Лекция 13. Вычислимые функции, перечислимые и разрешимые множества – 2

Дискретная математика, ВШЭ, факультет компьютерных наук

(Осень 2014 – весна 2015)

1 Перечислимые множества в терминах вычислимых функций

Какие множества естественно связаны с некоторым классом функций из $\mathbb{N}$ в $\mathbb{N}$? Есть по меньшей мере 4 (вообще говоря, различных) класса множеств:

1) множества значений функций из данного класса;
2) множества значений всюду определённых функций из данного класса;
3) области определения функций из данного класса;
4) графики функций, то есть подмножества $\mathbb{N} \times \mathbb{N}$ вида $\Gamma_f = \{(x, y) : y = f(x)\}$, где f принадлежит данному классу функций.

В случае вычислимых функций эти классы множеств выражаются через перечислимые множества.

Начнём с двух простых наблюдений.

Утверждение 1. *Если множество S перечислимо, то оно является множеством значений некоторой вычислимой функции.*

Доказательство. Пусть S — перечислимо. Превратим алгоритм перечисления его элементов в алгоритм вычисления некоторой функции: на входе n алгоритм запускает алгоритм перечисления элементов S и подсчитывает количество «напечатанных» элементов (печатать элементы ему для этого необязательно). Когда напечатан n-й элемент, алгоритм выдаёт его в качестве результата работы и останавливается.

Множеством значений такой функции будут в точности числа, которые печатает алгоритм перечисления, то есть множество S. □

Утверждение 2. *Если множество S является множеством значений всюду определённой вычислимой функции, то оно перечислимо.*

Доказательство. Пусть $S = f(\mathbb{N})$ для некоторой всюду определённой функции f. Алгоритм перечисления множества S устроен так: для каждого натурального числа, начиная с 0, n вычисляем $f(n)$ и печатаем результат. □

Задача 1. Какое множество перечисляет алгоритм из доказательства утверждения 2, если f не всюду определена?

Мы уже использовали в доказательстве теоремы Поста разбиение работы алгоритма на шаги. В доказательствах нам будет удобно использовать формальное определение количества шагов алгоритма.

Свойство алгоритмов 3. *Для некоторой универсальной вычислимой функции $U\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ существует всюду определённая вычислимая функция $F\colon \mathbb{N} \times \mathbb{N} \times \mathbb{N} \to \{0, 1\}$ (отладочная функция), обладающая следующими свойствами:*

- *при любых p, x функция $F(p, x, t)$ — неубывающая функция от аргумента t;*
- *$U(p, x)$ не определена тогда и только тогда, когда $F(p, x, t) = 0$ для всех t.*

Неформально говоря, значение функции $F(p, x, t)$ равно 0 тогда и только тогда, когда программа p на входе x не закончила работу за количество шагов t. В противном случае значение функции $F(p, x, t)$ равно 1.

Теорема 1. *Для непустого множества $S \subseteq \mathbb{N}$ следующие свойства равносильны:*

1) S — перечислимое;

2) S — множество значений некоторой вычислимой функции;

3) S — множество значений некоторой всюду определённой вычислимой функции;

4) S — область определения некоторой вычислимой функции.

Доказательство. 1) $\Rightarrow$ 2) — это утверждение 1.

2) $\Rightarrow$ 3). Пусть $S = f(\mathbb{N})$ для некоторой вычислимой функции f. Зафиксируем некоторое $a \in S$ (мы предполагаем, что множество S непусто).

Функция f не всюду определена и может не иметь вычислимого всюду определённого продолжения. Поэтому используем отладочную функцию.

Пусть $f(x) = U(p, x)$, где U — некоторая универсальная вычислимая функция, для которой существует отладочная функция.

Опишем алгоритм вычисления всюду определённой функции $g\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$. На входе (x, t) алгоритм вычисляет $F(p, x, t)$ и сравнивает результат с 1. Если $F(p, x, t) = 1$, то алгоритм выдаёт $U(p, x)$. В противном случае результат равен a.

По определению отладочной функции из $F(p, x, t) = 1$ следует, что $U(p, x)$ определена. Поэтому функция g всюду определена. Её множество значений совпадает с S. В одну сторону: если $y = g(x, t)$, то $y = a \in S$ или $y = U(p, x) = f(x) \in S$. В другую: пусть $y = f(x) = U(p, x)$. На паре (p, x) функция U определена, поэтому для некоторого t значение отладочной функции $F(p, x, t) = 1$. Но тогда $y = g(x, t)$.

