517	4519	4523	4547	4549	4561	4567	4583
4591	4597	4603	4621	4637	4639	4643	4649	4651	4657
4663	4673	4679	4691	4703	4721	4723	4729	4733	4751
4759	4783	4787	4789	4793	4799	4801	4813	4817	4831
4861	4871	4877	4889	4903	4909	4919	4931	4933	4937
4943	4951	4957	4967	4969	4973	4987	4993	4999	5003
5009	5011	5021	5023	5039	5051	5059	5077	5081	5087
5099	5101	5107	5113	5119	5147	5153	5167	5171	5179
5189	5197	5209	5227	5231	5233	5237	5261	5273	5279
5281	5297	5303	5309	5323	5333	5347	5351	5381	5387
5393	5399	5407	5413	5417	5419	5431	5437	5441	5443
5449	5471	5477	5479	5483	5501	5503	5507	5519	5521
5527	5531	5557	5563	5569	5573	5581	5591	5623	5639
5641	5647	5651	5653	5657	5659	5669	5683	5689	5693
5701	5711	5717	5737	5741	5743	5749	5779	5783	5791
5801	5807	5813	5821	5827	5839	5843	5849	5851	5857
5861	5867	5869	5879	5881	5897	5903	5923	5927	5939
5953	5981	5987	6007	6011	6029	6037	6043	6047	6053
6067	6073	6079	6089	6091	6101	6113	6121	6131	6133

Простые p_n, $801 \le n \le 1200$

6143	6151	6163	6173	6197	6199	6203	6211	6217	6221
6229	6247	6257	6263	6269	6271	6277	6287	6299	6301
6311	6317	6323	6329	6337	6343	6353	6359	6361	6367
6373	6379	6389	6397	6421	6427	6449	6451	6469	6473
6481	6491	6521	6529	6547	6551	6553	6563	6569	6571
6577	6581	6599	6607	6619	6637	6653	6659	6661	6673
6679	6689	6691	6701	6703	6709	6719	6733	6737	6761
6763	6779	6781	6791	6793	6803	6823	6827	6829	6833
6841	6857	6863	6869	6871	6883	6899	6907	6911	6917
6947	6949	6959	6961	6967	6971	6977	6983	6991	6997
7001	7013	7019	7027	7039	7043	7057	7069	7079	7103
7109	7121	7127	7129	7151	7159	7177	7187	7193	7207
7211	7213	7219	7229	7237	7243	7247	7253	7283	7297
7307	7309	7321	7331	7333	7349	7351	7369	7393	7411
7417	7433	7451	7457	7459	7477	7481	7487	7489	7499
7507	7517	7523	7529	7537	7541	7547	7549	7559	7561
7573	7577	7583	7589	7591	7603	7607	7621	7639	7643
7649	7669	7673	7681	7687	7691	7699	7703	7717	7723
7727	7741	7753	7757	7759	7789	7793	7817	7823	7829
7841	7853	7867	7873	7877	7879	7883	7901	7907	7919
7927	7933	7937	7949	7951	7963	7993	8009	8011	8017
8039	8053	8059	8069	8081	8087	8089	8093	8101	8111
8117	8123	8147	8161	8167	8171	8179	8191	8209	8219
8221	8231	8233	8237	8243	8263	8269	8273	8287	8291
8293	8297	8311	8317	8329	8353	8363	8369	8377	8387
8389	8419	8423	8429	8431	8443	8447	8461	8467	8501
8513	8521	8527	8537	8539	8543	8563	8573	8581	8597
8599	8609	8623	8627	8629	8641	8647	8663	8669	8677
8681	8689	8693	8699	8707	8713	8719	8731	8737	8741
8747	8753	8761	8779	8783	8803	8807	8819	8821	8831
8837	8839	8849	8861	8863	8867	8887	8893	8923	8929
8933	8941	8951	8963	8969	8971	8999	9001	9007	9011
9013	9029	9041	9043	9049	9059	9067	9091	9103	9109
9127	9133	9137	9151	9157	9161	9173	9181	9187	9199
9203	9209	9221	9227	9239	9241	9257	9277	9281	9283
9293	9311	9319	9323	9337	9341	9343	9349	9371	9377
9391	9397	9403	9413	9419	9421	9431	9433	9437	9439
9461	9463	9467	9473	9479	9491	9497	9511	9521	9533
9539	9547	9551	9587	9601	9613	9619	9623	9629	9631
9643	9649	9661	9677	9679	9689	9697	9719	9721	9733

