Лекция C1
Машины Шёнфилда

Вадим Пузаренко

7 декабря 2022 г.

Определение машины Шёнфилда

Команды.

$\mathrm{INC}\,\mathrm{I}$ Как содержимое I-го регистра, так и счётчик команд увеличивает на единицу; содержимое остальных регистров остаётся неизменным.

$\mathrm{DEC}\,\mathrm{I}, n$ Если содержимое I-го регистра больше нуля, то уменьшает содержимое I-го регистра на единицу и помещает в счётчик команд число n; если же содержимое I-го регистра равняется нулю, то содержимое I-го регистра не меняется, а счётчик команд увеличивается на единицу.

Определение машины Шёнфилда

Команды.

$\mathrm{INC\,I}$ Как содержимое I-го регистра, так и счётчик команд увеличивает на единицу; содержимое остальных регистров остаётся неизменным.

$\mathrm{DEC\,I}, n$ Если содержимое I-го регистра больше нуля, то уменьшает содержимое I-го регистра на единицу и помещает в счётчик команд число n; если же содержимое I-го регистра равняется нулю, то содержимое I-го регистра не меняется, а счётчик команд увеличивается на единицу.

В обоих случаях содержимое регистра $\mathrm{J} \neq \mathrm{I}$ остаётся неизменным.

Определение машины Шёнфилда

Команды.

$\mathrm{INC\,I}$ Как содержимое I-го регистра, так и счётчик команд увеличивает на единицу; содержимое остальных регистров остаётся неизменным.

$\mathrm{DEC\,I}, n$ Если содержимое I-го регистра больше нуля, то уменьшает содержимое I-го регистра на единицу и помещает в счётчик команд число n; если же содержимое I-го регистра равняется нулю, то содержимое I-го регистра не меняется, а счётчик команд увеличивается на единицу.

В обоих случаях содержимое регистра $\mathrm{J} \neq \mathrm{I}$ остаётся неизменным.

Счётчик команд принимает в качестве значений натуральные числа, а имена регистров закодированы натуральными числами. Регистр, закодированный числом $\imath$, будем обозначать как $[\imath]$.

Определение машины Шёнфилда

Программа.

Программа имеет вид

$$\begin{array}{rl} 0: & P_0 \\ 1: & P_1 \\ \ldots & \ldots \\ n: & P_n \end{array}$$

Здесь число k в записи $k:$ означает значение счётчика команд, а P_k — одна из команд, описанных выше ($0 \leqslant k \leqslant n$).

Определение машины Шёнфилда

Программа.

Программа имеет вид

$$\begin{array}{rl} 0: & P_0 \\ 1: & P_1 \\ \ldots & \ldots \\ n: & P_n \end{array}$$

Здесь число k в записи k : означает значение счётчика команд, а P_k — одна из команд, описанных выше ($0 \leqslant k \leqslant n$).

Машина Шёнфилда: описание.

Однозначно задаётся следующими атрибутами:
1) потенциально бесконечным множеством **регистров**, занумерованными натуральными числами. Каждый регистр — это ячейка памяти, способная содержать любое натуральное число.

Определение машины Шёнфилда

Машина Шёнфилда: описание.

Содержимое регистров может меняться в процессе вычислений. Отметим, что каждая фиксированная машина Шёнфилда использует в своих вычислениях толь ко конечное число регистров. Основное назначение регистровой памяти — это хранение входных, промежуточных и выходных данных.

2) **счётчиком команд**, являющимся особой ячейкой памяти, которая в каждый момент времени содержит некоторое натуральное число. Счётчик команд указывает на номер команды, которая исполняется в данный момент. В начальный момент времени счётчик команд равняется нулю.

3) **программой**, содержащейся в выделенной ячейке памяти машины. Программа не меняется в процессе вычисления.

Шаг машины состоит в выполнении команды, на которую указывает счётчик команд. Если команды с таким номером нет, то программа останавливается.

Макропрограммы

Макрос.

Каждой обычной программе P может быть сопоставлен оператор P^*, который будем называть **макросом**. Данный оператор может быть использован как отдельная команда.

Макропрограммы

Макрос.

Каждой обычной программе P может быть сопоставлен оператор P^*, который будем называть **макросом**. Данный оператор может быть использован как отдельная команда.

Макропрограмма.

Программы, содержащие макросы, будем называть **макропрограммами**.

Макропрограммы

Макрос.

Каждой обычной программе P может быть сопоставлен оператор P^*, который будем называть **макросом**. Данный оператор может быть использован как отдельная команда.

Макропрограмма.

Программы, содержащие макросы, будем называть **макропрограммами**.

Исполнение макроса.

Пусть макрос P^* находится в строке с номером m в макропрограмме; тогда выполняется следующее:
1) если в счётчике команд находится m, то вызывается программа P в качестве подпрограммы;
2) P исполняется с тем содержимым регистров, которым сформировался в данный момент в результате исполнения макропрограммы;

Макропрограммы

Исполнение макроса.

3) если работа программы P закончилась, то в счётчик команд помещается $m+1$; если программа P не останавливается, то и вся макропрограмма не останавливается.

Макропрограммы

Исполнение макроса.

3) если работа программы P закончилась, то в счётчик команд помещается $m+1$; если программа P не останавливается, то и вся макропрограмма не останавливается.

Замечание C1.1.

Любой макрос может использовать конечный набор параметров, каждый из которых является некоторым регистром. Макросы не могут использовать в качестве своих параметров номера строк!!!

Макропрограммы

Исполнение макроса.

3) если работа программы P закончилась, то в счётчик команд помещается $m+1$; если программа P не останавливается, то и вся макропрограмма не останавливается.

Замечание C1.1.

Любой макрос может использовать конечный набор параметров, каждый из которых является некоторым регистром. Макросы не могут использовать в качестве своих параметров номера строк!!!

Пример C1.1.

Программа $0 : \mathrm{INC}\,0$ и $1 : \mathrm{DEC}\,0, n$ имитирует $\mathrm{GOTO}\,n$, однако макросом она не является (хотя и демонстрирует наличие определённого свойства).

