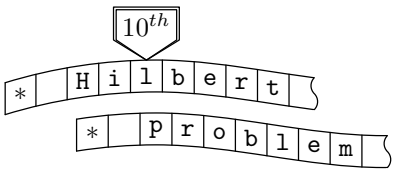


1

Десятая проблема Гильберта

**Ю. В. Матиясевич, Я. Абрамов, А. Я. Белов-Канель,
И. А. Иванов-Погодаев, А. С. Малистов**

В математике часто встречаются уравнения, обе части которых представляют собой многочлены с целыми коэффициентами, а решения требуется найти в целых числах. Такие уравнения называются *диофантовыми*.

На Втором Международном конгрессе математиков в Париже [1900] Давид Гильберт сделал свой знаменитый доклад «Математические проблемы», содержащий 23 проблемы или, точнее, 23 группы родственных проблем, которые 19-й век оставлял в наследие 20-му. Проблема под номером десять была посвящена диофантовым уравнениям.

10. Задача о разрешимости диофантова уравнения

Пусть задано диофантово уравнение с произвольными неизвестными и целыми рациональными коэффициентами. Указать способ, при помощи которого возможно после конечного числа операций установить, разрешимо ли это уравнение в целых рациональных числах.

Под «способом», который предлагает найти Д. Гильберт, в настоящее время подразумевают «алгоритм». В начале века, когда проблемы формулировались, еще не было математически строгого *общего* понятия алгоритма. Отсутствие такого понятия не могло само по себе служить препятствием к положительному решению 10-й проблемы Гильберта, поскольку про *конкретные* алгоритмы всегда было ясно, что они действительно дают требуемый общий способ решения соответствующих проблем.

В 30-е годы в работах К. Геделя, А. Ччрча, А.М. Тьюринга и других логиков было выработано строгое общее понятие алгоритма, которое дало принципиальную возможность устанавливать *алгоритмическую неразрешимость*, то есть доказывать невозможность алгоритма с требуемыми свойствами. Тогда же были найдены первые примеры алгоритмически неразрешимых проблем,сначала в самой математической логике, а затем и в других разделах математики.

Таким образом, теория алгоритмов создала необходимые предпосылки для попыток доказать неразрешимость 10-й проблемы Гильберта. Первые работы в этом направлении были опубликованы в начале 50-х годов, а в 1970 году исследования завершились «отрицательным решением» 10-й проблемы Гильберта, полученным Ю. В. Матиясевичем.

В настоящем цикле задач мы следуем пути доказательста Ю. В. Матиясевича.

A. Диофантовы множества

В этой задаче, решая уравнения, мы будем искать только целые неотрицательные решения.

Рассмотрим равенство вида

$$D(a_1, \ldots, a_n, x_1, \ldots, x_m) = 0,$$

где D — многочлен с целыми коэффициентами относительно переменных a_1, …, a_n, x_1, …, x_m. Будем считать, что переменные разбиты на две части: *параметры* $a_1, \ldots, a_n$ и неизвестные $x_1, \ldots, x_m$. При фиксированных целых значениях параметров получаются конкретные диофантовы уравнения, составляющие семейство. (Заметим, коэффициенты у нас просто целые, а решения мы ищем целые неотрицательные). Например, если $D(a_1, a_2, x) = a_1x^2 + a_2x$, то в семейство диофантовых уравнений войдут уравнения $x^2 = 0$, $2x^2 + 6x = 0$, $3x^2 - 17x = 0$ и т. п. В первом случае значения параметров выбраны $a_1 = 1$, $a_2 = 0$, во втором — $a_1 = 2$, $a_2 = 6$, в третьем — $a_1 = 3$, $a_2 = -17$.

Семейство диофантовых уравнений $D(a_1, \ldots, a_n, x_1, \ldots, x_m) = 0$ определяет некоторое множество M, состоящее из всех таких наборов значений параметров $a_1, \ldots, a_n$, для которых существуют целые

неотрицательные значения неизвестных $x_1, \ldots, x_m$, удовлетворяющие уравнению. В этом случае говорят, что множество M имеет *диофантово представление*. Число n называют *размерностью* множества. Множества, имеющие диофантовы представления, будем называть *диофантовыми*.

Множество чётных чисел диофантово: уравнение $2x = a$ имеет целочисленное решение только при чётных значениях параметра a. Аналогично диофантовым является множество пар (a, b), таких, что a – нечётное, а b – чётное, большее 2. Для доказательства достаточно рассмотреть представление $2(2x + 1) = a^2(b^3 - 14)$.

♦ **A1.** *Установите диофантовость следующих множеств:*
a) *множество всех положительных чётных чисел;*
b) *множество всех нечётных чисел;*
c) *множество всех чисел, являющихся квадратами;*
d) *множество всех чисел, являющихся кубами.*

♦ **A2.** *Докажите, что система диофантовых уравнений эквивалентна некоторому диофантову уравнению* (*т. е. множества решений у них совпадают*).

