Funkcje rekurencyjne

Jerzy Pogonowski

Zakład Logiki Stosowanej UAM
www.logic.amu.edu.pl
pogon@amu.edu.pl

Funkcje rekurencyjne

Plan na dziś:

- definicja funkcji pierwotnie, częściowo i ogólnie rekurencyjnych;
- przykłady i proste własności funkcji rekurencyjnych;
- definicje przez schematy rekursji;
- funkcje kodujące liczby naturalne.

Będziemy korzystać z definicji oraz przykładów zamieszczonych w:

- I.A. Ławrow, L.L. Maksimowa *Zadania z teorii mnogości, logiki matematycznej i teorii algorytmów*. Wydawnictwo Naukowe PWN, Warszawa 2004. Z języka rosyjskiego przełożył Jerzy Pogonowski.

Definicja funkcji rekurencyjnych

Częściowe funkcje liczbowe $f^n(x_1, \ldots, x_n)$ (dla $n = 1, 2, \ldots$), to funkcje określone na pewnym podzbiorze zbioru $\mathcal{N}^n$ o wartościach będących liczbami naturalnymi.

Dla dowolnych liczb $a_1, \ldots, a_n \in \mathcal{N}$ oraz funkcji f^k i g^s piszemy

$$f^k(a_{j_1}, \ldots, a_{j_k}) = g^s(a_{j_1}, \ldots, a_{j_s}),$$

jeśli: albo wartości $f^k(a_{j_1}, \ldots, a_{j_k})$ oraz $g^s(a_{j_1}, \ldots, a_{j_s})$ są nieokreślone albo są obie określone i identyczne.

n-argumentowa funkcja $f^n(x_1, \ldots, x_n)$ jest *całkowita*, jeśli jej dziedziną jest cały zbiór $\mathcal{N}^n$, czyli gdy $\delta_{f^n} = \mathcal{N}^n$.

Funkcje proste, złożenie i podstawienie

Następujące funkcje całkowite nazywamy *prostymi*:

- $s^1(x) = x + 1$,
- $o^1(x) = 0$,
- $I^n_m(x_1, \ldots, x_n) = x_m \quad (\text{dla} \quad 1 \leqslant m \leqslant n)$.

Funkcja $h^n(x_1, \ldots, x_n) = g^m(f^n_1(x_1, \ldots, x_n), \ldots, f^n_m(x_1, \ldots, x_n))$ otrzymywana jest z funkcji $g^m, f^n_1, \ldots, f^n_m$ przez operację *złożenia*.

Funkcję $h^n(x_1, \ldots, x_n) = g^m(t_1, \ldots, t_m)$ otrzymujemy z pomocą operacji *podstawienia* z funkcji $g^m, f_1, \ldots, f_k$, gdy $t_i = f_j(x_{j_1}, \ldots, x_{j_s})$, gdzie każde x_{j_l} jest jedną ze zmiennych $x_1, \ldots, x_n$ lub t_i jest jedną ze zmiennych $x_1, \ldots, x_n$.

Schemat rekursji prostej

Funkcję $f^{n+1}(x_1, \ldots, x_n, y)$ otrzymujemy z funkcji $g^n(x_1, \ldots, x_n)$ oraz $h^{n+2}(x_1, \ldots, x_n, y, z)$ za pomocą *operatora rekursji prostej*, gdy może ona być określona następującym *schematem rekursji prostej*:

- $f^{n+1}(x_1, \ldots, x_n, 0) = g^n(x_1, \ldots, x_n)$,
- $f^{n+1}(x_1, \ldots, x_n, y+1) = h^{n+2}(x_1, \ldots, x_n, y, f^{n+1}(x_1, \ldots, x_n, y))$.

Dla $n = 0$ schemat rekursji prostej przyjmuje następującą postać:

- $f(0) = a$,
- $f(y+1) = g(y, f(y))$,

gdzie a jest jednoargumentową funkcją stałą o wartości a.