Примеры макросов

$\mathrm{ZERO\,I}$

$0 : \mathrm{DEC\,I}, 0$
(обнуляет содержимое I-го регистра).

Примеры макросов

ZERO I

$0 : \mathrm{DEC}\, \mathrm{I}, 0$
(обнуляет содержимое I-го регистра).

$[i] \to [j], (k)$

Пусть натуральные числа i, j, k таковы, что $i \neq k$ и $j \neq k$. Данный макрос будет копировать содержимое $[i]$-го регистра в $[j]$-ый регистр, используя в качестве вспомогательного $[k]$-ый регистр (не перемещается, а именно копируется). Пусть сначала $i \neq j$; тогда

$0 : \mathrm{ZERO}\, \mathrm{J}$	$3 : \mathrm{DEC}\, 0, 6$	$6 : \mathrm{DEC}\, \mathrm{I}, 4$	$9 : \mathrm{INC}\, \mathrm{I}$
$1 : \mathrm{ZERO}\, \mathrm{K}$	$4 : \mathrm{INC}\, \mathrm{J}$	$7 : \mathrm{INC}\, 0$	$10 : \mathrm{DEC}\, \mathrm{K}, 9$
$2 : \mathrm{INC}\, 0$	$5 : \mathrm{INC}\, \mathrm{K}$	$8 : \mathrm{DEC}\, 0, 10$	

При $i = j$ можно взять программу, которая работает впустую:
$0 : \mathrm{INC}\, 0$
$1 : \mathrm{DEC}\, 0, 2$

Элиминация макросов

Определение C1.1.

Макропрограммы P и Q называются **эквивалентными**, если при работе машин Шёнфилда с этими макропрограммами с одними и теми же входными данными (содержимыми регистров) они обе либо остановятся с одинаковыми выходными данными, либо обе не останавливаются.

Элиминация макросов

Определение C1.1.

Макропрограммы P и Q называются **эквивалентными**, если при работе машин Шёнфилда с этими макропрограммами с одними и теми же входными данными (содержимыми регистров) они обе либо остановятся с одинаковыми выходными данными, либо обе не останавливаются.

Теорема C1.1.

Любая макропрограмма эквивалентна некоторой программе, не содержащей макросов.

Элиминация макросов

Определение C1.1.

Макропрограммы P и Q называются **эквивалентными**, если при работе машин Шёнфилда с этими макропрограммами с одними и теми же входными данными (содержимыми регистров) они обе либо остановятся с одинаковыми выходными данными, либо обе не останавливаются.

Теорема C1.1.

Любая макропрограмма эквивалентна некоторой программе, не содержащей макросов.

Доказательство.

Достаточно построить эквивалентную макропрограмму, содержащую на один макрос меньше.
Пусть P — программа и P^* — используемый макрос (скажем, $m : P^*$). Проделаем следующую процедуру:

Элиминация макросов

Доказательство (окончание).

1) Заменим данное вхождение оператора $m : P^*$ на последовательность команд программы P, причём номер команды k следует поменять на номер $t_1(k) = m + k$ (предположим, что программа P имеет p_0 команд). Кроме того, все номера команд $k > m$ следует заменить на $t_2(k) = k + p_0 - 1$.

2) Операторы вида $\mathrm{INC\,I}$ и остальные макросы оставляем неизменными.

3) Пусть оператор имеет вид $l : \mathrm{DEC\,I}, n$; тогда проводим следующую замену:

- если $l < m$, то $l : \mathrm{DEC\,I}, n$ при $n \leqslant m$; $l : \mathrm{DEC\,I}, t_2(n)$ при $n > m$;
- если $m \leqslant l < m + p_0$, то $l : \mathrm{DEC\,I}, t_1(n)$ при $0 \leqslant n < p_0$; $l : \mathrm{DEC\,I}, m + p_0$ при $n > p_0$;
- если $l \geqslant m + p_0$, то $l : \mathrm{DEC\,I}, n$ при $n \leqslant m$; $l : \mathrm{DEC\,I}, t_2(n)$ при $n > m$.

$\square$

Вычислимость на МШ

Определение С1.2.

Частичная числовая функция $f(x_1, x_2, \ldots, x_k)$ называется **вычислимой на машине Шёнфилда** с программой P, если выполняются следующие условия (здесь $n_1, n_2, \ldots, n_k \in \omega$):

1. если $f(n_1, n_2, \ldots, n_k)\downarrow$, то машина P, начиная работу с содержимым $[i]$-го регистра n_i $(1 \leqslant i \leqslant k)$ и остальными регистрами, содержащими 0, останавливается и $f(n_1, n_2, \ldots, n_k)$ находится в $[0]$-м регистре;
2. если $f(n_1, n_2, \ldots, n_k)\uparrow$, то машина P, начиная работу с содержимым $[i]$-го регистра n_i $(1 \leqslant i \leqslant k)$ и остальными регистрами, содержащими 0, не останавливается и работает бесконечно.

$f([i_1], [i_2], \dots, [i_k]) \to [j], (s)$

Описание макроса.

Пусть f — k-местная частичная функция, вычислимая на машине Шёнфилда с программой P. Данный макрос вычисляет функцию $f(n_1, n_2, \dots, n_k)$ так, что n_t является содержимым регистра $[i_t]$ $(1 \leqslant t \leqslant k)$, а значение функции выдаёт в регистре с номером j и при этом не меняет содержимое всех регистров до s включительно (за исключением, возможно, регистра с номером j), если программа останавливается и функция определена; программа не останавливается, если функция не определена.

Частично вычислимые функции

Простейшие функции.

1. $0(x) = 0$ (тождественно нулевая функция);
2. $\mathrm{s}(x) = x + 1$ (функция следования);
3. $\mathrm{I}^n_m(x_1, x_2, \ldots, x_n) = x_m$, $1 \leqslant m \leqslant n$ (функции проекции).

Частично вычислимые функции

Простейшие функции.

1. $0(x) \equiv 0$ (тождественно нулевая функция);
2. $\mathrm{s}(x) = x + 1$ (функция следования);
3. $\mathrm{I}^n_m(x_1, x_2, \dots, x_n) = x_m$, $1 \leqslant m \leqslant n$ (функции проекции).