♦ **A3.** *Докажите, что объединение и пересечение двух диофантовых множеств одинаковой размерности тоже диофантово.*

♦ **A4.** *Докажите, что если множество n-ок $\mathfrak{M}$ диофантово, то множество m-ок $(a_1, \ldots, a_m)$ таких, что существуют $a_{m+1}, \ldots, a_n$, что $(a_1, \ldots, a_n) \in \mathfrak{M}$, тоже диофантово.*

В дальнейшем можно говорить не о диофантовости каких-либо множеств, а о диофантовости свойств и отношений. Например, можно говорить не о принадлежности к диофантову множеству чётных чисел, а о диофантовости свойства *быть чётным числом*. Аналогично, вместо того чтобы говорить о диофантовости множества с представлением $(a_1 - a_2)^2 = x + 1$, можно говорить о диофантовости отношения $\neq$. Формально, будем называть свойство P натуральных чисел диофантовым, если множество всех чисел, обладающих свойством P, диофантово. Аналогично, отношение R между n натуральными числами называется диофантовым, если диофантовым является множество n-ок (последовательностей из n элементов) натуральных чисел, находящихся в отношении R. Функцию $a = F(b_1, \ldots b_k)$ будем называть диофантовой, если множество последовательностей из $k + 1$ элемента $[a, b_1, \ldots, b_k]$, удовлетворяющих равенству $a = F(b_1, \ldots b_k)$, является диофантовым.

♦ **A5.** *Докажите, что следующие отношения диофантовы:*

a) отношение «больше» $(a > b)$;

b) отношение «делимости» (a делит b).

c) Докажите, что диофантовым является множество троек чисел (a, b, c), таких что a — остаток от деления b на c (**обозначение: $\mathbf{a = b \pmod c}$**);

d) Докажите, что диофантовым является множество троек чисел (a, b, c), таких что $a = \min\{b \pmod c, (c - b) \pmod c\}$ (**обозначение: $\mathbf{a = b \pmod c}$**)
(Т.е. расстояние до ближайшего числа, делящегося на c);
e) Докажите, что диофантовым является множество троек (a, b, c), таких что $a = \left[\frac{b}{c}\right]$;

♦ **A6.** *Докажите, что отношение «взаимной простоты» а также функции НОК и НОД диофантовы.*

♦ **A7.** *Докажите, что диофантовым является:*
а) множество всех чисел, не являющихся квадратами;
б) множество пар (a, b), таких что a — не степень b.

Уравнение Пелля

Определение. *Уравнением Пелля* называется уравнение вида $x^2 - dy^2 = 1, d \in \mathbb{N}$.

♦ **A8.** *Уравнение Пелля.*
а) Решение (x, y) назовчм нетривиальным, если $y \neq 0$. Пусть d – квадрат. Докажите, что нет нетривиальных решений.
б) Пусть (u_1, v_1) и (u_2, v_2) – решения уравнения Пелля $x^2 - dy^2 = 1$. Докажите, что если $u_3 + \sqrt{d}v_3 = (u_1 + \sqrt{d}v_1) \cdot (u_2 + \sqrt{d}v_2)$, то (u_3, v_3) — также решение. В частности, если (x, y) — решение, то (x_n, y_n) — решение, где $x_n + \sqrt{d}y_n = (x + \sqrt{d}y)^n$.
в) Решение назовчм минимальным, если оно нетривиально, и среди нетривиальных $|x+y|$ минимально. Докажите, что любое решение — степень минимального.

Известно, что если d не квадрат, то существует нетривиальное решение. Это достаточно сложное утверждение, поэтому мы не приводим его доказательство и не включаем его в список задач. Доказательство см., например, в книге В. О. Бугаенко «Уравнение Пелля».

♦ **A9.** *Специальный случай уравнения Пелля*
а) $d = k^2 - 1$. Докажите, что $(k, 1)$ — минимальное решение.
б) $d = k^2 - 1$, (x_1, y_1) — минимальное решение, $(x_n, y_n) = (x_1, y_1)^n$. Докажите, что $y_n \equiv n \pmod{(k-1)}$.
с) Другие виды уравнения Пелля. Решите уравнение $x^2 - (\frac{b^2}{4} - 1)y^2 = 1$

Диофантовость степени

Сейчас мы будем рассматривать решения специального семейства диофантовых уравнений, называемых уравнениями Пелля. Решения этих уравнений получаются из частного их решения с помощью последовательного многократного применения одного и того же преобразования плоскости.

Рассмотрим последовательность, заданную так: $\alpha_0(b) = 0$, $\alpha_1(b) = 1$, $\alpha_{n+2}(b) = b\alpha_{n+1}(b) - \alpha_n(b)$, где $b \geqslant 2$.

♦ **A10.** *Докажите, что $x^2 - bxy + y^2 = 1, x, y \geqslant 0$, тогда и только тогда, когда*

$$\textit{либо} \begin{cases} x = \alpha_{m+1}(b), \\ y = \alpha_m(b), \end{cases} \qquad \textit{либо} \begin{cases} x = \alpha_m(b), \\ y = \alpha_{m+1}(b), \end{cases}$$

для некоторого m.

♦ **A11.** *Докажите, что $\alpha_n(2) = n$;*

♦ **A12.** *Докажите, что $\alpha_{k+\ell}(b) = \alpha_k(b) \cdot \alpha_{\ell+1}(b) - \alpha_{k-1}(b) \cdot \alpha_\ell(b)$.*

♦ **A13.** *Докажите, что $\alpha_n(b) \equiv \alpha_{n+4m}(b) \pmod{v}$, где $v = \alpha_{m+1}(b) - \alpha_{m-1}(b)$;*

♦ **A14.** *Докажите, что если $b_1 \equiv b_2 \pmod{q}$, то $\alpha_n(b_1) \equiv \alpha_n(b_2) \pmod{q}$.*

♦ **A15.** *Докажите, что наименьшим числом k, таким что при фиксированном n*

$$\alpha_n(w) \pmod{v} = \alpha_{n+k}(w) \pmod{v},$$

$$\textit{где} \quad w \equiv b \pmod{v}, \quad v = \alpha_{m+1}(b) - \alpha_{m-1}(b),$$

является $2m$.