Minimum efektywne

Funkcję $f^n(x_1, \ldots, x_n)$ otrzymujemy z funkcji $g^{n+1}(x_1, \ldots, x_n, y)$ za pomocą operacji *minimum efektywnego* (za pomocą *μ-operatora*), co zaznaczamy następująco:

$$f^n(x_1, \ldots, x_n) = \mu y[g^{n+1}(x_1, \ldots, x_n, y) = 0],$$

gdy spełniony jest warunek:

$f^n(x_1, \ldots, x_n)$ jest określone i równe y wtedy i tylko wtedy, gdy $g(x_1, \ldots, x_n, 0), \ldots, g(x_1, \ldots, x_n, y-1)$ są wszystkie określone i różne od 0, zaś $g(x_1, \ldots, x_n, y) = 0$.

Funkcje: pierwotnie, częściowo i ogólnie rekurencyjne

- Funkcja $f(x_1, \ldots, x_n)$ jest *pierwotnie rekurencyjna* (*prf*), jeśli może być otrzymana z funkcji prostych za pomocą skończonej liczby zastosowań operacji złożenia oraz rekursji prostej.
- Funkcja $f(x_1, \ldots, x_n)$ jest *częściowo rekurencyjna* (*crf*), jeśli może być otrzymana z funkcji prostych za pomocą skończonej liczby zastosowań operacji złożenia, rekursji prostej oraz minimum efektywnego.
- Funkcja $f(x_1, \ldots, x_n)$ jest *ogólnie rekurencyjna* (*orf*), gdy jest ona całkowitą funkcją częściowo rekurencyjną.

Każda funkcja pierwotnie rekurencyjna jest też ogólnie rekurencyjna (lecz nie na odwrót).

Ograniczony μ-operator

Funkcję $f^n(x_1, \ldots, x_n)$ otrzymujemy z funkcji

$$g^{n+1}(x_1, \ldots, x_n, y) \text{ oraz } h^n(x_1, \ldots, x_n)$$

za pomocą *ograniczonego μ-operatora*, jeśli dla wszystkich $x_1, \ldots, x_n$:

$$\mu y[g^{n+1}(x_1, \ldots, x_n, y) = 0]$$

jest określone i nie większe niż $h^n(x_1, \ldots, x_n)$ oraz

$$f^n(x_1, \ldots, x_n) = \mu y[g^{n+1}(x_1, \ldots, x_n, y) = 0].$$

Inne schematy rekursji

Funkcję f^{n+1} otrzymujemy z $g^n, h^{n+s+1}, t_1^1, \ldots, t_s^1$ z pomocą schematu *rekursji zwrotnej*, gdy może ona być określona schematem:

$$
\begin{aligned}
&f^{n+1}(x_1, \ldots, x_n, 0) = g^n(x_1, \ldots, x_n),\\
&f^{n+1}(x_1, \ldots, x_n, y+1) =\\
&\qquad = h^{n+s+1}(x_1, \ldots, x_n, y, f(x_1, \ldots, x_n, t_1(y+1)), \ldots\\
&\qquad\qquad \ldots, f(x_1, \ldots, x_n, t_s(y+1))),
\end{aligned}
$$

gdzie $t_1(y+1) \leqslant y, \ldots, t_s(y+1) \leqslant y$.

Jeśli funkcje $g, h, t_1, \ldots t_s$ są pierwotnie rekurencyjne, to funkcja f jest pierwotnie rekurencyjna.

Inne schematy rekursji

Niech $f_1^{n+1}, \ldots, f_k^{n+1}$ będą zdefiniowane przez *rekursję jednoczesną*, tzn. za pomocą następującego schematu:

$$\begin{cases} f_i^{n+1}(x_1, \ldots, x_n, 0) = g_i^n(x_1, \ldots, x_n), \\ f_i^{n+1}(x_1, \ldots, x_n, y+1) = \\ = h_i^{n+k+1}(x_1, \ldots, x_n, y, f_1(x_1, \ldots, x_n, y), \ldots, f_k(x_1, \ldots, x_n, y)) \end{cases}$$

dla wszystkich $1 \leqslant i \leqslant k$.