Оператор S суперпозиции.

Пусть $h^n(y_1, y_2, \dots, y_m)$ — частичная n-арная функция, а $g_1^m(x_1, x_2, \dots, x_m)$, $g_2^m(x_1, x_2, \dots, x_m)$, …, $g_n^m(x_1, x_2, \dots, x_m)$ — частичные m-арные функции. Определим частичную m-арную функцию $f^m = S(h, g_1, g_2, \dots, g_m)$ следующим образом:
$f(x_1 x_2, \dots, x_m) =$
$h(g_1(x_1, x_2, \dots, x_m), g_2(x_1, x_2, \dots, x_m), \dots, g_n(x_1, x_2, \dots, x_m))$.

Частично вычислимые функции

Оператор R примитивной рекурсии.

Пусть $g^n(x_1, x_2, \ldots, x_n)$ — частичная n-арная функция и пусть $h^{n+2}(x_1, x_2, \ldots, x_n, y, z)$ — частичная $n+2$-арная функция. Определим $n+1$-арную функцию $f^{n+1} = R(g, h)$ следующим образом:

$$\left[\begin{array}{l} f(x_1, x_2, \ldots, x_n, 0) = g(x_1, x_2, \ldots, x_n); \\ f(x_1, x_2, \ldots, x_n, y+1) = h(x_1, x_2, \ldots, x_n, y, f(x_1, x_2, \ldots, x_n, y)) \end{array} \right.$$

Частично вычислимые функции

Оператор R примитивной рекурсии.

Пусть $g^n(x_1, x_2, \ldots, x_n)$ — частичная n-арная функция и пусть $h^{n+2}(x_1, x_2, \ldots, x_n, y, z)$ — частичная $n+2$-арная функция. Определим $n+1$-арную функцию $f^{n+1} = R(g, h)$ следующим образом:

$$\left[\begin{array}{l} f(x_1, x_2, \ldots, x_n, 0) = g(x_1, x_2, \ldots, x_n); \\ f(x_1, x_2, \ldots, x_n, y+1) = h(x_1, x_2, \ldots, x_n, y, f(x_1, x_2, \ldots, x_n, y)) \end{array}\right.$$

Оператор R примитивной рекурсии для унаров.

Пусть $a \in \omega$ и пусть $h(y, z)$ — частичная бинарная функция. Тогда $f^1 = R(a, h)$ определяется следующим образом:

$$\left[\begin{array}{l} f(0) = a; \\ f(y+1) = h(y, f(y)) \end{array}\right.$$

Частично вычислимые функции

Оператор M минимизации.

Пусть $g^{n+1}(x_1, x_2, \ldots, x_n, y)$ — частичная $n+1$-арная функция. Определим n-арную функцию $f^n = M(g)$ следующим образом:

$$f(x_1, x_2, \ldots, x_n) = \begin{cases} y, & \text{если } g(x_1 x_2, \ldots, x_n, y) = 0\& \\ & \forall i < y[g(x_1, x_2, \ldots, x_n, i) \downarrow\neq 0]; \\ \uparrow & \text{в противном случае.} \end{cases}$$

Частично вычислимые функции

Оператор M минимизации.

Пусть $g^{n+1}(x_1, x_2, \ldots, x_n, y)$ — частичная $n+1$-арная функция. Определим n-арную функцию $f^n = M(g)$ следующим образом:

$$f(x_1, x_2, \ldots, x_n) = \begin{cases} y, & \text{если } g(x_1 x_2, \ldots, x_n, y) = 0\& \\ & \forall i < y[g(x_1, x_2, \ldots, x_n, i)\downarrow \neq 0]; \\ \uparrow & \text{в противном случае.} \end{cases}$$

Пример C1.2.

Пусть $g(0,0) = 1$, $g(0,1)\uparrow$, $g(0,2) = 0$; тогда $M(g)(0)\uparrow$, а не $M(g)(0) = 2$, поскольку $g(0,1)\uparrow$.

Частично вычислимые функции

Определение С1.3.

Частичная функция f называется **примитивно рекурсивной**, если существует последовательность $f_1, f_2, \ldots, f_n = f$ функций такая, что каждая функция этой последовательности либо простейшая, либо получена из предыдущих с помощью операторов S и R. Другими словами, класс примитивно рекурсивных функций является замыканием класса простейших функций относительно операторов S и R.

Частично вычислимые функции

Определение С1.3.

Частичная функция f называется **примитивно рекурсивной**, если существует последовательность $f_1, f_2, \ldots, f_n = f$ функций такая, что каждая функция этой последовательности либо простейшая, либо получена из предыдущих с помощью операторов S и R. Другими словами, класс примитивно рекурсивных функций является замыканием класса простейших функций относительно операторов S и R.

Определение С1.4.

Частичная n-арная функция $f(x_1, x_2, \ldots, x_n)$ называется **всюду определённой**, если её область задания $\delta f = \omega^n$.

Частично вычислимые функции

Определение C1.3.

Частичная функция f называется **примитивно рекурсивной**, если существует последовательность f_1, f_2, ..., $f_n = f$ функций такая, что каждая функция этой последовательности либо простейшая, либо получена из предыдущих с помощью операторов S и R. Другими словами, класс примитивно рекурсивных функций является замыканием класса простейших функций относительно операторов S и R.

Определение C1.4.

Частичная n-арная функция $f(x_1, x_2, \ldots, x_n)$ называется **всюду определённой**, если её область задания $\delta f = \omega^n$.

Предложение C1.1.

Любая примитивно рекурсивная функция всюду определена.

Частично вычислимые функции

Упражнение C1.1.

Докажите предложение C1.1.

Частично вычислимые функции

Упражнение C1.1.

Докажите предложение C1.1.

Определение C1.5.

Частичная функция f называется **частично вычислимой**, если существует последовательность $f_1, f_2, \ldots, f_n = f$ функций такая, что каждая функция этой последовательности либо простейшая, либо получена из предыдущих с помощью операторов S, R и M. Другими словами, класс примитивно рекурсивных функций является замыканием класса простейших функций относительно операторов S, R и M.