Мы представили S как множество значений всюду определённой функции от двух натуральных аргументов. Чтобы перейти к функциям одного аргумента, используем вычислимую биекцию $c\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ и выразим S как $S = g \circ c^{-1}(\mathbb{N})$.

3) $\Rightarrow$ 1) — это утверждение 2. Таким образом, первые три свойства равносильны.

3) $\Rightarrow$ 4). Пусть $S = f(\mathbb{N})$ для некоторой всюду определённой функции f. Опишем алгоритм вычисления функции g, который получает на вход x: перебираем все числа, начиная с 0; для каждого числа n вычисляем $f(n)$ и сравниваем с x; в случае равенства выдаём результат 1.

Если $x \in S$, то $g(x) = 1$, так как $x = f(n)$ для некоторого n.

Если $x \notin S$, то $g(x)$ не определена, так как для любого n выполняется $x \neq f(n)$, то есть алгоритм вычисления g не даёт никакого результата.

4) $\Rightarrow$ 2). Пусть S — область определения некоторой вычислимой функции f, а p — номер программы, вычисляющей f. Опишем алгоритм вычисления функции g из $\mathbb{N} \times \mathbb{N}$ в $\mathbb{N}$ на входе (x, t): вычисляем $F(p, x, t)$ и сравниваем с 1; если $F(p, x, t) = 1$, то выдаём результат x, иначе не выдаём никакого результата.

Если $x \in S$, то $x = g(x, t)$ для некоторого t. И обратно, если $x = g(x, t)$ для некоторого t, то $U(p, x)$ определена, а значит, определена и $f(x)$.

Мы представили S как множество значений функции от двух натуральных аргументов. Чтобы перейти к функциям одного аргумента, используем вычислимую биекцию $c\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ и выразим S как $S = g \circ c^{-1}(\mathbb{N})$. $\square$

2 Проекции и графики

Проекцией множества $D \subseteq \mathbb{N} \times \mathbb{N}$ назовём множество $\text{Пр}\, D = \{x : \exists y (x, y) \in D\}$. Название становится понятным, если нарисовать условную картинку на декартовой плоскости.

Теорема 2. *Перечислимые множества — это в точности проекции разрешимых.*

Доказательство. Проекция пустого множества пуста. Дальше рассматриваем только непустые множества.

Пусть $D \subseteq \mathbb{N} \times \mathbb{N}$ разрешимо, $(a, b) \in D$. По определению, индикаторная функция χ_D вычислима. Но тогда вычислима и функция

$$f(x, y) = \begin{cases} x, & \text{если } \chi_D(x, y) = 1, \\ a, & \text{в противном случае,} \end{cases}$$

для которой $f(\mathbb{N}\times\mathbb{N}) = \text{Пр}\, D$. Поэтому $\text{Пр}\, D$ перечислимо (как всегда, нужно ещё позаботиться о замене функций от двух аргументов функцией от одного аргумента).

Пусть S — перечислимо. Тогда S — область определения некоторой вычислимой функции f, которая имеет номер p в универсальной нумерации. Построим такое множество:

$$D = \{(x, t) : F(p, x, t) = 1\}.$$

Докажем, что $\text{Пр}\, D = S$. Пусть $x \in S$. Это значит, что на x функция f определена, тем самым определена и функция $U(p, x)$. Но тогда по определению отладочной функции $F(p, x, t) = 1$ для некоторого t, то есть $x \in \text{Пр}\, D$.

В обратную сторону аналогично: если $x \in \text{Пр}\, D$, то для некоторого t выполняется $F(p, x, t)$, то есть $U(p, x) = f(x)$ определена. Таким образом, $x \in S$. $\square$

Теорема 3. *Функция вычислима тогда и только тогда, когда её график перечислим.*

Доказательство. Ясно, что график является образом $f(\mathbb{N})$ функции из $\mathbb{N}$ в $\mathbb{N} \times \mathbb{N}$, которая определена как $x \mapsto (x, f(x))$. Поэтому график вычислимой функции перечислим.

Пусть график $\Gamma = \{(x, y) : y = f(x)\}$ перечислим. Тогда следующий алгоритм вычисляет f: на входе x запускаем перечисление элементов графика; когда найден очередной элемент (y, z), проверяем $x = y$; в случае равенства выдаём результат z.

Из определения графика ясно, что на входе x такой алгоритм может выдать лишь результат $f(x)$. С другой стороны, если x принадлежит области определения f, то пара $(x, f(x))$ рано или поздно будет перечислена. В этот момент алгоритм и выдаст результат $f(x)$. $\square$

3 Перечислимые неразрешимые множества

Из доказанных выше теорем очень легко построить пример перечислимого неразрешимого множества.

Пусть U — некоторая универсальная функция. Обозначим через H множество

$$\{x : U(x, x) \text{ определена}\}.$$

Теорема 4. *H перечислимо, но неразрешимо.*

Доказательство. Множество H является областью определения вычислимой функции $x \mapsto U(x, x)$. Поэтому оно перечислимо.

Неразрешимость доказывается диагональным аргументом. Предположим, что H разрешимо. Рассмотрим такой алгоритм вычисления функции f: на входе x он вызывает алгоритм разрешения множества H для x. Если $x \in H$, то алгоритм не даёт никакого результата, а если $x \notin H$, то выдаёт результат 1.

Пусть p — номер функции f в универсальной нумерации, то есть $f(x) = U(p, x)$.

Предположим, что $p \in H$. Тогда алгоритм вычисления f, описанный выше, не выдаёт никакого результата, то есть $f(p) = U(p,p)$ не определено. По определению множества H это означает, что $p \notin H$.

Если $p \notin H$, то алгоритм вычисления f даёт результат 1, то есть $1 = f(p) = U(p,p)$. Следовательно, $p \in H$.

Пришли к противоречию. Значит, множество H неразрешимо. $\square$

Из теоремы Поста получаем простое следствие.

Следствие 1. *Множество $\bar{H}$ неперечислимо.*

Заметим, что любое множество

$$H_a = \{x : U(x,x) = a\}$$

также перечислимо как область определения следующей вычислимой функции:

$$f_a = \begin{cases} 1, & \text{если } U(x,x) = a, \\ & \text{не определена в противном случае.} \end{cases}$$

Каждое такое множество также неразрешимо. На самом деле, выполняется даже более сильное свойство: при $a \neq b$ множества H_a и H_b *неотделимы*. Это значит, что не найдётся такого разрешимого множества D, для которого

$$H_a \subseteq D, \quad H_b \cap D = \varnothing.$$

Следствие 2. *При $a \neq b$ множества H_a и H_b неотделимы.*

Доказательство. Предположим, D — разрешимое множество, которое отделяет H_a и H_b. Рассмотрим такой алгоритм вычисления функции f: на входе x он вызывает алгоритм разрешения множества D для x. Если $x \in D$, то алгоритм даёт результат b, а если $x \notin D$, то выдаёт результат a.

Пусть p — номер функции f в универсальной нумерации, то есть $f(x) = U(p,x)$.

Предположим, что $p \in D$. Тогда $b = f(p) = U(p,p)$ и $p \in H_b$, то есть $p \notin D$.

Если $p \notin H$, то алгоритм вычисления f даёт результат $a = f(p) = U(p,p)$, то есть $p \in H_a \subseteq D$. Следовательно, $p \in D$.

Пришли к противоречию. Значит, пара H_a, H_b неотделима. $\square$

Ещё одним примером перечислимых неразрешимых множеств являются *универсальные перечислимые множества*, Множество $W \subseteq \mathbb{N} \times \mathbb{N}$ называется универсальным, если для любого перечислимого множества S найдётся номер p такой, что

$$S = \{x : (p,x) \in W\}.$$

(Определение универсального подмножества натуральных чисел получается применением вычислимой биекции.)

По универсальной функции U легко построить универсальное перечислимое множество. Это просто область определения U.

Утверждение 3. *Область определения универсальной функции является универсальным перечислимым множеством.*

Доказательство. Пусть S перечислимо. Тогда S — область определения некоторой вычислимой функции f. Обозначим через p номер этой функции в нумерации U. По определению получаем

$$S = \{x : U(p,x) \text{ определена}\},$$

что и требовалось. $\square$

Теорема 5. *Универсальное перечислимое множество неразрешимо.*

Доказательство. Если бы универсальное перечислимое множество было разрешимым, то из алгоритма его разрешения получался бы алгоритм разрешения любого перечислимого множества (первым элементом пары такой алгоритм получает номер перечислимого множества в данной нумерации). □