Można udowodnić, że jeśli funkcje $g_1, \ldots, g_k, h_1, \ldots, h_k$ są pierwotnie rekurencyjne, to funkcje $f_1, \ldots, f_k$ są pierwotnie rekurencyjne.

Inne schematy rekursji

Schemat rekursji ograniczonej ma postać następującą:

$$f(x_1, \ldots, x_n, 0) = g(x_1, \ldots, x_n)$$

$$f(x_1, \ldots, x_n, x + 1) = h(x_1, \ldots, x_n, x, f(x_1, \ldots, x_n, x))$$

$$f(x_1, \ldots, x_n, x) \leqslant j(x_1, \ldots, x_n, x).$$

Możliwe są różne dalsze schematy rekursji.
Definiowanie przez rekursję to ważne narzędzie w językach programowania.

Funkcje elementarnie rekurencyjne

Klasa funkcji elementarnie rekurencyjnych to najmniejsza klasa funkcji zawierająca funkcje:

- odejmowania $\dot{-}$,
- funkcję wykładniczą,
- funkcję następnika,

oraz zamknięta ze względu na operacje:

- złożenia,
- minimum ograniczonego.

Można udowodnić, że klasa wszystkich funkcji elementarnie rekurencyjnych jest zawarta w klasie wszystkich funkcji pierwotnie rekurencyjnych (i ta inkluzja jest właściwa).

Hierarchia Grzegorczyka

Hierarchia Grzegorczyka. Niech: $f_0(x,y) = y+1$, $f_1(x,y) = x+y$, $f_2(x,y) = (x+1)\cdot(y+1)$, i dla $n \geqslant 2$:
$f_{n+1}(0,y) = f_n(x+1,y+1)$
$f_{n+1}(x+1,y) = f_{n+1}(x, f_{n+1}(x,y))$.

Dla dowolnego n niech E_n będzie najmniejszą klasą funkcji zawierającą funkcje: I^2_1, I^2_2, funkcję następnika oraz funkcję f_n i zamkniętą ze względu na złożenie i schemat rekursji ograniczonej. Wtedy:

- E_3 jest równa klasie funkcji elementarnie rekurencyjnych;
- dla każdego n mamy: $E_n \subset E_{n+1}$ (wszystkie inkluzje właściwe);
- $\bigcup\limits_n E_n$ jest równy klasie wszystkich funkcji pierwotnie rekurencyjnych;
- dla każdego n funkcje f_n są ściśle rosnące względem każdego z argumentów;
- dla każdego n funkcja $f_{n+1}(x,x)$ rośnie szybciej niż wszystkie funkcje klasy E_n.

Oznaczenia

Stosujemy oznaczenie:

$$f^n(x_1, \ldots, x_n) = \mu y[g(x_1, \ldots, x_n, y) = h(x_1, \ldots, x_n, y)],$$

gdy spełniony jest warunek: $f^n(x_1, \ldots, x_n)$ jest określone i równe y wtedy i tylko wtedy, gdy $g(x_1, \ldots, x_n, i)$ oraz $h(x_1, \ldots, x_n, i)$ są określone dla $i = 0, 1, \ldots, y$, $g(x_1, \ldots, x_n, i) \neq h(x_1, \ldots, x_n, i)$ dla $i < y$ oraz $g(x_1, \ldots, x_n, y) = h(x_1, \ldots, x_n, y)$.

W podobny sposób rozumiemy oznaczenia:

- $\mu y[\varphi(x_1, \ldots, x_n, y) \neq g(x_1, \ldots, x_n, y)]$,
- $\mu y[\varphi(x_1, \ldots, x_n, y) \leqslant g(x_1, \ldots, x_n, y)]$,
- $\mu y[\varphi(x_1, \ldots, x_n, y) < g(x_1, \ldots, x_n, y)]$, itd.