Частично вычислимые функции

Упражнение C1.1.

Докажите предложение C1.1.

Определение C1.5.

Частичная функция f называется **частично вычислимой**, если существует последовательность $f_1, f_2, \ldots, f_n = f$ функций такая, что каждая функция этой последовательности либо простейшая, либо получена из предыдущих с помощью операторов S, R и M. Другими словами, класс примитивно рекурсивных функций является замыканием класса простейших функций относительно операторов S, R и M.

Определение C1.6.

Частично вычислимая функция f называется **вычислимой**, если она всюду определена.

Частично вычислимые функции

Замечание C1.2.

Нетрудно видеть, что любая примитивно рекурсивная функция (ПРФ) является вычислимой (ВФ); в свою очередь, любая ВФ является частично вычислимой (ЧВФ). Однако существуют частично вычислимые функции, не являющиеся вычислимыми; можно построить вычислимую функцию, не являющуюся ПРФ (последний тезис отложим до лучших времён).

ЧВФ $\mapsto$ МШ

Теорема С1.2.

Любая частично вычислимая функция вычислима на некоторой машине Шёнфилда.

ЧВФ $\mapsto$ МШ

Теорема C1.2.

Любая частично вычислимая функция вычислима на некоторой машине Шёнфилда.

Доказательство.

Простейшие функции

$$\begin{array}{rl} \mathbf{0}(x) & 0 : \mathrm{ZERO}\,0 \\ s(x) & 0 : \mathrm{INC}\,1;\ 1 : [1] \to [0] \\ \mathrm{I}^n_m & 0 : [m] \to [0] \end{array}$$

Оператор S ($f^m = S(g^n, h^m_1, h^m_2, \ldots, h^m_n)$)

$0 : h_1([1], [2], \ldots, [m]) \to [m+1]$
$1 : h_2([1], [2], \ldots, [m]) \to [m+2]$
$\ldots$
$n-1 : h_n([1], [2], \ldots, [m]) \to [m+n]$
$n : g([m+1], [m+2], \ldots, [m+n]) \to [0]$

ЧВФ $\mapsto$ МШ

Доказательство (окончание).

Оператор R $(f^{n+1} = R(g^n, h^{n+2}))$

$0 : g([1],[2],\dots,[n]) \to [0]$

$1 : [n+1] \to [n+2]$

$2 : \mathrm{ZERO}\, n+1$

$3 : \mathrm{INC}\, 0$

$4 : \mathrm{DEC}\, 0, 7$

$5 : h([1],[2],\dots,[n],[n+1],[0]) \to [0]$

$6 : \mathrm{INC}\, n+1$

$7 : \mathrm{DEC}\, n+2, 5$

Оператор M $(f^n = M(g^{n+1}))$

$0 : \mathrm{INC}\, 0$

$1 : \mathrm{DEC}\, 0, 3$

$2 : \mathrm{INC}\, 0$

$3 : g([1],[2],\dots,[n],[0]) \to [n+1]$

$4 : \mathrm{DEC}\, n+1, 2$

$\square$

Примитивно Рекурсивные Функции

Лемма С1.1.

Следующие функции примитивно рекурсивны:

1. $f_n(x) \equiv n,\ n \in \omega$;
2. $g_k(x) = x + k,\ k \in \omega$;
3. $h_{n,m}(x_1, x_2, \ldots, x_m) \equiv n,\ n, m - 1 \in \omega$.

Доказательство.

$f_0(x) = 0(x)$; $f_{l+1}(x) = s(f_l(x))$.

Остальные функции оставляются в качестве упражнения. $\square$

Примитивно Рекурсивные Функции

Лемма C1.2.

Следующие функции примитивно рекурсивны:

1) $f_1(x, y) = x + y$;

2) $f_2(x, y) = x \cdot y$;

3) $f_3(x, y) = x^y$ $(0^0 = 1)$;

4) $\mathrm{sg}(x) = \begin{cases} 0, & \text{если } x = 0; \\ 1, & \text{если } x > 0; \end{cases}$

5) $\overline{\mathrm{sg}}(x) = \begin{cases} 0, & \text{если } x > 0; \\ 1, & \text{если } x = 0; \end{cases}$

6) $f_4(x) = x \dot{-} 1 = \begin{cases} 0, \text{если } x = 0; \\ x - 1, & \text{если } x > 0; \end{cases}$

7) $f_5(x, y) = x \dot{-} y = \begin{cases} 0, \text{если } x \leqslant y; \\ x - 1, & \text{если } x > y; \end{cases}$

8) $f_6(x, y) = |x - y|$.

Примитивно Рекурсивные Функции

Доказательство.

В самом деле,

$$\left[\begin{array}{l} f_1(x,0) = x + 0 = x, \\ f_1(x, y+1) = x + (y+1) = (x+y)+1 = s(f_1(x,y)); \end{array}\right.$$

поэтому $f_1 = R(g,h)$, где $g(x) = \mathrm{I}^1_1(x)$, $h(x,y,z) = s(\mathrm{I}^3_3(x,y,z))$.
Доказательство для остальных функций предлагается в качестве упражнения. □

Примитивно Рекурсивные Функции

Доказательство.

В самом деле,

$$\left[\begin{array}{l} f_1(x,0) = x+0 = x, \\ f_1(x,y+1) = x+(y+1) = (x+y)+1 = s(f_1(x,y)); \end{array}\right.$$

поэтому $f_1 = R(g,h)$, где $g(x) = \mathrm{I}_1^1(x)$, $h(x,y,z) = s(\mathrm{I}_3^3(x,y,z))$. Доказательство для остальных функций предлагается в качестве упражнения. □

Лемма С1.3.

Если $g(x_1,x_2,\ldots,x_n,y)$ — чвф (прф), то и функции

$f(x_1,x_2,\ldots,x_n,y) = \sum\limits_{i=0}^{y} g(x_1,x_2,\ldots,x_n,i)$,

$h(x_1,x_2,\ldots,x_n,y) = \prod\limits_{i=0}^{y} g(x_1,x_2,\ldots,x_n,i)$ также являются чвф (прф).

Примитивно Рекурсивные Функции

Доказательство.