♦ **A16.** *Докажите, что если*

$$w \equiv b \pmod{v}, \quad w \equiv 2 \pmod{u}, \qquad \text{где } v > 2\alpha_k(b),\ u > 2k,$$

то первые k членов последовательности

$$\big(\alpha_0(b), 0\big), \ldots, \big(\alpha_n(b), n\big), \ldots$$

совпадают с первыми k членами последовательности

$$\Big(\alpha_0(w) \pmod{v}, \alpha_0(w) \pmod{u}\Big), \ldots, \Big(\alpha_n(w) \pmod{v}, \alpha_n(w) \pmod{u}\Big), \ldots$$

♦ **A17.** *Докажите, что если $\alpha_m(b)$ делится на $(\alpha_k(b))^2$, то m делится на $\alpha_k(b)$.*

♦ **A18.** *Докажите, что если $2\alpha_k(b) < u$, то $2k < u$.*

♦ **A19.** *Докажите, что множество $\{(a,\ b,\ c) \mid\ a = \alpha_c(b),\ b > 3\}$ диофантово.*

♦ **A20.** *Докажите, что $(k-1)^n \leqslant \alpha_{n+1}(k) \leqslant k^n$;*

♦ **A21.** *Докажите, что $(1+s)^n \geqslant 1 + ns$ при $s \in \mathbb{R}$, $s > -1$, n — целое неотрицательное;*

♦ **A22.** *Докажите, что* $b^c = \lim\limits_{n\to\infty} \dfrac{\alpha_{c+1}(bn+4)}{\alpha_{c+1}(n)}$.

♦ **A23.** *Докажите диофантовость множества* $\big\{(a,b,c) \mid a = b^c\big\}$.

B. Кодирование

Данное Кантором классическое доказательство счётности счётного объединения счётных множеств использует следующее линейное упорядочивание множества всех пар натуральных чисел:

$$\langle 0,0\rangle, \langle 0,1\rangle, \langle 1,0\rangle, \langle 0,2\rangle, \langle 1,1\rangle, \langle 2,0\rangle, \langle 0,3\rangle \ldots$$

Заметим, что номер пары $\langle a,b\rangle$ в указанной последовательности полиномиально выражается через a и b, а именно, он равен $\mathbf{Cantor}(a,b) = ((a+b)^2+3a+b)/2$. Соответственно, диофантовыми оказываются функции $\mathbf{ElemA}(c)$ и $\mathbf{ElemB}(c)$, дающие по номеру пары ее первый и второй элементы:

$$a = \mathbf{ElemA}(c) \iff \exists y\colon [(a+y)^2 + 3a + y = 2c];$$
$$b = \mathbf{ElemB}(c) \iff \exists x\colon [(x+b)^2 + 3x + b = 2c]$$

Нумерацию пар легко обобщить на нумерацию троек, четвёрок и т.д. Можно, например, положить

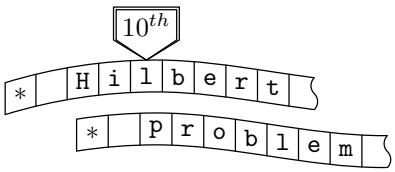
$$\mathbf{Cantor}_1(a_1) = a_1, \qquad \mathbf{Cantor}_{n+1}(a_1, \ldots a_{n+1}) = \mathbf{Cantor}_n(a_1, \ldots a_{n-1}, \mathbf{Cantor}(a_n, a_{n+1}))$$

и называть число $\mathbf{Cantor}_n(a_1, \ldots a_n)$ *канторовым номером* кортежа $\langle a_1, \ldots a_n\rangle$. Аналогично, функция $\mathbf{Elem}_{n,m}(c)$, дающая по числу c, рассматриваемому как номер n-ки натуральных чисел, значение ее m-ой компоненты, будет диофантовой:

$$a = \mathbf{Elem}_{n,m}(c) \iff \exists x_1 \ldots x_{m-1} x_{m+1} \ldots x_n\colon [2^{2^n}\, \mathbf{Cantor}_n(x_1, \ldots x_{m-1}, a, x_{m+1} \ldots x_n) = 2^{2^n} c]$$

(появление множителя 2^{2^n} связано с тем, что $\mathbf{Cantor}_n$ не является полиномом с целыми коэффициентами).

Заметим однако, что введенная таким образом нумерация кортежей натуральных чисел обладает одним существенным недостатком: обратная функция $\mathbf{Elem}_{n,m}(c)$ является диофантовой функцией одной переменной при любых фиксированных значениях n и m, но не видно пути установить диофантовость $\mathbf{Elem}_{n,m}(c)$ как функции трех аргументов c, m, n. Для работы с кортежами, длина которых не фиксирована заранее, нам придется использовать другие методы.



5

Позиционный код

Пусть $\langle a_1, \ldots, a_n\rangle$ — последовательность натуральных чисел (кортеж). Выберем $b > a_i$, для всех i. Пусть

$$a = a_n b^{n-1} + a_{n-1} b^{n-2} + \cdots + a_1 b^0.$$

Иначе говоря, $a_1, \ldots, a_n$ — это цифры в записи числа a в позиционной системе счисления с основанием b, и значит, кортеж $\langle a_1, \ldots, a_n\rangle$ однозначно восстанавливается по тройке (a, b, n). Тройку (a, b, n) будем называть *позиционным кодом* кортежа $\langle a_1, \ldots, a_n\rangle$. Позиционным кодом пустого кортежа будет тройка $(0, b, 0)$. Заметим, что не каждая тройка является позиционным кодом какого-нибудь кортежа. Легко установить диофантовость отношения «*быть позиционным кодом*»:

$$\mathrm{Code}(a, b, c) \quad \Longleftrightarrow \quad \begin{cases} b \geqslant 2, \\ a < b^c. \end{cases}$$

♦ **B1.** *Докажите диофантовость множества четверок (a, b, k, c), таких что c равен k-му члену последовательности, заданной a и b.*

♦ **B2.** *а) Закодируйте последовательность $c_i = \binom{i}{n} = C_i^n$ и докажите, что множество троек (c, m, n), таких что $c = \binom{m}{n} = C_m^n$, диофантово.*
б) Докажите, что

$$m! = \lim_{n\to\infty} \frac{n^m}{\binom{n}{m}}$$

и что множество чисел, являющихся факториалами, диофантово.

♦ **B3.** *Докажите диофантовость множества простых чисел.*

♦ **B4.** *Докажите, что уравнение $D(a, x_1, \ldots, x_m) = 0$ имеет решение в неизвестных $x_1, \ldots, x_m$ тогда и только тогда, когда имеет решение в неизвестных $x_0, x_1, \ldots, x_m$ уравнение*

$$a = (x_0 + 1)(1 - D(x_0, x_1, \ldots, x_m)^2) - 1.$$

♦ **B5.** *Докажите, что существует многочлен с целыми коэффициентами, положительные значения которого суть все простые числа.*

Далее мы установим возможность объединять две последовательности в одну, сравнивать их поэлементно, проверять, кодируют ли две тройки чисел одну и ту же последовательность, — и все это при помощи решения соответствующего диофантова уравнения.

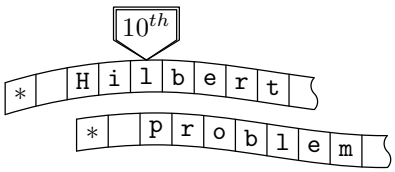
♦ **B6.** (**теорема Куммера**) *а) Докажите, что число k, такое что $n!$ делится на p^k, но не делится на p^{k+1}, равно*

$$\left[\frac{n}{p}\right] + \left[\frac{n}{p^2}\right] + \left[\frac{n}{p^3}\right] + \ldots$$

б) Докажите, что число l, такое что $\binom{m+n}{n}$ делится на p^l, но не делится на p^{l+1}, равно количеству переносов из разряда в разряд при сложении чисел m и n в p-ичной системе счисления.

♦ **B7.** *Установите диофантовость множества троек (a_1, a_2, p), где p — простое, а пары (a_1, p) и (a_2, p) задают последовательности, такие что k-й элемент первой не превосходит k-го элемента второй для любого k.*
Подсказка. *Что будет, если сложить a_1 и $a_2 - a_1$?*

♦ **B8.** *Установите диофантовость множества четверок (a, p, n, e), где p — простое, и тройка (a, p, n) задает последовательность, каждый элемент которой не превосходит числа e.*



♦ **B9.** *Докажите, что если* (a_1, b_1, n) *и* (a_2, b_2, n) *кодируют одну и ту же последовательность, причем* $b_1 < b_2$, *то* $a_1 \equiv a_2 \pmod{b_2 - b_1}$.

♦ **B10.** *Докажите, что если в условиях предыдущей задачи* $b_1^n < b_2 - b_1$, *то число* a_1 *однозначно определяется по числам* $a_2,\ b_1,\ b_2,\ n$.

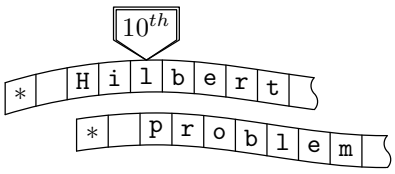
♦ **B11.** *Установите, диофантовость множества* (a_1, b_1, a_2, b_2), *таких что* (a_1, b_1) *и* (a_2, b_2) *кодируют одну и ту же последовательность.*

♦ **B12.** *Докажите диофантовость множества четверок* (a_1, b_1, a_2, b_2), *таких что пары* (a_1, b_1) *и* (a_2, b_2) *кодируют последовательности, причем* k*-й член первой не больше* k*-го члена второй для любого* k.

♦ **B13.** *Докажите диофантовость множества четверок* (a, b, n, e), *где тройка* (a, b, n) *задает последовательность, все члены которой не превосходят числа* e.

♦ **B14.** *Докажите диофантовость множества восьмерок* $(A, B, a_1, b_1, n_1, a_2, b_2, n_2)$ *таких, что пара* (A, B) *кодирует последовательность, полученную из последовательности, которую кодирует тройка* (a_1, b_1, n_1), *дописыванием после нее последовательности, которую кодирует тройка* (a_2, b_2, n_2).

♦ **B15.** *Покажите, как по кодам последвательностей* $p_1, \ldots, p_n$ *и* $q_1, \ldots, q_m$ *закодировать* p_1+q_1, p_1+q_2, $\ldots$, $p_1 + q_m$, $p_2 + q_1$, $p_2 + q_2$, $\ldots$, $p_2 + q_m, \ldots,\ p_n + q_m$ *и* $p_1 \cdot q_1,\ p_1 \cdot q_2, \ldots,\ p_1 \cdot q_m,\ p_2 \cdot q_1,\ p_2 \cdot q_2, \ldots$, $p_2 \cdot q_m, \ldots,\ p_n \cdot q_m$.



7

Десятая проблема Гильберта

**Ю. Матиясевич, Я. Абрамов, А. Белов-Канель,
И. Иванов-Погодаев, А. Малистов, И. Нетай**

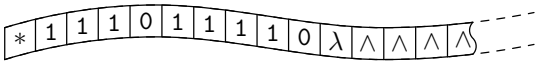
C. Машины Тьюринга

Развитая теоретико-числовая техника дачт возможность приступить к доказательству отсутствия универсального *способа* решения диофантовых уравнений. Но сначала надо определиться, что мы будем понимать под этим «способом». Для этого необходимо уточнить, в каком случае мы считаем, что обладаем *методом* для решения какого-либо типа задач.

Основная идея любого такого уточнения состоит в следующем. Мы можем считать, что имеем метод для решения какого-либо типа задач, если мы можем находить решение любой задачи этого типа без приложения творческих усилий, как говорят, механически, с помощью компьютера.

С компьютером мы сталкиваемся не только в реальной жизни, но и в математике. Наша цель – формализовать понятие метода с помощью абстрактной машины, позволяющей решать все задачи данного типа автоматически.

Теперь нам предстоит разобраться, как устроена *машина Тьюринга*. Машина имеет память в виде ленты, разделенной на клетки. Лента имеет один конец, будем считать, что левый. Вправо лента является потенциально бесконечной. То есть, в отличие от реальных машин, машина Тьюринга всегда обладает достаточным количеством памяти. В то же время, любое конкретное вычисление использует лишь конечный отрезок ленты.



Каждая клетка может быть либо пустой, либо содержать один символ из некоторого конечного множества символов $A = \{a_1, a_2, \dots a_w\}$, называемого *алфавитом*. Разные машины имеют, вообще говоря, разные алфавиты. Один символ играет выделенную роль — им всегда будет помечена самая левая клетка и только она. В качестве такого маркера мы будем использовать символ «*». Кроме того, удобно использовать специальный символ для обозначения пустой клетки, по традиции мы будем использовать для этой цели букву $\wedge$.

Машина снабжена специальным считывающим и пишущим устройством — *головкой*, которая в каждый момент дискретного времени обозревает одну из клеток ленты. Головка может перемещаться вдоль ленты налево и направо. В каждый момент времени головка находится в одном из конечного числа состояний. Будем обозначать эти состояния $q_1, \dots, q_v$. Вообще, «состояние машины» будет значить то же самое, что и «состояние головки». Одно из состояний объявляется *начальным*, у нас это всегда будет q_1. Кроме того, одно или несколько состояний объявляются *заключительными*.

Очередное действие машины полностью определяется состоянием, в котором она находится, и символом, который «видит» головка. За один шаг машина может изменить символ в клетке, переместить головку влево или вправо и перейти в другое состояние. При этом, все эти действия (изменить символ в клетке, переместить головку, перейти в другое состояние) машина делает *в зависимости* от текущего состояния головки. То есть, действия машины полностью определяются набором инструкций вида

$$\text{«состояние } q_i + \text{символ } a_j \to \texttt{Left}(\texttt{Right}, \texttt{Stop}) + q_k + a_\ell\text{»}.$$

Такая инструкция означает следующее «если головка находится в состоянии q_i над клеткой в которой записан символ a_j, то машина производит следующие действия: 1) символ a_j меняется на a_ℓ, 2)

головка сдвигается налево (направо или остается на месте), 3) состояние головки меняется на q_k». Такие инструкции должны быть выписаны для всевозможных сочетаний символа a_j и состояния машины q_i. Весь набор таких инструкций называется *программой* машины.

Все машины, которые мы будем строить, будут иметь один и тот же алфавит $\{*, 0, 1, 2, 3, \lambda\}$.

Кроме того, будет два заключительных состояния q_2 и q_3 и мы будем интерпретировать q_2 как ответ «ДА», а q_3 как ответ «НЕТ». Клетки, содержащие символ λ будут играть роль дублеров пустых клеток, а именно: справа от клетки с символом «λ» может находиться только клетка с символом «λ» либо пустая клетка; для любого состояния q_i, у инструкций с левыми частями $q_i\lambda$ и $q_j\wedge$ будут совпадать правые части.

Пример простейшей машины. Машина LEFT с инструкциями

$$\begin{aligned}
q_1* &\to *Sq_2\\
q_10 &\to 0Lq_2\\
q_11 &\to 1Lq_2\\
q_12 &\to 2Lq_2\\
q_13 &\to 3Lq_2\\
q_1\lambda &\to \lambda Lq_2\\
q_1\wedge &\to \lambda Lq_2
\end{aligned}$$

сдвигает головку на одну клетку влево, если конечно, головка не находилась на самой левой клетке, помеченной символом «*».

Машина WRITE(0) с инструкциями

$$\begin{aligned}
q_1* &\to *Sq_2\\
q_10 &\to 0Sq_2\\
q_11 &\to 0Sq_2\\
q_12 &\to 0Sq_2\\
q_13 &\to 0Sq_2\\
q_1\lambda &\to 0Sq_2\\
q_1\wedge &\to 0Sq_2
\end{aligned}$$

вписывает в клетку, где находится головка, символ «0», если, конечно, это не самая левая клетка с маркером «*». Ясно, что аналогично можно построить машины WRITE(1), WRITE(2), WRITE(3) инструкции которых получаются заменой всюду в правых частях символа «0» на «1», «2» и «3» соответственно.