Iteracja, odwrócenie, funkcja uniwersalna

Będziemy mówić, że funkcję $f(x)$ otrzymujemy z funkcji $g(x)$ przez *iterację* i oznaczać ten fakt przez $f(x) = \mathrm{i}g(x)$, gdy

- $f(0) = 0$,
- $f(x+1) = g(f(x))$.

Będziemy mówić, że funkcję $f(x)$ otrzymujemy z funkcji $g(x)$ przez *odwrócenie* i zaznaczać ten fakt przez $f(x) = g^{-1}(x)$, gdy

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

Niech $\boldsymbol{G}$ będzie pewną rodziną n-argumentowych funkcji częściowych. Funkcję F^{n+1} nazwiemy *funkcją uniwersalną* dla $\boldsymbol{G}$, jeśli

$$\boldsymbol{G} = \{F(0, x_1, \ldots, x_n), F(1, x_1, \ldots, x_n), \ldots\}.$$

Przykłady funkcji rekurencyjnych

Następujące funkcje są pierwotnie rekurencyjne:

- (1) $f(x) = x + n$;
- (2) $f(x) = n$;
- (3) $f(x, y) = x + y$;
- (4) $f(x, y) = x \cdot y$;
- (5) $f(x, y) = x^y$ (przyjmujemy $0^0 = 1$);
- (6) $f(x) = x!$ (przyjmujemy $0! = 1$).

Przykłady funkcji rekurencyjnych

Dowód.

- (1) $f(x) = \underbrace{s(s(\ldots s}_{n \text{ razy}}(x)\ldots))$.
- (2) $f(x) = \underbrace{s(s(\ldots s}_{n \text{ razy}}(o(x))\ldots))$.
- (3) $f(x, y)$ otrzymujemy przez rekursję prostą z funkcji $g(x) = I^1_1(x)$ oraz $h(x, y, z) = s(I^3_3(x, y, z))$.
- (4) $f(x, y)$ otrzymujemy przez rekursję prostą z $g(x) = o(x)$ i $h(x, y, z) = I^3_1(x, y, z) + I^3_3(x, y, z)$.
- (5) $f(x, 0) = 1,\ f(x, y + 1) = x \cdot f(x, y)$.
- (6) $f(0) = 1,\ f(x + 1) = s(x) \cdot f(x)$.

Przykłady funkcji rekurencyjnych

Następujące funkcje są pierwotnie rekurencyjne:

- (7) $sg(x) = \begin{cases} 0, & \text{gdy} \quad x = 0 \\ 1, & \text{gdy} \quad x > 0; \end{cases}$
- (8) $\overline{sg}(x) = \begin{cases} 0, & \text{gdy} \quad x > 0 \\ 1, & \text{gdy} \quad x = 0; \end{cases}$
- (9) $x \dot{-} 1 = \begin{cases} 0, & \text{gdy} \quad x = 0 \\ x - 1, & \text{gdy} \quad x > 0; \end{cases}$
- (10) $x \dot{-} y = \begin{cases} 0, & \text{gdy} \quad x \leqslant y \\ 1, & \text{gdy} \quad x > y; \end{cases}$
- (11) $|x - y|$;
- (12) $\max(x, y)$;
- (13) $\min(x, y)$.

Przykłady funkcji rekurencyjnych

Dowód.

- (7) $\mathrm{sg}\,0 = 0$; $\mathrm{sg}(x+1) = s(o(x))$.
- (8) $\overline{\mathrm{sg}}0 = 1$, $\overline{\mathrm{sg}}(x+1) = o(x)$.
- (9) $0 \dot{-} 1 = 0$, $(x+1) \dot{-} 1 = x$.
- (10) $x \dot{-} 0 = x$, $x \dot{-} (y+1) = (x \dot{-} y) \dot{-} 1$.
- (11) $|x - y| = (x \dot{-} y) + (y \dot{-} x)$.
- (12) $\max(x, y) = x \cdot \mathrm{sg}(x \dot{-} y) + y \cdot \overline{\mathrm{sg}}(x \dot{-} y)$.
- (13) $\min(x, y) = x \cdot \mathrm{sg}(y \dot{-} x) + y \cdot \overline{\mathrm{sg}}(y \dot{-} x)$.