В самом деле,

$$\left[\begin{array}{l} f(x_1, x_2, \ldots, x_n, 0) = g(x_1, x_2, \ldots, x_n, 0), \\ f(x_1, x_2, \ldots, x_n, y+1) = \sum\limits_{i=0}^{y} g(x_1, x_2, \ldots, x_n, i) + \\ +g(x_1, x_2, \ldots, x_n, y+1) = f(x_1, x_2, \ldots, x_n, y) + \\ +g(x_1, x_2, \ldots, x_n, y+1); \end{array}\right.$$

поэтому $f = R(g_1, h_1)$, где

$g_1(x_1, x_2, \ldots, x_n) = g(\mathrm{I}_1^n(\overrightarrow{x}), \mathrm{I}_2^n(\overrightarrow{x}), \ldots, \mathrm{I}_n^n(\overrightarrow{x}), 0(\overrightarrow{x}))$,

$h_1(x_1, x_2, \ldots, x_n, y, z) =$

$f_1(\mathrm{I}_{n+2}^{n+2}(\overrightarrow{x}, y, z), g(\mathrm{I}_1^{n+2}(\overrightarrow{x}, y, z), \mathrm{I}_2^{n+2}(\overrightarrow{x}, y, z), \ldots, \mathrm{I}_n^{n+2}(\overrightarrow{x}, y, z),$

$s(\mathrm{I}_{n+1}^{n+2}(\overrightarrow{x}, y, z))))$.

Функция h рассматривается аналогично и оставляется в качестве упражнения. $\square$

Примитивно Рекурсивные Функции

Определение C1.7.

Будем говорить, что функция f^n получена из всюду определённых функций g^{n+1} и h^n с помощью **ограниченной минимизации** (используется обозначение $f(x_1, x_2, \ldots, x_n) = \mu y \leqslant h(x_1, x_2, \ldots, x_n)[g(x_1, x_2, \ldots, x_n, y) = 0]$), если для всех x_1, x_2, ..., x_n выполняется следующее соотношение:

$$f(\overrightarrow{x}) = \begin{cases} y, & \text{если } g(\overrightarrow{x}, y) = 0 \& (y \leqslant h(\overrightarrow{x})) \& \forall i < y[g(\overrightarrow{x}, i) \neq 0]; \\ h(\overrightarrow{x}) + 1 & \text{в противном случае.} \end{cases}$$

Примитивно Рекурсивные Функции

Определение C1.7.

Будем говорить, что функция f^n получена из всюду определённых функций g^{n+1} и h^n с помощью **ограниченной минимизации** (используется обозначение $f(x_1, x_2, \ldots, x_n) = \mu y \leqslant h(x_1, x_2, \ldots, x_n)[g(x_1, x_2, \ldots, x_n, y) = 0]$), если для всех x_1, x_2, ..., x_n выполняется следующее соотношение:

$$f(\overrightarrow{x}) = \begin{cases} y, & \text{если } g(\overrightarrow{x}, y) = 0 \& (y \leqslant h(\overrightarrow{x})) \& \forall i < y[g(\overrightarrow{x}, i) \neq 0]; \\ h(\overrightarrow{x}) + 1 & \text{в противном случае.} \end{cases}$$

Предложение C1.2.

Если функции g и h примитивно рекурсивны (вычислимы), то и функция, полученная из них с помощью ограниченной минимизации, также примитивно рекурсивна (вычислима).

Примитивно Рекурсивные Функции

Доказательство.

В самом деле, $f(\overrightarrow{x}) = \sum\limits_{i=0}^{h(\overrightarrow{x})} \mathrm{sg}(\prod\limits_{j=0}^{i} g(\overrightarrow{x}, i))$, а по леммам С1.2(1,2,4) и С1.3, получаем требуемое (плюс, разумеется, оператор суперпозиции). $\square$

Вычислимые отношения

Определение C1.8.

Отношение $R \subseteq \omega^n$ называется **вычислимым (примитивно рекурсивным)**, если его характеристическая функция

$$\chi_R(x_1, x_2, \ldots, x_n) = \begin{cases} 0, & \text{если } R(x_1, x_2, \ldots, x_n); \\ 1, & \text{если } \neg R(x_1, x_2, \ldots, x_n); \end{cases}$$

является вычислимой (примитивно рекурсивной).

Вычислимые отношения

Определение C1.8.

Отношение $R \subseteq \omega^n$ называется **вычислимым (примитивно рекурсивным)**, если его характеристическая функция

$$\chi_R(x_1, x_2, \ldots, x_n) = \begin{cases} 0, & \text{если } R(x_1, x_2, \ldots, x_n); \\ 1, & \text{если } \neg R(x_1, x_2, \ldots, x_n); \end{cases}$$

является вычислимой (примитивно рекурсивной).

Обозначение C1.1.

Пусть $P \subseteq \omega^n$, $Q \subseteq \omega^n$; тогда

- $P\&Q = P \wedge Q = P \cup Q = \{\langle x_1, x_2, \ldots, x_n\rangle | P(x_1, x_2, \ldots, x_n) \wedge Q(x_1, x_2, \ldots, x_n)\}$;
- $P \vee Q = P \cup Q = \{\langle x_1, x_2, \ldots, x_n\rangle | P(x_1, x_2, \ldots, x_n) \vee Q(x_1, x_2, \ldots, x_n)\}$;
- $\neg P = \omega^n \setminus P = \overline{P} = \{\langle x_1, x_2, \ldots, x_n\rangle | \neg P(x_1, x_2, \ldots, x_n)\}$;
- $P \to Q = \{\langle x_1, x_2, \ldots, x_n\rangle | \neg P(x_1, x_2, \ldots, x_n) \vee Q(x_1, x_2, \ldots, x_n)\}$.

Вычислимые отношения

Предложение C1.3.

Пусть $P, Q \subseteq \omega^n$ — вычислимые (примитивно рекурсивные) отношения. Тогда $P \wedge Q$, $P \vee Q$, $\neg P$ и $P \to Q$ также вычислимы (примитивно рекурсивны).

Вычислимые отношения

Предложение C1.3.