Машина READ(0) с инструкциями

$$\begin{aligned}
q_1* &\to *Sq_3\\
q_10 &\to 0Sq_2\\
q_11 &\to 1Sq_3\\
q_12 &\to 2Sq_3\\
q_13 &\to 3Sq_3\\
q_1\lambda &\to \lambda Sq_3\\
q_1\wedge &\to \lambda Sq_3
\end{aligned}$$

распознает, записан ли в клетке, где находится головка, символ «0» или нет. В зависимости от этого она заканчивает работу в состоянии q_2 или q_3. Аналогично можно построить машины READ(1), READ(2), READ(3), READ(*).

♦ **C0.** *a) Постройте машину Тьюринга* STOP, *которая не сдвигая головки сразу переходит в заключительное состояние.*
b) Постройте машину Тьюринга READNOT(x), *которая распознает отстутствие символа «x» в клетке.*

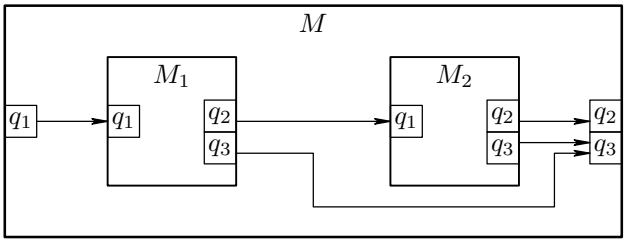
Два способа композиции машин.

Первый способ построения по двум машинам Тьюринга M_1 м M_2 новой машины M состоит в следующем:

1. Во всех инструкциях машины M_1 заключительное состояние q_2 заменяется на q_{v+1}, где v – количество состояний машины M_1. (Заключительные состояния могут стоять только в правых частях инструкций);

2. Во всех инструкциях машины M_2 каждое незаключительное состояние q_i заменяется на q_{v+i}. (В частности, q_1 заменяется на q_{v+1})

3. Модифицированные инструкции обеих машин объединяются и образуют систему инструкций новой машины M.



Действие машины M, построенной описанным выше способом, состоит в последовательном выполнении действий машин M_1 и M_2 при условии, что машина M_1 заканчивает работу в состоянии q_2. Для обозначения машины M мы будем испоьзовать одну из трех форм записи

$$M_1; M_2 \qquad M_1 \textbf{ and } M_2 \qquad \text{или} \qquad \textbf{if } M_1 \textbf{ then } M_2.$$

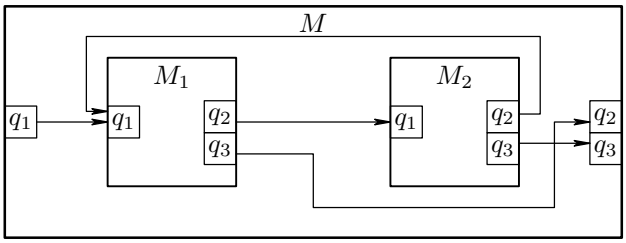
Второй способ построения по двум машинам Тьюринга M_1 и M_2 новой машины M позволяет конструировать циклы типа `FOR i=1 TO N` или `WHILE` с которыми вы встречались при программировании. Такие циклы встречаются и в математике. Например, чтобы вычислить функцию $f(n) = 2^{2^n}$ нужно число 2 последовательно возвести в квадрат n раз.

Этот способ состоит в следующем:

1. Во всех инструкциях машины M_1 заключительное состояние q_2 заменяется на q_{v+1}, где v – количество состояний машины M_1, а заключительное состояние q_3 заменяется на q_2;

2. Во всех инструкциях машины M_2 каждое незаключительное состояние q_j заменяется на q_{v+i}, а заключительное состояние q_2 заменяется на q_1;

3. Модифицированные инструкции обеих машин объединяются и образуют систему инструкций новой машины M.



Действие машины M, построенной описанным выше способом, состоит в поочередном выполнении действий машин M_1 и M_2 до тех пор, пока одна из машин не перейдет в состояние q_3. Построенную таким образом машину мы будем обозначать посредством

$$\textbf{while } M_1 \textbf{ do } M_2 \textbf{ od}.$$

Введчнные нами обозначения выглядят как некоторый примитивный язык программирования. (Следует отметить, однако, что каждая «программа» является на самом деле обозначением конкретной машины Тьюринга.) Однако, этот язык не столь примитивен, как кажется. Он позволяет эмулировать любой компьютер, с которым вы имели дело. Знаменитый *тезис Черча* гласит, что любой алгоритм

может быть реализован на машине Тьюринга. Это понятие адекватно формализует понятие механической работы. В дальнейшем под существованием алгоритмов мы понимаем существование машины Тьюринга. То есть, эти понятия отожествляются.

Машина

$$\texttt{STAR} = \textbf{while } \texttt{READNOT}(*) \textbf{ do } \texttt{LEFT} \textbf{ od}$$

устанавливает головку на левый конец ленты (помеченный символом «*»).

Машина

$$\texttt{VACANT} = \texttt{STAR};\ \textbf{while } \texttt{READNOT}(\lambda) \textbf{ do } \texttt{RIGHT} \textbf{ od}$$

устанавливает головку на самую левую пустую клетку или ее дублера - клетку с символом «λ»).