Przykłady funkcji rekurencyjnych

Niech funkcje $f_0^n, f_1^n, \ldots, f_s^n$ mają następującą własność: dla dowolnych argumentów $x_1, \ldots, x_n$ jedna i tylko jedna z tych funkcji równa jest 0. Powiemy, że funkcja g^n jest *określona warunkowo*, gdy

$$g^n(x_1, \ldots, x_n) = \begin{cases} h_0^n(x_1, \ldots, x_n), & \text{gdy} \quad f_0^n(x_1, \ldots, x_n) = 0, \\ \ldots & \ldots \quad \ldots \\ h_s^n(x_1, \ldots, x_n), & \text{gdy} \quad f_s^n(x_1, \ldots, x_n) = 0. \end{cases}$$

Jeśli funkcje $h_0^n, \ldots, h_s^n, f_0^n, \ldots, f_s^n$ są pierwotnie rekurencyjne, to g^n jest pierwotnie rekurencyjna.

Dowód. $g(x_1, \ldots, x_n) = h_0(x_1, \ldots, x_n) \cdot \overline{\mathrm{sg}}\, f_0(x_1, \ldots, x_n) + \ldots$
$\ldots + h_s(x_1, \ldots, x_n) \cdot \overline{\mathrm{sg}}\, f_s(x_1, \ldots, x_n).$

Przykłady funkcji rekurencyjnych

Niech $g^{n+1}, \alpha^m, \beta^m$ będą funkcjami pierwotnie rekurencyjnymi. Wtedy następujące funkcje są pierwotnie rekurencyjne:

(14) $f^{n+1}(x_1, \ldots, x_n, x_{n+1}) = \sum\limits_{i=0}^{x_{n+1}} g(x_1, \ldots, x_n, i)$;

(15) $f^{n+2}(x_1, \ldots, x_n, y, z) = \begin{cases} \sum\limits_{i=y}^{z} g(x_1, \ldots, x_n, i), & \text{gdy} \quad y \leqslant z, \\ 0, & \text{gdy} \quad y > z; \end{cases}$

(16) $f^{n+m}(x_1, \ldots, x_n, y_1, \ldots, y_m) =$

$$= \begin{cases} \sum\limits_{i=\alpha(y_1,\ldots,y_m)}^{\beta(y_1,\ldots,y_m)} g(x_1, \ldots, x_n, i), & \text{gdy } \alpha(y_1, \ldots, y_m) \leqslant \beta(y_1, \ldots, y_m), \\ 0, & \text{w pozostałych przypadkach}; \end{cases}$$

Przykłady funkcji rekurencyjnych

(17) $f^{n+1}(x_1, \ldots, x_n, x_{n+1}) = \prod_{i=0}^{x_{n+1}} g(x_1, \ldots, x_n, i);$

(18) $f^{n+2}(x_1, \ldots, x_n, y, z) = \begin{cases} \prod_{i=y}^{z} g(x_1, \ldots, x_n, i), & \text{gdy} \quad y \leqslant z \\ 0, & \text{gdy} \quad y > z; \end{cases}$

(19) $f^{n+m}(x_1, \ldots, x_n, y_1, \ldots, y_m) =$

$$= \begin{cases} \prod_{i=\alpha(y_1,\ldots,y_m)}^{\beta(y_1,\ldots,y_m)} g(x_1, \ldots, x_n, i), & \text{gdy } \alpha(y_1, \ldots, y_m) \leqslant \beta(y_1, \ldots, y_m), \\ 0, & \text{w pozostałych przypadkach.} \end{cases}$$

Jeśli funkcję f otrzymujemy z funkcji pierwotnie rekurencyjnych g i h za pomocą ograniczonego μ-operatora, to f jest pierwotnie rekurencyjna.