Простые p_n, $1201 \le n \le 1600$

9739	9743	9749	9767	9769	9781	9787	9791	9803	9811
9817	9829	9833	9839	9851	9857	9859	9871	9883	9887
9901	9907	9923	9929	9931	9941	9949	9967	9973	10007
10009	10037	10039	10061	10067	10069	10079	10091	10093	10099
10103	10111	10133	10139	10141	10151	10159	10163	10169	10177
10181	10193	10211	10223	10243	10247	10253	10259	10267	10271
10273	10289	10301	10303	10313	10321	10331	10333	10337	10343
10357	10369	10391	10399	10427	10429	10433	10453	10457	10459
10463	10477	10487	10499	10501	10513	10529	10531	10559	10567
10589	10597	10601	10607	10613	10627	10631	10639	10651	10657
10663	10667	10687	10691	10709	10711	10723	10729	10733	10739
10753	10771	10781	10789	10799	10831	10837	10847	10853	10859
10861	10867	10883	10889	10891	10903	10909	10937	10939	10949
10957	10973	10979	10987	10993	11003	11027	11047	11057	11059
11069	11071	11083	11087	11093	11113	11117	11119	11131	11149
11159	11161	11171	11173	11177	11197	11213	11239	11243	11251
11257	11261	11273	11279	11287	11299	11311	11317	11321	11329
11351	11353	11369	11383	11393	11399	11411	11423	11437	11443
11447	11467	11471	11483	11489	11491	11497	11503	11519	11527
11549	11551	11579	11587	11593	11597	11617	11621	11633	11657
11677	11681	11689	11699	11701	11717	11719	11731	11743	11777
11779	11783	11789	11801	11807	11813	11821	11827	11831	11833
11839	11863	11867	11887	11897	11903	11909	11923	11927	11933
11939	11941	11953	11959	11969	11971	11981	11987	12007	12011
12037	12041	12043	12049	12071	12073	12097	12101	12107	12109
12113	12119	12143	12149	12157	12161	12163	12197	12203	12211
12227	12239	12241	12251	12253	12263	12269	12277	12281	12289
12301	12323	12329	12343	12347	12373	12377	12379	12391	12401
12409	12413	12421	12433	12437	12451	12457	12473	12479	12487
12491	12497	12503	12511	12517	12527	12539	12541	12547	12553
12569	12577	12583	12589	12601	12611	12613	12619	12637	12641
12647	12653	12659	12671	12689	12697	12703	12713	12721	12739
12743	12757	12763	12781	12791	12799	12809	12821	12823	12829
12841	12853	12889	12893	12899	12907	12911	12917	12919	12923
12941	12953	12959	12967	12973	12979	12983	13001	13003	13007
13009	13033	13037	13043	13049	13063	13093	13099	13103	13109
13121	13127	13147	13151	13159	13163	13171	13177	13183	13187
13217	13219	13229	13241	13249	13259	13267	13291	13297	13309
13313	13327	13331	13337	13339	13367	13381	13397	13399	13411
13417	13421	13441	13451	13457	13463	13469	13477	13487	13499