♦ **C1.** *a) Постройте машину* JUMP*, передвигающую головку до ближайшей клетки с символом «0».*
b) Постройте серию машин Тьюринга FIND(k)*, устанавливающих головку на k-ый от левого края символ «0».*

На ленте машины Тьюринга можно запоминать наборы (кортежи) чисел: для записи кортежа $(a_1, \ldots, a_n)$ используются символы 1, разделенные символами 0. Например, кортеж $(3, 1, 2, 0, 2)$ записывается как $*0111010110011\lambda\ldots$. Таким образом, машина Тьюринга может получать на входе какой-то кортеж и осуществлять его преобразование.

♦ **C2.** *Постройте машины Тьюринга, преобразующие кортеж $(a_1, \ldots, a_n)$ в кортеж:*

a) $(a_1, \ldots, a_n, 0)$;
b) $(a_1, \ldots, a_n + 1)$;

♦ **C3.** *a) Постройте машину Тьюринга, преобразующую кортеж $(a_1, \ldots, a_n)$ в кортеж $(a_1, \ldots, a_n - 1)$, если $a_n > 0$, или останавливающуюся в состоянии q_3, если $a_n = 0$.*
b) Постройте машину Тьюринга, усекающую кортеж $(a_1, \ldots, a_n)$ до кортежа $(a_1, \ldots, a_{n-1})$.

♦ **C4.** *Постройте машины Тьюринга, преобразующие кортеж $(a_1, \ldots, a_n)$ в кортеж:*
a) $(a_1, \ldots, a_n, a_k + a_l)$, для заданных $1 \leqslant k, l \leqslant n$;
b) $(a_1, \ldots, a_n, a_k \times a_l)$, для заданных $1 \leqslant k, l \leqslant n$;

♦ **C5.** *Постройте машину* NOTGREATER(k, l)*, сравнивающую элементы a_k и a_l кортежа $(a_1, \ldots, a_n)$ и останавливающуюся в состоянии q_2 или q_3, в соответствии с тем, которое из неравенств $a_k \leqslant a_l$ или $a_k > a_l$ имеет место.*

♦ **C6.** *Постройте машины Тьюринга, преобразующие кортеж $(a_1, \ldots, a_n)$ в кортеж:*
a) $(a_1, \ldots, a_n, b, c)$, где (b, c) пара чисел, следующая за парой (a_{n-1}, a_n) в канторовой нумерации;
b) $(a_1, \ldots, a_n, b, c)$, где (b, c) пара чисел, имеющая номер a_n в канторовой нумерации;

♦ **C7.** *Пусть имеется уравнение $D(a_1, \ldots, a_n, x_1, \ldots x_{m+1}) = 0$. Постройте машину Тьюринга, которая по кортежу $(a_1, \ldots, a_n, y_0)$ проверяет, является ли y_0 канторовым номером кортежа $(x_1, \ldots, x_{m+1})$, дающего решение уравнения.*

♦ **C8.** *Пусть имеется уравнение $D(a_1, \ldots, a_n, x_1, \ldots x_{m+1}) = 0$. Постройте машину Тьюринга, получающую на входе кортеж $(a_1, \ldots, a_n)$ и заканчивающую свою работу в том и только том случае, когда уравнение $D = 0$ разрешимо относительно $x_1, \ldots, x_{m+1}$.*

Будем называть множество ω, состоящее из упорядоченных наборов из n натуральных чисел (n-ок) *полуразрешимым по Тьюрингу*, если существует машина Тьюринга M, которая начав работу над каноническим изображением кортежа $(a_1, \ldots, a_n)$ в состоянии q_1 и головкой, установленной в левом конце ленты, окончит свою работу в том и только том случае, когда $(a_1, \ldots, a_n) \in \omega$. (Однако может не быть никакого способа оценить время работы и распознать ситуацию, когда $(a_1, \ldots, a_n) \notin \omega$).

Итак, из результата задачи C7 следует, что всякое диофантово множество является полуразрешимым по Тьюрингу. Целью следующих задач будет доказательство обратного, то есть того что каждое полуразрешимое по Тьюрингу множество является диофантовым.

Пусть дана машина Тьюринга M, полуразрешающая некоторое множество ω, состоящее из n натуральных чисел. Пусть $\{\alpha_1, \ldots, \alpha_w\}$ — алфавит этой машины. На каждом шаге работы машины M заполнен только конечный отрезок ленты, скажем, длины l. Мы можем представить его кортежем $(s_1, s_2, \ldots, s_m, \ldots, s_{l-1}, s_l)$, где s_i – номер символа записанного в ячейке i.

Текущее состояние q_i машины и положение головки можно представлять кортежем такой же длины $(0, \ldots, 0, i, 0, \ldots, 0)$, все элементы которого, кроме одного, равны нулю, единственный ненулевой элемент указывает положение головки и равен номеру состояния машины.

Тройку, состоящую из текущего содержимого ленты, состояния машины и положения головки, будем называть *конфигурацией*. Конфигурация определяется двумя кортежами $(s_1, s_2, \ldots, s_m, \ldots, s_{l-1}, s_l)$ и $(0, \ldots, 0, i, 0, \ldots, 0)$. Для представления кортежей мы будем использовать позиционную систему кодирования с каким либо фиксированным основанием $\beta > v, w$. *Кодом конфигурации* будем называть пару (p, t), где p и t – шифры по основанию β указанных кортежей.

Итак, необходимо построить диофантово уравнение $D(p, t, x_1, \ldots, x_m) = 0$, такое что, если (p, t) — код некоторой конфигурации, то уравнение разрешимо относительно $x_1, \ldots, x_m$ тогда и только тогда, когда машина M начав работу в этой конфигурации, остановится через конечное число шагов. Вопрос о разрешимости уравнения в случае, когда (p, t) не является кодом никакой конфигурации, нас не интересует.

♦ **C9.** *Пусть из конфигурации с кодом (p, t) машина M непосредственно переходит в конфигурацию с кодом* $[\mathrm{NextP}\,(p, t), \mathrm{NextT}\,(p, t)]$. *Докажите, что функции* NextP *и* NextT *диофантовы.*

♦ **C10.** *Пусть из конфигурации с кодом (p, t) машина M за k шагов переходит в конфигурацию с кодом* $[\mathrm{AfterP}\,(p, t, k), \mathrm{AfterT}\,(p, t, k)]$. *Докажите, что функции* AfterP *и* AfterT *диофантовы.*

♦ **C11.** *Постройте диофантово уравнение с параметрами $a_1, \ldots, a_n$, разрешимое в тех и только тех случаях, когда данная машина Тьюринга M, начав работу на кортеже $a_1, \ldots, a_n$, останавливается.*

D. Универсальная машина Тьюринга

В информатике (Computer Science) вы встречались с операционными системами (Windows, Dos, Unix), позволяющими запускать произвольные исполняющие файлы (то есть работать с произвольными алгоритмами). При этом, операционная система сама является программой. В математике аналогичным понятием является понятие *универсального алгоритма.*

До сих пор мы конструировали свою машину Тьюринга для каждого алгоритма. Но можно поступить и по-другому: придумать язык, на котором описывается, что нужно сделать (алгоритм), и записать эту информацию с помощью какой-то кодировки на ленте машины Тьюринга. Работающая с этой лентой машина «разархивирует» с ленты алгоритм и реализует его. При этом, у данной машины своя фиксированная программа, которая не меняется от того, какой алгоритм записан на ленте. Такая машина называется *универсальной машиной Тьюринга.*

Ясно, что мы можем пронумеровать все алгоритмы (машины Тьюринга) и говорить о алгоритме (машине) с номером n.

♦ **D1. Кодировка**. *Придумайте способ записи алгоритма (машины Тьюринга) на ленте.*

♦ **D2.** *Докажите, что существует машина Тьюринга, считывающая с ленты пару чисел (m, n) и выполняющая алгоритм с номером n, получающий на входе число m.*

♦ **D3. Самоприменение алгоритма.** *Докажите, что существует такая машина Тьюринга U которая на числе n работает также как n-ый алгоритм на числе n.*

♦ **D4. Проблема останова.** a) *Предположим, что существует универсальная машина Тьюринга, $U(n, m)$ проверяющая, останавливается ли n-ая машина Тьюринга на числе m. Докажите, что тогда существует алгоритм $V(n)$, устанавливающий, останавливается или работает бесконечно n-ый алгоритм на числе n. Докажите также, что существует алгоритм $T(n)$, который останавливается, если $V(n)$ работает бесконечно, и работает бесконечно если $V(n)$ останавливается.*
b) *Пусть k – номер алгоритма T. Останавливается ли $T(k)$?*

E. Универсальные диофантовы уравнения

Рассмотрим семейство уравнений $U(a_1, \ldots, a_n, k_1, \ldots k_\ell, y_1, \ldots, y_v) = 0$. Будем выделять в нем две группы параметров: параметры-элементы $a_1, \ldots, a_n$ и параметры-коды $k_1, \ldots, k_\ell$. Будем называть такое уравнение универсальным, если для любого диофантового уравнения с n параметрами: $D(a_1, \ldots, a_n, x_1, \ldots, x_m) = 0$ можно так выбрать значения параметров кодов $k_1(D), \ldots, k_u(D)$, что уравнение $D = 0$ разрешимо относительно $x_1, \ldots x_n$ при тех и только тех наборах параметров $a_1, \ldots, a_n$, при которых разрешимо уравнение $U(a_1, \ldots, a_n, k_1(D), \ldots, k_\ell(D), y_1, \ldots, y_v) = 0$ относительно $y_1, \ldots y_v$. Можно сказать, каждое универсальное диофантово уравнение порождает кодирование диофантовых множеств *данной размерности*, и для данного уравнения $D = 0$ этим кодом можно считать кортеж $[k_1(D), \ldots, k_\ell(D)]$.

♦ **E1.** *Покажите, что ценой увеличения количества неизвестных любое универсальное уравнение может быть преобразовано в уравнение с тем же количеством параметров-элементов, но всего с одним параметром-кодом.*

♦ **E2.** *Покажите, что если существует универсальное диофантово уравнение, кодирующее одномерные диофантовы множества (то есть, для $n = 1$), то существует константа m, такая, что для любого n имеется универсальное диофантово уравнение с $u = 1$ и $v = m$.*

♦ **E3.** *Придумайте способ кодировки уравнения и потенциального решения с помощью нескольких натуральных чисел, такой, что функция проверки, является ли данный кортеж решением данного уравнения, была диофантовой.*

♦ **E4.** *Постройте универсальное диофантово уравнение.*

♦ **E5.** *Постройте диофантово множество M с недиофантовым дополнением (то есть такое, что дополнение $\overline{M}$ до множества $\mathbb{N}$ не является диофантовым).*

♦ **E6.** *Докажите, что множество кодов диофантовых уравнений без параметров, имеющих решения, является множеством из E5.*

F. Заключительная задача

♦ **F1.** *Докажите, что нельзя построить машину Тьюринга, которая, начав работу над изображением числа k, останавливалась бы через конечное число шагов в состоянии q_2 или q_3 в соответствии с тем, разрешимо или неразрешимо диофантово уравнение с кодом k.*