Пусть $P, Q \subseteq \omega^n$ — вычислимые (примитивно рекурсивные) отношения. Тогда $P \wedge Q$, $P \vee Q$, $\neg P$ и $P \to Q$ также вычислимы (примитивно рекурсивны).

Доказательство.

В самом деле,
$\chi_{P\wedge Q}(x_1, x_2, \ldots, x_n) = \mathrm{sg}(\chi_P(x_1, x_2, \ldots, x_n) + \chi_Q(x_1, x_2, \ldots, x_n))$.
Остальные случаи оставляются в качестве упражнений. $\square$

Вычислимые отношения

Предложение C1.3.

Пусть $P, Q \subseteq \omega^n$ — вычислимые (примитивно рекурсивные) отношения. Тогда $P \wedge Q$, $P \vee Q$, $\neg P$ и $P \rightarrow Q$ также вычислимы (примитивно рекурсивны).

Доказательство.

В самом деле,
$\chi_{P \wedge Q}(x_1, x_2, \ldots, x_n) = \mathrm{sg}(\chi_P(x_1, x_2, \ldots, x_n) + \chi_Q(x_1, x_2, \ldots, x_n))$.
Остальные случаи оставляются в качестве упражнений. $\square$

Обозначение C1.2.

Пусть $P \subseteq \omega^n$, $Q \subseteq \omega^m$; тогда $P \times Q =$
$\{\langle x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m \rangle | P(x_1, x_2, \ldots, x_n) \& Q(y_1, y_2, \ldots, y_m)\}$.

Вычислимые отношения

Обозначение C1.3.

Пусть $R \subseteq \omega^{n+1}$; тогда
$\exists i \leqslant y R(x_1, x_2, \ldots, x_n, i) =$
$\{\langle x_1, x_2, \ldots, x_n, y\rangle | R(x_1, x_2, \ldots, x_n, i)$ для некоторого $i \leqslant y\}$;
$\exists i < y R(x_1, x_2, \ldots, x_n, i) =$
$\{\langle x_1, x_2, \ldots, x_n, y\rangle | R(x_1, x_2, \ldots, x_n, i)$ для некоторого $i < y\}$;
$\forall i \leqslant y R(x_1, x_2, \ldots, x_n, i) =$
$\{\langle x_1, x_2, \ldots, x_n, y\rangle | R(x_1, x_2, \ldots, x_n, i)$ для всех $i \leqslant y\}$;
$\forall i < y R(x_1, x_2, \ldots, x_n, i) = \{\langle x_1, x_2, \ldots, x_n, y\rangle | R(\overrightarrow{x}, i)$ для всех $i < y\}$.

Вычислимые отношения

Обозначение C1.3.

Пусть $R \subseteq \omega^{n+1}$; тогда
$\exists i \leqslant yR(x_1, x_2, \ldots, x_n, i) =$
$\{\langle x_1, x_2, \ldots, x_n, y\rangle | R(x_1, x_2, \ldots, x_n, i)$ для некоторого $i \leqslant y\}$;
$\exists i < yR(x_1, x_2, \ldots, x_n, i) =$
$\{\langle x_1, x_2, \ldots, x_n, y\rangle | R(x_1, x_2, \ldots, x_n, i)$ для некоторого $i < y\}$;
$\forall i \leqslant yR(x_1, x_2, \ldots, x_n, i) =$
$\{\langle x_1, x_2, \ldots, x_n, y\rangle | R(x_1, x_2, \ldots, x_n, i)$ для всех $i \leqslant y\}$;
$\forall i < yR(x_1, x_2, \ldots, x_n, i) = \{\langle x_1, x_2, \ldots, x_n, y\rangle | R(\overrightarrow{x}, i)$ для всех $i < y\}$.

Предложение C1.4.

Пусть $R \subseteq \omega^{n+1}$ — вычислимое (примитивно рекурсивное) отношение. Тогда $\exists i \leqslant yR(\overrightarrow{x}, i)$, $\exists i < yR(\overrightarrow{x}, i)$, $\forall i \leqslant yR(\overrightarrow{x}, i)$, $\forall i < yR(\overrightarrow{x}, i)$ также вычислимы (примитивно рекурсивны).

Вычислимые отношения

Доказательство.

В самом деле, $\chi_{\exists i\leqslant yR}(x_1, x_2, \ldots, x_n, y) = \prod\limits_{i=0}^{y} \chi_R(x_1, x_2, \ldots, x_n, i)$.

Остальные случаи предлагаются в качестве упражнения. $\square$

Вычислимые отношения

Доказательство.

В самом деле, $\chi_{\exists i \leqslant y R}(x_1, x_2, \ldots, x_n, y) = \prod_{i=0}^{y} \chi_R(x_1, x_2, \ldots, x_n, i)$.

Остальные случаи предлагаются в качестве упражнения. □

Предложение C1.5.

Пусть $P \subseteq \omega^n$, $Q \subseteq \omega^m$ — вычислимые (примитивно рекурсивные) отношения. Тогда $P \times Q$ также вычислимо (примитивно рекурсивно).

Доказательство.

В самом деле, $\chi_{P \times Q}(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m) = \mathrm{sg}(\chi_P(x_1, x_2, \ldots, x_n) + \chi_Q(y_1, y_2, \ldots, y_m))$. □

Вычислимые отношения

Доказательство.

В самом деле, $\chi_{\exists i \leqslant y R}(x_1, x_2, \ldots, x_n, y) = \prod_{i=0}^{y} \chi_R(x_1, x_2, \ldots, x_n, i)$.

Остальные случаи предлагаются в качестве упражнения. □

Предложение C1.5.

Пусть $P \subseteq \omega^n$, $Q \subseteq \omega^m$ — вычислимые (примитивно рекурсивные) отношения. Тогда $P \times Q$ также вычислимо (примитивно рекурсивно).

Доказательство.

В самом деле, $\chi_{P \times Q}(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m) = \mathrm{sg}(\chi_P(x_1, x_2, \ldots, x_n) + \chi_Q(y_1, y_2, \ldots, y_m))$. □

Предложение C1.6.

Бинарные отношения $=$, $\neq$, $<$, $>$, $\leqslant$, $\geqslant$ примитивно рекурсивны.