Простые p_n, $1601 \le n \le 2000$

13513	13523	13537	13553	13567	13577	13591	13597	13613	13619
13627	13633	13649	13669	13679	13681	13687	13691	13693	13697
13709	13711	13721	13723	13729	13751	13757	13759	13763	13781
13789	13799	13807	13829	13831	13841	13859	13873	13877	13879
13883	13901	13903	13907	13913	13921	13931	13933	13963	13967
13997	13999	14009	14011	14029	14033	14051	14057	14071	14081
14083	14087	14107	14143	14149	14153	14159	14173	14177	14197
14207	14221	14243	14249	14251	14281	14293	14303	14321	14323
14327	14341	14347	14369	14387	14389	14401	14407	14411	14419
14423	14431	14437	14447	14449	14461	14479	14489	14503	14519
14533	14537	14543	14549	14551	14557	14561	14563	14591	14593
14621	14627	14629	14633	14639	14653	14657	14669	14683	14699
14713	14717	14723	14731	14737	14741	14747	14753	14759	14767
14771	14779	14783	14797	14813	14821	14827	14831	14843	14851
14867	14869	14879	14887	14891	14897	14923	14929	14939	14947
14951	14957	14969	14983	15013	15017	15031	15053	15061	15073
15077	15083	15091	15101	15107	15121	15131	15137	15139	15149
15161	15173	15187	15193	15199	15217	15227	15233	15241	15259
15263	15269	15271	15277	15287	15289	15299	15307	15313	15319
15329	15331	15349	15359	15361	15373	15377	15383	15391	15401
15413	15427	15439	15443	15451	15461	15467	15473	15493	15497
15511	15527	15541	15551	15559	15569	15581	15583	15601	15607
15619	15629	15641	15643	15647	15649	15661	15667	15671	15679
15683	15727	15731	15733	15737	15739	15749	15761	15767	15773
15787	15791	15797	15803	15809	15817	15823	15859	15877	15881
15887	15889	15901	15907	15913	15919	15923	15937	15959	15971
15973	15991	16001	16007	16033	16057	16061	16063	16067	16069
16073	16087	16091	16097	16103	16111	16127	16139	16141	16183
16187	16189	16193	16217	16223	16229	16231	16249	16253	16267
16273	16301	16319	16333	16339	16349	16361	16363	16369	16381
16411	16417	16421	16427	16433	16447	16451	16453	16477	16481
16487	16493	16519	16529	16547	16553	16561	16567	16573	16603
16607	16619	16631	16633	16649	16651	16657	16661	16673	16691
16693	16699	16703	16729	16741	16747	16759	16763	16787	16811
16823	16829	16831	16843	16871	16879	16883	16889	16901	16903
16921	16927	16931	16937	16943	16963	16979	16981	16987	16993
17011	17021	17027	17029	17033	17041	17047	17053	17077	17093
17099	17107	17117	17123	17137	17159	17167	17183	17189	17191
17203	17207	17209	17231	17239	17257	17291	17293	17299	17317
17321	17327	17333	17341	17351	17359	17377	17383	17387	17389

Пары близнецов #1–200

3,	5	5,	7	11,	13	17,	19	29,	31
41,	43	59,	61	71,	73	101,	103	107,	109
137,	139	149,	151	179,	181	191,	193	197,	199
227,	229	239,	241	269,	271	281,	283	311,	313
347,	349	419,	421	431,	433	461,	463	521,	523
569,	571	599,	601	617,	619	641,	643	659,	661
809,	811	821,	823	827,	829	857,	859	881,	883
1019,	1021	1031,	1033	1049,	1051	1061,	1063	1091,	1093
1151,	1153	1229,	1231	1277,	1279	1289,	1291	1301,	1303
1319,	1321	1427,	1429	1451,	1453	1481,	1483	1487,	1489
1607,	1609	1619,	1621	1667,	1669	1697,	1699	1721,	1723
1787,	1789	1871,	1873	1877,	1879	1931,	1933	1949,	1951
1997,	1999	2027,	2029	2081,	2083	2087,	2089	2111,	2113
2129,	2131	2141,	2143	2237,	2239	2267,	2269	2309,	2311
2339,	2341	2381,	2383	2549,	2551	2591,	2593	2657,	2659
2687,	2689	2711,	2713	2729,	2731	2789,	2791	2801,	2803
2969,	2971	2999,	3001	3119,	3121	3167,	3169	3251,	3253
3257,	3259	3299,	3301	3329,	3331	3359,	3361	3371,	3373
3389,	3391	3461,	3463	3467,	3469	3527,	3529	3539,	3541
3557,	3559	3581,	3583	3671,	3673	3767,	3769	3821,	3823
3851,	3853	3917,	3919	3929,	3931	4001,	4003	4019,	4021
4049,	4051	4091,	4093	4127,	4129	4157,	4159	4217,	4219
4229,	4231	4241,	4243	4259,	4261	4271,	4273	4337,	4339
4421,	4423	4481,	4483	4517,	4519	4547,	4549	4637,	4639
4649,	4651	4721,	4723	4787,	4789	4799,	4801	4931,	4933
4967,	4969	5009,	5011	5021,	5023	5099,	5101	5231,	5233
5279,	5281	5417,	5419	5441,	5443	5477,	5479	5501,	5503
5519,	5521	5639,	5641	5651,	5653	5657,	5659	5741,	5743
5849,	5851	5867,	5869	5879,	5881	6089,	6091	6131,	6133
6197,	6199	6269,	6271	6299,	6301	6359,	6361	6449,	6451
6551,	6553	6569,	6571	6659,	6661	6689,	6691	6701,	6703
6761,	6763	6779,	6781	6791,	6793	6827,	6829	6869,	6871
6947,	6949	6959,	6961	7127,	7129	7211,	7213	7307,	7309
7331,	7333	7349,	7351	7457,	7459	7487,	7489	7547,	7549
7559,	7561	7589,	7591	7757,	7759	7877,	7879	7949,	7951
8009,	8011	8087,	8089	8219,	8221	8231,	8233	8291,	8293
8387,	8389	8429,	8431	8537,	8539	8597,	8599	8627,	8629
8819,	8821	8837,	8839	8861,	8863	8969,	8971	8999,	9001
9011,	9013	9041,	9043	9239,	9241	9281,	9283	9341,	9343
9419,	9421	9431,	9433	9437,	9439	9461,	9463	9629,	9631

Пары близнецов #201–400

9677, 9679	9719, 9721	9767, 9769	9857, 9859	9929, 9931
10007, 10009	10037, 10039	10067, 10069	10091, 10093	10139, 10141
10271, 10273	10301, 10303	10331, 10333	10427, 10429	10457, 10459
10499, 10501	10529, 10531	10709, 10711	10859, 10861	10889, 10891
10937, 10939	11057, 11059	11069, 11071	11117, 11119	11159, 11161
11171, 11173	11351, 11353	11489, 11491	11549, 11551	11699, 11701
11717, 11719	11777, 11779	11831, 11833	11939, 11941	11969, 11971
12041, 12043	12071, 12073	12107, 12109	12161, 12163	12239, 12241
12251, 12253	12377, 12379	12539, 12541	12611, 12613	12821, 12823
12917, 12919	13001, 13003	13007, 13009	13217, 13219	13337, 13339
13397, 13399	13679, 13681	13691, 13693	13709, 13711	13721, 13723
13757, 13759	13829, 13831	13877, 13879	13901, 13903	13931, 13933
13997, 13999	14009, 14011	14081, 14083	14249, 14251	14321, 14323
14387, 14389	14447, 14449	14549, 14551	14561, 14563	14591, 14593
14627, 14629	14867, 14869	15137, 15139	15269, 15271	15287, 15289
15329, 15331	15359, 15361	15581, 15583	15641, 15643	15647, 15649
15731, 15733	15737, 15739	15887, 15889	15971, 15973	16061, 16063
16067, 16069	16139, 16141	16187, 16189	16229, 16231	16361, 16363
16451, 16453	16631, 16633	16649, 16651	16691, 16693	16829, 16831
16901, 16903	16979, 16981	17027, 17029	17189, 17191	17207, 17209
17291, 17293	17387, 17389	17417, 17419	17489, 17491	17579, 17581
17597, 17599	17657, 17659	17681, 17683	17747, 17749	17789, 17791
17837, 17839	17909, 17911	17921, 17923	17957, 17959	17987, 17989
18041, 18043	18047, 18049	18059, 18061	18119, 18121	18131, 18133
18251, 18253	18287, 18289	18311, 18313	18521, 18523	18539, 18541
18911, 18913	18917, 18919	19079, 19081	19139, 19141	19181, 19183
19211, 19213	19379, 19381	19421, 19423	19427, 19429	19469, 19471
19541, 19543	19697, 19699	19751, 19753	19841, 19843	19889, 19891
19961, 19963	19991, 19993	20021, 20023	20147, 20149	20231, 20233
20357, 20359	20441, 20443	20477, 20479	20507, 20509	20549, 20551
20639, 20641	20717, 20719	20747, 20749	20771, 20773	20807, 20809
20897, 20899	20981, 20983	21011, 21013	21017, 21019	21059, 21061
21191, 21193	21317, 21319	21377, 21379	21491, 21493	21521, 21523
21557, 21559	21587, 21589	21599, 21601	21611, 21613	21647, 21649
21737, 21739	21839, 21841	22037, 22039	22091, 22093	22109, 22111
22157, 22159	22271, 22273	22277, 22279	22367, 22369	22481, 22483
22541, 22543	22571, 22573	22619, 22621	22637, 22639	22697, 22699
22739, 22741	22859, 22861	22961, 22963	23027, 23029	23039, 23041
23057, 23059	23201, 23203	23291, 23293	23369, 23371	23537, 23539
23561, 23563	23627, 23629	23669, 23671	23687, 23689	23741, 23743

Пары близнецов #401–600

23831, 23833	23909, 23911	24107, 24109	24179, 24181	24371, 24373
24419, 24421	24917, 24919	24977, 24979	25031, 25033	25169, 25171
25301, 25303	25307, 25309	25409, 25411	25469, 25471	25577, 25579
25601, 25603	25799, 25801	25847, 25849	25931, 25933	25997, 25999
26111, 26113	26249, 26251	26261, 26263	26681, 26683	26699, 26701
26711, 26713	26729, 26731	26861, 26863	26879, 26881	26891, 26893
26951, 26953	27059, 27061	27107, 27109	27239, 27241	27281, 27283
27407, 27409	27479, 27481	27527, 27529	27539, 27541	27581, 27583
27689, 27691	27737, 27739	27749, 27751	27791, 27793	27917, 27919
27941, 27943	28097, 28099	28109, 28111	28181, 28183	28277, 28279
28307, 28309	28349, 28351	28409, 28411	28547, 28549	28571, 28573
28619, 28621	28661, 28663	28751, 28753	29021, 29023	29129, 29131
29207, 29209	29387, 29389	29399, 29401	29567, 29569	29669, 29671
29759, 29761	29879, 29881	30011, 30013	30089, 30091	30137, 30139
30269, 30271	30389, 30391	30467, 30469	30491, 30493	30557, 30559
30839, 30841	30851, 30853	30869, 30871	31079, 31081	31121, 31123
31151, 31153	31181, 31183	31247, 31249	31319, 31321	31391, 31393
31511, 31513	31541, 31543	31721, 31723	31727, 31729	31769, 31771
31847, 31849	32027, 32029	32057, 32059	32117, 32119	32141, 32143
32189, 32191	32297, 32299	32321, 32323	32369, 32371	32411, 32413
32441, 32443	32531, 32533	32561, 32563	32609, 32611	32717, 32719
32801, 32803	32831, 32833	32909, 32911	32939, 32941	32969, 32971
33071, 33073	33149, 33151	33179, 33181	33287, 33289	33329, 33331
33347, 33349	33587, 33589	33599, 33601	33617, 33619	33749, 33751
33767, 33769	33809, 33811	33827, 33829	34031, 34033	34127, 34129
34157, 34159	34211, 34213	34259, 34261	34301, 34303	34367, 34369
34469, 34471	34499, 34501	34511, 34513	34589, 34591	34649, 34651
34757, 34759	34841, 34843	34847, 34849	34961, 34963	35051, 35053
35081, 35083	35279, 35281	35447, 35449	35507, 35509	35531, 35533
35591, 35593	35729, 35731	35801, 35803	35837, 35839	35897, 35899
36011, 36013	36107, 36109	36341, 36343	36467, 36469	36527, 36529
36779, 36781	36791, 36793	36899, 36901	36929, 36931	37019, 37021
37199, 37201	37307, 37309	37337, 37339	37361, 37363	37547, 37549
37571, 37573	37589, 37591	37691, 37693	37781, 37783	37811, 37813
37991, 37993	38237, 38239	38327, 38329	38447, 38449	38459, 38461
38567, 38569	38609, 38611	38651, 38653	38669, 38671	38711, 38713
38747, 38749	38921, 38923	39041, 39043	39161, 39163	39227, 39229
39239, 39241	39341, 39343	39371, 39373	39509, 39511	39827, 39829
39839, 39841	40037, 40039	40127, 40129	40151, 40153	40427, 40429
40529, 40531	40637, 40639	40697, 40699	40847, 40849	41141, 41143

Пары близнецов #601–800

41177, 41179 41201, 41203 41231, 41233 41387, 41389 41411, 41413
41519, 41521 41609, 41611 41759, 41761 41849, 41851 41957, 41959
41981, 41983 42017, 42019 42071, 42073 42179, 42181 42221, 42223
42281, 42283 42407, 42409 42461, 42463 42569, 42571 42641, 42643
42701, 42703 42839, 42841 42899, 42901 43049, 43051 43319, 43321
43397, 43399 43541, 43543 43577, 43579 43607, 43609 43649, 43651
43781, 43783 43787, 43789 43889, 43891 43961, 43963 44027, 44029
44087, 44089 44129, 44131 44201, 44203 44267, 44269 44279, 44281
44381, 44383 44531, 44533 44621, 44623 44699, 44701 44771, 44773
45119, 45121 45137, 45139 45179, 45181 45317, 45319 45341, 45343

45587, 45589 45821, 45823 46049, 46051 46091, 46093 46181, 46183
46271, 46273 46307, 46309 46349, 46351 46439, 46441 46589, 46591
46679, 46681 46769, 46771 46817, 46819 46829, 46831 47057, 47059
47147, 47149 47351, 47353 47387, 47389 47417, 47419 47657, 47659
47699, 47701 47711, 47713 47741, 47743 47777, 47779 47807, 47809
48119, 48121 48311, 48313 48407, 48409 48479, 48481 48539, 48541
48647, 48649 48677, 48679 48731, 48733 48779, 48781 48821, 48823
48857, 48859 48869, 48871 48989, 48991 49031, 49033 49121, 49123
49169, 49171 49199, 49201 49277, 49279 49331, 49333 49367, 49369
49391, 49393 49409, 49411 49529, 49531 49547, 49549 49667, 49669

49739, 49741 49787, 49789 49919, 49921 49937, 49939 49991, 49993
50021, 50023 50051, 50053 50129, 50131 50261, 50263 50459, 50461
50549, 50551 50591, 50593 50891, 50893 50969, 50971 51059, 51061
51131, 51133 51197, 51199 51239, 51241 51341, 51343 51347, 51349
51419, 51421 51437, 51439 51479, 51481 51719, 51721 51767, 51769
51827, 51829 51869, 51871 51971, 51973 52067, 52069 52181, 52183
52289, 52291 52361, 52363 52541, 52543 52709, 52711 52859, 52861
52901, 52903 53087, 53089 53147, 53149 53171, 53173 53231, 53233
53267, 53269 53279, 53281 53549, 53551 53591, 53593 53609, 53611
53717, 53719 53897, 53899 54011, 54013 54401, 54403 54419, 54421