Dowód. $f(x_1, \ldots, x_n) = \sum_{i=0}^{h(x_1,\ldots,x_n)} \mathrm{sg}(\prod_{j=0}^{i} g(x_1, \ldots, x_n, j)).$

Przykłady funkcji rekurencyjnych

Następujące funkcje są pierwotnie rekurencyjne:

- (20) $[\frac{x}{y}]$ — część całkowita z dzielenia x przez y (przyjmujemy, że $[\frac{x}{0}] = x$);
- (21) $\mathrm{rest}(x, y)$ — reszta z dzielenia x przez y (przyjmujemy, że $\mathrm{rest}(x, 0) = x$);
- (22) $\tau(x)$ — liczba dzielników liczby x, gdzie $\tau(0) = 0$;
- (23) $\sigma(x)$ — suma dzielników liczby x, gdzie $\sigma(0) = 0$;
- (24) $\mathrm{lh}(x)$ — liczba dzielników liczby x, które są liczbami pierwszymi (przyjmujemy $\mathrm{lh}(0) = 0$);
- (25) $\pi(x)$ — liczba liczb pierwszych nie większych niż x;
- (26) $k(x, y)$ — najmniejsza wspólna wielokrotność liczb x i y, gdzie $k(x, 0) = k, \ k(0, y) = 0$;
- (27) $d(x, y)$ — największy wspólny dzielnik liczb x i y, gdzie $d(0, 0) = 0$.

Przykłady funkcji rekurencyjnych

(20) $[\frac{x}{y}] = \sum\limits_{i=1}^{x} \overline{\text{sg}}\,(iy \dot{-} x)$.

(21) $\text{rest}\,(x, y) = x \dot{-} y[\frac{x}{y}]$.

(22) $\tau(x) = \sum\limits_{i=1}^{x} \overline{\text{sg}}\,(\text{rest}\,(x, i))$.

(23) $\sigma(x) = \sum\limits_{i=1}^{x} i\overline{\text{sg}}\,(\text{rest}\,(x, i))$.

(24) x jest liczbą pierwszą $\Leftrightarrow \tau(x) = 2$ (zob. (22));

$\text{lh}\,(x) = \sum\limits_{i=1}^{x} \overline{\text{sg}}\,(|\tau(i) - 2| + \text{rest}\,(x, i))$.

(25) $\pi(x) = \sum\limits_{i=1}^{x} \overline{\text{sg}}\,(|\tau(i) - 2|)$ (zob. (22)).

(26) $k(x, y) = \mu z[z \cdot \overline{\text{sg}}\,(x \cdot y) + \text{sg}(x \cdot y)(\overline{\text{sg}}\,z +$

$+\text{rest}\,(z, x) + \text{rest}\,(z, y)) = 0] \leqslant x \cdot y$.

(27) $d(x, y) = [\frac{xy}{k(x,y)}] + x \cdot \overline{\text{sg}}\,y + y \cdot \overline{\text{sg}}\,x$ (zob. (26)).

Przykłady funkcji rekurencyjnych

Następujące funkcje są pierwotnie rekurencyjne:

- (28) $p(x)$ – x-ta liczba pierwsza $(p(0) = 2, p(1) = 3, p(2) = 5, \ldots)$;
- (29) $\mathrm{long}(x)$ – numer największego dzielnika liczby x, będącego liczbą pierwszą;
- (30) $\mathrm{ex}(x, y)$ – wykładnik potęgi x-tej liczby pierwszej $p(x)$ w kanonicznym rozkładzie liczby y na czynniki pierwsze; przyjmujemy, że $\mathrm{ex}(x, 0) = 0$;
- (31) $[\sqrt{x}\,]$;
- (32) $[\sqrt[y]{x}\,]$, gdzie $[\sqrt[0]{x}\,] = x$;
- (33) $[x\sqrt{2}\,]$.