Доказательство.

В самом деле, $\chi_{=}(x, y) = \mathrm{sg}|x - y|$. Остальные случаи предлагаются в качестве упражнения. □

Вычислимые Функции и Множества

Доказательство.

В самом деле, $\chi_{=}(x,y) = \mathrm{sg}|x-y|$. Остальные случаи предлагаются в качестве упражнения. □

Предложение C1.7.

Пусть $R \subseteq \omega^n$ — вычислимое (примитивно рекурсивное) отношение и пусть f_1, f_2, …, f_n — m-арные вычислимые (примитивно рекурсивные) функции. Тогда отношение $Q(x_1, x_2, \ldots, x_m)(\Leftrightarrow R(f_1(x_1, x_2, \ldots, x_m), f_2(x_1, x_2, \ldots, x_m), \ldots, f_n(x_1, x_2, \ldots, x_m)))$ также вычислимо (примитивно рекурсивно).

Вычислимые Функции и Множества

Доказательство.

В самом деле, $\chi_=(x, y) = \mathrm{sg}|x - y|$. Остальные случаи предлагаются в качестве упражнения. □

Предложение С1.7.

Пусть $R \subseteq \omega^n$ — вычислимое (примитивно рекурсивное) отношение и пусть f_1, f_2, …, f_n — m-арные вычислимые (примитивно рекурсивные) функции. Тогда отношение $Q(x_1, x_2, \ldots, x_m)(\Leftrightarrow R(f_1(x_1, x_2, \ldots, x_m), f_2(x_1, x_2, \ldots, x_m), \ldots, f_n(x_1, x_2, \ldots, x_m)))$ также вычислимо (примитивно рекурсивно).

Доказательство.

В самом деле, $\chi_Q(x_1, x_2, \ldots, x_m) = \chi_R(f_1(x_1, x_2, \ldots, x_m), f_2(x_1, x_2, \ldots, x_m), \ldots, f_n(x_1, x_2, \ldots, x_m))$. □

Вычислимые Функции и Множества

Предложение C1.8.

Пусть R_1, R_2, ..., $R_k \subseteq \omega^n$ — дизъюнктная последовательность вычислимых (примитивно рекурсивных) отношений такая, что $R_1 \cup R_2 \cup \ldots \cup R_k = \omega^n$. Пусть также f_1, f_2, ..., f_k — n-арные вычислимые (примитивно рекурсивные) функции. Тогда

$$f(x_1, x_2, \ldots, x_n) = \begin{cases} f_1(x_1, x_2 \ldots, x_n), & \text{если } R_1(x_1, x_2, \ldots, x_n); \\ f_2(x_1, x_2 \ldots, x_n), & \text{если } R_2(x_1, x_2, \ldots, x_n); \\ \ldots, & \ldots \\ f_k(x_1, x_2 \ldots, x_n), & \text{если } R_k(x_1, x_2, \ldots, x_n); \end{cases}$$

также является вычислимой (примитивно рекурсивной) функцией.

Вычислимые Функции и Множества

Предложение C1.8.

Пусть R_1, R_2, ..., $R_k \subseteq \omega^n$ — дизъюнктная последовательность вычислимых (примитивно рекурсивных) отношений такая, что $R_1 \cup R_2 \cup \ldots \cup R_k = \omega^n$. Пусть также f_1, f_2, ..., f_k — n-арные вычислимые (примитивно рекурсивные) функции. Тогда

$$f(x_1, x_2, \ldots, x_n) = \begin{cases} f_1(x_1, x_2 \ldots, x_n), & \text{если } R_1(x_1, x_2, \ldots, x_n); \\ f_2(x_1, x_2 \ldots, x_n), & \text{если } R_2(x_1, x_2, \ldots, x_n); \\ \ldots, & \ldots \\ f_k(x_1, x_2 \ldots, x_n), & \text{если } R_k(x_1, x_2, \ldots, x_n); \end{cases}$$

также является вычислимой (примитивно рекурсивной) функцией.

Доказательство.

В самом деле, $f(\overrightarrow{x}) =$
$f_1(\overrightarrow{x}) \cdot \overline{\mathrm{sg}}\chi_{R_1}(\overrightarrow{x}) + f_2(\overrightarrow{x}) \cdot \overline{\mathrm{sg}}\chi_{R_2}(\overrightarrow{x}) + \ldots + f_k(\overrightarrow{x}) \cdot \overline{\mathrm{sg}}\chi_{R_k}(\overrightarrow{x})$. $\square$

Вычислимые Функции и Множества

Обозначение С1.4.

Пусть R — $n+1$-арное отношение на ω и пусть h — n-арная всюду определённая функция. Положим
$\mu y.R(x_1, x_2, \ldots, x_n, y) = \mu y.[\chi_R(x_1, x_2, \ldots, x_n, y) = 0]$,
$\mu y \leqslant h(x_1, x_2, \ldots, x_n).R(x_1, x_2, \ldots, x_n, y) = \mu y \leqslant h(x_1, x_2, \ldots, x_n).[\chi_R(x_1, x_2, \ldots, x_n, y) = 0]$.

Вычислимые Функции и Множества

Обозначение C1.4.

Пусть R — $n+1$-арное отношение на ω и пусть h — n-арная всюду определённая функция. Положим
$\mu y.R(x_1, x_2, \ldots, x_n, y) = \mu y.[\chi_R(x_1, x_2, \ldots, x_n, y) = 0]$,
$\mu y \leqslant h(x_1, x_2, \ldots, x_n).R(x_1, x_2, \ldots, x_n, y) = \mu y \leqslant h(x_1, x_2, \ldots, x_n).[\chi_R(x_1, x_2, \ldots, x_n, y) = 0]$.

Предложение C1.9.

Если R — $n+1$-арное вычислимое (примитивно рекурсивное) отношение, то $\mu y.R(x_1 x_2, \ldots, x_n, y)$ будет ч.в.ф. Если, к тому же, h — n-арная вычислимая (примитивно рекурсивная) функция, то $\mu y \leqslant h(x_1, x_2, \ldots, x_n).R(x_1, x_2, \ldots, x_n, y)$ также будет вычислимой (примитивно рекурсивной) функцией.

