

# Пределы доказуемости

## Грегори Чейтин



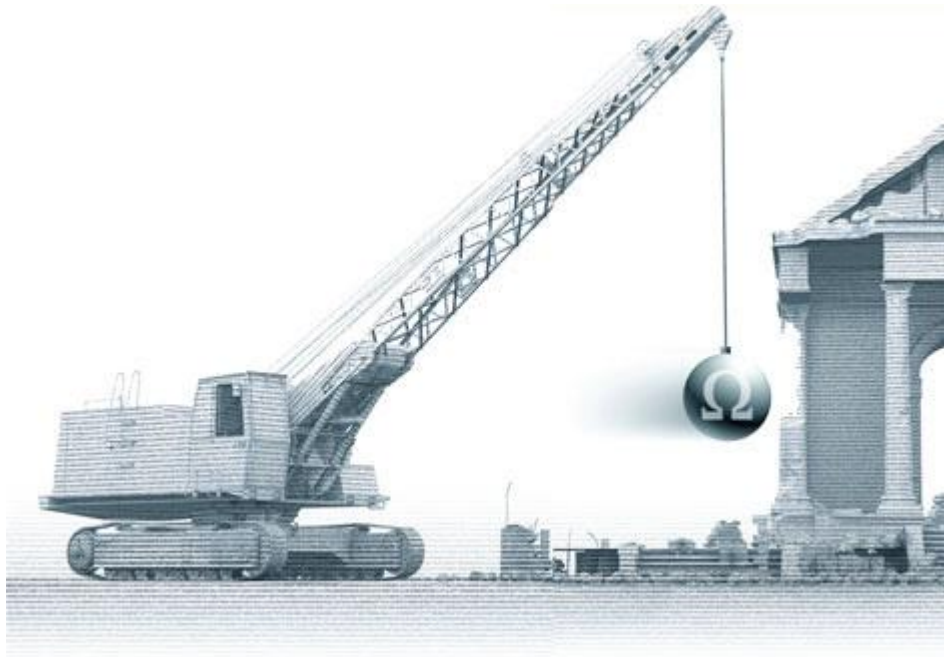
**Грегори Чейтин** (Gregory Chaitin) — научный сотрудник Исследовательского центра им. Томаса Уотсона корпорации IBM, почетный профессор Университета Буэнос-Айреса и Оклендского университета. Параллельно с А.Н. Колмогоровым занимался разработкой теории алгоритмической информации. Написал 9 книг, в том числе две научно-популярные: «Беседы с математиком» (*Conversations with a Mathematician*) и «Метаматематика!» (*Meta Math!*).

**Из идей сложности и случайности, впервые высказанных Готфридом Лейбницем в его «Рассуждении о метафизике» (1686), и их подтверждения в современной теории информации следует, что невозможно создать «самую общую теорию всего» в математике.**

В 1956 году журнал *Scientific American* опубликовал статью Эрнста Нагеля (Ernest Nagel) и Джеймса Ньюмана (James R. Newman) «Доказательство Гёделя». Через два года ее авторы выпустили одноименную книгу, которая переиздается до сих пор. В те дни я был еще ребенком, но до сих пор помню трепет, который испытал, открыв ее в Нью-йоркской публичной библиотеке.

Меня поразило то, как Курт Гёдель (Kurt Gödel) использовал математику, чтобы показать, что ее собственные возможности ограничены. Он опроверг высказанное около столетия назад Давидом Гильбертом утверждение о существовании полной теории математики, т.е. конечной совокупности принципов, из которых с помощью последовательного использования правил математической логики можно вывести все положения математики. Гёдель показал, что существуют истинные математические утверждения, которые не могут быть доказаны таким образом. Его выводы основаны на двух самоотносимых парадоксах: «данное утверждение ложно» и «данное утверждение недоказуемо». (Более подробные сведения о теореме неполноты Гёделя [можно найти](#) на сайте *Scientific American*.)

Существование специфического строго определенного числа  $\Omega$ , которое невозможно вычислить с помощью конечной компьютерной программы, разбивает надежду на создание всеобъемлющей математической системы, в рамках которой можно строго доказать любое истинное утверждение (изображение: [www.sciam.ru](http://www.sciam.ru))



Всю жизнь я разбирался с доказательством Гёделя и теперь, полвека спустя, издал собственную книжку. В какой-то степени это моя версия книги Нагеля и Ньюмана, однако доказательство Гёделя — не главная ее тема. Моя работа основана на измерении информации и доказательстве того, что некоторые математические факты не удается втиснуть в теорию, потому что они слишком сложны. Согласно моему подходу, Гёдель открыл только верхушку айсберга: существует бесконечное множество верных математических теорем, которые невозможно доказать, исходя из конечной системы аксиом.

### **Сложность и законы науки**



Готфрид Лейбниц, которому в Лейпциге поставлен памятник, еще 300 лет назад предвидел многие свойства алгоритмической информации (фото с сайта [www.uni-leipzig.de](http://www.uni-leipzig.de))

В 1686 году было издано философское эссе Готфрида Лейбница (Gottfried W. Leibniz) «Рассуждения о метафизике» (*Discours de métaphysique*), в котором поставлен вопрос: как отличить факты, которые можно описать неким законом, от фактов, никаким законам не подчиняющихся? **В четвертом разделе своего эссе Лейбниц высказал очень простую и глубокую мысль: теория должна быть проще данных, которые она объясняет, иначе она не объясняет ничего.** Концепция научного закона становится бессмысленной, если допускает неограниченный уровень математической сложности, потому что в таком случае всегда можно сформулировать закон независимо от того, насколько случайны и беспорядочны факты. И наоборот, если единственный закон, объясняющий какие-то данные, оказывается слишком сложным, то рассматриваемые данные на самом деле не подчиняются никакому закону.

Современная математическая теория алгоритмической информации позволила дать точные количественные определения понятиям сложности и простоты. Обычная теория информации определяет объем информации числом битов, необходимых для ее кодирования. Например, для кодирования простого ответа «да/нет» нужен один бит. В отличие от этого, объем алгоритмической информации определяется длиной компьютерной программы, необходимой для генерации данных. Минимальное число битов, необходимых для хранения программы, называется количеством алгоритмической информации данных. Например, бесконечный ряд натуральных чисел 1, 2, 3,... содержит очень мало алгоритмической информации: все числа ряда можно получить с помощью коротенькой компьютерной программы. Не имеет значения, сколько времени понадобится для выполнения вычислений и какой объем памяти придется использовать, важна лишь длина программы в битах. (Разумеется, точное значение количества алгоритмической информации зависит от выбранного языка программирования, однако для рассматриваемых в данной статье вопросов это несущественно.)

В качестве другого примера возьмем число  $\pi$ , равное 3,14159... Количество алгоритмической информации в нем тоже невелико: для последовательного вычисления всех его знаков нужен довольно короткий алгоритм. А вот случайное число, содержащее всего миллион знаков, скажем, 1,341285...64, характеризуется гораздо большим количеством алгоритмической информации. Поскольку в таком числе нет определяющей структуры, длина самой короткой программы, необходимой для его написания, будет близка к длине самого числа:

*Начать*

*Напечатать «1,341285...64»*

*Конец*

(В программу должны быть включены все цифры, замененные многоточием.) Никакая более короткая программа не позволит рассчитать подобную последовательность цифр: ее невозможно сжать, в ней нет избыточности. Самое лучшее, что можно сделать, — просто передать ее, как она есть. Такие последовательности называются неприводимыми или алгоритмически случайными.



Научная теория подобна компьютерной программе, предсказывающей результаты наблюдений. Полезная теория представляет собой сублимацию экспериментальных данных: с помощью нескольких законов и уравнений можно описать целый мир различных явлений (изображение: [www.sciam.ru](http://www.sciam.ru))

Как же соотносятся вышесказанное с научными законами и фактами? Идея заключается в том, чтобы взглянуть на науку глазами программиста: научная теория подобна компьютерной программе, предсказывающей результаты наблюдений, т.е. экспериментальные данные. Такая точка зрения опирается на два фундаментальных принципа. Согласно первому («бритва Оккама»), из двух теорий, объясняющих некоторые данные, следует предпочесть более простую. Иначе говоря, наилучшей теорией является самая короткая программа, позволяющая рассчитать результаты наблюдений. Второй принцип, изложенный Лейбницем, в современных понятиях звучит так: теория, объем которой в битах равен количеству объясняемых ею данных, бесполезна, поскольку теорией такого размера можно описать совершенно случайные данные. Полезная теория обеспечивает сокращение количества информации: осмысление данных — это их сжатие в краткие алгоритмические описания. Чем проще теория, тем лучше понимание сути явления.

### **Достаточная причина**

Лейбниц, живший за два с половиной века до появления компьютерных программ, очень близко подошел к современному понятию алгоритмической информации. Лейбниц знал, что все можно представить в виде двоичных кодов, и создал одно из первых вычислительных устройств; рассматривая понятия сложности и простоты, он осознавал огромный потенциал вычислений. Если бы Лейбниц объединил все известные ему элементы, то, скорее всего, усомнился бы в одном из устоев своей философии — принципе достаточной причины, согласно которому все происходящее имеет причину. Более того, если какое-то положение истинно, то оно истинно по какой-то причине. Бывает, что в суесть и хаосе повседневной жизни в это трудно поверить. Даже если мы не всегда можем увидеть причину (возможно потому, что цепочка рассуждений слишком длинна и запутанна), ее видит Бог. Вот и всё! Здесь Лейбниц соглашался с древнегреческими авторами этой идеи.

Математики, несомненно, безоговорочно принимают принцип достаточной причины Лейбница, потому что всегда стремятся всё доказать. Даже если истинность теоремы очевидна, и миллионы примеров подтверждают ее, математики все равно требуют обобщенного доказательства, на меньшее они не согласны. И здесь концепция алгоритмической информации может внести удивительный вклад в философские рассуждения об источниках и пределах познания. Она показывает, что некоторые математические факты истинны безо всяких причин, и бросает вызов принципу достаточной причины. Как будет показано ниже, существует бесконечное число неприводимых математических фактов, истинность которых нельзя объяснить никакой теорией. Они неприводимы не только вычислительно, но и логически. «Доказать» эти факты можно только одним способом: признать их аксиомами без всяких рассуждений.

Понятие «аксиома» тесно связано с логической неприводимостью. Аксиомы — это математические положения, которые мы считаем самоочевидными и не пытаемся доказать, исходя из более простых принципов. Все математические теории основаны на аксиомах, из которых выводятся следствия, называемые теоремами. Именно так поступал Евклид два тысячелетия назад: его труды по геометрии стали классическим примером математического изложения.

В древней Греции, чтобы убедить сограждан проголосовать именно так, а не иначе, вы должны были бы привести им свои доводы. Вероятно, именно поэтому греки пришли к мысли, что математические положения нужно доказывать, а не выводить опытным путем. (В отличие от греков, древнейшие цивилизации Месопотамии и Египта, похоже, полагались на эксперимент.) Метод логических рассуждений оказался чрезвычайно плодотворным: с его помощью были созданы современная математика, математическая физика и все точные науки, включая технологию создания компьютеров — в высшей степени математичных и логичных машин. Утверждаю ли я, что подход, на котором математика и вся наука строились в течение двух тысячелетий, терпит крах? В каком-то смысле да. Моим контрпримером, иллюстрирующим ограниченность возможностей логики и рассуждений, моим источником бесконечного потока недоказуемых математических положений является число, которое я назвал «омега» ( $\Omega$ ).

## Число $\Omega$

Первый шаг к открытию числа  $\Omega$  был сделан в знаменитой статье, опубликованной ровно через 250 лет после издания эссе Лейбница. В 1936 году на страницах «Трудов Лондонского математического общества» (*Proceedings of the London Mathematical Society*) Алан Тьюринг впервые представил математическую модель простой универсальной вычислительной машины. Кроме того, он задался вопросом: можно ли определить, остановится когда-нибудь компьютерная программа или нет? Так была сформулирована знаменитая проблема останова.



Число  $\Omega$  представляет собой непознаваемую часть математики. Компьютерная программа конечной длины позволяет определить лишь конечное число битов этого числа; все последующие остаются во мраке неизвестности (изображение: [www.sciam.ru](http://www.sciam.ru))

Разумеется, запустив программу, вы можете со временем обнаружить, что она остановилась. Фундаментальная проблема заключается в том, чтобы решить, когда вы сдадитесь и перестанете ждать, если программа не останавливается. Для множества частных случаев она может быть решена, но Тьюринг показал, что общего решения не существует. Никакой алгоритм и никакая математическая теория не позволят определить, какая программа остановится, а какая нет. (Современное доказательство данного положения Тьюринга [можно найти](#) на сайте Scientific American.) Кстати, употребляя слово

«программа» в современном смысле, я имею в виду совокупность самой компьютерной программы и данных, которые она обрабатывает.

Следующим шагом на пути к числу  $\Omega$  становится рассмотрение множества всех возможных программ. Остановится ли когда-нибудь выбранная случайным образом программа? Вероятность останова и есть  $\Omega$ . Определим сначала, как осуществить случайный выбор программы. Программа представляет собой последовательность битов, поэтому для выбора значения каждого последующего бита будем просто бросать монету. Сколько битов должна содержать программа? Будем бросать монету до тех пор, пока компьютер не перестанет запрашивать следующий бит. Число  $\Omega$  есть вероятность того, что при введении такой случайной последовательности битов машина когда-нибудь остановится. (Численное значение  $\Omega$  зависит от выбора языка программирования, но удивительные свойства этого числа с ним не связаны. Когда же язык выбран,  $\Omega$  приобретает определенную величину, так же, как число  $\pi$  или число 5.)

Поскольку число  $\Omega$  выражает вероятность, оно должно быть больше нуля, но меньше единицы, т.к. некоторые программы останавливаются, а некоторые – нет. Число  $\Omega$ , записанное в двоичном коде, будет иметь вид вроде 0,1110100... Последовательность битов после запятой неприводима, а сами они оказываются неприводимыми математическими фактами (каждый факт состоит в том, является ли данный бит нулем или единицей).

### Как определяется число $\Omega$

Чтобы понять, как определяется число  $\Omega$ , рассмотрим упрощенный пример. Допустим, что из всех программ некоего компьютера останавливаются всего три, которые представляют собой 110, 11100 и 11110, длиной 3, 5 и 5 битов соответственно. Если для выбора программ мы будем бросать монету, чтобы определять значение каждого очередного бита случайным образом, вероятности получения каждой из них будут равны соответственно  $1/2^3$ ,  $1/2^5$  и  $1/2^5$ , поскольку вероятность получения единицы или нуля для каждого бита равна  $1/2$ . Тогда вероятность останова программы для такого компьютера будет определяться выражением:

$$\Omega = 1/2^3 + 1/2^5 + 1/2^5 = 0,001 + 0,00001 + 0,00001 = 0,00110.$$

Здесь двоичные числа выражают вероятность случайного выбора одной из трех останавливающихся программ. Поскольку программа 110 останавливается, мы не рассматриваем программы длиной больше трех битов, начинающиеся с 110, например, 1100 и 1101, т. е. мы не добавляем к сумме 0,0001 для каждой из них. Мы считаем все более длинные программы (1100 и т. д.) с таким началом включенными в программу 110, которая останавливается. Другими словами, данные программы являются самоограничивающимися, поскольку после их остановки последующие биты не запрашиваются.

Число  $\Omega$  можно определить как бесконечную сумму, и каждая программа длиной  $N$  битов вносит в нее свой вклад, равный  $1/2^N$ . Иными словами, каждая  $N$ -битовая программа, которая останавливается, добавляет единицу к  $N$ -ному биту двоичного представления числа  $\Omega$ . Сложив все биты, соответствующие остановившимся программам, мы можем получить точное значение  $\Omega$ . Создается впечатление, что  $\Omega$  можно вычислить точно, как  $\sqrt{2}$  или  $\pi$ . Однако это не так: число  $\Omega$  строго определено и имеет вполне конкретное значение, но рассчитать его невозможно, поскольку это позволило бы решить проблему останова, у которой действительно нет решения. Если говорить конкретнее, знание первых  $N$  битов числа  $\Omega$  позволяет определить, остановится ли когда-нибудь любая программа длиной до  $N$  битов, из чего следует, что для нахождения  $N$  битов числа  $\Omega$  требуется программа длиной

не менее  $N$  битов. Заметьте, я не утверждаю, что нельзя определить некоторое число битов числа  $\Omega$ . Например, зная, что компьютерные программы 0, 10 и 110 останавливаются, мы можем говорить, что с точностью до первых трех битов  $\Omega$  имеет вид 0,111. Дело в том, что первые  $N$  битов  $\Omega$  нельзя вычислить с помощью программы, которая была бы существенно короче  $N$  битов.

Самое главное, что  $\Omega$  дает нам бесконечное число неприводимых битов. Любая программа конечной длины, сколько миллиардов битов она бы ни содержала, не поможет нам определить оставшиеся биты, которых бесконечно много. Иными словами, при любом конечном наборе аксиом мы имеем бесконечное число истин, которые не могут быть доказаны с помощью этого набора.

### Почему число $\Omega$ несжимаемо?

Попробуем доказать, что число несжимаемо, т. е. его первые  $N$  битов невозможно определить с помощью программы длиной существенно меньше  $N$  битов. Проанализируем свойства  $\Omega$  в свете поставленной Тьюрингом проблемы останова. Используем утверждение, что для программ длиной до  $N$  битов задача не может быть решена с помощью программы меньшей длины.

Для демонстрации несжимаемости числа  $\Omega$  покажем, что знание первых его  $N$  битов позволяет решить задачу Тьюринга для программ длиной до  $N$  битов. Из этого будет следовать, что никакая программа длиной меньше  $N$  битов не может вычислить первые  $N$  битов  $\Omega$ . (Если бы такая программа существовала, с ее помощью можно было бы вычислить первые  $N$  битов  $\Omega$ , а затем использовать их в решении задачи Тьюринга для программ длиной  $N$  битов — задача невыполнимая с помощью столь короткой программы.)

Посмотрим теперь, как знание  $N$  битов  $\Omega$  позволит решить задачу останова и определить, какие из всех программ длиной до  $N$  битов будут останавливаться. Сделаем это поэтапно. На  $K$ -м этапе будем прогонять каждую программу в течение  $K$  секунд и по числу остановившихся программ определять вероятность останова  $\Omega_K$ . Она окажется меньше  $\Omega$ , поскольку будет получена с использованием лишь подмножества программ, которые в конце концов останавливаются, тогда как  $\Omega$  рассчитывается с использованием всех программ. С увеличением  $K$  значение  $\Omega_K$  будет приближаться к  $\Omega$ , и все больше первых битов  $\Omega_K$  будут становиться равными соответствующим битам  $\Omega$ . Когда первые  $N$  битов  $\Omega_K$  и  $\Omega$  совпадут, это будет означать, что учтены все программы длиной до  $N$  битов, которые рано или поздно останавливаются. (Если бы существовала еще какая-то программа длиной  $N$  битов, она остановилась бы на более позднем этапе  $K$ , и тогда  $\Omega_K$  оказалось бы больше  $\Omega$ , что невозможно.)

Итак, зная первые  $N$  битов  $\Omega$ , можно решить задачу останова для всех программ длиной до  $N$  битов. Предположим теперь, что первые  $N$  битов  $\Omega$  можно определить с помощью программы длиной существенно меньше  $N$  битов. Тогда ее можно объединить с программой вычисления  $\Omega_K$  и получить в итоге программу длиной меньше  $N$  битов для решения проблемы останова для всех программ длиной до  $N$  битов. Однако, как сказано выше, такой программы существовать не может. Следовательно, для вычисления первых  $N$  битов  $\Omega$  требуется программа длиной почти  $N$  битов. Этого достаточно, чтобы признать число  $\Omega$  несжимаемым, т.е. неприводимым. (Для больших  $N$  сокращение длины с  $N$  битов до почти  $N$  битов несущественно.)

Из неприводимости числа  $\Omega$  следует, что всеобъемлющей математической теории существовать не может. Бесконечное множество битов  $\Omega$  составляет бесконечное

множество математических фактов (является ли каждый выбранный бит единицей или нулем), которые не могут быть выведены из каких бы то ни было принципов, более простых, чем сама последовательность битов. Значит, сложность математики бесконечна, тогда как любая отдельная теория «всего на свете» характеризуется конечной сложностью и, следовательно, не может охватить все богатство мира математических истин. Из сказанного отнюдь не следует, что от доказательств нет никакого толка, и я ни в коем случае не против логических рассуждений. На самом деле, неприводимые принципы (аксиомы) всегда составляли часть математики. Просто число  $\Omega$  показывает, что их гораздо больше, чем предполагалось ранее.

### **Обзор: неприводимая сложность**

\* Курт Гедель показал неизбежную неполноту математики: в ней существуют истинные положения, которые невозможно строго доказать. Особое число  $\Omega$  выявляет еще бóльшую неполноту и свидетельствует о существовании бесконечного множества теорем, которые нельзя вывести из конечного набора аксиом.

\* Число  $\Omega$  строго определено и имеет вполне конкретное значение, но вычислить его с помощью конечной компьютерной программы невозможно.

\* Анализ свойств числа  $\Omega$  показывает, что математикам иногда следует постулировать новые аксиомы. Именно так поступают физики, которые обобщают результаты экспериментов и выводят фундаментальные законы, недоказуемые с помощью логики.

Возможно, математикам не нужно пытаться все доказать. Иногда им следует просто добавлять новые аксиомы, когда дело доходит до неприводимых фактов. Проблема в том, чтобы понять, что они неприводимы, и признать, что их невозможно доказать. Однако математики никогда не сдадутся, в отличие от физиков, которые всегда готовы обойтись правдоподобными рассуждениями вместо строгих доказательств, и охотно выводят новые законы, чтобы осмыслить свежие экспериментальные данные. Возникает интересный вопрос: похожа ли математика на физику?

### **Математика и физика**

Принято считать, что математика и физика совершенно не похожи друг на друга. Физики описывают мир, исходя из результатов экспериментов и наблюдений. Законы, управляющие Вселенной, будь то законы Ньютона или Стандартная модель физики элементарных частиц, должны устанавливаться эмпирически и затем приниматься за аксиомы, которые невозможно доказать логическим путем, а можно лишь проверить экспериментально. Математики же в некотором смысле независимы от мира. Их выводы и теоремы, например, свойства целых или вещественных чисел, никак не зависят от окружающей нас реальности. Математические истины должны быть верны в любом мире. И все же определенное сходство есть. В физике, и вообще в естественных науках, ученые формулируют законы, сублимируя результаты наблюдений. Затем они показывают, как результаты наблюдений могут быть выведены из получившихся законов. В математике происходит нечто подобное: математики сжимают результаты вычислительных экспериментов в аксиомы, а затем выводят из них теоремы.

Если бы Гильберт оказался прав, то математика была бы замкнутой системой, в которой нет места новым идеям. Существовала бы статичная замкнутая теория, объясняющая в математике все, и это было бы похоже на диктатуру. Чтобы математика развивалась, нужны новые идеи и простор для творчества. Недостаточно усердно работать, выводя все



возможные следствия из фиксированного числа базовых принципов. Лично мне больше нравятся открытые системы, я не люблю жестких, авторитарных способов мышления.

Имре Лакатош (*Imre Lakatos*), бежавший в 1956 году из Венгрии и впоследствии занимавшийся философией науки в Англии, тоже считал, что математика похожа на физику. Он ввел понятие квазиэмпиричности, чтобы показать, что и математике не чужды эксперименты. Например, еще в 1742 году Кристиан Гольдбах опытным путем пришел к предположению, что любое четное число больше двух можно представить в виде суммы двух простых чисел. Предположение Гольдбаха успешно проверено для чисел до  $10^{14}$ , но строго не доказано. Мне кажется, что математика квазиэмпирична. Иными словами, она отличается от физики (которая истинно эмпирична), но, вероятно, не так сильно, как полагает большинство людей.

## Новые аксиомы

Идея добавления новых аксиом не чужда математикам. Возьмем для примера пятый постулат Евклида: через выбранную точку, лежащую вне прямой, можно провести только одну прямую, параллельную данной. Столетиями геометры ломали голову, пытаясь доказать это, исходя из остальных постулатов Евклида. Не удалось. Наконец, математики поняли, что пятую аксиому можно заменить и получить неевклидову геометрию криволинейных пространств, в частности сферического и седлообразного. Другими примерами могут служить закон исключенного среднего в логике и аксиома выбора в теории множеств, которыми охотно пользуется в своих доказательствах большинство математиков. Но ведь есть ученые, которые их не признают и исследуют так называемую интуиционистскую логику и конструктивистскую математику. Оказывается, математика пока не стала монолитной системой абсолютных истин!

Другой очень интересной аксиомой может стать утверждение « $P$  не равно  $NP$ », где  $P$  и  $NP$  – названия классов задач. К классу  $NP$  относятся задачи, для которых предлагаемое решение можно проверить очень быстро. Например, для задачи «найти множители числа 8 633» предлагаемое решение «97 и 89» быстро проверяется простым перемножением. (Существует строгое определение понятия «быстро», но подробности здесь не имеют значения.) Класс  $P$  составляют задачи, которые можно быстро решить, не имея предварительного предположения. Вопрос, ответа на который не знает никто, состоит в том, можно ли быстро решить любую задачу класса  $NP$ . (Есть ли способ быстро найти множители числа 8 633?) Иначе говоря, тождественны ли классы  $P$  и  $NP$ ? Это один из пунктов списка «Проблем тысячелетия» Математического института Клэя (*Clay Millennium Prize Problem*), за решение каждой из которых назначена награда в \$1 млн.

Большинство специалистов по вычислительной технике убеждено, что  $P$  не равно  $NP$ , но строгое доказательство пока не найдено. Истинность такого предположения подтверждается множеством эмпирических свидетельств, но можно ли на этом основании принять его в качестве аксиомы? Специалисты по вычислительной технике именно так и поступили. Правда, остается вопрос о надежности некоторых широко применяемых криптографических систем: считается, что взломать их невозможно, но никто не может этого доказать.

## Экспериментальная математика

На стыке физики и математики возникла экспериментальная математика: открытие новых математических закономерностей путем компьютерной обработки большого числа примеров. Такой подход не столь убедителен, как короткое доказательство, но может быть

убедительнее длинного, сложного доказательства и в некоторых случаях вполне приемлем. В прошлом данную концепцию отстаивали и Дьёрдь Пойа (*George Pólya*), и Лакатош, убежденные сторонники эвристических методов и квазиэмпирической природы математики. Он применяется и обосновывается в книге «Новый вид науки» (*A New Kind of Science*) Стивена Вольфрама (*Stephen Wolfram*), вышедшей в 2002 году.

Масштабные компьютерные вычисления могут быть очень убедительными, но избавляют ли они от необходимости доказательств? И да, и нет. Вычисления и доказательства дают свидетельства разного рода. В особо важных случаях я считаю необходимыми и те, и другие, поскольку доказательства могут содержать ошибки, а компьютерные вычисления могут, по несчастью, быть остановлены как раз перед обнаружением контрпримера, который опроверг бы предполагаемый вывод.

Рассмотренные вопросы чрезвычайно интересны, но далеки от решения. Со времени публикации статьи о доказательстве Гёделя прошло 50 лет, а сейчас, в 2006 году, мы все еще не знаем, насколько серьезна неполнота, и не следует ли из-за нее пересмотреть математические методы. Возможно, через 50 лет ответ будет найден.

#### Дополнительная литература:

- Главу о Лейбнице см. в книге: **Men of Mathematics**. E.T. Bell. Reissue. Touchstone, 1986.
- Более полные сведения о квазиэмпирическом взгляде на математику см.: **New Directions in the Philosophy of Mathematics**. Edited by Thomas Tymoczko. Princeton University Press, 1998.
- **Gödel's Proof**. Revised edition. E. Nagel, J.R. Newman and D.R. Hofstadter. New York University Press, 2002.
- **Mathematics by Experiment: Plausible Reasoning in the 21st Century**. J. Borwein and D. Bailey. A.K. Peters, 2004.
- О философии Гёделя и связи его работ с трудами Лейбница см.: **Incompleteness: The Proof and Paradox of Kurt Gödel**. Rebecca Goldstein. W.W. Norton 2005.
- **Meta Math!: The Quest for Omega**. Gregory Chaitin. Pantheon Books, 2005.
- [Краткие биографии математиков](#) доступны на сайте Школы математики и статистики Университета Св. Эндрю (Шотландия).
- [Домашняя страница Грегори Чейтина](#).