54497, 54499 54539, 54541 54581, 54583 54629, 54631 54917, 54919
55049, 55051 55217, 55219 55331, 55333 55337, 55339 55439, 55441
55619, 55621 55631, 55633 55661, 55663 55817, 55819 55901, 55903
55931, 55933 56039, 56041 56099, 56101 56207, 56209 56237, 56239
56267, 56269 56477, 56479 56501, 56503 56531, 56533 56597, 56599
56711, 56713 56807, 56809 56891, 56893 56909, 56911 56921, 56923
57191, 57193 57221, 57223 57269, 57271 57329, 57331 57347, 57349
57527, 57529 57557, 57559 57791, 57793 57899, 57901 58109, 58111
58151, 58153 58169, 58171 58229, 58231 58367, 58369 58391, 58393
58439, 58441 58451, 58453 58601, 58603 58787, 58789 58907, 58909

Пары близнецов #801–1000

59009, 59011	59021, 59023	59051, 59053	59207, 59209	59219, 59221
59357, 59359	59417, 59419	59441, 59443	59471, 59473	59627, 59629
59669, 59671	60089, 60091	60101, 60103	60167, 60169	60257, 60259
60647, 60649	60659, 60661	60761, 60763	60887, 60889	60899, 60901
60917, 60919	61151, 61153	61331, 61333	61379, 61381	61469, 61471
61559, 61561	61979, 61981	62129, 62131	62141, 62143	62189, 62191
62297, 62299	62927, 62929	62969, 62971	62981, 62983	62987, 62989
63029, 63031	63197, 63199	63311, 63313	63389, 63391	63419, 63421
63587, 63589	63599, 63601	63647, 63649	63689, 63691	63839, 63841
64151, 64153	64187, 64189	64301, 64303	64451, 64453	64577, 64579
64661, 64663	64781, 64783	64877, 64879	64919, 64921	65027, 65029
65099, 65101	65171, 65173	65267, 65269	65447, 65449	65519, 65521
65537, 65539	65579, 65581	65699, 65701	65717, 65719	65729, 65731
65837, 65839	65927, 65929	65981, 65983	66107, 66109	66359, 66361
66569, 66571	66749, 66751	66851, 66853	66947, 66949	67139, 67141
67187, 67189	67211, 67213	67217, 67219	67271, 67273	67409, 67411
67427, 67429	67577, 67579	67757, 67759	67931, 67933	68111, 68113
68207, 68209	68279, 68281	68447, 68449	68489, 68491	68711, 68713
68819, 68821	68879, 68881	68897, 68899	69029, 69031	69149, 69151
69191, 69193	69257, 69259	69401, 69403	69491, 69493	69497, 69499
69737, 69739	69761, 69763	69827, 69829	69857, 69859	69929, 69931
70001, 70003	70121, 70123	70139, 70141	70181, 70183	70199, 70201
70379, 70381	70457, 70459	70487, 70489	70571, 70573	70619, 70621
70841, 70843	70877, 70879	70919, 70921	70949, 70951	70979, 70981
70997, 70999	71261, 71263	71327, 71329	71339, 71341	71387, 71389
71411, 71413	71471, 71473	71549, 71551	71711, 71713	71807, 71809
71879, 71881	72089, 72091	72101, 72103	72167, 72169	72221, 72223
72227, 72229	72251, 72253	72269, 72271	72467, 72469	72647, 72649
72671, 72673	72869, 72871	73037, 73039	73061, 73063	73361, 73363
73607, 73609	73679, 73681	73847, 73849	74099, 74101	74159, 74161
74201, 74203	74381, 74383	74411, 74413	74507, 74509	74609, 74611
74717, 74719	74729, 74731	74759, 74761	75011, 75013	75167, 75169
75209, 75211	75389, 75391	75401, 75403	75539, 75541	75617, 75619
75707, 75709	75989, 75991	76001, 76003	76079, 76081	76157, 76159
76259, 76261	76367, 76369	76421, 76423	76541, 76543	76649, 76651
76829, 76831	76871, 76873	76961, 76963	77237, 77239	77261, 77263
77267, 77269	77417, 77419	77477, 77479	77489, 77491	77549, 77551
77687, 77689	77711, 77713	78137, 78139	78191, 78193	78437, 78439
78509, 78511	78539, 78541	78569, 78571	78779, 78781	78887, 78889
78977, 78979	79151, 79153	79229, 79231	79397, 79399	79559, 79561