Теория алгоритмов

В. Е. Плиско

1 Основные понятия теории алгоритмов

Первые примеры *алгоритмов* встречаются уже в средней школе: алгоритм сложения натуральных чисел столбиком, алгоритм умножения двух натуральных чисел, алгоритм деления с остатком, процесс нахождения наибольшего общего делителя двух целых чисел, известный под названием *алгоритма Евклида*. Из других алгоритмов можно указать алгоритм разложения натурального числа на простые множители, извлечения квадратного корня из натурального числа, решения системы линейных уравнений методом Гаусса и т. д. Каждый из этих алгоритмов представляет собой некоторую вычислительную процедуру, выполнение которой не требует изобретательности или сообразительности, а состоит лишь в строгом следовании инструкциям.

Общие черты алгоритмов:

1. *Дискретность.* Алгоритм — это процесс последовательного построения объектов, идущий в дискретном времени таким образом, что в начальный момент задается исходный набор объектов, а в каждый следующий момент из набора объектов, имевшихся в предыдущий момент, по определенному закону (программе) получается новый набор объектов. Выполнение всякого алгоритма состоит из отдельных шагов. Каждый шаг обязательно завершается. Если выполнение алгоритма никогда не закончится, это означает, что алгоритм совершает бесконечное число шагов.

2. *Детерминированность.* Набор объектов, получаемых в какой-то момент, однозначно определяется набором объектов, полученных в предшествующие моменты.

3. *Элементарность шагов.* Закон получения последующего набора объектов из предшествующего должен быть простым и локальным.

4. *Направленность.* Должно быть указано, что́ надо считать результатом алгоритма.

5. *Массовость.* Начальный набор объектов может выбираться из потенциально бесконечного множества — множества *возможных исходных данных* этого алгоритма. При этом исхлдными данными для алгоритма могут быть лишь так называемые *конструктивные объекты*. Интуитивно, конструктивный объект — это такой объект, который построен (сконструирован) из некоторых исходных элементарных неделимых элементов фиксированного точно очерченного конечного множества по фиксированным правилам, так что строение этого объекта может быть полностью описано некоторым текстом на подходящем языке. Если фиксированы набор исходных элементов и правила конструирования объектов из них, говорят, что задан *ансамбль клнструктивных объектов*.

Типичным примером ансамбля конструктивных объектов является множество слов в данном конечном *алфавите*. При этом исходными элементами являются *буквы* алфавита, а способ конструирования состоит в построении конечных цепочек букв, т. е. *слов*. В частности, множество натуральных чисел $\mathbf{N} = \{0, 1, 2, \ldots\}$ может рассматриваться как ансамбль конструктивных объектов, ибо натуральные числа можно изображать, например, словами в однобуквенном алфавите $\{|\}$: 0 — *пустое слово*; 1 — слово $|$; 2 — слово $||$ и т. д. Множества целых и рациональных чисел также могут рассматриваться как ансамбли конструтивных объектов.

Всякий ансамбль конструктивных объектов счетен. Более того, его всегда можно эффективно занумеровать, т. е. установить такое взаимно однозначное соответствие между ансамблем конструктивных объектов и множеством натуральных чисел, что по любому объекту можно вычислить его номер, а по любому натуральному числу n можно эффективно построить объект с номером n. Свойство эффективной нумеруемости мы примем за основной признак ансамбля конструктивных объектов. Итак, множество возможных исходных данных любого алгоритма — это некоторый ансамбль конструктивных объектов.

Понятие алгоритма, определяемое требованиями 1–5, не строгое. Это нестрогое понятие алгоритма обычно называется *интуитивным* понятием алгоритма.

При применении данного алгоритма к исходному данному возиожны следующие три варианта. 1) Применение алгоритма к исходному данному завершится через конечное число шагов, и алгоритм выдаст результат. В этом случае говорят, что алгоритм примени́м к исходному данному. 2) Применение алгоритма к исходному данному завершится через конечное число шагов, но алгоритм не выдаст никакого результата. 3) Применение алгоритма к исходному данному никогда не закончится, т. е. алгоритм совершает бесконечное число

шагов. В случаях 2) и 3) говорят, что алгоритм не применим к исходному данному. Множество возможных исходных данных, к которым алгоритм применим, называется *областью применимости* этого алгоритма.

Пусть X — множество возможных исходных данных алгоритма $\mathfrak{A}$, а его результаты принадлежат ансамблю конструктивных объектов Y. Пусть $D \subseteq X$ — область применимости алгоритма $\mathfrak{A}$. Будем говорить, что алгоритм $\mathfrak{A}$ *вычисляет функцию* $f : D \to Y$, которая каждому $x \in D$ сопоставляет результат применения алгоритма $\mathfrak{A}$ к x. *Частичной функцией из* X *в* Y будем называть любую функцию $f : D \to Y$, где $D \subseteq X$. Таким образом, каждый алгоритм с множеством возможных исходных данных X, результаты которого принадлежат ансамблю конструктивных объектов Y, вычисляет некоторую частичную функцию из X в Y.

В дальнейшем мы будем иметь дело с выражениями (частичными именными формами), в которых встречаются обозначения для частичных функций, и которые, следовательно, определены не при всех значениях свободных переменных. В связи с эти мы будем употреблять символ условного равенства $\simeq$ и будем считать, что предложение $s \simeq t$ истинно при тех значениях свободных переменных, при которых оба выражения s и t определены и имеют одинаковые значения или же оба не определены.

Интуитивное понятие алгоритма удовлетворяло математиков до тех пор, пока возникающие в математике проблемы, требующие нахождения алгоритма для решения некоторой совокупности родственных задач (или алгоритмические проблемы), удавалось решить путем указания конкретных алгоритмов. Положение изменилось в XX в., когда на первый план выдвинулись алгоритмические проблемы, положительное решение которых было сомнительным. Среди них — так называемая 10-я проблема Гильберта о разрешимости алгебраических уравнений в целых числах. Чтобы доказать несуществование алгоритма, надо иметь точное определение понятия алгоритма. Математические уточнения понятия алгоритма были получены в середине тридцатых годов в работах Гильберта, Гёделя, Чёрча, Клини и Поста.

2 Машина Тьюринга

Пост (1936) и Тьюринг (1937) независимо друг от друга предложили уточнение понятия алгоритма как процесса, который может совершаться подходяще устроенной "машиной". Машины, введенные Постом и Тьюрингом, отличались не очень существенно и в дальнейшем стали называться машинами Тьюринга. Мы рассмотрим вариант машин Тьюринга, близкий к тому, который был предложен Постом.

Машина Тьюринга содержит следующие части:

1. *Конечная лента*, разбитая на конечное число клеток (или ячеек). В процессе работы машины к ленте могут пристраиваться новые ячейки, так что лента может считаться потенциально бесконечной. В каждой ячейке ленты записан (или содержится) один из конечного числа символов, составляющих *внешний алфавит* $A = \{a_0, a_1, \ldots, a_m\}$. Клетки, в которых записан символ a_0, обычно обозначаемый 0, называются пустыми. Все вновь пристраиваемые ячейки являются пустыми.

2. *Внутренняя память* — некоторе устройство, которое в каждый момент находится в одном из конечного числа "состояний", составляющих *внутренний алфавит* $Q = \{q_0, q_1, \ldots, q_n\}$. Состояние q_0 называется *заключительным*.

3. *Управляющая головка* — некоторое устройство, которое может перемещаться вдоль ленты так, что в каждый момент находится в определенной ячейке, или "обозревает"эту ячейку.

4. *Механическое устройство*, которое в зависимости от содержимого обозреваемой ячейки и состояния внутренней памяти может изменить состояние внутренней памяти и при этом либо изменить содержимое обозреваемой ячейки, либо сдвинуть управляющую головку в соседнюю ячейку слевы или в соседнюю ячейку справа. Если управляющая головка находится в крайней ячейке и должна сдвинуться в отсутствующую соседнюю ячейку, то машина пристраивает недостающую пустую ячейку.

Машинным словом машины Тьюринга (или *конфигурацией*) называется слово

$$a_{j_1} a_{j_2} \ldots q a_{j_k} \ldots a_{j_r},$$

содержащее совокупность всех данных о состоянии машины в данный момент: $a_{j_1} a_{j_2} \ldots a_{j_k} \ldots a_{j_r}$ — последовательность букв, записанных в ячейках ленты, q_i — внутреннее состояние, k — номер обозреваемой ячейки.

Работа машины состоит в последовательном переходе от одной конфигурации к другой в результате выполнения одного из следующих действий:

1. Имея внутреннее состояние q_i и обозревая ячейку, в которой записан символ a_j, машина переводит внутреннюю память в состояние q_s и одновременно заменяет символ в обозреваемой ячейке на a_r. Это действие выражается *командой*

$$q_i a_j \to q_s a_r.$$

2. Имея внутреннее состояние q_i и обозревая ячейку, в которой записан символ a_j, машина переводит внутреннюю память в состояние q_s и одновременно передвигает управляющую головку в соседнюю ячейку справа. Это действие выражается командой

$$q_i a_j \to q_s R.$$

3. Имея внутреннее состояние q_i и обозревая ячейку, в которой записан символ a_j, машина переводит внутреннюю память в состояние q_s и одновременно передвигает управляющую головку в соседнюю ячейку слева. Это действие выражается командой

$$q_i a_j \to q_s L.$$

Совокупность всех команд, которые может выполнять машина, называется ее *программой*. Для каждого незаключительного внутреннего состояния q_i и каждого символа a_j из внешнего алфавита программа должна содержать ровно одну команду, начинающуюся с $q_i a_j$.

Выполнение каждой из команд может быть описано как процесс построения конфигурации $\mathfrak{m}'$ по заданной конфигурации $\mathfrak{m}$.

Пусть Σ, Δ — некоторые алфавиты, не содержащие символ a_0, F — частичная n-местная функция, определенная на некотором подмножестве множества Σ^* всех слов в алфавите Σ и со значениями в множестве Δ^*. Будем говорить, что данная машина Тьюринга *вычисляет функцию* F, если ее внешний алфавит включает алфавиты Σ, Δ и, каковы бы ни были слова $\mathfrak{r}_1, \ldots, \mathfrak{r}_n$ в алфавите Σ, если значение $F(\mathfrak{r}_1, \ldots, \mathfrak{r}_n)$ определено, то эта машина перерабатывает конфигурацию $q_1 0 \mathfrak{r}_1 0 \ldots 0 \mathfrak{r}_n$ в конфигурацию $0 \ldots 0 q_0 0 \mathfrak{a} 0 \ldots 0$, причем $F(\mathfrak{r}_1, \ldots, \mathfrak{r}_n) = \mathfrak{a}$, а если значение $F(\mathfrak{r}_1, \ldots, \mathfrak{r}_n)$ не определено, то в процессе переработки конфигурации $q_1 0 \mathfrak{r}_1 0 \ldots 0 \mathfrak{r}_n$ машина никогда не придет в заключительное состояние. Функция F называется *вычислимой по Тьюрингу*, если существует машина Тьюринга, которая вычисляет функцию F.

Натуральные числа изображаются как слова в однобуквенном алфавите $\{|\}$, так что можно считать опрделенным понятие вычислимой по Тьюрингу частичной числовой функции.

Задачи

Построить машины Тьюринга, вычисляющие следующие функции:

1) $s(x) = x + 1$; 2) $o(x) = 0$; 3) $I_m^n(x_1, \ldots, x_n) = x_m$ $(1 \le m \le n)$; 4) $x + y$;

5) $p(x) = \begin{cases} x - 1, & \text{если } x > 0; \\ 0, & \text{если } x = 0; \end{cases}$ 6) $\operatorname{sg}(x) = \begin{cases} 1, & \text{если } x > 0; \\ 0, & \text{если } x = 0; \end{cases}$ 7) $\overline{\operatorname{sg}}(x) = \begin{cases} 0, & \text{если } x > 0; \\ 1, & \text{если } x = 0; \end{cases}$

8) $d(x) = \begin{cases} x - y, & \text{если } x \ge y; \\ 0, & \text{если } x < y; \end{cases}$ 9) $x - y$; 10) $\frac{x}{2}$; 11) $\left[\frac{x}{2}\right]$.

3 Частично-рекурсивные функции

С помощью машин Тьюринга мы получили некоторое уточнение понятия вычислимой частичной функции из множества Σ^* в множество Δ^* для произвольных алфавитов Σ и Δ. В частности, мы имеем понятие вычислимой по Тьюрингу числовой функции, т. е. частичной многоместной функции из $\mathbf{N}$ в $\mathbf{N}$. Посредством $\mathcal{C}^{(n)}$ будем обозначать класс всех вычислимых по Тьюрингу частичных функций из $\mathbf{N}^n$ в $\mathbf{N}$. Вместо $\mathcal{C}^{(1)}$ будем писать просто $\mathcal{C}$.

Насколько хорошо понятие функции, вычислимой по Тьюрингу, соответствует интуитивному понятию вычислимой функции? В настоящее время известно много других формальных описаний вычислимых функций. Оказалось, что все они задают один и тот же класс n-местных вычислимых частичных числовых функций, совпадающий с $\mathcal{C}^{(n)}$. Кроме того, до сих пор никому не удалось привести пример вычислимой функции, которая не была бы вычислима по Тьюрингу. Это, а также многие другие наблюдения, позволили сформулировать следующий тезис, обычно называемый *тезисом Чёрча*: для любой вычислимой частичной функции из Σ^* в Δ^* существует машина Тьюринга, которая вычисляет эту функцию. В частности, интуитивно понимаемый класс n-местных вычислимых частичных числовых функций совпадает с $\mathcal{C}^{(n)}$. Заметим, что тезис Чёрча невозможно ни доказать, ни опровергнуть, так как понятие вычислимой функции не имеет строгого математического определения. Таким образом, тезис Чёрча — это своего рода естественно-научная гипотеза, подтверждаемая математическим опытом.

Классы $\mathcal{C}^{(n)}$ $(n = 1, 2, \ldots)$ допускают и другое, внутреннее описание, которое полезно знать всякому, кто изучает теорию алгоритмов.

Следующие тотальные (т. е. всюду определенные) числовые функции будем называть простейшими или *базисными*: $s(x) = x + 1$; $o(x) = 0$; $I_m^n(x_1, \ldots, x_n) = x_m$ $(1 \le m \le n)$. Очевидно, что все базисные функции вычислимы. Более того, решив задачи 1) -3) из раздела 2, мы доказали, что все они вычислимы по Тьюрингу.

Будем говорить, что n-местная ($n \geq 1$) частичная функция h получается с помощью *операции подстановки* из k-местной функции f и n-местных функций $g_1, \ldots, g_k$, если для любых $x_1, \ldots, x_n \in \mathbf{N}$ имеет место условное равенство

$$h(x_1, \ldots, x_n) \simeq f(g_1(x_1, \ldots, x_n), \ldots, g_k(x_1, \ldots, x_n)).$$

Например, функция $f(x) = 1$ получается операцией подстановки из функций $s(x)$ и $o(x)$. Функция

$$f(x, y, z) = z + 1$$

получается подстановкой из функций $s(x)$ и $I_3^3(x, y, z)$. Очевидно, что если функции $f, g_1, \ldots, g_k$ всюду определены, то h — тотальная функция. Кроме того, если функции $f, g_1, \ldots, g_k$ вычислимы, то и функция h вычислима.

Скажем, что $(n+1)$-местная ($n \geq 1$) частичная функция h получается с помощью *операции рекурсии* из n-местной функции f и $(n+2)$-местной функции g, если для любых $x_1, \ldots, x_n, y \in \mathbf{N}$ выполняются следующие условные равенства:

$$h(x_1, \ldots, x_n, 0) \simeq f(x_1, \ldots, x_n);$$

$$h(x_1, \ldots, x_n, y+1) \simeq g(x_1, \ldots, x_n, y, h(x_1, \ldots, x_n, y)).$$

Для $n = 0$ операция рекурсии определяется следующим образом. Одноместная частичная функция h получается операцией рекурсии из двуместной частичной функции g и натурального числа a, если для любого натурального y выполняются условные равенства

$$h(0) = a;$$

$$h(y+1) \simeq g(y, h(y)).$$

Например, функция $h(x, y) = x + y$ получается рекурсией из функций $I_1^1(x)$ и $g(x, y, z) = z + 1$. Очевидно, что если функция h получается рекурсией из вычислимых функции f и g, то и функция h вычислима. Кроме того, если функции f и g всюду определены, то h — тотальная функция.

Функция называется *примитивно репурсивной*, если она может быть получена из базисных функций с помощью конечного числа применений операций подстановки и рекурсии. Иными словами, функция f является примитивно рекурсивной, если существует конечная последовательность функций $f_0, \ldots, f_n$, в которой каждая функция f_i ($i \leq n$) либо является базисной, либо получается из каких-нибудь предшествующих функций с помощью подстановки или рекурсии, и при этом f_n есть функция f. Как показывают приведенные выше примеры, функции $f(x) = 1$, $g(x, y) = x + y$ являются примитивно рекурсивными. Из отмеченных выше свойств операций подстановки и рекурсии немедленно следует, что всякая примитивно рекурсивная функция вычислима и является тотальной.

Будем говорить, что n-местная ($n \geq 1$) частичная функция g получается с помощью *операции минимизации* (или *μ-оператора*) из $(n+1)$-местной частичной функции f, если для любых $x_1, \ldots, x_n, y \in \mathbf{N}$ значение $g(x_1, \ldots, x_n)$ определено и равно y тогда и только тогда, когда для любого $z < y$ значение $f(x_1, \ldots, x_n, z)$ определено и не равно 0, а $f(x_1, \ldots, x_n, y) = 0$. В этом случае пишут

$$g(x_1, \ldots, x_n) \simeq \mu y[f(x_1, \ldots, x_n, y) = 0].$$

Заметим, что если функция g получается операцией минимизации из вычислимой функции f, то g также вычислима, однако она может оказаться не тотальной, даже если функция f всюду определена.

Функция называется *частично-рекурсивной*, если она может быть получена из базисных функций с помощью конечного числа применений операций подстановки, рекурсии и минимизации. Иными словами, функция f является частично-рекурсивной, если существует конечная последовательность функций, заканчивающаяся функцией f, в которой каждая функция либо является базисной, либо получается из предыдущих функций с помощью операций подстановки, рекурсии или минимизации. Тотальные частично-рекурсивные функции называются *общерекурсивными*.

Теорема 3.1 *Класс всех n-местных частично-рекурсивных функций совпадает с классом $\mathcal{C}^{(n)}$.*

Эта теорема доказывается довольно рутинным способом. Мы этого делать не будем. Заметим лишь, что в силу тезиса Чёрча понятие частично-рекурсивной функции может рассматриваться как еще одно уточнение понятия вычислимой числовой функции.

Задачи

1. Доказать, что если функция $f(x_1, \ldots, x_n)$ примитивно рекурсивна, то примитивно рекурсивна также функция $g(x_1, \ldots, x_n) = f(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$, где σ — произвольная тотальная функция из $\{1, \ldots, n\}$ в $\{1, \ldots, n\}$.

2. Доказать, что если функция $f(x_1, \ldots, x_n)$ примитивно рекурсивна, то примитивно рекурсивна также функция $h(x_1, \ldots, x_n, x_{n+1}) = f(x_1, \ldots, x_n)$.

3. Доказать, что следующие функции примитивно рекурсивны:

   а) $f(x, y) = x \cdot y$; б) $f(x, y) = x^y$ (здесь $0^0 = 1$); в) $f(x) = x!$ (здесь $0! = 1$);

   г) $\mathrm{sg}(x) = \begin{cases} 1, & \text{если } x > 0; \\ 0, & \text{если } x = 0; \end{cases}$ д) $\overline{\mathrm{sg}}(x) = \begin{cases} 0, & \text{если } x > 0; \\ 1, & \text{если } x = 0; \end{cases}$ е) $p(x) = \begin{cases} x - 1, & \text{если } x > 0; \\ 0, & \text{если } x = 0; \end{cases}$

   ж) $d(x) = \begin{cases} x - y, & \text{если } x \geq y; \\ 0, & \text{если } x < y; \end{cases}$ з) $|x - y|$; и) $\max(x, y)$; к) $\min(x, y)$.

4. Доказать, что следующие функции частично-рекурсивны:

   а) $x - y$; б) $\frac{x}{y}$; в) $\sqrt[y]{x}$; г) $\frac{x}{2}$; д) $\left[\frac{x}{2}\right]$.

4 Машина с неограниченными регистрами

Рассмотрим еще одно уточнение интуитивного понятия алгоритма, работающего с натуральными числами.

Машина с неограниченными регистрами (МНР) — это своего рода идеализированный компьютер. МНР имеет бесконечно много регистров $R_1, R_2, R_3, \ldots$, в каждом из которых в любой момент времени записано некоторое натуральное число. Число содержащееся в регистре R_n, мы будем обозначать r_n. Содержимое регистров может меняться при выполнении некоторой команды. Конечный список команд образует программу. Команды бывают следующих четырех типов:

1) *Команда обнуления* имеет вид $Z(n)$ ($n \in \{1, 2, 3, \ldots\}$). Выполнение такой команды заменяет содержимое регистра R_n на 0, не затрагивая другие регистры.

2) *Команда прибавления единицы* имеет вид $S(n)$, где $n \in \{1, 2, 3, \ldots\}$. Выполнение такой команды увеличивает содержимое регистра R_n на 1, не затрагивая другие регистры.

3) *Команда переадресации* имеет вид $T(m, n)$ (здесь $m, n \in \{1, 2, 3, \ldots\}$). Выполнение такой команды заменяет содержимое регистра R_n числом r_m, содержащимся в регистре R_m, не затрагивая другие регистры (включая R_m).