Вычислимые Функции и Множества

Обозначение C1.4.

Пусть R — $n+1$-арное отношение на ω и пусть h — n-арная всюду определённая функция. Положим
$\mu y.R(x_1, x_2, \ldots, x_n, y) = \mu y.[\chi_R(x_1, x_2, \ldots, x_n, y) = 0]$,
$\mu y \leqslant h(x_1, x_2, \ldots, x_n).R(x_1, x_2, \ldots, x_n, y) = \mu y \leqslant h(x_1, x_2, \ldots, x_n).[\chi_R(x_1, x_2, \ldots, x_n, y) = 0]$.

Предложение C1.9.

Если R — $n+1$-арное вычислимое (примитивно рекурсивное) отношение, то $\mu y.R(x_1 x_2, \ldots, x_n, y)$ будет ч.в.ф. Если, к тому же, h — n-арная вычислимая (примитивно рекурсивная) функция, то $\mu y \leqslant h(x_1, x_2, \ldots, x_n).R(x_1, x_2, \ldots, x_n, y)$ также будет вычислимой (примитивно рекурсивной) функцией.

Доказательство.

Непосредственно вытекает из определения C1.8 и предложения C1.2. $\square$

Вычислимые Функции и Множества

Лемма С1.4.

1. Функция $\left[\frac{x}{y}\right]$ взятия неполного частного при делении x на y является примитивно рекурсивной (полагаем $\left[\frac{x}{0}\right] = x$).
2. Отношение $\mathrm{Div}(x, y)(\Leftrightarrow x|y)$ является примитивно рекурсивным.
3. Множество $\mathrm{Prime}(x)$ всех простых чисел является примитивно рекурсивным.

Вычислимые Функции и Множества

Лемма C1.4.

1. Функция $\left[\frac{x}{y}\right]$ взятия неполного частного при делении x на y является примитивно рекурсивной (полагаем $\left[\frac{x}{0}\right] = x$).
2. Отношение $\mathrm{Div}(x, y)(\Leftrightarrow x|y)$ является примитивно рекурсивным.
3. Множество $\mathrm{Prime}(x)$ всех простых чисел является примитивно рекурсивным.

Доказательство.

1) $\left[\frac{x}{y}\right] = z \iff ((y = 0) \wedge (z = x) \vee (y > 0) \wedge (z \leqslant \frac{x}{y} < z + 1)) \iff$ $((y = 0) \wedge (z = x) \vee (y > 0) \wedge (y \cdot z \leqslant x < y \cdot (z + 1)))$. Кроме того, имеем $\left[\frac{x}{y}\right] \leqslant x$. Следовательно,

$$\left[\frac{x}{y}\right] = \mu z \leqslant x.((x = z) \vee (x < y \cdot (z + 1))).$$

Вычислимые Функции и Множества

Доказательство (окончание).

2) $x|y \iff \exists z \leqslant y(z \cdot x = y)$.

3) $\mathrm{Prime}(x) \iff ((x > 1) \wedge \forall y \leqslant x(y|x \to ((y = 1) \vee (y = x))))$. $\square$

Вычислимые Функции и Множества

Доказательство (окончание).

2) $x|y \iff \exists z \leqslant y(z \cdot x = y)$.
3) $\mathrm{Prime}(x) \iff ((x > 1) \wedge \forall y \leqslant x(y|x \to ((y = 1) \vee (y = x))))$. $\square$

Лемма C1.5.

1. Функция $f(x) = p_x$, где $p_0 = 2$, $p_1 = 3$, $p_2 = 5$, $\ldots$ — перечисление без повторений простых чисел в порядке возрастания, является примитивно рекурсивной функцией.
2. Функция $\mathrm{ex}(i, x)$, показатель простого числа p_i в разложении числа x (считаем $\mathrm{ex}(i, 0) = 0$), является примитивно рекурсивной.

Вычислимые Функции и Множества

Доказательство.

1) $\left[\begin{array}{l} p(0) = 2, \\ p(x+1) = \mu y \leqslant s(p(x)!).(\mathrm{Prime}(y) \wedge (y > p(x))); \end{array}\right.$
в этом случае $f = R(a, g)$, где $a = 2$ и
$g(x, z) = \mu y \leqslant s(z!)(\mathrm{Prime}(y) \wedge (y > z))$ (из доказательства теоремы Евклида о бесконечности простых чисел вытекает, что $p_{x+1} \leqslant p_0 \cdot p_1 \cdot \ldots \cdot p_x + 1 \leqslant (p_x)! + 1$).
2) $\mathrm{ex}(i, x) = \mu y \leqslant x.(\neg \mathrm{Div}(p_i^{y+1}, x) \vee (x = 0))$ (действительно, $y < p_i^y \leqslant x$). $\square$

Вычислимые Функции и Множества

Доказательство.

1) $\left[\begin{array}{l} p(0) = 2, \\ p(x+1) = \mu y \leqslant s(p(x)!).(\mathrm{Prime}(y) \wedge (y > p(x))); \end{array}\right.$
в этом случае $f = R(a, g)$, где $a = 2$ и
$g(x, z) = \mu y \leqslant s(z!)(\mathrm{Prime}(y) \wedge (y > z))$ (из доказательства теоремы Евклида о бесконечности простых чисел вытекает, что $p_{x+1} \leqslant p_0 \cdot p_1 \cdot \ldots \cdot p_x + 1 \leqslant (p_x)! + 1$).
2) $\mathrm{ex}(i, x) = \mu y \leqslant x.(\neg \mathrm{Div}(p_i^{y+1}, x) \vee (x = 0))$ (действительно, $y < p_i^y \leqslant x$). □

Замечание C1.3.

В доказательстве п. 1 использовано неравенство $p_{x+1} \leqslant (p_x!) + 1$. Применяя теорему Чебышева, гласящую, что для любого $n \geqslant 1$ среди чисел n, $n+1$, …, $2n$ найдётся простое число, можно получить более точную оценку $p_x \leqslant 2^{x+1}$.

Спасибо за внимание.