Przykłady funkcji rekurencyjnych

- (28) $p(x) = \mu y[|\pi(y) - (x+1)| = 0] \leqslant 2^{2^x}$ (zob. (25)).
- (29) $\operatorname{long}(x) = \mu y[\sum_{i=y+1}^{x} \overline{\operatorname{sg}}(\operatorname{rest}(x, p(i))) = 0] \leqslant x$.
- (30) $\operatorname{ex}(x, y) = \mu z[(\overline{\operatorname{sg}}\operatorname{rest}(y, (p(x))^{z+1})) \cdot \operatorname{sg} y = 0] \leqslant x$ (zob. (28)).
- (31) $[\sqrt{x}\,] = \mu z[\overline{\operatorname{sg}}((z+1)^2 \dot{-} x) = 0] \leqslant x$.
- (32) $[\,\sqrt[y]{x}\,] = \mu z[\overline{\operatorname{sg}}((z+1)^y \dot{-} x) \cdot \operatorname{sg} y = 0] + \overline{\operatorname{sg}}\, y \cdot x \leqslant x$.
- (33) $[x\sqrt{2}\,] = \mu z[\overline{\operatorname{sg}}((z+1)^2 - 2x^2) = 0] \leqslant 2x$.

Jest nieskończenie wiele funkcji pierwotnie (elementarnie, częściowo, ogólnie) rekurencyjnych.
Każda z tych klas zawiera jednak tylko $\aleph_0$ funkcji.
Ponieważ wszystkich funkcji z $\mathcal{N}^n$ w $\mathcal{N}$ jest kontinuum, więc prawie wszystkie funkcje są poza klasą funkcji ogólnie rekurencyjnych.

Niektóre własności funkcji rekurencyjnych

Funkcja numerująca Cantora $c(x, y) = \frac{(x+y)^2+3x+y}{2}$ ustanawia wzajemnie jednoznaczną odpowiedniość między $\mathcal{N}^2$ a $\mathcal{N}$ (koduje pary liczb naturalnych).

Niech $l(x)$ i $r(x)$ będą takie, że $c(l(x), r(x)) = x$. Wtedy $l(x)$ i $r(x)$ są pierwotnie rekurencyjne oraz $l(c(x, y)) = x$, $r(c(x, y)) = y$.

$l(x) + r(x) = \mu z[\overline{\mathrm{sg}}\,([\frac{(z+1)(z+2)}{2}] \dot{-} x) = 0] = z_0(x) \leqslant 2x$;

$$l(x) = x \dot{-} [\frac{z_0(x) \cdot (z_0(x) + 1)}{2}];$$

$$r(x) = z_0(x) \dot{-} l(x).$$

Niektóre własności funkcji rekurencyjnych

Dla każdego $n \geqslant 1$ zdefiniujmy funkcje:$c^1(x) = x_1$

$$c^{n+1}(x_1, x_2, x_3, \ldots, x_{n+1}) = c^n(c(x_1, x_2), x_3, \ldots, x_{n+1}).$$

Niech c_{ni} $(1 \leqslant i \leqslant n)$ będą takie, że $c^n(c_{n1}(x), \ldots c_{nn}(x)) = x$. Wtedy:

- Zachodzą równości: $c_{ni}(c^n(x_1, \ldots, x_n)) = x_i$ dla $1 \leqslant i \leqslant n$.
- Funkcje c^n i c_{ni} są pierwotnie rekurencyjne.
- Funkcje $c^n(x_1, \ldots, x_n)$ ustanawiają wzajemnie jednoznaczne odpowiedniości między $\mathcal{N}^n$ oraz $\mathcal{N}$ (numerują ciągi liczb naturalnych długości n).
- Z jednoargumentowych funkcji częściowo rekurencyjnych oraz z funkcji $c^n(x_1, \ldots, x_n)$ otrzymać można wszystkie funkcje częściowo rekurencyjne.