4) *Команда условного перехода* имеет вид $J(m, n, q)$ ($m, n, q \in \{1, 2, 3, \ldots\}$). Выполнение такой команды состоит в следующем. Сравнивается содержимое регистров R_m и R_n. Если $r_m = r_n$, то машина переходит к выполнению q-й команды выполняемой программы; если $r_m \neq r_n$, то машина переходит к выполнению следующей команды. Если требуется выполнение команды с номером, превосходящим число команд в программе, машина прекращает работу.

Команды обнуления, прибавления единицы и переадресации называются *арифметическими*.

Пусть $n \in \{1, 2, 3, \ldots\}$. Каждую программу можно рассматривать как алгоритм с множеством возможных исходных данных $\mathbf{N}^n$. Применение такого алгоритма к исходному данному $\langle x_1, \ldots, x_n \rangle$ состоит в следующем. В начальный момент числа $x_1, \ldots, x_n$ помещаются соответственно в регистры $R_1, \ldots, R_n$, при этом во всех остальных регистрах содержится 0. Затем выполняются команды данной программы, начиная с первой. Один шаг работы алгоритма состоит в выполнении одной команды. Последовательность шагов работы алгоритма называется *вычислением*. Вычисление завершается, когда в программе нет команды, которую можно было бы выполнить. Это может произойти, если

1) выполнена последняя команда программы, и эта команда была арифметической;

2) при выполнении команды $J(m, n, q)$ оказалось, что $r_m = r_n$, но q превосходит число команд в программе;

3) при выполнении команды $J(m, n, q)$ оказалось, что $r_m \neq r_n$, но это была последняя команда программы.

Завершение работы алгоритма всегда считается результативным. Результатом работы алгоритма является натуральное число, записанное в регистре R_1 в момент завершения вычисления. Таким образом, каково бы ни было $n \in \{1, 2, 3, \ldots\}$, каждая программа вычисляет частичную n-местную числовую функцию.

Частичная функция из $\mathbf{N}^n$ в $\mathbf{N}$ называется *МНР-вычислимой*, если существует программа для МНР, которая вычисляет эту функцию.

Примеры.

1. Функция $f(x, y) = x + y$ вычисляется следующей программой:

1. $J(3,2,5)$;
2. $S(1)$;
3. $S(3)$;
4. $J(1,1,1)$.

2. Каково бы ни было $n \in \{1,2,3,\ldots\}$, нигде не определенная функция из $\mathbf{N}^n$ в $\mathbf{N}$ вычисляется, например, следующей программой:

1. $J(1,1,1)$.

Задачи.

1. Доказать, что следующие функции МНР-вычислимы:

   а) $\mathrm{sg}(x)$; б) $f(x)=5$; в) $f(x,y)=\begin{cases} 0, & \text{если } x=y,\\ 1, & \text{если } x\neq y; \end{cases}$ г) $f(x,y)=\begin{cases} 0, & \text{если } x\leq y,\\ 1, & \text{если } x>y; \end{cases}$

   д) $f(x)=\frac{x}{3}$; е) $f(x)=\left[\frac{2x}{3}\right]$ ($[z]$ обозначает целую часть числа z).

2. Доказать, что если функция f вычисляется программой без команд условного перехода, то найдется такое число m, что либо $(\forall x)f(x)=m$, либо $(\forall x)f(x)=x+m$.

3. Доказать, что для любой МНР-вычислимой функции существует программа без команд переадресации, вычисляющая эту функцию.

5 МНР-вычислимость частично-рекурсивных функций

Мы докажем, что всякая частично-рекурсивная функция вычислима на машине с неограниченными регистрами.

5.1 Соединение программ

В дальнейшем нам придется рассматривать программы, которые содержат другие программы в качестве *подпрограмм*. Рассмотрим некоторые технические средства, позволяющие строить сложные программы из более простых. Для этого нам понадобится некоторая стандартизация программ. Пусть программа P состоит из команд $I_1,\ldots,I_s$. Будем говорить, что программа P имеет *стандартный вид*, если во всякой команде условного перехода $J(m,n,q)$ из P выполняется неравенство $q\leq s+1$. Очевидно, что каждая программа P может быть приведена к стандартному виду путем замены в ней всякой команды вида $J(m,n,q)$, где $q>s+1$ на $J(m,n,s+1)$, выполняющую точно такое же действие, а именно, остановку выполнения программы P.

Пусть теперь $P=I_1,\ldots,I_s$ и $Q=I'_1,\ldots,I'_t$ — программы стандартного вида. *Соединением* программ P и Q называется программа

$$PQ=I_1,\ldots,I_s,I_{s+1},\ldots,I_{s+t},$$

где $I_{s+1},\ldots,I_{s+t}$ — команды полученные из команд программы Q заменой каждой команды условного перехода $J(m,n,q)$ на $J(m,n,s+q)$. Очевидно, что результат выполнения программы PQ такой же, как и результат последовательного выполнения программ P и Q. Можно говорить также о соединении трех и более программ, понимая, например, программу PQR как $(PQ)R$.

В том случае, когда одна программа используется как подпрограмма в другой программе, важно позаботиться о том, чтобы в ходе выполнения подпрограммы не изменилось содержимое регистров, используемых основной программой. Этого нетрудно добиться следующим образом. Пусть мы собираемся использовать программу P как подпрограмму при составлении новой программы Q. Поскольку программа P конечна, в ней используется конечное число регистров, так что найдется такое наименьшее число u (обозначим его $\rho(P)$), что регистры R_v при $v>u$ не используются в этой программе. Тогда при составлении программы Q, использующей P в качестве своей части, т. е. подпрограммы, нужно использовать в качестве рабочих только регистры R_v при $v>u$.

Если программа P предназначена для вычисления некоторой функции $f(x_1,\ldots,x_n)$, то, в соответствии с общими соглашениями, исходные данные записываются в регистры $R_1,\ldots,R_n$, а результат — в регистр R_1. В тоже время, при использовании программы P в качестве подпрограммы в другой программе, исходные данные для P могут быть записаны в каких-то других регистрах $R_{l_1},\ldots,R_{l_n}$, а результат требуется поместить

в регистр R_l. Чтобы обеспечить возможность такого использования программы P, мы можем преобразовать ее в следующую программу, являющуюся соединением трех программ:

$\left\{\begin{array}{l} T(l_1,1); \\ \ldots \\ T(l_n,n); \\ Z(n+1); \\ \ldots \\ Z(\rho(P)); \end{array}\right.$
P
$\{T(1,l)$.

Модифицированную таким образом программу P будем обозначать

$$P[l_1,\ldots,l_n \to l].$$

5.2 Реализация подстановки на МНР

Сейчас мы докажем, что класс всех МНР-вычислимых функций замкнут относительно операции подстановки.

Теорема 5.1 *Пусть функция $h(\mathbf{x})$, где $\mathbf{x}=(x_1,\ldots,x_n)$, получена подстановкой из МНР-вычислимых функций $f(y_1,\ldots,y_k)$ и $g_1(\mathbf{x}),\ldots,g_k(\mathbf{x})$. Тогда и функция $h(\mathbf{x})$ МНР-вычислима.*

Доказательство. Пусть программы $F,G_1,\ldots,G_k$, имеющие стандартный вид, вычисляют соответственно функции $f,g_1,\ldots,g_k$. Напишем программу H для вычисления функции h. Пусть

$$m=\max(n,k,\rho(F),\rho(G_1),\ldots,\rho(G_k)).$$

Тогда программа H будет выглядеть как следующее соединение нескольких программ:

$\left\{\begin{array}{l} T(1,m+1); \\ \ldots \\ T(n,m+n); \end{array}\right.$
$G_1[m+1,m+2,\ldots,m+n \to m+n+1]$
$\ldots$
$G_k[m+1,m+2,\ldots,m+n \to m+n+k]$
$F[m+n+1,\ldots,m+n+k \to 1]$

Вспомнив только что введенные обозначения для модифицированных программ, читатель без труда убедится, что программа H действительно вычисляет функцию h. Теорема 5.1 доказана.

5.3 Реализация рекурсии на МНР

Сейчас мы докажем, что класс всех МНР-вычислимых функций замкнут относительно операции рекурсии.

Теорема 5.2 *Пусть функция $h(\mathbf{x},y)$, где $\mathbf{x}=(x_1,\ldots,x_n)$, получена рекурсией из МНР-вычислимых функций $f(\mathbf{x})$ и $g(\mathbf{x},y,z)$. Тогда и функция $h(\mathbf{x},y)$ МНР-вычислима.*

Доказательство. Пусть программы F и G, имеющие стандартный вид, вычисляют соответственно функции $f(\mathbf{x})$ и $g(\mathbf{x},y,z)$. Напишем программу H для вычисления функции $h(\mathbf{x},y)$. Пусть

$$m=\max(n+2,\rho(F),\rho(G)).$$

Обозначим сумму $m+n$ через t. Тогда программа H будет выглядеть как следующее соединение нескольких программ:

$\left\{\begin{array}{l} T(1,m+1); \\ \ldots \\ T(n+1,m+n+1); \end{array}\right.$
$F[1,2,\ldots,n \to t+3]$
$\{I_q : J(t+2,t+1,p)$
$G[m+1,m+2,\ldots,m+n,t+2,t+3 \to t+3]$
$\left\{\begin{array}{l} S(t+2); \\ J(1,1,q); \\ I_p : T(t+3,1) \end{array}\right.$

Вспомнив введенные выше обозначения для модифицированных программ, читатель без труда убедится, что программа H действительно вычисляет функцию h. Теорема 5.2 доказана.

5.4 Реализация минимизации на МНР

Сейчас мы докажем, что класс всех МНР-вычислимых функций замкнут относительно операции минимизации.

Теорема 5.3 *Пусть* $g(\mathbf{x}) = \mu y[f(\mathbf{x}, y) = 0]$, *где* $\mathbf{x} = (x_1, \ldots, x_n)$, *а* $f(\mathbf{x}, y)$ *есть МНР-вычислимая функция. Тогда функция* $g(\mathbf{x})$ *также МНР-вычислима.*

Доказательство. Пусть программа F, имеющая стандартный вид, вычисляет функцию $f(\mathbf{x}, y)$. Напишем программу G для вычисления функции $g(\mathbf{x})$. Пусть $m = \max(n+1, \rho(F))$. Тогда программа G будет выглядеть как следующее соединение нескольких программ:

$\begin{cases} T(1, m+1); \\ \ldots \\ T(n, m+n); \end{cases}$

$J_p : F[m+1, m+2, \ldots, m+n+1 \to 1]$

(т. е. J_p является номером первой команды)

$\begin{cases} J(1, m+n+2, q); \\ S(m+n+1); \\ J(1, 1, p); \\ I_q : T(m+n+1, 1) \end{cases}$

Вспомнив введенные выше обозначения для модифицированных программ, читатель без труда убедится, что программа G действительно вычисляет функцию g. Теорема 5.3 доказана.

5.5 Основной результат

Теорема 5.4 *Всякая частично-рекурсивная функция является МНР-вычислимой.*

Доказательство. По определению, любая частично-рекурсивная функция может быть получена из базисных функций

$$o(x), s(x), I_m^n(x_1, \ldots, x_n)(n = 1, 2, \ldots; 1 \leq m \leq n)$$

с помощью операций подстановки, рекурсии и минимизации. Базисные функции, очевидно, МНР-вычислимы. Так, функция $o(x)$ вычисляется программмой, состоящей из одной команды

$$Z(1),$$

функция $s(x)$ вычисляется программмой, состоящей из одной команды

$$S(1),$$

а функция $I_m^n(x_1, \ldots, x_n)$ вычисляется программмой, состоящей из одной команды

$$T(m, 1).$$

Отсюда и из теорем 5.1–5.3 немедленно следует, что всякая частично-рекурсивная функция МНР-вычислима. Теорема 5.4 доказана.

Теорема 5.5 *Всякая МНР-вычислимая функция является частично-рекурсивной.*

Эта теорема доказывается более сложно с помощью такого рассуждения. Пусть функция $f(\mathbf{x})$, где $\mathbf{x} = (x_1, \ldots, x_n)$, вычисляется программой P. Обозначим через $c(\mathbf{x}, t)$ содержимое регистра R_1 после t шагов работы программы P на исходных данных $\mathbf{x}$, если она не завершилась раньше, и заключительное содержимое регистра R_1, если работа программы завершилась за $< t$ шагов. Через $j(\mathbf{x}, t)$ обозначим номер следующей команды после того как сделано ровно t шагов работы программы P на исходных данных $\mathbf{x}$, если она не завершилось раньше, и 0, если работа программы завершилась за $\leq t$ шагов. Тогда, очевидно,

$$f(\mathbf{x}) \simeq c(\mathbf{x}, \mu t[j(\mathbf{x}, t) = 0]).$$

Затем доказывается, что функции c и j частично-рекурсивны. Это дает частичную рекурсивность функции f.

Таким образом, классы числовых функций вычислимых на машинах Тьюринга и на машинах с неограниченными регистрами, совпадают с классом всех частично-рекурсивных функций. Этот факт может рассматриваться как еще один довод в пользу тезиса Чёрча, в силу которого мы теперь имеем, что числовые функции, вычислимые в интуитивном смысле, — это в точности МНР-вычислимые функции.

Задачи

Составить программы для МНР, вычисляющие следующие числовые функции:
1) $f(x,y,z) = x+y+z$; 2) $f(x) = x!$ (факториал); 3) $f(x,y) = x \cdot y$; 4) $f(x,y) = x^y$; 5) $f(x,y) = |x-y|$;
6) $f(x,y) = \max(x,y)$; 7) $f(x,y) = \min(x,y)$; 8) $f(x,y) = \frac{x}{y}$.

6 Нумерация вычислимых функций

Напомним, что $\mathbf{N} = \{0,1,2,\ldots\}$. Пусть $\mathbf{N}^+ = \{1,2,\ldots\}$. Нетрудно убедиться, что функция

$$\pi(m,n) = 2^m \cdot (2n+1) - 1$$

задает вычислимое взимно-однозначное соответствие между множествами $\mathbf{N} \times \mathbf{N}$ и $\mathbf{N}$. Аналогично, функция

$$\zeta(m,n,q) = \pi(\pi(m-1,n-1),q-1)$$

задает вычислимое взимно-однозначное соответствие между множествами $\mathbf{N}^+ \times \mathbf{N}^+ \times \mathbf{N}^+$ и $\mathbf{N}$. Наконец, рассмотрим функцию

$$\tau(\langle a_1,\ldots,a_k\rangle) = 2^{a_1} + 2^{a_1+a_2+1} + 2^{a_1+a_2+a_3+2} + \ldots + 2^{a_1+a_2+\ldots+a_k+k-1} - 1$$

(в дальнейшем вместо $\tau(\langle a_1,\ldots,a_k\rangle)$ мы будем писать просто $\tau(a_1,\ldots,a_k)$). Функция τ задает вычислимое взаимно-однозначное соответствие между множеством всех кортежей натуральных чисел и множеством $\mathbf{N}$. Реально это соответствие устанавливается так. Для данного кортежа $\langle a_1,\ldots,a_k\rangle$ будем строить конечную последовательность нулей и единиц справа налево следующим образом. Первую единицу поставим на (a_1+1)-м месте справа, т. е пишем a_1 нулей, а за ними — одну единицу. Вторую единицу поставим на (a_2+1)-м месте влево от первой единицы, и так далее. Наконец, последнюю, k-ю единицу поставим на (a_k+1)-м месте слева от $(k-1)$-й единицы. Полученная последовательность нулей и единиц является двоичной записью некоторого положительного натурального числа a. Тогда $a-1$ как раз и есть $\tau(a_1,\ldots,a_k)$. Пусть, например, дан кортеж $\langle 3,7,1\rangle$. Тогда, по определению, $\tau(3,7,1) = 2^3 + 2^{11} + 2^{13} - 1$. Построим последовательность нулей и единиц, как сказано выше. Получим двоичное число 10100000001000, очевидно, равное числу $2^3 + 2^{11} + 2^{13}$. Вычитая из него 1, как раз и получаем $\tau(3,7,1)$.

Используя функции π и ζ, каждой команде I для МНР поставим в соответствие число $\beta(I)$ по следующему правилу:
$\beta(Z(n)) = 4(n-1)$;
$\beta(S(n)) = 4(n-1)+1$;
$\beta(T(m,n)) = 4\pi(m-1,n-1)+2$;
$\beta(J(m,n,q)) = 4\zeta(m,n,q)+3$.

Очевидно, функция β задает вычислимое взвимно-однозначное соответствие между командами для МНР и натуральными числами. Число $\beta(I)$ будем называть *гёделевым номером* команды I. Наконец, для любой программы $P = I_1, I_2, \ldots, I_s$ положим

$$\gamma(P) = \tau(\beta(I_1),\ldots,\beta(I_s)).$$

Очевидно, что функция γ задает вычислимое взаимно-однозначное соответствие между программами для МНР и натуральными числами. Число $\gamma(P)$ будем называть *гёделевым номером* программы P или просто ее *номером*. Программу с номером n будем обозначать P_n.

Как известно, при любом $n \geq 1$ всякая программа для МНР вычисляет некоторую частичную n-местную числовую функцию. n-местную функцию, вычисляемую программой P_a, будем обозначать $\varphi_a^{(n)}$. Одноместную функцию, вычисляемую программой P_a, будем обозначать просто φ_a.

Очевидно, что вычислимые числовые функции образуют счетное множество. Поскольку множество всех числовых функций имеет мощность континуума, очевидно существование невычислимых числовых функций. Теперь мы можем привести конкретный пример такой функции. Пусть

$$f(n) = \begin{cases} \varphi_n(n)+1, & \text{если значение } \varphi_n(n) \text{ определено;} \\ 0, & \text{если значение } \varphi_n(n) \text{ не определено.} \end{cases}$$

Допустим, что эта функция вычислима. Тогда она МНР-вычислима. Пусть f вычисляется программой P_m, т. е. f есть φ_m. Очевидно, что значение $\varphi_m(m)$ определено, так что имеем:

$$f(m) = \varphi_m(m) + 1 = f(m) + 1.$$

Это явный абсурд. Полученное противоречие доказывает, что на самом деле функция f невычислима.

Задачи.

1. Найти гёделев номер команды $J(3,4,2)$.
2. Найти команду с номером 503.
3. Найти номер следующей программы:
   1. $T(3,4)$;
   2. $S(3)$;
   3. $Z(1)$.
4. Найти программу P_{100}.
5. Доказать, что невычислима функция $f(n) = \begin{cases} \varphi_n(n) + 27^n, \text{ если } \varphi_n(n) \text{ определено} \\ n^2, \text{ если } \varphi_n(n) \text{ не определено} \end{cases}$.
6. Пусть $f(x,y)$ — тотальная вычислимая функция. Для каждого m обозначим через g_m такую функцию, что $(\forall y) g_m(y) = f(m,y)$. Построить такую тотальную вычислимую функцию h, что $(\forall m) h \neq g_m$.

7 Теорема о параметризации

Условимся о некоторых терминах и обозначениях, связанных с функциями. В этой и следующих лециях мы будем рассматривать только числовые функции, т. е. многоместные частичные функции из $\mathbf{N}$ в $\mathbf{N}$. Всюду определенные функции будем называть *тотальными функциями*.

Иногда функция задается путем выражения ее значений через значения аргументов в виде числовой формы. Например, говорят о функции x^2. Чтобы выражаться более аккуратно и различать числовые (и вообще, именные) формы и задаваемые ими функции, обычно используют так называемые λ-обозначения. Например, упомянутую функцию x^2 можно обозначить так: $\lambda x.x^2$. Вообще, если дана именная форма $f(x)$ с единственным параметром x, то выражение $\lambda x.f(x)$ считается именем (т. е. обозначением) функции, которая каждому значению a переменной x сопоставляет значение $f(a)$ формы $f(x)$ при этом значении переменной x. В выражении $\lambda x.f(x)$ переменная x оказывается связанной. Если именная форма $f(x,\mathbf{y})$ наряду с x содержит другие параметры из списка $\mathbf{y}$, то выражение $\lambda x.f(x,\mathbf{y})$ является функциональной формой с параметрами $\mathbf{y}$, значением которой при данных значениях $\mathbf{a}$ параметров $\mathbf{y}$ является функция $\lambda x.f(x,\mathbf{a})$. Аналогично вводятся обозначения для многоместных функций. Пусть дана именная форма $f(x_1,\ldots,x_n)$ с параметрами $x_1,\ldots,x_n$. Тогда выражение $\lambda x_1 \ldots x_n.f(x_1,\ldots,x_n)$ считается обозначением n-местной функции, которая каждому набору $a_1,\ldots,a_n$ значений переменных $x_1,\ldots,x_n$ сопоставляет объект, именем которого является выражение $f(a_1,\ldots,a_n)$. Если именная форма $f(x_1,\ldots,x_n,\mathbf{y})$ наряду с $x_1,\ldots,x_n$ содержит другие параметры из списка $\mathbf{y}$, то выражение $\lambda x_1 \ldots x_n.f(x_1,\ldots,x_n,\mathbf{y})$ является функциональной формой с параметрами $\mathbf{y}$, значением которой при данных значениях $\mathbf{a}$ параметров $\mathbf{y}$ является функция

$$\lambda x_1 \ldots x_n.f(x_1,\ldots,x_n,\mathbf{a}).$$

Теорема 7.1 (теорема о параметризации) *Пусть f — двуместная вычислимая функция. Существует одноместная тотальная вычислимая функция k такая, что $f(x,y) \simeq \varphi_{k(x)}(y)$.*

Доказательство. Пусть программа F вычисляет функцию f. Для каждого x построим программу Q_x, являющуюся соединением двух программ:

$$\left\{ \begin{array}{l} T(1,2); \\ Z(1); \\ S(1); \\ \ldots \\ S(1) \end{array} \right. \\ F,$$

где команда $S(1)$ написана x раз. Пусть $k(x)$ — гёделев номер программы Q_x. Очевидно, что функция k вычислима. Нетрудно видеть, что программа Q_x вычисляет функцию $\lambda y.f(x,y)$, т. е. имеет место

$$f(x,y) \simeq \varphi_{k(x)}(y),$$

что и требовалось доказать. Теорема 7.1 доказана.

Только что доказанная теорема имеет следующее естественное обобщение.

Теорема 7.2 (s-m-n-теорема) *Для любых $m, n \geq 1$ существует тотальная вычислимая $(m+1)$-местная функция s_n^m такая, что для любых натуральных чисел $e, x_1, \ldots, x_m, y_1, \ldots, y_n$ имеет место*

$$\varphi_e^{(m+n)}(x_1, \ldots, x_m, y_1, \ldots, y_n) \simeq \varphi^{(n)}_{s_n^m(e,x_1,\ldots,x_m)}(y_1, \ldots, y_n).$$

Эта теорема имеет прозрачный смысл: по программе с гёделевым номером e, вычисляющей $(m+n)$-местную функцию $\varphi_e^{(m+n)}$, и произвольному набору натуральных чисел $x_1, \ldots, x_m$ можно найти (с помощью функции s_n^m) гёделев номер программы, вычисляющей функцию

$$\lambda y_1 \ldots y_n . \varphi_e^{(m+n)}(x_1, \ldots, x_m, y_1, \ldots, y_n).$$

Теорема доказывается с помощью того же программистского приема, что и теорема о параметризации.

8 Универсальная вычислимая функция

Пусть $\mathcal{K}$ — некоторый класс n-местных ($n \geq 1$) функций. Тогда $(n+1)$-местная функция U называется *универсальной для класса $\mathcal{K}$*, если выполнены следующие условия:

- для каждого натурального e функция $\lambda x_1 \ldots x_n . U(e, x_1, \ldots, x_n)$ принадлежит классу $\mathcal{K}$;
- для любой функции f из класса $\mathcal{K}$ существует такое натуральное число e, что

$$f = \lambda x_1 \ldots x_n . U(e, x_1, \ldots, x_n).$$

Очевидно, что универсальная функция для класса $\mathcal{K}$ существует в том и только в том случае, когда класс $\mathcal{K}$ не более чем счетен. Более интересен вопрос о существовании вычислимых универсальных функций для классов, состоящих только из вычислимых функций.

Теорема 8.1 (существование вычислимой универсальной функции) *Для любого натурального числа $n \geq 1$ существует $(n+1)$-местная вычислимая функция, универсальная для класса $\mathcal{C}^{(n)}$ всех n-местных вычислимых функций.*

Доказательство. $(n+1)$-местную функцию $\Phi^{(n)}$ определим как $\lambda e x_1 \ldots x_n . \varphi_e^{(n)}(x_1, \ldots, x_n)$. Функция $\Phi^{(n)}$ вычисляется следующим неформальным алгоритмом:

"Пусть даны натуральные числа $e, x_1, \ldots, x_n$. Найдите программу P_e с номером e. Затем поместите в регистры $R_1, \ldots, R_n$ соответственно числа $x_1, \ldots, x_n$ и запустите выполнение программы P_e. Если вычисление заканчивается, требуемое значение $\Phi^{(n)}(e, x_1, \ldots, x_n)$ содержится в регистре R_1."

Таким образом, функция $\Phi^{(n)}$ вычислима.

Докажем, что $\Phi^{(n)}$ — универсальная функция для класса $\mathcal{C}^{(n)}$. Действительно, для любого натурального e функция $\lambda x_1 \ldots x_n . \Phi^{(n)}(e, x_1, \ldots, x_n)$ совпадает с функцией φ_e и, следовательно, принадлежит классу $\mathcal{C}^{(n)}$. С другой стороны, любая функция f из класса $\mathcal{C}^{(n)}$ вычисляется некоторой программой P_e и потому имеет вид φ_e для некоторого e. Но тогда, очевидно, $f = \lambda x_1 \ldots x_n . \Phi^{(n)}(e, x_1, \ldots, x_n)$. Теорема доказана.

Задачи.

1. Доказать, что не существует вычислимой функции, универсальной для класса всех одноместных тотальных вычислимых функций.
2. Доказать, что существует вычислимая функция, универсальная для класса всех одноместных примитивно-рекурсивных функций.
3. Доказать, что не существует примитивно-рекурсивной функции, универсальной для класса всех одноместных примитивно-рекурсивных функций.
4. Доказать, что существует общерекурсивная функция, не являющаяся примитивно рекурсивной.

9 Разрешимые и перечислимые множества

Пусть X — некоторый ансамбль конструктивных объектов, A — его подмножество. *Характеристической функцией множества* A называется функция $\chi_A : X \to \{0,1\}$, определяемая следующим образом:

$$\chi_A(x) = \begin{cases} 1, \text{если } x \in A; \\ 0, \text{если } x \notin A. \end{cases}$$

Определение. Множество $A \subseteq X$ называется *разрешимым* (или *рекурсивным*), если его характеристическая функция χ_A вычислима.

Иными словами, множество A разрешимо, если существует алгоритм, который для каждого $x \in X$ вычисляет истинностное значение высказывания $x \in A$. Очевидно, что множества X и $\emptyset$ разрешимы.

Теорема 9.1 *Если множества $A, B \subseteq X$ разрешимы, то множества $A \cup B$, $A \cap B$, $A \setminus B$ разрешимы.*

Доказательство. Доказываемое утверждение совершенно очевидно. Действительно, так как множества $A, B \subseteq X$ разрешимы, то их характеристические функции χ_A и χ_B вычислимы. Тогда функция $\chi_{A\cup B}$ вычисляется следующим алгоритмом: "если $\chi_A(x) = 1$ и $\chi_B(x) = 1$, то $\chi_{A\cup B}(x) = 1$; в противном случае $\chi_{A\cup B}(x) = 0$". Аналогично находятся значения $\chi_{A\cap B}(x)$ и $\chi_{A\setminus B}(x)$, если найдены значения $\chi_A(x)$ и $\chi_B(x)$. Теорема доказана.

Из теоремы 9.1, в частности, получается, что дополнение $X \setminus A$ разрешимого множества A разрешимо.

Полухарактеристической функцией множества A называется частичная функция ψ_A из X в $\{1\}$, определяемая следующим образом:

$$\psi_A(x) = \begin{cases} 1, \text{если } x \in A; \\ \text{не определено, если } x \notin A. \end{cases}$$

Определение. Множество $A \subseteq X$ называется *полуразрешимым*, если его полухарактеристическая функция ψ_A вычислима.

Теорема 9.2 *Всякое разрешимое множество полуразрешимо.*

Доказательство. Пусть множество $A \subseteq X$ разрешимо. Тогда его характеристическая функция χ_A вычислима. Полухарактеристическая функция ψ_A множества A вычисляется следующим неформальным алгоритмом: "Пусть дан элемент $x \in X$. Вычислить $\chi_A(x)$. Если $\chi_A(x) = 1$, то положить $\psi_A(x) = 1$; в противном случае значение $\psi_A(x)$ не определено."Теорема доказана.

Теорема 9.3 (теорема Поста) *Множество $A \subseteq X$ разрешимо тогда и только тогда, когда оба множества A и $X \setminus A$ полуразрешимы.*

Доказательство. Если множество A разрешимо, то множество $X \setminus A$ также разрешимо, и по теореме 9.2 оба они полуразрешимы.

Обратно, пусть оба множества A и $X\setminus A$ полуразрешимы. Тогда их полухарактеристические функции ψ_A и $\psi_{X\setminus A}$ вычислимы. Характеристическая функция χ_A множества A вычисляется следующим неформальным алгоритмом: "Пусть дан элемент $x \in X$. Запустить параллельное выполнение программ для вычисления значений $\psi_A(x)$ и $\psi_{X\setminus A}(x)$. Если первая программа выдала результат, то $\chi_A(x) = 1$. Если вторая программа выдала результат, то $\chi_A(x) = 0$. Теорема доказана.

Последовательностью называется любая функция, область определения которой — множество всех натуральных чисел $\mathbf{N}$. Следующее понятие в некотором смысле является алгоритмическим аналогом понятия не более чем счетного множества.

Пусть X — некоторый ансамбль конструктивных объектов. Множество $A \subseteq X$ называется *перечислимым* (или *рекурсивно перечислимым*), если $A = \emptyset$ или A является множеством значений некоторой вычислимой последовательности.

Теорема 9.4 *Всякое перечислимое множество полуразрешимо.*

Доказательство. Пусть множество $A \subseteq X$ перечислимо. Если $A = \emptyset$, то A разрешимо и, следовательно, полуразрешимо. Пусть $A \neq \emptyset$. Тогда A является множеством значений некоторой вычислимой последовательности $f : \mathbf{N} \to X$. Полухарактеристическая функция ψ_A множества A вычисляется следующим неформальным алгоритмом: "Пусть дан элемент $x \in X$. Для каждого натурального n, начиная с 0, вычислять $f(n)$ и проверять условие $f(n) = x$. Если нашлось такое n, что $f(n) = x$, то $\psi_A(x) = 1$."Теорема доказана.

Пусть X и Y — некоторые ансамбли конструктивных объектов, f — частичная функция из X в Y. *Графиком функции* f называется множество $\Gamma_f \subseteq X \times Y$, определяемое следующим образом:

$$\Gamma_f = \{\langle x, y\rangle | y = f(x)\}.$$

Теорема 9.5 (теорема о графике) *Частичная функция f из X в Y вычислима тогда и только тогда, когда ее график Γ_f перечислим.*

Доказательство. Пусть f — частичная вычислимая функция из X в Y. Если функция f нигде не определена, то ее график пуст и, следовательно, перечислим по определению. Рассмотрим случай, когда f определена хотя бы в одной точке. Пусть f определена в точке $a \in X$. Тогда пара $\langle a, f(a)\rangle$ принадлежит графику Γ_f. Будем считать, что фиксировано некоторое кодирование ансамбля конструктивных объектов X натуральными числами, т. е. фиксирована некоторая вычислимая биекция $\alpha : X \to \mathbf{N}$. Для каждого натурального числа i положим $x_i = \alpha^{-1}(i)$. Последовательность $g : \mathbf{N} \to X \times Y$ определим следующим образом. Пусть дано натуральное число n. Оно является номером некоторой пары натуральных чисел $\langle l, r\rangle$ при рассмотренной нами нумерации всех пар натуральных чисел τ, т. е. $\tau(l, r) = n$, причем числа l и r можно эффективно вычислить по n. Запустим алгоритм вычисления функции f на исходном данном x_l и дождемся выполнения $r + 1$ шагов работы этого алгоритма. Если за $r + 1$ шагов или раньше получен результат $f(x_l)$, полагаем $g(n) = \langle x_l, f(x_l)\rangle$. Если же за $r + 1$ шагов работа алгоритма еще не завершилась или же произошла безрезультатная остановка, полагаем $g(n) = \langle a, f(a)\rangle$. Очевидно, что последовательность g вычислима. Очевидно также, что множество ее значений $Ran(g)$ является подмножеством графика Γ_f. Докажем, что $\Gamma_f \subseteq Ran(g)$. Пусть пара $\langle x, y\rangle$ принадлежит графику функции f. Это означает, что алгоритм вычисления функции f на исходном данном x заканчивает работу, скажем, за $r > 0$ шагов и выдает результат y. Пусть $\alpha(x) = l$. Положим $n = \tau(l, r - 1)$. Из определения функции g видно, что $g(n) = \langle x, y\rangle$, т. е.

$$\langle x, y\rangle \in Ran(f).$$

Таким образом, график функции f является множеством значений вычислимой последовательности g, следовательно, он перечислим по определению.

Пусть график Γ_f функции f перечислим. Если он пуст, то функция f нигде не определена и, следовательно, вычислима. Если же график не пуст и перечислим, то он является множеством значений некоторой вычислимой последовательности $g : \mathbf{N} \to X \times Y$. Функция f вычисляется следующим алгоритмом. Пусть дано натуральное число n. Последовательно для каждого натурального i, начиная с 0, вычисляем значение $g(i) = \langle a, f(a)\rangle$ и проверяем условие $a = n$. Если это условие выполнено, то $f(n) = f(a)$, и вычисление закончено. Теорема доказана.

Теорема 9.6 *Пусть f — частичная вычислимая функция из X в Y. Тогда ее область определения $Dom(f) \subseteq X$ и множество значений $Ran(f) \subseteq Y$ суть перечислимые множества.*

Доказательство. Пусть функция f вычислима. По теореме 9.5 ее график перечислим. Если f нигде не определена, то $Dom(f)$ и $Ran(f)$ пусты и, следовательно, перечислимы. В противном случае Γ_f является множеством значений некоторой вычислимой последовательности $g : \mathbf{N} \to X \times Y$. Последовательности $d : \mathbf{N} \to X$ и $r : \mathbf{N} \to Y$ определим следующим образом. Пусть дано натуральное число n. Вычисляем

$$g(n) = \langle a, b\rangle$$

и полагаем $d(n) = a$, $r(n) = b$. Очевидно, что последовательности d и r обе вычислимы, причем $Ran(d)$ есть в точности множество первых компонент пар из Γ_f, т. е. область определения фунции f, а $Ran(r)$ — это в точности множество вторых компонент пар из Γ_f, т. е. множество значений функции f. Таким образом, множество $Dom(f)$ является множеством значений вычислимой последовательности d и перечислимо по определению. Множество $Ran(f)$ является множеством значений вычислимой последовательности r и потому также перечислимо. Теорема доказана.

Теорема 9.7 *Множество $A \subseteq X$ перечислимо тогда и только тогда, когда оно является множеством значений некоторой вычислимой функции.*

Доказательство. Пусть множество A перечислимо. Докажем, что A является множеством значений некоторой вычислимой функции. Если оно пусто, то является множеством значений нигде не определенной функции. Если же A не пусто, то, по определению, оно является множеством значений вычислимой посседовательности, и утверждение доказано.

Обратно, если A является множеством значений некоторой вычислимой функции, то по теореме 9.6 оно перечислимо. Теорема доказана.

Теорема 9.8 *Множество $A \subseteq X$ перечислимо тогда и только тогда, когда оно является областью определения некоторой вычислимой функции.*

Доказательство. Пусть множество A перечислимо. Тогда, по теореме 9.4, оно полуразрешимо. Следовательно, его полухарактеристическая функция ψ_A вычислима. Но всякое множество является областью определения своей полухарактеристической функции. Таким образом, A является областью определения вычислимой функции ψ_A.

Обратно, если множество A является областью определения некоторой вычислимой функции, то по теореме 9.6 оно перечислимо. Теорема доказана.

Теорема 9.9 *Всякое полуразрешимое множество перечислимо.*

Доказательство. Полуразрешимое множество A является областью определения вычислимой функции ψ_A и по теореме 9.6 перечислимо. Теорема доказана.

Теорема 9.10 *Пусть X — произвольный ансамбль конструктивных объектов, A — произвольное его подмножество. Тогда следующие условия эквивалентны:*

1) множество A перечислимо;

2) множество A полуразрешимо;

3) множество A является областью определения некоторой вычислимой функции;

4) множество A является множеством значений некоторой вычислимой функции.

Доказательство. Эквивалентность условий 1) и 2) вытекает из теорем 9.4 и 9.9. В силу теорем 9.8 и 9.7 каждое из условий 3), 4) эквивалентно условию 1). Теорема доказана.

Теорема 9.11 *Пусть X — произвольный ансамбль конструктивных объектов, $A, B \subseteq X$ — перечислимые множества. Тогда множества $A \cap B$ и $A \cup B$ перечислимы.*

Доказательство. Пусть $A, B \subseteq X$ — перечислимые множества. Докажем перечислимость множества $A \cup B$. Если хотя бы одно из множеств A, B пусто, то $A \cup B$ совпадает с другим множеством и потому перечислимо. Рассмотрим случай, когда оба множества A, B непусты. Тогда A является множеством значений некоторой вычислимой последовательности f, а B — множеством значений некоторой вычислимой последовательности g. Определим последовательность h следующим образом: для любого натурального числа n положим

$$h(n) = \begin{cases} f(\frac{n}{2}), \text{если } n \text{ четно;} \\ g(\frac{n-1}{2}), \text{если } n \text{ нечетно.} \end{cases}$$

Очевидно, что последовательность h вычислима, а множество ее значений есть в точности множество $A \cup B$.

Докажем, что множество $A \cap B$ перечислимо. Для этого достаточно доказать, что оно полуразрешимо. Множества A, B полуразрешимы, следовательно, их полухарактеристичекие функции ψ_A и ψ_B вычислимы. Полухарактеристическая функция $\psi_{A \cap B}$ множества $A \cap B$ вычисляется следующим алгоритмом. Пусть дан элемент $x \in X$. Вычислять $\psi_A(x)$. Если вычисление завершилось результативно, вычислять $\psi_B(x)$. Если и это вычисление завершилось результативно, то $\psi_{A \cap B}(x) = 1$. Теорема доказана.

Задачи.

1. Доказать, что существует множество натуральных чисел, не являющееся перечислимым.
2. Доказать, что бесконечное множество $A \subseteq \mathbf{N}$ разрешимо тогда и только тогда, когда A является множеством значений строго возрастающей тотальной вычислимой функции.
3. Доказать, что непустое множество $A \subseteq \mathbf{N}$ разрешимо тогда и только тогда, когда A является множеством значений монотонно (не обязательно строго) возрастающей тотальной вычислимой функции.

10 Теоремы о разрешимых и перечислимых множествах

Функция $f : \mathbf{N} \to \mathbf{N}$ называется *неубывающей*, если

$$(\forall x, y)[x < y \Rightarrow f(x) \leq f(y)].$$

Будем говорить, что множество $A \subseteq \mathbf{N}$ *перечислимо в порядке неубывания*, если существует такая неубывающая вычислимая функция $f : \mathbf{N} \to \mathbf{N}$, что $A = Ran(f)$.

Теорема 10.1 *Пусть $A \subseteq \mathbf{N}$. Множество A разрешимо и непусто тогда и только тогда, когда оно перечислимо в порядке неубывания.*

Доказательство. Пусть множество $A \subseteq \mathbf{N}$ разрешимо и непусто. Пользуясь разрешимостью множества A, найдем его наименьший элемент n_0. Теперь определим функцию $f : \mathbf{N} \to \mathbf{N}$ следующим образом:

$$f(0) = n_0;$$

$$f(x+1) = \begin{cases} x+1, \text{ если } x+1 \in A, \\ f(x), \text{ в противном случае.} \end{cases}$$

Нетрудно убедиться, что f — неубывающая вычислимая функция, и

$$A = Ran(f).$$

Обратно, пусть существует такая неубывающая вычислимая функция $f : \mathbf{N} \to \mathbf{N}$, что $A = Ran(f)$. Очевидно, что в этом случае A непусто. Если множество A конечно, то оно, очевидно, разрешимо. Если же A бесконечно, то для выяснения вопроса, принадлежит ли произвольное данное число x множеству A, будем вычислять последовательные значения $f(0), f(1), f(2), \ldots$ функции f, пока среди них не появится число, большее, чем x. Если к этому моменту число x уже появилось среди значений функции f, то $x \in A$; в противном случае $x \notin A$. Итак, мы располагаем алгоритмом для распознавания принадлежности произвольного числа множеству A, т. е. A разрешимо. Теорема 10.1 доказана.

Функция $f : \mathbf{N} \to \mathbf{N}$ называется *возрастающей*, если

$$(\forall x, y)[x < y \Rightarrow f(x) < f(y)].$$

Будем говорить, что множество $A \subseteq \mathbf{N}$ *перечислимо в порядке возрастания*, если существует такая возрастающая вычислимая функция $f : \mathbf{N} \to \mathbf{N}$, что $A = Ran(f)$.

Теорема 10.2 *Пусть $A \subseteq \mathbf{N}$. Множество A разрешимо и бесконечно тогда и только тогда, когда оно перечислимо в порядке возрастания.*

Доказательство. Пусть множество $A \subseteq \mathbf{N}$ разрешимо и бесконечно. Пользуясь разрешимостью множества A, найдем его наименьший элемент n_0. Теперь определим функцию $f : \mathbf{N} \to \mathbf{N}$ следующим образом:

$$f(0) = n_0;$$

$$f(x+1) = \mu y[y \in A \,\&\, f(x) < y].$$

Нетрудно убедиться, что f — тотальная возрастающая вычислимая функция, и $A = Ran(f)$.

Обратно, пусть существует такая возрастающая вычислимая функция $f : \mathbf{N} \to \mathbf{N}$, что $A = Ran(f)$. Очевидно. что в этом случае множество A бесконечно. Для выяснения вопроса, принадлежит ли произвольное данное число x множеству A, будем вычислять последовательные значения $f(0), f(1), f(2), \ldots$ функции f, пока среди них не появится число, большее, чем x. Если к этому моменту число x уже появилось среди значений функции f, то $x \in A$; в противном случае $x \notin A$. Итак, мы располагаем алгоритмом для распознавания принадлежности произвольного числа множеству A, т. е. A разрешимо. Теорема 10.2 доказана.

Теорема 10.3 *Всякое бесконечное перечислимое множество содержит бесконечное перечислимое подмножество.*

Доказательство. Пусть множество $A \subseteq \mathbf{N}$ перечислимо и бесконечно. Тогда, по определению, A является множеством значений некоторой вычислимой последовательности f. Определим функцию g следующим образом:

$$g(0) = f(0);$$

$$g(x+1) = f(\mu y[f(y) > g(x)]).$$

Нетрудно убедиться, что g — тотальная возрастающая вычислимая функция. Положим $B = Ran(g)$. По теореме 10.2 множество B разрешимо. Так как при этом, очевидно, $B \subseteq A$, то теорема 10.3 доказана.

11 Нумерация перечислимых множеств

Как было доказано в разделе 9, каждое перечислимое множество является областью определения некоторой вычислимой функции. Это позволяет задать следующую нумерацию всех перечислимых подмножеств $\mathbf{N}$. Пусть $W_x = Dom(\varphi_x)$. Число x будем называть *гёделевым номером* множества W_x. Очевидно, что всякое перечислимое множество имеет бесконечно много гёделевых номеров.

В разделе 9 было также установлено, что каждое перечислимое множество является множеством значений некоторой вычислимой функции. Используя нумерацию перечислимых множеств, этот факт можно выразить в следующей более сильной форме.

Теорема 11.1 *Существует одноместная тотальная вычислимая функция f такая, что для любого x $Dom(\varphi_x) = Ran(\varphi_{f(x)})$.*

Доказательство. Пусть двуместная функция h определена следующим образом:

$$h(x,y) = \begin{cases} y, \text{ если } !\varphi_x(y); \\ \text{не определено в противном случае.} \end{cases}$$

Очевидно, что функция h вычислима. По теореме о параметризации (раздел 7) существует такая одноместная тотальная вычислимая функция f, что $h(x,y) \simeq \varphi_{f(x)}(y)$ для любых x, y. Из этого условного равенства и определения функции h следует, что $y \in Ran(\varphi_{f(x)}) \Leftrightarrow !\varphi_x(y) \Leftrightarrow y \in Dom(\varphi_x)$. Это как раз и означает, что f — искомая функция. Теорема 11.1 доказана.

12 Неразрешимые алгоритмические проблемы

Пусть f — частичная функция. Функция g называется *продолжением* функции f, если $Dom(f) \subseteq Dom(g)$ и $g(x) = f(x)$ для любого $x \in Dom(f)$.

Теорема 12.1 *Существует частичная вычислимая функция, не имеющая тотального вычислимого продолжения.*

Доказательство. Рассмотрим следующую частичную функцию f из $\mathbf{N}$ в $\mathbf{N}$:

$$f(n) = \begin{cases} \varphi_n(n) + 1, \text{ если значение } \varphi_n(n) \text{ определено;} \\ \text{не определено, если значение } \varphi_n(n) \text{ не определено.} \end{cases}$$

Очевидно, что функция f вычислима. Докажем, что никакая тотальная вычислимая функция не является продолжением функции f. Пусть дана произвольная тотальная вычислимая функция $g : \mathbf{N} \to \mathbf{N}$, и пусть m — гёделев номер программы, вычисляющей функцию g. Тогда $g = \varphi_m$, и

$$g(m) = \varphi_m(m). \tag{1}$$

Так как значение $\varphi_m(m)$ определено, то

$$f(m) = \varphi_m(m) + 1. \tag{2}$$

Равенства (1) и (2) показывают, что функция g не является продолжением функции f. □

Теорема 12.2 *Существует неразрешимое перечислимое множество.*

Доказательство. В силу теоремы 12.1 существует частичная вычислимая функция f, не имеющая тотального вычислимого продолжения. Область определения этой функции $Dom(f)$ — перечислимое множество. Докажем, что оно неразрешимо. Это вытекает из следующего довольно очевидного общего факта: если область определения частичной вычислимой функции разрешима, то эта функция имеет тотальное вычислимое продолжение. Действительно, пусть g — вычислимая частичная функция из X в Y, причем множество $Dom(g) \subseteq X$ разрешимо. Рассмотрим тотальную функцию $h : X \to Y$, определенную следующим образом: для любого $x \in X$

$$h(x) = \begin{cases} g(x), \text{ если } x \in Dom(g); \\ 0, \text{ если } x \notin Dom(g). \end{cases}$$

Очевидно, что функция h вычислима и является продолжением функции g. □

Теорема 12.3 *Множество $K = \{n|\varphi_n(n)$ определено$\}$ перечислимо, но не разрешимо.*

Доказательство. Множество K является областью определения функции f, построенной при доказательстве теоремы 12.1. Его неразрешимость была установлена при доказательстве теоремы 12.2. □

Теорема 12.3 означает алгоритмическую неразрешимость так называемой проблемы самоприменимости программ: не существует алгоритма, который по любой программе для МНР давал бы правильный ответ на вопрос, завершается ли работа этой программы, когда исходным данным является гёделев номер этой программы. Очевидно, что это явление характерно не только для МНР, но и для любого другого способа программирования.

Теорема 12.4 (неразрешимость проблемы остановки) *Не существует алгоритма, который по любой программе P для МНР и любому исходному данному x давал бы правильный ответ на вопрос, завершается ли работа программы P на исходном данном x.*

Доказательство. Если бы существовал алгоритм, о котором идет речь в формулировке теоремы, то, очевидно, была бы разрешима проблема самоприменимости программ, что, как мы видели, невозможно. □

13 Теорема Райса

Проблема самоприменимости и проблема остановки — это лишь два примера неразрешимых алгоритмических проблем в теории алгоритмов. На самом деле оказывается неразрешимой любая нетривиальная алгоримическая проблема, связанная с распознаванимем свойств вычислимых функций по программам, вычисляющим эти функции.

Пусть $\mathcal{F}$ — произвольное семейство одноместных частичных числовых функций, т. е. $\mathcal{F} \subseteq \mathcal{C}$. *Индексным множеством* семейства $\mathcal{F}$ называется множество $I_{\mathcal{F}} = \{n|\varphi_n \in \mathcal{F}\}$. Иными словами, индексное множество семейства $\mathcal{F}$ состоит в точности из всех номеров программ, вычисляющих функции из семейства $\mathcal{F}$. Семейство $\mathcal{F} \subseteq \mathcal{C}$ называется *нетривиальным*, если $\mathcal{F} \neq \emptyset$ и $\mathcal{F} \neq \mathcal{C}$.

Теорема 13.1 (теорема Райса) *Индексное множество $I_{\mathcal{F}}$ всякого нетривиального семейства функций $\mathcal{F} \subseteq \mathcal{C}$ неразрешимо.*

Доказательство. Пусть ξ — нигде не определенная функция. Допустим, что $\xi \in \mathcal{F}$. Так как семейство $\mathcal{F}$ нетривиальное, существует функция $g \in \mathcal{C} \setminus \mathcal{F}$. Двуместную частичную функцию f определим следующим образом: для произвольных натуральных чисел n и x положим

$$f(n,x) \simeq \begin{cases} g(x), & \text{если } \varphi_n(n) \text{ определено;} \\ \text{не определено,} & \text{если } \varphi_n(n) \text{ не определено.} \end{cases}$$

Функция f вычисляется следующим неформальным алгоритмом:

"Вычислять $\varphi_n(n)$. Если будет получен результат, перейти к вычислению $g(x)$. Если вычисление завершится результативно, положить $f(n,x) = g(x)$."

В силу теоремы о параметризации существует одноместная тотальная вычислимая функция k такая, что $f(n,x) \simeq \varphi_{k(n)}(x)$. Очевидно, что если $n \in K$, т. е. значение $\varphi_n(n)$ определено, то $\varphi_{k(n)} = g$, и $k(n) \notin I_{\mathcal{F}}$. Если же $n \notin K$, т. е. значение $\varphi_n(n)$ не определено, то $\varphi_{k(n)} = \xi$, и $k(n) \in I_{\mathcal{F}}$. Таким образом,

$$n \in K \Leftrightarrow k(n) \notin I_{\mathcal{F}}. \tag{3}$$

Из условия (3) непосредственно вытекает, что если бы множество $I_{\mathcal{F}}$ было разрешимым, то разрешимо было бы и множество K, что невозможно в силу теоремы 12.3.

Мы доказали неразрешимость множества $I_{\mathcal{F}}$ при условии, что $\xi \in \mathcal{F}$. Если же $\xi \notin \mathcal{F}$, то из доказанного вытекает неразрешимость множества $I_{\mathcal{C}\setminus\mathcal{F}} = \mathbf{N} \setminus I_{\mathcal{F}}$, откуда уже следует неразрешимость множества $I_{\mathcal{F}}$, так как дополнение неразрешимого множества неразрешимо. □

14 Десятая проблема Гильберта

Мы видим, что в теории алгоритмов имеется очень много неразрешимых алгоритмических проблем. Используя результаты о неразрешимости некоторых проблем в теории алгоритмов, удалось доказать алгоритмическую неразрешимость некоторых проблем, имеющих общематематическое значение.

В 1900 году на Международном математическом конгрессе в Париже знаменитый немецкий математик Д. Гильберт сформулировал ряд математических проблем, решение которых, по его мнению, наиболее актуально для математики 20-го века. Одна из них, под номером 10, касалась так называемых диофантовых уравнений, т. е. уравнений вида

$$p(x_1, x_2, \ldots, x_n) = 0,$$

где $p(x_1, x_2, \ldots, x_n)$ — диофантов многочлен, т. е. многочлен от переменных $x_1, x_2, \ldots, x_n$ с целыми коэффициентами, причем ищутся только целые решения такого уравнения. Десятая проблема Гильберта состояла в том, чтобы установить, существует ли алгоритм, с помощью которого можно определить, имеет ли решение произвольное наперед заданное диофантово уравнение. В 1970 году советский математик Ю. В. Матиясевич доказал, что такого алгоритма не существует. Суть его доказательства состоит в том, что для любого перечислимого множества $A \subseteq \mathbf{N}$ можно написать такой диофантов многочлен $p(a, x_1, x_2, \ldots, x_n)$ от переменных $a, x_1, x_2, \ldots, x_n$, что для любого натурального a уравнение $p(a, x_1, x_2, \ldots, x_n) = 0$ как уравнение относительно неизвестных $x_1, x_2, \ldots, x_n$ имеет целые решения тогда и только тогда, когда $a \in A$. Теперь решение десятой проблемы Гильберта непосредственно вытекает из существования неразрешимого перечислимого множества.

Задачи.

1. Пусть тотальная одноместная вычислимая функция $f : \mathbf{N} \to \mathbf{N}$ удовлетворяет условию $(\forall x) f(x) \geq x$. Доказать, что множество значений $Ran(f)$ функции f разрашимо.

2. Доказать, что каждое бесконечное перечислимое множество $A \subseteq \mathbf{N}$ является множеством значений некоторой взаимно-однозначной тотальной вычислимой функции $f : \mathbf{N} \to \mathbf{N}$.

3. Доказать, что график тотальной вычислимой функции разрешим.

4. Доказать, что полный прообраз $f^{-1}(A)$ разрешимого множества A относительно тотальной вычислимой функции f разрешим.

5. Пусть $A \subseteq \mathbf{N}$ — разрешимое множество, $f : \mathbf{N} \to \mathbf{N}$ — тотальная вычислимая функция, причем $Ran(f) = \mathbf{N}$, $f(A) \cap f(\mathbf{N} \setminus A) = \emptyset$. Доказать, что множество $f(A)$ разрешимо.

6. Пусть A, B — перечислимые множества, C — разрешимое множество, причем $A \cap B = \emptyset$, $A \subseteq C \subseteq A \cup B$. Доказать, что множество A разрешимо.

7. Пусть $\mathcal{A}$ — произвольное семейство перечислимых множеств натуральных чисел. Индексным множеством семейства $\mathcal{A}$ называется множество $I_{\mathcal{A}} = \{n | W_n \in \mathcal{A}\}$. Семейство $\mathcal{A}$ называется нетривиальным, если оно не пусто и содержит не все перечислимые множества натуральных чисел. Доказать следующий вариант теоремы Райса для перечислимых множеств: индексное множество $I_{\mathcal{A}}$ всякого нетривиального семейства перечислимых множеств $\mathcal{A}$ неразрешимо.

15 Индексы разрешимых и конечных множеств

Каждое разрешимое, в частности, каждое конечное множество перечислимо, следовательно, имеет вид W_x для некоторого x. Дополнение разрешимого множества также разрешимо, следовательно, перечислимо. Однако, как мы сейчас увидим, гёделев номер дополнения разрешимого множества W_x не может быть найден по числу x с помощью какого-либо алгоритма.

Теорема 15.1 *Не существует частичной вычислимой функции ψ такой, что*

$$(\forall x)[W_x \text{ разрешимо } \Rightarrow (!\psi(x) \,\&\, W_{\psi(x)} = \overline{W_x})].$$

Доказательство. Предположим, что такая функция ψ существует. Рассмотрим двуместную функцию

$$f(x, y) \simeq \varphi_x(x).$$

Очевидно, что функция f вычислима. По теореме о параметризации существует одноместная тотальная вычислимая функция g такая, что $f(x, y) \simeq \varphi_{g(x)}(y)$ для любых x, y. Из этого условного равенства и определения функции f следует, что

$$W_{g(x)} = \begin{cases} \mathbf{N}, \text{ если } x \in K; \\ \emptyset, \text{ если } x \notin K. \end{cases}$$

Таким образом, для любого x множество $W_{g(x)}$ разрешимо. Тогда при любом x определено значение $\psi(g(x))$, причем

$$W_{\psi(g(x))} = \begin{cases} \emptyset, \text{ если } x \in K; \\ \mathbf{N}, \text{ если } x \notin K. \end{cases}$$

Значит, $\overline{K} = \{x | W_{\psi(g(x)} \neq \emptyset\}$. Так как множество K перечислимо, но не разрешимо, то по теореме Поста его дополнение неперечислимо. С другой стороны, множество $\{x | W_{\psi(g(x))} \neq \emptyset\}$ перечислимо. Действительно, оно полуразрешимо, так как его полухарактеристическая функция вычисляется следующим неформальным алгоритмом: "Пусть дано число x. Вычислить $\psi(g(x))$. Найти программу с номером $\psi(g(x))$ и запустить параллельные вычисления этой программы на исходных данных $0, 1, 2, \ldots$. Если какое-то из этих вычислений завершилось результативно, выдать 1."Полученное противоречие показывает, что на самом деле вычислимой функции ψ не существует. Теорема 15.1 доказана.

По определению, множество разрешимо тогда и только тогда, когда вычислима его характеристическая функция. Гёделев номер характеристической функции χ_A разрешимого множества $A \subseteq \mathbf{N}$ будем называть *характеристическим индексом* множества A.

Теорема 15.2 *Существует вычислимая функция $h : \mathbf{N} \to \mathbf{N}$ такая, что для любого x, если φ_x — характеристическая функция некоторого множества A, то $\varphi_{h(x)}$ — характеристическая функция его дополнения $\mathbf{N} \setminus A$.*

Доказательство. Двуместную частичную функцию f определим следующим образом:

$$f(x, y) \simeq \overline{\text{sg}}(\varphi_x(y)).$$

Очевидно, что функция f вычислима. По теореме о параметризации существует такая одноместная тотальная вычислимая функция h, что $f(x, y) \simeq \varphi_{h(x)}(y)$ для любых x, y. Из этого условного равенства и определения функции f следует, что $\varphi_{h(x)}(y) \simeq \overline{\text{sg}}(\varphi_x(y))$. В частности, если φ_x — характеристическая функция некоторого множества A, то $\varphi_{h(x)}$ — характеристическая функция его дополнения $\mathbf{N} \setminus A$, что и требовалось доказать. Теорема 15.2 доказана.

Следующая теорема показывает, что существует алгоритм, который по характеристическому индексу разрешимого множества A находит его гёделев номер как перечислимого множества.

Теорема 15.3 *Существует вычислимая функция $g : \mathbf{N} \to \mathbf{N}$ такая, что для любого x, если φ_x — характеристическая функция некоторого множества A, то $W_{g(x)} = A$.*

Доказательство. Двуместную частичную функцию f определим следующим образом:

$$f(x, y) = \begin{cases} 1, \text{ если } \varphi_x(y) = 1; \\ \text{не определено в противном случае.} \end{cases}$$

Очевидно, что функция f вычислима. По теореме о параметризации существует такая одноместная тотальная вычислимая функция g, что

$$f(x, y) \simeq \varphi_{g(x)}(y)$$

для любых x, y. Из этого условного равенства и определения функции f следует, что $W_{g(x)} = \{y | \varphi_x(y) = 1\}$. Таким образом, если φ_x — это характеристическая функция некоторого множества A, то $W_{g(x)} = A$, что и требовалось доказать. Теорема 15.3 доказана.

Оказывается, однако, что обратный переход — от гёделевых номеров разрешимых множеств к их характеристическим индексам — невозможно осуществить с помощью какого-либо алгоритма.

Теорема 15.4 *Не существует одноместной частичной вычислимой функции ϕ такой, что*

$$(\forall x)[W_x \textit{ разрешимо } \Rightarrow (!\phi(x) \,\&\, \varphi_{\phi(x)} = \chi_{W_x}].$$

Доказательство. Допустим, что такая функция ϕ существует. Для каждого x положим

$$\psi(x) \simeq g(h(\phi(x))),$$

где g и h — функции, существование которых утверждается соответственно в теоремах 15.3 и 15.2. Очеидно, что функция ψ вычислима. Тогда, если x — гёделев номер разрешимого множества A, то $\phi(x)$ — характеристический индекс множества A, а $h(\phi(x))$ — характеристический индекс его дополнения $\mathbf{N} \setminus A$. Наконец, $\psi(x)$ есть гёделев номер множества $\mathbf{N} \setminus A$. Таким образом, с помощью вычислимой функции ψ мы можем

по гёделеву номеру произвольного разрешимого множества A найти гёделев номер его дополнения, что противоречит теореме 15.1. Это доказывает, что на самом деле вычислимой функции ϕ не существует. Теорема 15.4 доказана.

Очевидно, что если мы знаем характеристический индекс разрешимого множества, то мы владеем и разрешающим алгоритмом для этого множества. Теорема 15.4 показывает, что по гёделеву номеру разрешимого множества в принципе невозможно найти разрешающий алгоритм для этого множества, так что в некотором смысле характеристические индексы разрешимых множеств обладают большей "информативностью"по сравнению с их гёделевыми номерами.

Каждое конечное множество, являясь разрешимым, имеет гёделев номер и характеристический индекс. Однако конечные множества натуральных чисел допускают еще и нумерацию третьего вида. А именно, если $A = \{x_1, x_2, \ldots, x_n\}$, где $x_1 < x_2 < \ldots < x_n$, то число $2^{x_1} + 2^{x_2} + \ldots + 2^{x_n}$ называется *каноническим индексом* множества A. Каноническим индексом пустого множества считается число 0. Через D_x будем обозначать конечное множество, каноническим индексом которого является число x. Очевидно, что всякое конечное множество имеет единственный канонический индекс, и каждое натуральное число есть канонический индекс некоторого конечного множества. Чтобы в явном виде найти все элементы множества D_x, нужно рассмотреть двоичную запись числа x и, если в этой записи цифра 1 стоит на $(i+1)$-м месте справа, то $i \in D_x$. Например, число 13 имеет двоичную запись 1101. Следовательно, $D_{13} = \{0, 2, 3\}$. Очевидно, что по каноническому индексу конечного множества может быть алгоритмически найден характеристический индекс этого множества. Сейчас мы увидим, что алгоритмический переход от характеристических индексов конечных множеств к их каноническим индексам невозможен. Сначала отметим следующий очевидный факт: существует тотальная одноместная вычислимая функция $f : \mathbf{N} \to \mathbf{N}$ такая, что $f(x)$ есть число элементов в множестве D_x, каково бы ни было x. Действительно, по числу x мы можем восстановить множество D_x, а затем подсчитать число элементов в нем. С другой стороны, имеет место следующая теорема.

Теорема 15.5 *Не существует одноместной частичной вычислимой функции ψ такой, что для всякого x, если φ_x — характеристическая функция некоторого конечного множества A, то значение $\psi(x)$ определено и равно числу элементов в множестве A.*

Доказательство. Допустим, что такая функция ψ существует. Двуместную функцию f зададим следующим образом:

$$f(x,y) = \begin{cases} 1, & \text{если работа программы } P_x \text{ на исходном} \\ & \text{данном } x \text{ завершается ровно за } y \text{ шагов;} \\ 0 & \text{в противном случае.} \end{cases}$$

Очевидно, что функция f вычислима. По теореме о параметризации существует такая одноместная тотальная вычислимая функция g, что

$$f(x,y) \simeq \varphi_{g(x)}(y)$$

для любых x, y. Из этого условного равенства и определения функции f следует, что $\varphi_{g(x)}$ является характеристичекой функцией одноэлементного множества, если $x \in K$, и пустого множества, если $x \notin K$. Таким образом, для любого x

$$\psi(g(x)) = \begin{cases} 1, \text{ если } x \in K; \\ 0, \text{ если } x \notin K. \end{cases}$$

Это противоречит неразрешимости множества K. Значит, на самом деле функция ψ не существует. Теорема 15.5 доказана.

Таким образом, по каноническому индексу конечного множества A мы можем найти мощность множества A, а по его характеристическому индексу — нет. Значит, алгоритмический переход от характеристических индексов конечных множеств к их каноническим индексам невозможен, и канонические индексы конечных множеств обладают большей "информативностью"по сравнению с их характеристическими индексами.

16 Рекурсивно неотделимые перечислимые множества

Пусть X — некоторый ансамбль конструктивных объектов. Говорят, что множества $A, B \subseteq X$ *рекурсивно отделимы*, если существует такое разрешимое множество $C \subseteq X$, что $A \subseteq C$, $B \subseteq (X \setminus C)$.

Теорема 16.1 *Существует пара рекурсивно неотделимых непересекающихся перечислимых множеств $A, B \subseteq \mathbf{N}$.*

Доказательство. Пусть

$$A = \{x|\varphi_x(x) = 0\},\ B = \{x|\varphi_x(x) = 1\}.$$

Очевидно, что множества A и B оба перечислимы и не пересекаются. Допустим, что они рекурсивно отделимы. Пусть $C \subseteq \mathbf{N}$ — такое разрешимое множество, что $A \subseteq C$, $B \subseteq (\mathbf{N} \setminus C)$. Характеристическая функция множества C вычисляется некоторой программой P_m. Допустим, что $m \in C$. Тогда $\varphi_m(m) = 1$, и $m \in B$, что противоречит включению $B \subseteq (\mathbf{N} \setminus C)$. Значит, $m \notin C$. Тогда $\varphi_m(m) = 0$, и $m \in A$, что противоречит включению $A \subseteq C$. Таким образом, множество C с указанными свойствами существовать не может. Теорема 16.1 доказана.

Задачи.

1. Доказать, что образ $f(A)$ перечислимого множества A относительно частичной вычислимой функции f перечислим.
2. Доказать, что полный прообраз $f^{-1}(A)$ перечислимого множества A относительно частичной вычислимой функции f перечислим.
3. Пусть A, B — перечислимые множества. Доказать, что существуют перечислимые множества A_1, B_1 такие, что
$$A_1 \subseteq A, B_1 \subseteq B, A_1 \cap B_1 = \emptyset, A_1 \cup B_1 = A \cup B.$$
4. Доказать, что следующие множества неразрешимы:
   (a) $A_1 = \{x|\varphi_x$ — константа$\}$;
   (b) $A_2 = \{x|\varphi_x(a) = b\}$, где a, b — фиксированные числа;
   (c) $A_3 = \{\langle x, y\rangle | y \in Dom(\varphi_x)\}$;
   (d) $A_4 = \{\langle x, y\rangle | y \in Ran(\varphi_x)\}$;
   (e) $A_5 = \{x|\varphi_x = \varphi_y\}$.

17 Теорема Райса – Шапиро

Семейство вычислимых функций $\mathcal{F} \subseteq \mathcal{C}$ называется *вполне разрешимым*, если его индексное множество разрешимо. Доказанная в разделе 13 теорема Райса утверждает, что никакое нетривиальное семейство $\mathcal{F}$ не является вполне разрешимым. Семейство вычислимых функций $\mathcal{F} \subseteq \mathcal{C}$ называется *вполне перечислимым*, если его индексное множество перечислимо. Существуют нетривиальные вполне перечислимые семейства вычислимых функций. Таково, например, семейство всех вычислимых функций, принимающих значение 0 в точке 0. Необходимое условие вполне перечислимости семейства вычислимых функций дает теорема Райса – Шапиро.

Конечной функцией называется функция с конечной областью определения. Запись $\theta \subseteq f$ будет означать, что функция f является продолжением функции θ.

Теорема 17.1 (теорема Райса – Шапиро) *Пусть $\mathcal{F}$ — вполне перечислимое семейство одноместных вычислимых числовых функций. Тогда*

$$\mathcal{F} = \{f \in \mathcal{C} |\ \textit{существует конечная функция } \theta \in \mathcal{F} \textit{ такая, что } \theta \subseteq f\}.$$

Доказательство. Пусть $\mathcal{F}$ — вполне перечислимое семейство, т. е. индексное множество $I_{\mathcal{F}}$ семейства $\mathcal{F}$ перечислимо. Семейство

$$\{f \in \mathcal{C} |\ \text{существует конечная функция } \theta \in \mathcal{F} \text{ такая, что } \theta \subseteq f\},$$

участвующее в формулировке теоремы, обозначим через $\mathcal{A}$. Теорему доказываем от противного. Допустим сначала, что неверно включение $\mathcal{F} \subseteq \mathcal{A}$. Тогда существует функция $f \in \mathcal{F}$ такая, что f не является продолжением никакой конечной функции θ из $\mathcal{F}$. Пусть P — программа для МНР, вычисляющая полухарактеристическую функцию множества K. Двуместную функцию g определим следующим образом:

$$g(z,t) \simeq \begin{cases} f(t), \text{ если вычисление } P(z) \text{ не завершается за } \leq t \text{ шагов;} \\ \text{не определено, если вычисление } P(z) \text{ завершается за} \\ \leq t \text{ шагов.} \end{cases}$$

Очевидно, что функция g вычислима. По теореме о параметризации существует такая одноместная тотальная вычислимая функция $s : \mathbf{N} \to \mathbf{N}$, что для любых z, t имеет место условное равенство $g(z,t) \simeq \varphi_{s(z)}(t)$. Отсюда и из определения функции g следует, что $\varphi_{s(z)} \subseteq f$. Заметим также, что если $z \in K$, то вычисление $P(z)$ завершается за конечное число шагов, скажем, t_0. Тогда, как видно из задания функции g, значения $g(z,t)$ для $t > t_0$ не определены, и функция $\varphi_{s(z)}$ конечна. Следовательно, $\varphi_{s(z)} \notin \mathcal{F}$ и $s(z) \notin I_{\mathcal{F}}$. Если же $z \notin K$, то $g(z,t) \simeq f(t)$ при любом t, и $\varphi_{s(z)} = f$. Следовательно, в этом случае $\varphi_{s(z)} \in \mathcal{F}$ и $s(z) \in I_{\mathcal{F}}$. Таким образом, мы видим, что для любого z

$$z \notin K \Leftrightarrow s(z) \in I_{\mathcal{F}},$$

откуда вытекает, что дополение множества K перечислимо. Полученное противоречие показывает, что допущение неверно, и на самом деле $\mathcal{F} \subseteq \mathcal{A}$.

Допустим теперь, что неверно включение $\mathcal{A} \subseteq \mathcal{F}$. Тогда существует конечная функция $\theta \in \mathcal{F}$ такая, что f является продолжением функции θ, но $f \notin \mathcal{F}$. Двуместную функцию g определим следующим образом:

$$g(z,t) \simeq \begin{cases} f(t), \text{ если } t \in Dom(\theta) \text{ или } z \in K; \\ \text{не определено в остальных случаях.} \end{cases}$$

Очевидно, что функция g вычислима. По теореме о параметризации существует такая одноместная тотальная вычислимая функция $s : \mathbf{N} \to \mathbf{N}$, что для любых z, t имеет место условное равенство $g(z,t) \simeq \varphi_{s(z)}(t)$. Отсюда и из определения функции g следует, что если $z \in K$, то $\varphi_{s(z)} = f$, следовательно, $\varphi_{s(z)} \notin \mathcal{F}$ и $s(z) \notin I_{\mathcal{F}}$. Если же $z \notin K$, то из того факта, что f является продолжением функции θ, следует, что $\varphi_{s(z)} = \theta$, следовательно, $\varphi_{s(z)} \in \mathcal{F}$ и $s(z) \in I_{\mathcal{F}}$. Таким образом, мы видим, что для любого z

$$z \notin K \Leftrightarrow s(z) \in I_{\mathcal{F}},$$

откуда вытекает, что дополнение множества K перечислимо. Полученное противоречие показывает, что допущение неверно, и на самом деле $\mathcal{A} \subseteq \mathcal{F}$. Таким образом, мы доказали оба включения $\mathcal{F} \subseteq \mathcal{A}$ и $\mathcal{A} \subseteq \mathcal{F}$. Следовательно, $\mathcal{F} = \mathcal{A}$, и теорема 17.1 доказана.

Теорема Райса – Шапиро позволяет легко доказывать неперечислимость некоторых множеств. Так, ранее мы доказали, что множество гёделевых номеров одноместных тотальных вычислимых числовых функций неперечислимо. Теперь этот же результат может быть получен с помощью теоремы Райса – Шапиро. Действительно, рассматриваемое множество — это индексное множество семейства $\mathcal{F}$ всех одноместных тотальных вычислимых числовых функций. Это семейство не содержит ни одной конечной функции, так что никакая функция из $\mathcal{F}$ не является продолжением конечной функции из $\mathcal{F}$, и по теореме Райса – Шапиро $\mathcal{F}$ не будет вполне перечислимым.

Теорема Райса – Шапиро может быть переформулирована и доказана и для семейств перечислимых множеств. Семейство $\mathcal{K}$ перечислимых подмножеств натурального ряда будем называть вполне перечислимым, если перечислимо его индексное множество. Класс конечных множеств $\mathcal{D}$ называется *канонически перечислимым*, если перечислимо множество канонических индексов всех множеств из $\mathcal{D}$.

Теорема 17.2 *Семейство $\mathcal{K}$ перечислимых подмножеств $\mathbf{N}$ вполне перечислимо тогда и только тогда, когда существует канонически перечислимое семейство конечных множеств $\mathcal{D}$ такое, что*

$$(\forall A)[A \in \mathcal{K} \Leftrightarrow (A \text{ перечислимо и } (\exists D)(D \in \mathcal{D} \,\&\, D \subseteq A))]. \tag{4}$$

Доказательство. Пусть $\mathcal{K}$ — вполне перечислимое семейство перечислимых множеств. Пусть $\mathcal{D}$ — семейство всех конечных множеств из $\mathcal{K}$. Докажем, что множество канонических индексов $\{x | D_x \in \mathcal{D}\}$ всех множеств из $\mathcal{D}$ перечислимо. А именно, убедимся, что оно полуразрешимо. Пусть дано натуральное число x. Как было отмечено в разделе 15, по числу x можно эффективно, т. е. с помощью алгоритма, найти характеристический индекс множества D_x, а затем — с помощью теорем, доказанных в разделе 15, — гёделев номер этого множества, т. е. такое число $\phi(x)$, что $D_x = W_{\phi(x)}$. Применим к этому числу алгоритм, вычисляющий полухарактеристическую функцию индексного множества семейства $\mathcal{K}$. Очевидно, что результат будет получен в том и только в том случае, когда множество $W_{\phi(x)}$ принадлежит семейству $\mathcal{K}$, т. е. множество D_x, принадлежит семейству $\mathcal{D}$. Таким образом, множество канонических индексов всех множеств из $\mathcal{D}$ перечислимо. Итак, $\mathcal{D}$ — канонически перечислимое семейство.

Рассмотрим теперь семейство вычислимых функций

$$\{f \in \mathcal{C} | Dom(f) \in \mathcal{K}\}.$$

Очевидно, что индексные множества семейств $\mathcal{K}$ и $\mathcal{F}$ совпадают. Так как семейство $\mathcal{K}$ вполне перечислимо, то таково же и семейство $\mathcal{F}$. По теореме 17.1 функция f принадлежит семейству $\mathcal{F}$ тогда и только тогда,

когда она является продолжением некоторой конечной функции из $\mathcal{F}$. В силу определения семейства $\mathcal{F}$ это означает, что произвольное перечислимое множество принадлежит семейству $\mathcal{K}$ тогда и только тогда, когда оно имеет конечное подмножество из $\mathcal{K}$, а значит, и из $\mathcal{D}$. Таким образом, семейство $\mathcal{D}$ является канонически перечислимым и удовлетворяет условию (4).

Докажем теперь, что если для данного семейства перечислимых множеств $\mathcal{K}$ существует канонически перечислимое семейство конечных множеств $\mathcal{D}$ такое, что выполняется условие (4), то семейство $\mathcal{K}$ вполне перечислимо. Если семейство $\mathcal{D}$ пусто, то, очевидно, семейство $\mathcal{K}$ также пусто, следовательно, вполне перечислимо. Если же семейство $\mathcal{D}$ не пусто, то множество канонических индексов всех множеств из $\mathcal{D}$ перечисляется некоторой вычислимой функцией. Опишем алгоритм, вычисляющий полухарактеристическую функцию индексного множества семейства $\mathcal{K}$. Пусть дано натуральное число x. Будем перечислять элемент за элементом множество канонических индексов всех множеств из $\mathcal{D}$. Получая очередной элемент y этого множества, выписываем элементы множества D_y и запускаем алгоритм проверки этих элементов на принадлежность их множеству W_x, т. е. на каждом из них начинаем вычислять значение функции φ_x. Пока процесс вычисления идет, мы находим следующий элемент y из множества канонических индексов всех множеств из $\mathcal{D}$ и запускаем алгоритм проверки его элементов на принадлежность их множеству W_x, и т. д. Если окажется, что все элементы какого-то из этих множеств D_y принадлежат множеству W_x, делаем вывод, что W_x принадлежит семейству $\mathcal{K}$, а x принадлежит индексному множеству семейства $\mathcal{K}$. Следовательно, индексное множество семейства $\mathcal{K}$ перечислимо, а само это семейство вполне перечислимо. Теорема 17.2 доказана.

Задачи.

1. Вывести теорему Райса из теоремы Райса – Шапиро.
2. Доказать, что множество гёделевых номеров всех нетотальных вычислимых функций неперечислимо.
3. Доказать, что множество гёделевых номеров всех вычислимых неинъективных функций перечислимо.
4. Выяснить, какие из следующих множеств разрешимы, какие перечислимы, какие имеют перечислимое дополнение:
   (a) $\{x|x \in W_x\}$;
   (b) $\{x|x$ есть полный квадрат$\}$;
   (c) $\{x|\varphi_x$ есть инъективная функция$\}$;
   (d) $\{x|$ в десятичном разложении числа π существуют x идущих друг за другом семерок$\}$;
   (e) $\{x|\varphi_m(x)$не определено$\}$ (m фиксировано).

18 Многозначная сводимость

Доказывая неразрешимость некоторых множеств, мы часто пользовались следующим приемом: мы показывали, что если бы существовал разрешающий алгоритм для данного множества, то мы имели бы разрешающий алгоритм и для некоторого заведомо неразрешимого множества, например, K. В общем случае этот прием называется *сведéнием* одной алгоритмической проблемы к другой. Понятию сведéния можно придать точный смысл многими различными способами. Мы рассмотрим один из них — многозначную сводимость.

Будем говорить, что множество $A \subseteq \mathbf{N}$ *многозначно сводится* (или *m-сводится*) к множеству $B \subseteq \mathbf{N}$ ($A \leq_m B$), если существует тотальная вычислимая функция $f : \mathbf{N} \to \mathbf{N}$ такая, что

$$(\forall x)[x \in A \Leftrightarrow f(x) \in B].$$

В этом случае будем говорить, что A сводимо к B *посредством* функции f, а функцию f будем называть *сводящей* A к B.

Примеры.

1) Пусть B есть множество всех гёделевых номеров функции, тождественно равной 0. Тогда $K \leq_m B$. Чтобы доказать это, рассмотрим двуместную функцию g, определяемую так:

$$g(x,y) = \begin{cases} 0, \text{ если } x \in K; \\ \text{не определено, если } x \notin K. \end{cases}$$

Очевидно, что функция g вычислима. По теореме о параметризации существует одноместная тотальная вычислимая функция f такая, что для любых x, y выполняется условное равенство $g(x,y) \simeq \varphi_{f(x)}(y)$. Отсюда

и из определения функции g немедленно следует, что если $x \in K$, то функция $\varphi_{f(x)}$ тождественно равна 0, т. е. $f(x) \in B$, а если $x \notin K$, то функция $\varphi_{f(x)}$ нигде не определена, и $f(x) \notin B$. Таким образом, $(\forall x)[x \in K \Leftrightarrow f(x) \in B]$, т. е. $K \leq_m B$.

2) Анализ доказательства теоремы Райса показывает, что множество K m-сводится к индексному множеству любого нетривиального семейства вычислимых функций, не содержащего нигде не определенную функцию.

3) Пусть A — множество всех гёделевых номеров тотальных одноместных вычислимых числовых функций, а B — множество всех гёделевых номеров функции, тождественно равной 0. Тогда $A \leq_m B$. Для доказательства рассмотрим двуместную функцию $g(x, y) \simeq 0 \cdot \varphi_x(y)$. Очевидно, что эта функция вычислима. По теореме о параметризации существует одноместная тотальная вычислимая функция f такая, что для любых x, y выполняется условное равенство $g(x, y) \simeq \varphi_{f(x)}(y)$. Отсюда и из определения функции g немедленно следует, что если $x \in A$, то функция $\varphi_{f(x)}$ тождественно равна 0, т. е. $f(x) \in B$, а если $x \notin A$, то функция $\varphi_{f(x)}$ не всюду определена, и $f(x) \notin B$. Таким образом, $(\forall x)[x \in A \Leftrightarrow f(x) \in B]$, т. е. $A \leq_m B$.

Теорема 18.1 *1. Отношение $\leq_m$ рефлексивно и транзитивно.*
2. $A \leq_m B$ тогда и только тогда, когда $\overline{A} \leq_m \overline{B}$.
3. Если множество B разрешимо, и $A \leq_m B$, то множество A разрешимо.
4. Если множество A разрешимо, $B \neq \emptyset$, $B \neq \mathbf{N}$, то $A \leq_m B$.
5. Если множество B перечислимо, и $A \leq_m B$, то множество A перечислимо.
6. $A \leq_m \mathbf{N} \Leftrightarrow A = \mathbf{N}$;
7. $A \leq_m \emptyset \Leftrightarrow A = \emptyset$;
8. $\mathbf{N} \leq_m A \Leftrightarrow A \neq \emptyset$;
9. $\emptyset \leq_m A \Leftrightarrow A \neq \mathbf{N}$.

Доказательство. 1. Очевидно, что A сводится к A посредством тождественной функции, так что $A \leq_m A$, и отношение $\leq_m$ рефлексивно. Очевидно также, что если A сводится к B посредством функции f, а B сводится к C посредством функции g, то A сводится к посредством композиции этих функций, т. е. такой функции h, что $(\forall x)h(x) = g(f(x))$.

2. Очевидно, что если A сводится к B посредством функции f, то $\overline{A}$ сводится к $\overline{B}$ посредством той же функции.

3. Очевидно, что если A сводится к B посредством функции f, то $(\forall x)\chi_A(x) = \chi_B(f(x))$, так что если множество B разрешимо, т. е. функция χ_B вычислима, то функция χ_A также вычислима, т. е. множество A разрешимо.

4. Если $B \neq \emptyset$, $B \neq \mathbf{N}$, возьмем $b \in B$, $c \notin B$. Если A разрешимо, то вычислима функция

$$f(x) = \begin{cases} b, \text{ если } x \in A; \\ c, \text{ если } x \notin A. \end{cases}$$

Очевидно, что функция f вычислима, и A сводится к B посредством f.

5. Очевидно, что если A сводится к B посредством функции f, то $(\forall x)\psi_A(x) = \psi_B(f(x))$, так что если множество B перечислимо, т. е. функция ψ_B вычислима, то функция ψ_A также вычислима, т. е. множество A перечислимо.

6. Согласно пункту 1, $\mathbf{N} \leq_m \mathbf{N}$. С другой стороны, если A сводится к $\mathbf{N}$ посредством функции f, то $x \in A \Leftrightarrow f(x) \in \mathbf{N}$. Правая часть этой эквивалентности выполняется для любого x, значит, левая тоже, и $A = \mathbf{N}$.

7. $A \leq_m \emptyset \Leftrightarrow \overline{A} \leq_m \mathbf{N} \Leftrightarrow \overline{A} = \mathbf{N} \Leftrightarrow A = \emptyset$.

8. Пусть $\mathbf{N}$ сводится к A посредством функции f. Тогда $A = Ran(f)$, так что $A \neq \emptyset$. С другой стороны, если $A \neq \emptyset$, возьмем произвольное $c \in A$. Тогда $\mathbf{N}$ сводится к A посредством функции, тождественно равной c.

9. $\emptyset \leq_m A \Leftrightarrow \mathbf{N} \leq_m \overline{A} \Leftrightarrow \overline{A} \neq \emptyset \Leftrightarrow A \neq \mathbf{N}$.

Теорема 18.1 доказана.

Следующая теорема показывает, что множество K в некотором смысле играет ключевую роль среди перечислимых множеств.

Теорема 18.2 *Множество $A \subseteq \mathbf{N}$ перечислимо тогда и только тогда, когда $A \leq_m K$.*

Доказательство. Тот факт, что если $A \leq_m K$, то A перечислимо, вытекает из пункта 5 теоремы 18.1 и того факта, что множество K перечислимо. Обратно, пусть $A \subseteq \mathbf{N}$ — произвольное перечислимое множество. Двуместную числовую функцию g определим следующим образом:

$$g(x, y) = \begin{cases} 1, \text{ если } x \in A; \\ \text{не определено, если } x \notin A. \end{cases}$$

Очевидно, что функция g вычислима. По теореме о параметризации существует одноместная тотальная вычислимая функция f такая, что для любых x, y выполняется условное равенство $g(x,y) \simeq \varphi_{f(x)}(y)$. Отсюда и из определения функции g немедленно следует, что если $x \in A$, то функция $\varphi_{f(x)}$ всюду определена, в частности, определено значение $\varphi_{f(x)}(f(x))$, т. е. $f(x) \in K$, а если $x \notin A$, то функция $\varphi_{f(x)}$ нигде не определена, и $f(x) \notin K$. Таким образом, $(\forall x)[x \in A \Leftrightarrow f(x) \in K]$, т. е. $A \leq_m K$. Теорема 18.2 доказана.

Задачи.

1. Пусть c — фиксированное число, $B = \{x | c \in W_x\}$. Доказать, что $K \leq_m B$.
2. Пусть A — множество всех гёделевых номеров тотальных одноместных вычислимых числовых функций. Доказать, что A не m-сводится к K.
3. Доказать, что если A — перечислимое множество, не являющееся разрешимым, то $\overline{A}$ не m-сводится к A, и A не m-сводится к $\overline{A}$.

19 Продуктивные множества

Как мы знаем, дополнение $\overline{K}$ множества K неперечислимо. Этот факт может быть выражен в более сильной форме: существует алгоритм, который позволяет по гёделеву номеру x любого перечислимого подмножества W_x множества $\overline{K}$ найти натуральное число из $\overline{K}$, не принадлежащее W_x. А именно, в качестве такого числа можно взять x. Действительно, пусть $W_x \subseteq \overline{K}$. Предположим, что $x \in W_x$. Тогда $x \in K$, что невозможно. Значит, $x \notin W_x$, следовательно, $x \in \overline{K}$ и $x \in (\overline{K} \setminus W_x)$. Это свойство множества $\overline{K}$ приводит к следующему общему определению: множество $A \subseteq \mathbf{N}$ называется *продуктивным*, если существует одноместная вычислимая числовая функция ψ такая, что

$$(\forall x)[W_x \subseteq A \Rightarrow (!\psi(x) \,\&\, \psi(x) \in A \setminus W_x)].$$

В этом случае функция ψ называется *продуктивной функцией* для множества A. Непосредственно из определения вытекает, что продуктивное множество неперечислимо. Как показано выше, множество $\overline{K}$ является продуктивным, причем тождественная функция $\psi(x) = x$ является продуктивной функцией для $\overline{K}$. Много других примеров продуктивных множеств можно получить с помощью следующей теоремы.

Теорема 19.1 *Пусть множество A продуктивно, и $A \leq_m B$. Тогда множество B продуктивно.*

Доказательство. Пусть A сводимо к B посредством функции f. Рассмотрим двуместную функцию $h(x,y) \simeq \varphi_x(f(y))$. Очевидно, что эта функция вычислима. По теореме о параметризации существует одноместная тотальная вычислимая функция k такая, что для любых x, y выполняется условное равенство $h(x,y) \simeq \varphi_{k(x)}(y)$. Отсюда и из определения функции h следует, что значение $\varphi_{k(x)}(y)$ определено тогда и только тогда, когда определено значение $\varphi_x(f(y))$. Это означает, что $y \in W_{k(x)} \Leftrightarrow f(y) \in W_x$, т. е. $W_{k(x)} = f^{-1}(W_x)$. Допустим теперь, что $W_x \subseteq B$. Тогда из определения m-сводимости следует, что

$$W_{k(x)} = f^{-1}(W_x) \subseteq A.$$

Пусть g — продуктивная функция для A. Тогда $g(k(x)) \in (A \setminus W_{k(x)})$, откуда следует, что

$$f(g(k(x))) \in (B \setminus W_x).$$

Следовательно, функция $\psi(x) = f(g(k(x)))$ является продуктивной для множества B. Теорема 19.1 доказана.

Рассмотрим некоторые применения теоремы 19.1.

Пример 1. В разделе 18 было доказано, что множество K m-сводимо к множеству гёделевых номеров функции $\mathbf{0}$, тождественно равной нулю. Следовательно, множество $\overline{K}$ m-сводимо к множеству $\{x | \varphi_x \neq \mathbf{0}\}$, которое, таким образом, оказывается продуктивным.

Пример 2. Как было замечено в разделе 18, множество K m-сводимо к индексному множеству $I_{\mathcal{F}}$ любого нетривиального семейства вычислимых функций $\mathcal{F}$, не содержащего нигде не определенную функцию. Отсюда следует, что если нетривиальное семейство вычислимых функций $\mathcal{F}$ содержит нигде не определенную функцию, то $\overline{K} \leq_m I_{\mathcal{F}}$, следовательно, в этом случае множество $I_{\mathcal{F}}$ является продуктивным.

Теорема 19.2 *Любое продуктивное множество содержит бесконечное перечислимое подмножество.*

Доказательство. Пусть A — продуктивное множество, g — его продуктивная функция. Последовательность $f : \mathbf{N} \to \mathbf{N}$ определим следующим образом. Пусть e_0 — гёделев номер программы для МНР, состоящей из единственной команды

$$J(1, 1, 1).$$

Очевидно, что в этом случае $W_{e_0} = \emptyset$, следовательно, $W_{e_0} \subseteq A$, и значение $y_0 = g(e_0)$ определено. В качестве $f(0)$ возьмем y_0. При этом, очевидно, $y_0 \in A$. Допустим теперь, что мы уже вычислили значения $f(i) = y_i$ для $i = 0, \ldots, n$, причем $\{y_0, \ldots, y_n\} \subseteq A$. Пусть e_{n+1} — гёделев номер программы для МНР, завершающей работу только на исходных данных $y_0, \ldots, y_n$. Очевидно, что в этом случае $W_{e_{n+1}} = \{y_0, \ldots, y_n\}$, следовательно, $W_{e_{n+1}} \subseteq A$, и значение $y_{n+1} = g(e_{n+1})$ определено. В качестве $f(n+1)$ возьмем y_{n+1}. При этом, очевидно, $y_{n+1} \in A$ и $y_{n+1} \notin \{y_0, \ldots, y_n\}$. Таким образом, f — вычислимая последовательность, множество ее значений $Ran(f)$ бесконечно, перечислимо и содержится в A. Теорема 19.2 доказана.

Задачи.

1. Доказать, что следующие множества продуктивны:
   (a) $\{x | c \notin W_x\}$ (c — фиксированное число);
   (b) $\{x | c \notin Ran(\varphi_x)\}$ (c — фиксированное число);
   (c) $\{x | W_x$ конечно$\}$;
   (d) $\{x | \varphi_x$ не сюръективна$\}$;
   (e) $\{x | \varphi_x$ инъективна$\}$;
   (f) $\{x | \varphi_x$ тотальна$\}$;
   (g) $\{x | \varphi_x$ является многочленом$\}$;
   (h) $\{x | \varphi_x$ не является многочленом$\}$.
2. Доказать, если A перечислимо, а множество $A \cap B$ продуктивно, то B также продуктивно.
3. Доказать, что любое продуктивное множество содержит бесконечное разрешимое подмножество.

20 Креативные множества

Множество $A \subseteq \mathbf{N}$ называется *креативным* (или *творческим*), если оно перечислимо, а его дополнение $\overline{A}$ продуктивно. Самым простым примером креативного множества является множество K. Другим примером может служить множество

$$A = \{x | \varphi_x(x) = 0\}.$$

Очевидно, что это множество перечислимо. Построим продуктивную функцию для его дополнения. Для этого рассмотрим двуместную функцию f, определяемую следующим образом:

$$f(x, y) = \begin{cases} 0, & \text{если значение } \varphi_x(y) \text{ определено;} \\ & \text{не определено в противном случае.} \end{cases}$$

Очевидно, что эта функция вычислима. По теореме о параметризации существует одноместная тотальная вычислимая функция g такая, что для любых x, y выполняется условное равенство $f(x, y) \simeq \varphi_{g(x)}(y)$. Отсюда и из определения функции f следует, что значение $\varphi_{g(x)}(g(x))$ определено (и равно 0) тогда и только тогда, когда определено значение $\varphi_x(g(x))$. Таким образом, $g(x) \in A \Leftrightarrow g(x) \in W_x$, так что если $W_x \subseteq \overline{A}$, то должно выполняться условие $g(x) \in (\overline{A} \setminus W_x)$, т. е. g — продуктивная функция для $\overline{A}$. Много других примеров креативных множеств можно получить с помощью следующей теоремы.

Теорема 20.1 *Пусть $\mathcal{F}$ — нетривиальное семейство одноместных числовых вычислимых функций, A — его индексное множество. Если множество A перечислимо, то оно креативно.*

Доказательство. Если нигде не определенная функция не принадлежит семейству $\mathcal{F}$, то, как отмечалось выше, из доказательства теоремы Райса следует, что множество A продуктивно, следовательно, не может быть перечислимым. Значит, нигде не определенная функция принадлежит дополнению семейства $\mathcal{F}$, а тогда дополнение множества A продуктивно. Следовательно, A креативно. Теорема 20.1 доказана.

Непосредственным применением этой теоремы доказывается, например, что множество $\{x | c \notin W_x\}$ и $\{x | c \notin Ran(\varphi_x)\}$ (c — фиксированное число) является креативным.

Задачи.

1. Доказать, что следующие множества креативны:
   (a) $\{x|W_x \neq \emptyset\}$;
   (b) $\{x|x \in Ran(\varphi_x)\}$;
   (c) $\{x|\varphi_x$ не инъективна$\}$;
   (d) $\{x|\varphi_x(x) = f(x)\}$, где f — произвольная тотальная вычислимая функция.
2. Доказать, что если множество A креативно, а множество B перечислимо, причем $A \cap B = \emptyset$, то множество $A \cup B$ креативно.

21 Простые множества

Множество $A \subseteq \mathbf{N}$ называется *простым*, если выполнены следующие условия:

- A перечислимо;
- $\overline{A}$ бесконечно;
- $\overline{A}$ не содержит бесконечного перечислимого подмножества.

Теорема 21.1 *Простое множество не является ни разрешимым, ни креативным.*

Доказательство. Пусть A — простое множество. В силу определения простого множества, его дополнение не перечислимо. Отсюда по теореме Поста получаем, что A не разрешимо. Из теоремы 19.2 и определения простого множества следует, что множество $\overline{A}$ не является простым, следовательно, A не является креативным. Теорема 21.1 доказана.

Теорема 21.2 *Существует простое множество.*

Доказательство. Одноместную числовую функцию f определим следующим алгоритмом ее вычисления. Пусть дано натуральное число x. Чтобы найти $f(x)$, будем последовательно вычислять значения

$$\varphi_x(0), \varphi_x(1), \ldots$$

(при этом вычисление $\varphi_x(n+1)$ начинается только после того, как вычисление $\varphi_x(n)$ завершилось); вычисления прекращаются, если найдено такое число n, что $\varphi_x(n) > 2n$; в этом случае полагаем $f(x) = \varphi_x(n)$. Таким образом, f — вычислимая функция. Тогда множество ее значений перечислимо. Докажем, что множество $A = Ran(f)$ является простым. Пусть B — произвольное бесконечное перечислимое множество. Тогда B является множеством значений некоторой вычислимой последовательности φ_b. Из определения функции f видно, что в этом случае значение $f(b)$ определено и $f(b) \in B$. Значит, B не содержится в дополнении множества A. Чтобы убедиться, что множество $\overline{A}$ бесконечно, заметим, что если значение $f(x)$ определено, то $f(x) > 2x$. Таким образом, для любого n в множестве $\{0, 1, 2, \ldots, 2n\}$ содержатся не менее n элементов из $\overline{A}$. Следовательно, множество $\overline{A}$ бесконечно. Теорема 21.2 доказана.

22 m-полные перечислимые множества

Множество $A \subseteq \mathbf{N}$ называется *m-полным*, если оно перечислимо, и любое другое перечислимое множество m-сводимо к A. В разделе 18 было показано, что множество K является m-полным. Отсюда и из транзитивности отношения $\leq_m$ следует, что перечислимое множество A является m-полным тогда и только тогда, когда $K \leq_m A$.

Множество K перечислимо, но не разрешимо. Возникает естественный вопрос: всякое ли неразрешимое перечислимое множество является m-полным? Сейчас мы получим ответ на этот вопрос.

Теорема 22.1 *Любое m-полное множество креативно.*

Доказательство. Если множество A m-полно, то $K \leq_m A$. Следовательно, $\overline{K} \leq_m \overline{A}$. Так как $\overline{K}$ продуктивно, то по теореме 19.1 множество $\overline{A}$ также продуктивно, а тогда A креативно. Теорема 22.1 доказана.

Имеет место и обратное утверждение: всякое креативное множество является m-полным. Однако доказательство этого утверждения требует более глубокого изучения теории алгоритмов и не входит в наши планы.

Теорема 22.2 *Простые множества не являются m-полными.*

Доказательство. По теореме 22.1 любое m-полное множество креативно, в то же время теорема 21.1 утверждает, что простые множества не являются креативными. Теорема 22.2 доказана.

Таким образом, простые множества перечислимы, не разрешимы, но не являются m-полными.

23 Язык формальной арифметики

Сейчас мы рассмотрим некоторые применения теории алгоритмов в математической логике. Как известно, математическая логика математическими методами изучает математические рассуждения. Всякое рассуждение есть в некотором смысле логически правильная последовательность утверждений. Чтобы сделать математические утверждения точными математическими объектами, для их записи в математической логике разработаны искусственные, формализованные языки. Мы рассмотрим один из таких языков — *язык формальной прифметики*, предназначенный для записи утверждений о натуральных числах. Как и всякий формализованный язык, язык формальной арифметики использует фиксированный *алфавит*. Алфавит языка формальной арифметики содержит следующие символы: *константы* 0 и 1, *символы для операций* сложения $+$ и умножения $\cdot$, *символ для отношения* равенства $=$, а также символы для *логических операций* $\neg$ (отрицание), $\&$ (конъюнкция), $\vee$ (дизъюнкция), $\supset$ (импликация) и *кванторов* всеобщности $\forall$ и существования $\exists$. Кроме того, в языке формальной арифметики имеются счетное множество *переменных* $V = \{v_0, v_1, v_2, \ldots\}$ и скобки (,).

Среди всех слов в описанном алфавите различаются два типа осмысленных выражений — термы и формулы. *Терм* — это формальный аналог числовой формы. Термы строятся по следующим правилам:

- константы 0 и 1 являются термами;
- всякая переменная является термом;
- если t_1 и t_2 — термы, то выражения $(t_1 + t_2)$ и $(t_1 \cdot t_2)$ являются термами.

Например, термами являются выражения 0, 1, (1+1), ((1+1)+1) и т. д. Очевидно, что эти термы являются записями чисел 0, 1, 2, 3 и т. д. В дальнейшем мы будем через $\overline{n}$ обозначать терм, являющийся записью натурального числа n. Например, $\overline{7}$ есть ((((((1+1)+1)+1)+1)+1)+1)+1).

Формула является формальным аналогом высказывания или высказывательной формы. Формулы строятся по следующим правилам:

- если t_1 и t_2 — термы, то выражение $t_1 = t_2$ является формулой;
- если Φ — формула, то выражение $\neg\Phi$ является формулой;
- если Φ и Ψ — формулы, то выражения $(\Phi \,\&\, \Psi)$, $(\Phi \vee \Psi)$, $(\Phi \supset \Psi)$ являются формулами;
- если Φ — формула, x — переменная, то выражения $\forall x\Phi$ и $\exists x\Phi$ являются формулами.

Например, формулой является выражение $\exists v_1(v_1 \cdot (1+1)) = v_0$. Очевидно, что эта формула является записью высказывательной формы "v_0 — четное число". Формула $\exists v_1(v_1 \cdot \overline{2}) = \overline{5}$ выражает ложное высказывание, а формула $\exists v_1(v_1 \cdot \overline{2}) = \overline{4}$ выражает истинное высказывание.

Выражает ли данная формула высказывание или высказывательную форму, зависит от присутствия в ней свободных вхождений переменных. Свободные и связанные вхождения переменных в формулу различаются следующим образом. В формуле вида $\forall x\Phi$ или $\exists x\Phi$ ее часть $\forall x$ или $\exists x$ называется *кванторной приставкой*, а Φ — ее *областью действия*. Вхождение переменной x называется *связанным*, если оно находится в кванторной приставке $\forall x$ или $\exists x$ или в ее области действия. Вхождение переменной называется *свободным*, если оно не является связанным.

Формула, не содержащая свободных вхождений переменных, называется *замкнутой формулой* или *высказыванием*. Очевидно, что всякое высказывание оказывается истинным или ложным при естественном понимании всех входящих в него символов. Обозначим через $\mathcal{T}$ множество всех истинных высказываний, а через $\mathcal{F}$ — множество всех ложных высказываний.

Если формула $\Phi(x_1, \ldots, x_n)$ не содержит свободных вхождений переменных, отличных от $x_1, \ldots, x_n$, то она превращается в высказывание $\Phi(\overline{k_1}, \ldots, \overline{k_n})$, если вместо этих свободных вхождений подставить соответственно термы $\overline{k_1}, \ldots, \overline{k_n}$, изображающие натуральные числа $k_1, \ldots, k_n$. Поэтому такая формула может рассматриваться как высказывательная форма. Например, отношение $x < y$ выражается формулой

$$\exists v_0(\neg v_0 = 0 \,\&\, (x + v_0) = y).$$

Поэтому в дальнейшем вместо такой формулы мы иногда будем писать просто $x < y$.

24 Арифметические множества и функции

Пусть $n \geq 1$ — натуральное число. Множество $A \subseteq \mathbf{N}^n$ называется *арифметическим*, если существует арифметическая формула $\Phi_A(v_1, \ldots, v_n)$ со свободными переменными только из списка $v_1, \ldots, v_n$ такая, что для любых натуральных чисел $k_1, \ldots, k_n$ выполняется следующее условие:

$$\langle k_1, \ldots, k_n \rangle \in A \Leftrightarrow \Phi(\overline{k_1}, \ldots, \overline{k_n}) \in \mathcal{T}.$$

В этом случае будем говорить, что формула $\Phi_A(v_1, \ldots, v_n)$ *определяет* множество A.

Теорема 24.1 *Если множества $A, B \subseteq \mathbf{N}^n$ являются арифметическими, то множества $\mathbf{N}^n \setminus A$, $A \cup B$, $A \cap B$ также являются арифметическими.*

Доказательство. Пусть множества A и B определяются соответственно формулами $\Phi_A(v_1, \ldots, v_n)$ и $\Phi_B(v_1, \ldots, v_n)$. Тогда, очевидно, множество $\mathbf{N}^n \setminus A$ определяется формулой $\neg\Phi_A(v_1, \ldots, v_n)$, множество $A \cup B$ определяется формулой $\Phi_A(v_1, \ldots, v_n) \vee \Phi_B(v_1, \ldots, v_n)$, а множество $A \cap B$ определяется формулой $\Phi_A(v_1, \ldots, v_n) \,\&\, \Phi_B(v_1$
Теорема 24.1 доказана.

Частичная функция f из множества $\mathbf{N}^n$ в множество $\mathbf{N}$ называется арифметической, если ее график Γ_f является арифметическим множеством. Формулу, определяющую график арифметической функции f, будем обозначать $\Phi_f(v_1, \ldots, v_n, v_{n+1})$.

Теорема 24.2 *Всякая вычислимая числовая функция является арифметической.*

Доказательство. В силу тезиса Чёрча всякая вычислимая числовая функция является частично-рекурсивной. Поэтому достаточно доказать, что всякая частично-рекурсивная функция является арифметической. Сначала докажем арифметичность базисных функций. Очевидно, что функция $s(x) = x + 1$ определяется формулой $v_2 = (v_1 + 1)$; функция $o(x) = 0$ определяется формулой $v_2 = 0$; функция $I_m^n(x_1, \ldots, x_n) = x_m$ определяется формулой $v_{n+1} = v_m$.

Докажем теперь, что если арифметичны k-местная функция f и n-местные функции $g_1, \ldots, g_k$, то n-местная функция h, получающаяся из них подстановкой, также арифметична. Пусть

$$\Phi_f(v_1, \ldots, v_k, v_{k+1}), \Phi_{g_i}(v_1, \ldots, v_n, v_{n+1})(i = 1, \ldots, k)$$

суть формулы, определяющие соответствующие функции. Тогда, очевидно, функция h определяется следующей формулой:

$$\exists u_1 \ldots \exists u_k(\Phi_{g_1}(v_1, \ldots, v_n, u_1) \,\&\, \ldots \,\&\, \Phi_{g_k}(v_1, \ldots, v_n, u_k) \,\&\, \Phi_f(u_1, \ldots, u_k, v_{n+1})).$$

Докажем, что если n-местная ($n \geq 1$) частичная функция g получается с помощью минимизации из $(n+1)$-местной частичной арифметической функции f, то функция g также арифметична. Пусть формула $\Phi_g(v_1, \ldots, v_n, v_{n+1}, v_{n+2})$ определяет функцию g. Тогда, очевидно, функция f определяется следующей формулой:

$$\forall v(v < v_{n+1} \supset \exists w(\Phi_g(v_1, \ldots, v_n, v, w) \,\&\, \neg w = 0)) \,\&\, \Phi_g(v_1, \ldots, v_n, v_{n+1}, 0).$$

Случай, когда $(n+1)$-местная ($n \geq 1$) частичная функция h получается с помощью операции рекурсии из n-местной арифметической функции f и $(n+2)$-местной арифметической функции g, требует некоторой изобретательности. Пусть x, y, z — натуральные числа. Говорят, что число z *сравнимо с числом y по модулю x* и пишут $z \equiv y (\mathrm{mod}\, x)$, если числа z и y дают одинаковые остатки при делении на x или, что то же самое, если их разность $z - y$ делится на x. Отметим следующий известный факт из алгебры: если числа w и x взаимно просты, то существует такое число z, что $wz \equiv 1 (\mathrm{mod}\, x)$.

Лемма 24.1 (китайская теорема об остатках) *Каковы бы ни были натуральные числа $y_1, \ldots, y_k$ и натуральные попарно взаимно простые числа $x_1, \ldots, x_k$, существует натуральное число z такое, что*

$$z \equiv y_1 (\mathrm{mod}\, x_1), \ldots, z \equiv y_k (\mathrm{mod}\, x_k).$$

Доказательство. Пусть $x = x_1 \ldots x_k$. Тогда $x = w_1 x_1 = \ldots w_k x_k$ для подходящих чисел $w_1, \ldots, w_k$. Для любого $i = 1, \ldots, k$ числа w_i и x_i взаимно просты, следовательно, существует такое число z_i, что $w_i z_i \equiv 1 (\mathrm{mod}\, x_i)$. Положим теперь $z = w_1 z_1 y_1 + \ldots + w_k z_k y_k$. Тогда $z \equiv y_1 (\mathrm{mod}\, x_i)$ для любого $i = 1, \ldots, k$. Лемма 24.1 доказана.

Пусть $\mathrm{rm}(x, y)$ обозначает остаток от деления числа y на число x. Рассмотрим следующую функцию: $\beta(x, y, z) = \mathrm{rm}(1 + (z+1) \cdot y, x)$ (обычно ее называют *β-функцией Гёделя*).

Лемма 24.2 *Для любой конечной последовательности натуральных чисел $k_0, k_1, \ldots, k_n$ существуют такие натуральные числа b и c, что для любого $i = 0, 1, \ldots, n$ имеет место равенство $\beta(b, c, i) = k_i$.*

Доказательство. Пусть $j = \max(n, k_0, k_1, \ldots, k_n)$ и $c = j!$. Рассмотрим числа $u_i = 1 + (i+1) \cdot c$ $(i = 0, 1, \ldots, n)$. Они попарно взаимно просты. Действительно, если бы числа u_l и u_m, где $1 \leq l < m \leq n$, имели простой общий делитель p, то p было бы делителем их разности $u_m - u_l = (m-l) \cdot c$, а значит, и делителем хотя бы одного из чисел $m - l$ и c. Но так как $m - l \leq j$, то в любом случае c делится на p. Однако очевидно, что тогда ни u_l, ни u_m не могут делится на p. Полученное противоречие показывает, что на самом деле числа u_l и u_m взаимно просты. Согласно китайской теореме об остатках (лемма 24.1), существует число b такое, что для любого $i = 0, 1, \ldots, n$ имеет место равенство $b \equiv k_i (\mathrm{mod}\, u_i)$, т. е. число b дает такой же остаток при делении на u_i, как и число k_i. Заметим, что для любого $i = 0, 1, \ldots, n$ выполняются условия $k_i \leq j \leq j! = c < 1 + (i+1) \cdot c = u_i$, т. е. $k_i < u_i$. Это означает, что $\mathrm{rm}(u_i, k_i) = k_i$. Теперь имеем:

$$\beta(b, c, i) = \mathrm{rm}(1 + (i+1) \cdot c, b) = \mathrm{rm}(u_i, b) = \mathrm{rm}(u_i, k_i) = k_i,$$

что и требовалось доказать. Лемма 24.2 доказана.

Заметим, что функция β является арифметической: она определима формулой

$$\exists v_5 v_1 = (((1 + ((v_3 + 1) \cdot v_2)) \cdot v_5) + v_4) \,\&\, v_4 < ((1 + ((v_3 + 1) \cdot v_2))),$$

которая, естественно, обозначается $\Phi_\beta(v_1, v_2, v_3, v_4)$.

Вернемся к доказательству теоремы 24.2. Пусть $(n+1)$-местная $(n \geq 1)$ частичная функция h получается с помощью рекурсии из n-местной арифметической функции f и $(n+2)$-местной арифметической функции g. Пусть $\Phi_f(v_1, \ldots, v_n, v_{n+1}, v_{n+2})$, $\Phi_g(v_1, \ldots, v_{n+2}, v_{n+3})$ — формулы, определяющие соответствующие функции. Тогда, очевидно, функция h определяется следующей формулой:

$$\exists u \exists v ((\exists w (\Phi_\beta(u, v, 0, w) \,\&\, \Phi_f(v_1, \ldots, v_n, w))) \,\&\, \Phi_\beta(u, v, v_{n+1}, v_{n+2}) \,\&$$

$$\&\, \forall w (w < v_{n+1} \supset \exists y \exists z (\Phi_\beta(u, v, w, y) \,\&$$

$$\&\, \Phi_\beta(u, v, (w+1), z) \,\&\, \Phi_g(v_1, \ldots, v_n, w, y, z)))).$$

Отдельно рассмотрим случай, когда одноместная функция h получается рекурсией из числа a и двуместной функции g. Пусть $\Phi_g(v_1, v_2, v_3)$ — формула, определяющая функцию g. Тогда функция h определяется следующей формулой:

$$\exists u \exists v (\Phi_\beta(u, v, 0, a) \,\&\, \Phi_\beta(u, v, v_1, v_2) \,\&\, \forall w (w < v_1 \supset \exists y \exists z (\Phi_\beta(u, v, w, y) \,\&$$

$$\&\, \Phi_\beta(u, v, (w+1), z) \,\&\, \Phi_g(w, y, z)))).$$

Итак, мы доказали, что все базисные функции являются арифметическими, а с помощью операций подстановки, рекурсии и минимизации из арифметических функций получаются только арифметические функции. Следовательно, всякая частично-рекурсивная функция арифметична. Теорема 24.2 доказана.

Теорема 24.3 *Всякое перечислимое множество $A \subseteq \mathbf{N}$ является арифметическим.*

Доказательство. Пусть множество $A \subseteq \mathbf{N}$ перечислимо. Тогда оно является областью определения некоторой вычислимой функции f. По теореме 24.2, функция f арифметична. Пусть $\Phi_f(v_1, v_2)$ — формула, определяющая функцию f. Так как A является областью определения функции f, то, очевидно, формула $\exists v_2 \Phi_f(v_1, v_2)$ определяет множество A. Теорема 24.3 доказана.

Теорема 24.4 *Существует неперечислимое арифметическое множество.*

Доказательство. Пусть A — какое-нибудь неразрешимое перечислимое множество. По теореме 24.3 оно является арифметическим. По теореме 24.1 его дополнение также арифметично, но оно, очевидно, неперечислимо. Теорема 24.4 доказана.

Задачи.

1. Доказать, что множество всех простых чисел является арифметическим.
2. Доказать, что если множества $A, B \subseteq \mathbf{N}$ арифметичны, то их разность $A \setminus B$ также арифметична.
3. Для следующих функций написать определяющие их арифметические формулы:

а) $f_1(x) = \begin{cases} x^2, \text{ если } x \text{ четно}, \\ x+1, \text{ если } x \text{ нечетно}; \end{cases}$ б) $f(x,y) = x^y$ (здесь $0^0 = 1$); в) $f(x) = x!$ (здесь $0! = 1$);

г) $\mathrm{sg}(x) = \begin{cases} 1, \text{ если } x > 0; \\ 0, \text{ если } x = 0; \end{cases}$ д) $\overline{\mathrm{sg}}(x) = \begin{cases} 0, \text{ если } x > 0; \\ 1, \text{ если } x = 0; \end{cases}$ е) $p(x) = \begin{cases} x-1, \text{ если } x > 0; \\ 0, \text{ если } x = 0; \end{cases}$

ж) $d(x) = \begin{cases} x-y, \text{ если } x \geq y; \\ 0, \text{ если } x < y; \end{cases}$ з) $|x-y|$; и) $\max(x,y)$; к) $\min(x,y)$; л) $x \dot{-} y$; м) $\frac{x}{y}$; н) $\sqrt[y]{x}$;

о) $\frac{x}{2}$; п) $\left[\frac{x}{2}\right]$.

25 Гёделева нумерация арифметических формул

Каждому символу α из алфавита языка формальной арифметики сопоставим натуральное число $g(\alpha)$, называемое *гёделевым номером* символа α: $g(0) = 3$, $g(1) = 5$, $g(+) = 7$, $g(\cdot) = 9$, $g(=) = 11$, $g(\neg) = 13$, $g(\&) = 15$, $g(\vee) = 17$, $g(\supset) = 19$, $g(\forall) = 21$, $g(\exists) = 23$, $g(() = 25$, $g()) = 27$, $g(v_i) = 29 + 2i$ $(i = 0, 1, 2, \ldots)$. Таким образом, различным символам поставлены в соответствие различные нечетные натуральные числа.

Каждому слову $w = \alpha_0\alpha_1 \ldots \alpha_n$ в алфавите языка формальной арифметики сопоставим натуральное число

$$g(w) = 2^{\alpha_0} \cdot 3^{\alpha_1} \cdot \ldots \cdot p_n^{\alpha_n},$$

где p_n есть n-е простое число, если считать $p_0 = 2$. Число $g(w)$ будем называть гёделевым номером слова w. В частности, каждый терм t имеет некоторый гёделев номер $g(t)$, и каждая формула Φ имеет некоторый гёделев номер $g(\Phi)$. Например, $g(v_0 = v_1) = 2^{29} \cdot 3^{11} \cdot 5^{31}$. Заметим, что в силу единственности разложения натуральных чисел в произведения степеней простых чисел различные выражения получают различные гёделевы номера. Кроме того, гёделевы номера выражений чётны и потому отличны от гёделевых номеров символов. Такая нумерация символов и выражений языка формальной арифметики была впервые введена Гёделем с целью замены высказываний о математических утверждениях на высказывания о натуральных числах.

Рассмотренный здесь способ построения гёделевых номеров не является единственно возможным. Главной отличительной особенностью этой и других гёделевых нумераций является существование алгоритма, позволяющего по любому выражению найти его гёделев номер, и алгоритма, который для любого натурального числа n позволяет определить, является ли n гёделевым номером какого-нибудь выражения и, если является, найти это выражение.

Если $\mathcal{X}$ — некоторое множество выражений языка формальной арифметики, ему соответствует множество $I_{\mathcal{X}} \subseteq \mathbf{N}$, состоящее в точности из гёделевых номеров выражений, входящих в $\mathcal{X}$. Будем называть $I_{\mathcal{X}}$ *индексным множеством* множества выражений $\mathcal{X}$. Теперь все эпитеты, которыми награждаются числовые множества, могут быть применены к множествам выражений языка формальной арифметики, так что множество выражений $\mathcal{X}$ будет называться разрешимым, перечислимым, продуктивным, креативным, арифметическим и т. д., если его индексное множество $I_{\mathcal{X}}$ является соответственно разрешимым, перечислимым, продуктивным, креативным, арифметическим и т. д.

26 Теорема Тарского

С чисто математической точки зрения наиболее интересен вопрос о свойствах множества $\mathcal{T}$ всех истинных высказываний. В некотором смысле окончательный ответ на этот вопрос дает следующая теорема.

Теорема 26.1 (теорема Тарского) *Множество всех истинных высказываний языка формальной арифметики не является арифметическим.*

Доказательство. Допустим, что множество $\mathcal{T}$ всех истинных высказываний языка формальной арифметики арифметично, и $\Phi_{\mathcal{T}}(v_1)$ — формула, определяющая множество $I_{\mathcal{T}}$, т. е. множество гёделевых номеров

всех истинных формул. Последнее означает, что каково бы ни было натуральное число n, формула $\Phi_{\mathcal{T}}(\overline{n})$ истинна тогда и только тогда, когда n является гёделевым номером истинной замкнутой формулы.

Рассмотрим следующую двуместную частичную числовую функцию:

$$sub(x,y) = \begin{cases} g(\Phi(\overline{y})), \text{ если } x = g(\Phi(v_1)) \text{ для некоторой формулы } \Phi; \\ \text{не определено в противном случае.} \end{cases}$$

Очевидно, что эта функция вычислима. Она вычисляется следующим алгоритмом: "Пусть даны натуральные числа x и y. Определите, является ли x гёделевым номером какой-либо формулы $\Phi(v_1)$. Если это так, то вместо всех свободных вхождений переменной v_1 в формулу $\Phi(v_1)$ подставьте терм, изображающий число y. Вычислите гёделев номер полученной формулы. Это и есть значение $sub(x,y)$."Поскольку функция sub вычислима, то в силу теоремы 24.2 она арифметична. Пусть $\Phi_{sub}(v_1,v_2,v_3)$ — формула, определяющая функцию sub. Рассмотрим формулу $\exists v_0(\Phi_{sub}(v_1,v_1,v_0)\,\&\,\neg\Phi_{\mathcal{T}}(v_0))$, которую обозначим $\Psi(v_1)$. Эта формула имеет свободные вхождения только переменной v_1 и, очевидно, утверждает, что замкнутая формула, полученная подстановкой терма $\overline{v_1}$ вместо свободных вхождений переменной "v_1"в формулу с гёделевым номером v_1, ложна. Пусть $m = g(\Psi(v_1))$. Тогда $sub(m,m) = g(\Psi(\overline{m}))$. Допустим, что высказывание $\Psi(\overline{m})$ истинно. Это высказывание имеет вид $\exists v_0(\Phi_{sub}(\overline{m},\overline{m},v_0)\,\&\,\neg\Phi_{\mathcal{T}}(v_0))$, и его истинность означает, что существует такое число a, для которого истинно высказывание $\Phi_{sub}(\overline{m},\overline{m},\overline{a})\,\&\,\neg\Phi_{\mathcal{T}}(\overline{a})$. Следовательно, $a = sub(m,m)$, и истинно высказывание $\neg\Phi_{\mathcal{T}}(\overline{sub(m,m)})$. Но это означает, что высказывание с номером $sub(m,m)$, т. е. $\Psi(\overline{m})$, ложно. Полученное противоречие показывает, что высказывание $\Psi(\overline{m})$ не может быть истинным. Значит, оно ложно. Ложность высказывания $\exists v_0(sub(\overline{m},\overline{m},v_0)\,\&\,\neg\Phi_{\mathcal{T}}(v_0))$ означает, что каково бы ни было натуральное число a, если истинно высказывание $\Phi_{sub}(\overline{m},\overline{m},\overline{a})$, то высказывание $\neg\Phi_{\mathcal{T}}(\overline{a})$ ложно. В частности, когда $a = sub(m,m)$, получаем, что ложно высказывание $\neg\Phi_{\mathcal{T}}(\overline{sub(m,m)})$. Но это означает, что истинно высказывание $\Phi_{\mathcal{T}}(\overline{sub(m,m)})$, а тогда истинно и высказывание с гёделевым номером $sub(m,m)$, т. е. $\Psi(\overline{m})$. Опять получили противоречие. Значит, высказывание $\Psi(\overline{m})$ не истинно и не ложно, чего не может быть. Итак, наше предположение об арифметичности множества $\mathcal{T}$ и существовании формулы $\Phi_{\mathcal{T}}(v_1)$ неверно, и на самом деле множество $\mathcal{T}$ неарифметично. Теорема 26.1 доказана.

Теорема 26.2 *Множество $\mathcal{T}$ всех истинных высказываний языка формальной арифметики не является перечислимым.*

Доказательство. По теореме 24.3, всякое перечислимое множество является арифметическим. В силу теоремы Тарского множество $\mathcal{T}$ неарифметично, значит, оно и неперечислимо. Теорема 26.2 доказана.

27 Первая теорема Гёделя о неполноте

Теорема 26.2 имеет самое непосредственное отношение к проблеме задания формальной арифметики в виде аксиоматической системы. *Аксиоматический метод* построения математической теории состоит в том, что некоторые исходные факты этой теории, называемые *аксиомами* или *постулатами*, принимаются "без доказательства", а все другие утверждения этой теории выводятся из них путем рассуждений. В 1891 г. итальянский математик Пеано предложил аксиоматику для натурального ряда. На языке формальной арифметики аксиомы Пеано записываются следующим образом:

1. $\forall x\forall y((x+1)=(y+1)\supset x=y)$;
2. $\forall x\neg(x+1)=0$;
3. $\forall x(x+0)=x$;
4. $\forall x\forall y(x+(y+1))=((x+y)+1)$;
5. $\forall x(x\cdot 0)=0$;
6. $\forall x\forall y(x\cdot(y+1))=((x\cdot y)+x)$;
7. $\Phi(0)\,\&\,\forall x(\Phi(x)\supset\Phi(x+1))\supset\forall x\Phi(x)$.

При этом 7) представляет собой не отдельную аксиому, а *схему аксиом*: какова бы ни формула $\Phi(x)$, выражение 7) является аксиомой.

В математической логике разработаны средства формального логического вывода, позволяющие из данных аксиом по определенным правилам получать новые высказывания, логически вытекающие из аксиом и называемые *теоремами*. Одним из таких средств является *исчисление предикатов*, изучаемое в курсе

математической логики. В рамках этого исчисления вводится понятие *доказательства* как конечной последовательности формул, строящейся по определенным правилам. При этом, если множество всех аксиом разрешимо (как, например, в случае аксиом Пеано), то множество всех доказательств также оказывается разрешимым. Формула Φ считается доказуемой, если существует доказательство, оканчивающееся этой формулой. Правила вывода исчисления предикатов позволяют на основе истинных аксиом доказывать только истинные высказывания. Нетрудно убедиться, что множество всех доказуемых высказываний является перечислимым. Следовательно, в силу теоремы 26.2, оно не совпадает с множеством всех истинных высказываний. Приведенное рассуждение представляет собой доказательство первой теоремы Гёделя о неполноте, которая в упрощенном виде формулируется следующим образом:

Теорема 27.1 *Пусть задана аксиоматическая теория в языке формальной арифметики с разрешимым множеством аксиом, в которой все доказуемые высказывания истинны. Тогда существует высказывание* Φ*, которое является истинным, но не доказуемым.*

Как построить конкретное истинное высказывание, которое не доказуемо в данной аксиоматической теории с разрешимым множеством аксиом? Ответ на этот вопрос получается из доказательства следующей теоремы.

Теорема 27.2 *Множество* $\mathcal{T}$ *всех истинных высказываний языка формальной арифметики является продуктивным.*

Доказательство. Пусть дано перечислимое подмножество $\mathcal{X}$ множества $\mathcal{T}$. Так как индексное множество $I_{\mathcal{X}}$ множества $\mathcal{X}$ перечислимо, то оно арифметично, причем по его гёделеву номеру можно построить формулу $\Phi_{\mathcal{X}}(v_1)$, определяющую множество $I_{\mathcal{X}}$. Рассмотрим формулу $\exists v_0(sub(v_1, v_1, v_0)\,\&\,\neg\Phi_{\mathcal{X}}(v_0))$, которую обозначим $\Psi(v_1)$. Пусть $m = g(\Psi(v_1))$. Тогда, очевидно, $sub(m, m) = g(\Psi(\overline{m}))$. Допустим, что высказывание $\Psi(\overline{m})$ ложно. Оно имеет вид $\exists v_0(sub(\overline{m}, \overline{m}, v_0)\,\&\,\neg\Phi_{\mathcal{X}}(v_0))$. Его ложность означает, что каково бы ни было натуральное число a, если истинно высказывание $\Phi_{sub}(\overline{m}, \overline{m}, \overline{a})$, то высказывание $\neg\Phi_{\mathcal{X}}(\overline{a})$ ложно. В частности, когда $a = sub(m, m)$, получаем, что ложно высказывание $\neg\Phi_{\mathcal{X}}(\overline{sub(m, m)})$. Но это означает, что истинно высказывание $\Phi_{\mathcal{X}}(\overline{sub(m, m)})$, а тогда оказывается, что высказывание с гёделевым номером $sub(m, m)$, т. е. $\Psi(\overline{m})$, принадлежит множеству $\mathcal{X}$, состоящему только из истинных высказываний. Значит, высказывание $\Psi(\overline{m})$ истинно и принадлежит множеству $\mathcal{T}$. С другой стороны, это высказывание означает, что существует такое число a, для которого истинно высказывание $\Phi_{sub}(\overline{m}, \overline{m}, \overline{a})\,\&\,\neg\Phi_{\mathcal{X}}(\overline{a})$. Отсюда следует, что $a = sub(m, m)$, и тогда истинно высказывание $\neg\Phi_{\mathcal{X}}(\overline{sub(m, m)})$. Но это означает, что высказывание с номером $sub(m, m)$, т. е. $\Psi(\overline{m})$, не принадлежит множеству $\mathcal{X}$. Итак, по гёделеву номеру перечислимого подмножества $I_{\mathcal{X}}$ множества $I_{\mathcal{T}}$ мы можем эффективно, т. е. с помощью алгоритма, построить число из $\mathcal{T} \setminus \mathcal{X}$, а именно, в качестве такого числа нужно взять гёделев номер построенной нами формулы $\Psi(\overline{m})$. Теорема 27.2 доказана.

Теорема 27.2 имеет и другое, более простое доказательство. В силу теоремы 19.1 достаточно доказать, что продуктивное множество $\overline{K}$ m-сводимо к множеству $I_{\mathcal{T}}$. Множество $\overline{K}$ является арифметическим. Пусть $\Phi_{\overline{K}}(v_1)$ — определяющая его формула. Для каждого натурального n положим $f(n) = g(\Phi_{\overline{K}}(\overline{n}))$. Множество $\overline{K}$ m-сводится к множеству $I_{\mathcal{T}}$ посредством функции f. Действительно, $n \in \overline{K}$ тогда и только тогда, когда формула $\Phi_{\overline{K}}(\overline{n})$ истинна, т. е. ее гёделев номер $g(\Phi_{\overline{K}}(\overline{n}))$ принадлежит множеству $I_{\mathcal{T}}$. Приведенное доказательство, однако, не дает явного построения продуктивной функции для множества $I_{\mathcal{T}}$ и в этом смысле уступает предыдущему доказательству.

Задачи.

1. Найти гёделев номер терма $\overline{2}$.
2. Найти гёделев номер формулы $(\overline{2} \cdot \overline{2}) = \overline{4}$.
3. Пусть $\mathcal{F}$ — множество всех ложных высказываний языка формальной арифметики. Доказать, что множество $\mathcal{F}$ неарифметично.
4. Доказать, что множество $\mathcal{F}$ продуктивно.

28 Меры сложности вычислений

В реальных вычислениях даже для вычислимой функции f весьма актуальным является вопрос: "Вычислима ли f практически?"Иными словами, существует ли программма, вычисляющая функцию f за время, которым мы располагаем? Конечно, многое зависит от мастерства программиста: неудачно составленная

программа может работать очень долго. Однако есть и объективные факторы, влияющие на скорость вычисления функций. Они изучаются в *теории сложности вычислений*.

Как известно, каждый алгоритм работает по шагам. В рассмотренных нами машинах Тьюринга и машинах с неограниченными регистрами шагом работы может считаться выполнение одной команды. Считая, что каждый шаг выполняется за единицу времени, мы можем определить *время вычисления* как число шагов алгоритма, совершаемых в процессе этого вычисления. Возможны и другие меры сложности вычисления. После того как выбрана мера сложности вычисления, возникает вопрос о сложности вычисления конкретных функций. В частности, представляется естественной проблема построения "наилучшего"алгоритма для вычисления данной функции.

В силу сказанного выше, для каждого натурального числа e и $n \geq 1$ определим функцию $t_e^{(n)}(x_1, \ldots, x_n)$ как число шагов, сделанное при вычислении значения $\varphi_e^{(n)}(x_1, \ldots, x_n)$ с помощью программы с номером e. (Если значение $\varphi_e^{(n)}(x_1, \ldots, x_n)$ не определено, то значение $t_e^{(n)}(x_1, \ldots, x_n)$ также считается неопределенным.) Так определенная функция $t_e^{(n)}$ называется *сигнализирующей функцией*. Отметим следующие простые, но важные свойства сигнализирующей функции $t_e^{(n)}$:

- $Dom(t_e^{(n)}) = Dom(\varphi_e^{(n)})$;
- для каждого n множество
$$\{\langle e, x_1, \ldots, x_n, y\rangle | t_e^{(n)}(x_1, \ldots, x_n) \simeq y\}$$
разрешимо.

Второе из этих свойств явно контрастирует с тем фактом, что множество

$$\{\langle e, x_1, \ldots, x_n, y\rangle | \varphi_e^{(n)}(x_1, \ldots, x_n) \simeq y\}$$

неразрешимо. При $n = 1$ вместо $t_e^{(n)}$ будем писать просто t_e.

Следующее понятие будет часто использоваться в формулировках теорем. Пусть задано некоторое свойство $M(n)$, которым может обладать или не обладать каждое натуральное число. Будем говорить, что свойство $M(n)$ выполняется *для почти всех* n или *почти всюду*, если

$$(\exists n_0)(\forall n \geq n_0) M(n)$$

(иными словами, если это свойство выполняется для всех натуральных чисел, кроме конечного их множества).

Следующая теорема показывает, что существуют сколь угодно сложно вычислимые функции.

Теорема 28.1 *Пусть b — одноместная тотальная вычислимая функция. Существует такая одноместная тотальная вычислимая функция f, принимающая только значения 0 и 1, что*

$$(\forall e)(f = \varphi_e \Rightarrow t_e(n) > b(n) \text{ почти всюду}).$$

Доказательство. Определим частичную одноместную функцию i следующим образом: для любого натурального n

$$i(n) = \begin{cases} \text{наименьшее } i \leq n \text{ такое, что } i \text{ отличается от всех} \\ \text{определенных значений } i(m) \text{ при } m < n, \text{ и } t_i(n) \leq b(n); \\ \text{не определено в противном случае.} \end{cases}$$

Функция i вычислима. Действительно, чтобы проверить, выполняется ли условие $t_i(n) \leq b(n)$, нужно вычислить $b(n)$ и сделать не более $b(n)$ шагов в вычислении $\varphi_i(n)$. Если за это время вычисление закончилось, то условие $t_i(n) \leq b(n)$ выполнено, в противном случае — нет. Вычислим, например, $i(0)$. Для этого вычислим $b(0)$ и сделаем не более чем $b(0)$ шагов в вычислении $\varphi_0(0)$ (т. е. в работе программы с номером 0 на исходном данном 0). Если за это время вычисление закончилось, полагаем $i(0) = 0$; в противном случае значение $i(0)$ считаем неопределенным. Заметим, что область определения функции i разрешима, и если значение $i(k)$ определено, то $i(k) \leq k$.

Покажем, что если число i таково, что $t_i(m) \leq b(m)$ для бесконечно многих m, то $i = i(n)$ для некоторого n. Пусть

$$p = 1 + \max\{k | i(k) \text{ определено и } i < i(k)\}$$

(положим $p = 0$, если нет ни одного значения $i(k) < i$). Возьмем такое $n \geq i$, что $n \geq p$ и $t_i(n) \leq b(n)$. Если $i = i(k)$ для некоторого $k < n$, то все доказано. Пусть $i \neq i(k)$ для всех $k < n$. Тогда получаем, что $i \leq n$, i

отличается от всех определенных значений $i(m)$ при $m < n$ и $t_i(n) \leq b(n)$. Значит, $i(n)$ определено, причем $i(n) \leq i$. Но поскольку $n \geq p$, то должно быть $i(n) \geq i$. Следовательно, $i(n) = i$, что и требовалось.

Теперь положим для любого натурального n

$$f(n) = \begin{cases} 1, \text{ если } i(n) \text{ определено и } \varphi_{i(n)}(n) = 0; \\ 0 \text{ в противном случае.} \end{cases}$$

Функция f вычислима. Действительно, как мы уже отметили выше, область определения функции i разрешима, так что с помощью алгоритма можно узнать, определено ли значение $i(n)$. Далее, если это значение определено, то обязательно определено и значение $\varphi_{i(n)}(n)$. Вычислив его, мы можем узнать, равно ли оно нулю. Таким образом, для любого n мы можем найти значение $f(n)$.

Покажем, что f — искомая функция. Пусть $f = \varphi_e$, т. е. программа с номером e вычисляет функцию f. Для любого n, если значение $i(n)$ определено, то, как видно из задания функции f, $e \neq i(n)$. С другой стороны, если бы неравенство $t_e(m) \leq b(m)$ выполнялось для бесконечно многих m, то, как мы показали ранее, должно быть $e = i(n)$ для некоторого n. Значит, для почти всех m выполняется неравенство

$$t_e(m) > b(m),$$

что и требовалось. Теорема 28.1 доказана.

Можно ли из формулировки теоремы исключить слова "почти всюду"и заменить их на "для всех n"? Очевидно, нет. Дело в том, что для каждой функции f можно написать программу, которая очень быстро вычисляет значение f в одной конкретной точке a. Пусть, например, $f(a) = 1$, и пусть программа P вычисляет функцию f. Составим новую программу P', вычисляющую функцию f:

$$\begin{array}{rl} 1) & S(2) \\ & \ldots \\ a) & S(2) \\ a+1) & Z(1) \\ a+2) & S(1) \\ a+3) & J(1,2,stop) \\ a+4) & Z(2) \\ & P \end{array}$$

(здесь *stop* — некоторое число, большее числа команд в программе P'). Очевидно, что если e — номер программы P', то $t_e(a) = a + 3$. Поэтому, если, например, $b(n) = n + 4$, то заключение теоремы 28.1 будет неверным для таких функций f и b при замене слов "почти всюду"на "для всех n".

Время вычисления является лишь одной из возможных мер сложности вычислений. При абстрактном подходе к сложности вычислений *мерой вычислительной сложности* (одноместных функций) называется семейство одноместных функций Φ_e ($e \in \mathbf{N}$) со следующими свойствами:

- $Dom(\Phi_e) = Dom(\varphi_e)$;
- множество $\{\langle e, x, y\rangle | \Phi_e(x) \simeq y\}$ разрешимо.

Очевидно, что рассмотренное выше семейство функций t_e является мерой вычислительной сложности в этом смысле. Вот другие примеры мер вычислительной сложности:

1. $\Phi_e(x)$ — количество выполнений команд условного перехода при вычислении $\varphi_e(x)$, если это вычисление завершается;
2. $\Phi_e(x)$ — максимальное число, содержащееся в каком-либо регистре МНР за все время вычисления $\varphi_e(x)$, если это вычисление завершается.

При вычислениях на машинах Тьюринга обычно рассматриваются временна́я сложность, определяемая как число шагов в вычислении, и пространственная сложность, определяемая как длина ленты, использованной в вычислении.

Заметим, что при доказательстве теоремы 28.1 мы использовали только те свойства функции t_e, которые сформулированы в определении меры вычислительной сложности. Следовательно, эта теорема остается верной при замене $t_e(n)$ на $\Phi_e(n)$ для любой меры вычислительной сложности Φ_e.

29 Теорема о неподвижной точке

В дальнейшем нам понадобится следующая теорема, представляющая и самостоятельный интерес.

Теорема 29.1 (теорема о неподвижной точке) *Каковы бы ни были натуральное число $m \geq 1$ и тотальная одноместная вычислимая функция f, существует число n такое, что $\varphi^{(m)}_{f(n)} = \varphi^{(m)}_n$.*

Доказательство. Рассмотрим следующую $(m+1)$-местную функцию:

$$\psi(u, x_1, \ldots, x_m) \simeq \varphi^{(m)}_{\varphi_u(u)}(x_1, \ldots, x_m).$$

Пусть e -гёделев номер функции ψ, т. е. $\psi = \varphi^{(m)}_e$. В силу s-m-n-теоремы (теорема 7.2) существует тотальная вычислимая двуместная функция s (функция s^1_m в обозначениях упомянутой теоремы) такая, что для любых $, u, x_1, \ldots, x_m$ имеет место условное равенство $\varphi^{(m+1)}_e(u, x_1, \ldots, x_m) \simeq \varphi^m_{s(e,u)}(x_1, \ldots, x_m)$. В нашем случае, при фиксированном e, положим $g(u) = s(e, u)$. Тогда

$$\varphi^m_{g(u)}(x_1, \ldots, x_m) \simeq \varphi^{(m)}_{\varphi_u(u)}(x_1, \ldots, x_m).$$

Пусть v — гёделев номер композиции функций f и g, т. е. $\varphi_v(x) = f(g(x))$ для любого x. Тогда для любых $x_1, \ldots, x_m$ имеем:

$$\varphi^m_{g(v)}(x_1, \ldots, x_m) \simeq \varphi^{(m)}_{\varphi_v(v)}(x_1, \ldots, x_m) \simeq \varphi^{(m)}_{f(g(v))}(x_1, \ldots, x_m),$$

и теорема доказана, если взять $n = g(v)$. $\square$

При $m = 1$ получаем, что для любой тотальной одноместной вычислимой функции f существует "неподвижная точка", т. е. такое число n, что $\varphi_{f(n)} = \varphi_n$. Эта теорема имеет многочисленные применения в теории алгоритмов.

Примеры. 1. Вот как, например, с помощью теоремы о неподвижной точке доказывается теорема Райса. Допустим, что индексное множество $I_{\mathcal{F}}$ нетривиального семейства одноместных вычислимых функций $\mathcal{F}$ разрешимо. Пусть $\varphi_a \in \mathcal{F}$, $\varphi_b \notin \mathcal{F}$. Если множество $I_{\mathcal{F}}$ разрешимо, то вычислима функция

$$f(x) = \begin{cases} b, & \text{если } x \in I_{\mathcal{F}}; \\ a, & \text{если } x \notin I_{\mathcal{F}}. \end{cases}$$

По теореме о неподвижной точке существует число n такое, что $\varphi_{f(n)} = \varphi_n$. Пусть $n \in I_{\mathcal{F}}$, т. е. $\varphi_n \in \mathcal{F}$. Тогда $f(n) = b$, что невозможно, так как в этом случае $\varphi_n = \varphi_{f(n)} = \varphi_b \notin \mathcal{F}$. Аналогично, если $n \notin I_{\mathcal{F}}$, т. е. $\varphi_n \notin \mathcal{F}$, то $f(n) = a$, что также невозможно, ибо в этом случае $\varphi_n = \varphi_{f(n)} = \varphi_a \in \mathcal{F}$. Значит, неверно ни одно из утверждений $n \in I_{\mathcal{F}}$ и $n \notin I_{\mathcal{F}}$. Полученное противоречие доказывает, что на самом деле множество $I_{\mathcal{F}}$ неразрешимо.

2) Если f — тотальная одноместная вычислимая функция, то существует такое число n, что $W_{f(n)} = W_n$. Действительно, по теореме о неподвижной точке существует число n такое, что $\varphi_{f(n)} = \varphi_n$. Тогда

$$W_{f(n)} = Dom(\varphi_{f(n)}) = Dom(\varphi_n) = W_n.$$

3) Существует такое число n, что $(\forall x)\varphi_n(x) = x^n$. Для доказательства этого факта рассмотрим двуместную вычислимую функцию $f(m, x) = x^m$. По теореме о параметризации существует тотальная одноместная вычислимая функция k такая, что $f(m, x) = \varphi_{k(m)}(x)$ для любых m, x. По теореме о неподвижной точке существует такое число n, что $\varphi_{k(n)} = \varphi_n$. Тогда для любого x имеем: $\varphi_n(x) = \varphi_{k(n)}(x) = f(n, x) = x^n$, что и требовалось.

Задачи.

1. Доказать, что существует такое число n, что

   а) $\varphi_n(0) = n$; б) $\varphi_n(n) = n$; в) $\varphi_n = \chi_{\{n\}}$; г) $\varphi_n = \psi_{\{n\}}$.

2. Доказать, что существует такое число n, что

   а) $W_n = \{n\}$; б) $W_n = \{n^2\}$; в) $W_n = \mathbf{N} \setminus \{n\}$.

30 Теорема об ускорении

Пусть P_a и P_b — две программы для вычисления тотальной функции f, причем $2t_b(x) < t_a(x)$ для любого x. В этом случае можно сказать, что программа P_b работает в два с лишним раза быстрее программы P_a. Оказывается, что существует тотальная функция f, обладающая следующим свойством: какова бы ни была программа P для вычисления функции f, существует другая программа, которая работает в два с лишним раза быстрее на почти всех исходных данных. Таким образом, для этой функции не существует "наилучшей"программы. Этот факт — частный случай следующей теоремы, доказанной Блюмом:

Теорема 30.1 (теорема об ускорении) *Пусть r — тотальная одноместная вычислимая функция. Существует тотальная одноместная вычислимая функция f такая, что по любой программе P_i, вычисляющей функцию f, можно построить другую программу P_j для вычисления f такую, что $r(t_j(x)) < t_i(x)$ для почти всех x.*

Доказательство. Теорему достаточно доказать в случае, когда r — возрастающая функция. Действительно, если r — произвольная тотальная вычислимая функция, рассмотрим функцию r', заданную так: $r'(0) = r(0)$ и

$$r'(n+1) = \begin{cases} r(n+1), \text{ если } r(n+1) \geq r'(n); \\ r'(n) \text{ в противном случае.} \end{cases}$$

Очевидно, что функция r' является возрастающей, вычислима и *мажорирует* функцию r в том смысле, что $r(n) \leq r'(n)$ для любого n. Теперь, если $r'(t_j(x)) < t_i(x)$ для почти всех x, то и $r(t_j(x)) < t_i(x)$ для почти всех x.

Пусть фиксировано натуральное число e. Определим двуместную вычислимую функцию $g(u,x)$ рекурсией по x при фиксированном u. При этом одновременно с вычислением $g(u,x)$ будут строиться вспомогательные конечные множества $C_{u,x}$. Итак, пусть значения $g(u,0),\ldots,g(u,x-1)$ и множества $C_{u,0},\ldots,C_{u,x-1}$ уже определены. В силу s-m-n-теремы существует тотальная вычислимая двуместная функция s (функция s^1_1 в обозначениях упомянутой теоремы) такая, что для любых $,x,y$ имеет место условное равенство

$$\varphi^{(2)}(x,y) \simeq \varphi_{s(e,x)}(y).$$

Зафиксируем такую функцию s. Положим

$$C_{u,x} = \{i | u \leq i < x \,\&\, i \notin \cup_{y<x} C_{u,y} \,\&\, t_i(x) \leq r(t_{s(e,i+1)}(x) + x)\},$$

если значение $t_{s(e,i+1)}(x)$ определено для всех $i = u,\ldots,x-1$; в противном случае множество $C_{u,x}$ не определено. Заметим, что если $x \leq u$, то $C_{u,x}$ определено и равно $\emptyset$. Кроме того, если $t_{s(e,i+1)}(x)$ определено, то можно решить, выполняется ли неравенство $t_i(x) \leq r(t_{s(e,i+1)}(x) + x)$. Теперь определим $g(u,x)$ следующим образом:

$$g(u,x) = \begin{cases} 1 + \max\{\varphi_i(x) | i \in C_{u,x}\}, \text{ если } C_{u,x} \text{ определено;} \\ \text{не определено в противном случае.} \end{cases}$$

Заметим, что если множество $C_{u,x}$ определено, то для каждого $i \in C_{u,x}$ значение $\varphi_i(x)$ определено, так что в этом случае значение $g(u,x)$ определено.

Итак, g — двуместная вычислимая функция. Вообще-то ее определение зависит от фиксированного в самом начале наших рассуждений числа e. Очевидно, что номер программы, вычисляющей функцию g, может быть вычислен по числу e, т. е. существует такая одноместная тотальная вычислимая функция k, что $g(u,x) \simeq \varphi^{(2)}_{k(e)}(u,x)$. В силу теоремы о неподвижной точке (теорема 29.1) существует такое число e, что

$$g(u,x) \simeq \varphi^{(2)}_e(u,x), \tag{5}$$

т. е. так построенная функция при этом фиксированном e сама вычисляется программой с номером e.

Покажем, что в этом случае функция g тотальна. Фиксируем x. Как мы уже отмечали, для $u \geq x$ множество $C_{u,x}$ определено и пусто, так что $g(u,x) = 1$. Предположим теперь, что все значения

$$g(x,x), g(x-1,x), \ldots, g(u+1,x)$$

определены. В силу (5) и определения функции s получаем, что значения

$$\varphi_{s(e,x)}(x), \varphi_{s(e,x-1)}(x), \ldots, \varphi_{s(e,u+1)}(x)$$

определены. Следовательно, для $i = u,\ldots,x-1$ определены значения $t_{s(e,i+1)}(x)$. Это означает, что множество $C_{u,x}$ определено; следовательно, определено и значение $g(u,x)$. Таким образом, $g(u,x)$ — тотальная функция.

Положим $g_u(x) = g(u,x)$. Тогда $g_u(x) = g(u,x) = \varphi_e^{(2)}(u,x) = \varphi_{s(e,x)}(x)$. Положим $f = g_0$. Покажем, что f является искомой функцией. Зафиксируем число u и убедимся, что функции g_0 и g_u отличаются только в конечном числе точек, т. е. $g(0,x) = g(u,x)$ для почти всех x. Из построения множества $C_{u,x}$ видно, что для всякого x

$$C_{u,x} = C_{0,x} \cap \{u, u+1, \ldots, x-1\}.$$

Так как по построению все множества $C_{0,x}$ при различных x попарно не пересекаются, то можно найти число $v = \max\{x | (\exists i < u) i \in C_{0,x}\}$. Тогда для $x > v$ имеем $C_{0,x} \subseteq \{u, u+1, \ldots, x-1\}$; следовательно, $C_{0,x} = C_{u,x}$, так что $g(0,x) = g(u,x)$ для всех $x > v$.

Пусть теперь $f = \varphi_i$. Тогда, если $j = s(e, i+1)$, то $\varphi_j = \varphi_{s(e,i+1)} = g_{i+1}$. Докажем, что $r(t_j(x) + x) < t_i(x)$ для почти всех x. Если бы это было не так, т. е. выполнялось бы неравенство $t_i(x) \leq r(t_{s(e,i+1)}(x) + x)$ для бесконечно многих x, то нашлось бы x, для которого

$$0 \leq u < x \,\&\, i \notin \cup_{y<x} C_{u,y} \,\&\, t_i(x) \leq r(t_{s(e,i+1)}(x) + x)\},$$

т. е. $i \in C_{0,x}$. Но тогда из определения функции g следует $g(0,x) \neq \varphi_i(x)$, что невозможно. Полученное противоречие доказывает, что неравенство $r(t_j(x) + x) < t_i(x)$ выполняется для почти всех x.

Итак, по данному i, для которого $f = \varphi_i$, мы построили такое число j, что функция φ_j совпадает с f почти всюду и $r(t_j(x) + x) < t_i(x)$ для почти всех x. Осталось только программу с номером j переделать в программу для вычисления функции f. Пусть $(\forall x > v)\varphi_j(x) = f(x)$ и $f(m) = b_m$ для $m \leq v$. Тогда программу P_j несколько модифицируем следующим образом:

1)	$J(1,2,l_0)$
2)	$S(2)$
3)	$J(1,2,l_1)$
	$\ldots$
$2v$)	$S(2)$
$2v+1$)	$J(1,2,l_v)$
$2v+2$)	P_j
	$\ldots$
	$J(1,1,stop)$
l_0)	Q_{b_0}
	$J(1,1,stop)$
	$\ldots$
l_v)	Q_{b_v}
	$J(1,1,stop)$

(Здесь через Q_b обозначена программа, помещающая число b в первый регистр.) Пусть j^* — номер только что построенной программы. Эта программа вычисляет функцию f. Дополнительные команды увеличивают время вычисления по сравнению с программой P_j не более чем на c шагов, где c — константа, не зависящая от исходного данного x. Таким образом, $(\forall x) t_{j^*}(x) \leq t_j(x) + c$. Поскольку r — возрастающая функция, $r(t_{j^*}(x)) \leq r(t_j(x) + c) \leq r(t_j(x) + x)$ для $x \geq c$. Учитывая также, что $r(t_j(x) + x) < t_i(x)$ для почти всех x, получаем, что $r(t_{j^*}(x)) < t_i(x)$ для почти всех x, что и требовалось доказать. Теорема 30.1 доказана.

Список литературы

[1] Катленд Н. Вычислимость. Введение в теорию рекурсивных функций. М., Мир, 1983.

[2] Лавров И. А., Максимова Л. Л. Задачи по теории множеств, математической логике и теории алгоритмов. М., Физматлит, 1995.

[3] Роджерс Х. Теория рекурсивных функций и эффективная вычислимость. М., Мир, 1972.

[4] Мендельсон Э. Введение в математическую логику. М., Наука, 1984.

[5] Успенский В. А. Лекции о вычислимых функциях. М., Физматлит, 1960.

[6] Успенский В. А., Верещагин Н. К., Плиско В. Е. Вводный курс математической логики. М., ФИЗМАТЛИТ, 2002.