Niektóre własności funkcji rekurencyjnych

Rozważmy następującą *funkcję Gödla*:

$$\beta(x, y, z) = \mathrm{rest}(x, 1 + y(z + 1)).$$

Można udowodnić, że dla dowolnego skończonego ciągu liczb naturalnych $a_0, \ldots a_n$ układ równań:

$$\begin{cases} \beta(x, y, 0) = a_0 \\ \ldots \\ \beta(x, y, n) = a_n \end{cases}$$

ma co najmniej jedno rozwiązanie x, y.

Funkcje kodujące zostaną wykorzystane w dowodach twierdzeń metalogicznych dotyczących Arytmetyki Peana.

Niektóre własności funkcji rekurencyjnych

Dla każdej liczby naturalnej n istnieją funkcje uniwersalne dla klas wszystkich n-argumentowych funkcji:

- pierwotnie rekurencyjnych;
- ogólnie rekurencyjnych.

Można udowodnić, że dla każdej liczby naturalnej n istnieje ogólnie rekurencyjna funkcja uniwersalna dla klasy wszystkich n-argumentowych funkcji pierwotnie rekurencyjnych.
W dowodzie tego twierdzenia wykorzystuje się możliwość kodowania liczb naturalnych.
W szczególności, wykazuje się, że zakodować można definiowanie przez schemat rekursji prostej oraz definiowanie przez złożenie.

Niektóre własności funkcji rekurencyjnych

Twierdzenie A. Nie istnieje funkcja pierwotnie rekurencyjna uniwersalna dla rodziny wszystkich n-argumentowych funkcji pierwotnie rekurencyjnych.

Twierdzenie B. Nie istnieje funkcja częściowo rekurencyjna uniwersalna dla rodziny wszystkich n-argumentowych funkcji ogólnie rekurencyjnych.

Dowód A. Niech $F(t, x_1, \ldots, x_n)$ będzie funkcją uniwersalną dla rodziny wszystkich n-argumentowych funkcji pierwotnie rekurencyjnych i przypuśćmy, że jest ona pierwotnie rekurencyjna. Wtedy $f(x_1, \ldots, x_n) = 1 + F(x_1, x_1, \ldots, x_n) = F(t_0, x_1, \ldots, x_n)$ dla pewnego t_0. Stąd $1 + F(t_0, t_0, \ldots, t_0) = F(t_0, t_0, \ldots, t_0)$. Dochodzimy do sprzeczności.

Dowód B. Zauważmy, że funkcja uniwersalna powinna być wszędzie określona, tzn. całkowita. Dalej, zobacz dowód Twierdzenia A.

Tak więc, choć można skonstruować funkcje uniwersalne dla rodziny wszystkich n-argumentowych funkcji:

- pierwotnie rekurencyjnych;
- ogólnie rekurencyjnych,

to z powyższych twierdzeń A i B otrzymujemy przykłady $n+1$-argumentowych funkcji, które **nie** są:

- pierwotnie rekurencyjne;
- ogólnie rekurencyjne.

Inna metoda pokazywania, iż jakaś funkcja nie należy do określonej klasy funkcji to dowód, że funkcja ta „rośnie szybciej" niż każda z funkcji tej klasy. W ten sposób pokazuje się np., że funkcja Ackermanna nie jest funkcją pierwotnie rekurencyjną.

Koniec

Można udowodnić, że:

- Dowolna funkcja częściowo rekurencyjna jest prawidłowo obliczalna w sensie Turinga.
- Dowolna funkcja obliczalna w sensie Turinga jest częściowo rekurencyjna.
- Istnieje funkcja pierwotnie rekurencyjna $S(z, x, y, w)$ taka, że:

$$S(z, x, y, w) = \begin{cases} 1, & \text{gdy } z = \lambda(T) \text{ i maszyna } T \text{ przetwarza} \\ & \text{słowo } q_1 01^x \text{ w słowo } q_0 01^y 0 \ldots 0 \\ & \text{w nie więcej niż } w \text{ krokach,} \\ 0, & \text{w pozostałych przypadkach.} \end{cases}$$

Koniec

Na dziś wystarczy, prawda?
Zabieraj zabawki i pędź cieszyć się Wiosną: