Teoria Programowania

Notatki

„Czyli tak właśnie wygląda bezrobocie."

Maciej Mikołajczak
Kraków, 2026/2027

Spis treści

1. Funkcje pierwotnie rekurencyjne **2**

2. Kodowanie ciągów **3**

3. Rekursja z historią i eliminacja rekursji **4**

4. Twierdzenie o formie normalnej rekursji **5**

5. Rachunek lambda **7**

6. Arytmetyka w rachunku lambda **8**

1. Funkcje pierwotnie rekurencyjne

2026-10-05

Przykład. Kilka trudnych (nierozstrzygalnych) problemów:

- Problem Stopu.
- Równoważność programów (M, N).
- Problem domina – mamy skończenie wiele typów kafelków, które możemy łączyć na ustalone sposoby; pytamy, czy da się pokryć całą przestrzeń.
- Dziesiąty problem Hilberta – instancją jest wielomian o współczynnikach całkowitych, pytamy, czy ma całkowite miejsce zerowe.
- Problem Posta – mamy słowa nad $\{a, b\}$, instancją jest $X = \{(w_1, v_1), \ldots, (w_n, v_n)\}$ dla $w_i, v_i \in \{a, b\}^*$, chcemy znaleźć skończony ciąg $\alpha_1, \ldots, \alpha_k$ taki, że $w_{\alpha_1} \cdots w_{\alpha_k} = v_{\alpha_1} \cdots v_{\alpha_k}$.
- Tautologiczność w rachunku predykatów.

Definicja 1. Funkcja $f : \mathbb{N}^{k+1} \to \mathbb{N}$ jest definiowana z $g : \mathbb{N}^k \to \mathbb{N}$ i $h : \mathbb{N}^{k+2} \to \mathbb{N}$ przez rekursję prostą, gdy $f(\overline{x}, 0) = g(\overline{x})$ i $f(\overline{x}, y+1) = h(\overline{x}, y, f(\overline{x}, y))$.

Definicja 2. Relacja $R \subseteq \mathbb{N}^{k+1}$ jest regularna, gdy $\forall_{\overline{x} \in \mathbb{N}^k}\ \exists_{y \in \mathbb{N}}\ R(\overline{x}, y)$.

Definicja 3. Funkcja $f : \mathbb{N}^k \to \mathbb{N}$ jest zdefiniowana przez μ-rekursję z regularnej relacji $R \subseteq \mathbb{N}^{k+1}$, gdy $f(\overline{x}) = \mu y.R(\overline{x}, y)$ (najmniejszy y spełniający relację).

Definicja 4. $g : \mathbb{N}^2 \to \mathbb{N}$ jest iteracją funkcji $f : \mathbb{N} \to \mathbb{N}$, gdy $g(x, 0) = x$ oraz $g(x, y+1) = f(g(x, y))$. Oznaczamy to $g(x, y) = f^{(y)}(x)$.

Definicja 5. Klasa $\mathcal{C}$ funkcji jest zamknięta na złożenia, jeżeli dla $g : \mathbb{N}^n \to \mathbb{N}$ i $f_1 : \mathbb{N}^k \to \mathbb{N}, \ldots, f_n : \mathbb{N}^k \to \mathbb{N}$ takich, że $g, f_1, \ldots, f_n \in \mathcal{C}$ funkcja $h(\overline{x}) = g(f_1(\overline{x}), \ldots, f_n(\overline{x}))$ również jest w $\mathcal{C}$.

Definicja 6. Klasa funkcji pierwotnie rekurencyjnych (primitive recursive), oznaczana PRec, to najmniejsza klasa zawierająca:

- $0^k(\overline{x}) = 0$ dla każdego $k \in \mathbb{N}$,
- $S : \mathbb{N} \ni x \mapsto x + 1 \in \mathbb{N}$,
- $I_i^n(x_1, \ldots, x_n) = x_i$ dla każdego $n \in \mathbb{N}_+$ i $i \leq n$

i zamknięta na rekursję prostą i złożenia.

Definicja 7. Klasa funkcji rekurencyjnych, oznaczana Rec, to najmniejsza klasa zawierająca funkcje wymienione wyżej i zamknięta na rekursję prostą, złożenia i μ-rekursję.

Przykład (Funkcja Ackermanna)**.**

$$A(0, n) = n + 1,$$
$$A(m, 0) = A(m - 1, 1),$$
$$A(m, n) = A(m - 1, A(m, n - 1)).$$

Taka funkcja jest rekurencyjna, ale nie pierwotnie rekurencyjna. Rośnie ona szybciej niż dowolna wieża.

Przykład. Pierwotnie rekurencyjne są następujące funkcje:

- funkcje stałe;
- suma $D(x, 0) = x$, $D(x, y+1) = S(D(x, y))$;

- produkt $M(x,0)=0$, $M(x,y+1)=D(M(x,y),x)$;
- potęga $P(x,0)=1$, $P(x,y+1)=M(P(x,y),x)$;
- monus $x \dot{-} 0 = x$, $x \dot{-} (y+1) = (x \dot{-} y) \dot{-} 1$;
- signum $\operatorname{sg}(x) = \begin{cases} 0, & x=0 \\ 1, & x>0 \end{cases}$;
- $\overline{\operatorname{sg}}(x) = \begin{cases} 1, & x=0 \\ 0, & x>0 \end{cases}$.

Lemat 8. Niech $f : \mathbb{N}^{k+1} \to \mathbb{N}$ będzie w PRec. Wtedy $g(\overline{x},z) = \sum_{y<z} f(\overline{x},y)$ oraz $h(\overline{x},z) = \prod_{y<z} f(\overline{x},y)$ też są w PRec.

Dowód. $g(\overline{x},0)=0$, $g(\overline{x},z+1) = g(\overline{x},z) + f(\overline{x},z)$.

$h(\overline{x},0)=1$, $h(\overline{x},z+1) = h(\overline{x},z) \cdot f(\overline{x},z)$. □

Definicja 9. $R \subseteq \mathbb{N}^k$ jest predykatem w klasie PRec gdy funkcja charakterystyczna tego predykatu $f_R : \mathbb{N}^k \ni x \mapsto \begin{cases} 1, & \overline{x} \in R \\ 0, & x \notin R \end{cases}$ jest w klasie PRec.

Lemat 10. Klasa predykatów PRec jest zamknięta na operacje logiczne i na kwantyfikatory ograniczone.

Dowód. Dla predykatów R, S w PRec mamy $f_{R\wedge S} = f_R \cdot f_S$ oraz $f_{\neg R} = 1 \dot{-} f_R$, co daje zamknięcie na operacje logiczne, bo $\{\wedge, \neg\}$ buduje wszystkie operacje.

Predykat $\forall_{y<z}\ R(\overline{x},y)$ ma funkcję charakterystyczną $\prod_{y<z} f_R(\overline{x},y)$. Kwantyfikator egzystencjalny również nie wyprowadza nas z PRec wobec $\exists_{y<z}\ R(\overline{x},y) \equiv \neg(\forall_{y<z}\ \neg R(\overline{x},y))$. □

Definicja 11. Funkcja $f : \mathbb{N}^k \to \mathbb{N}$ jest zdefiniowana przez ograniczony μ-operator z relacji $R \subseteq \mathbb{N}^{k+1}$, gdy

$$f(\overline{x}) = \mu y_{\leq z}.R(\overline{x},y) = \begin{cases} \mu y.R(\overline{x},y), & \exists_{y\leq z}\ R(\overline{x},y) \\ 0, & \text{wpp} \end{cases}.$$

Lemat 12. PRec jest zamknięta na ograniczony μ-operator.

2. Kodowanie ciągów

2026-10-12

Uwaga. Relacja $x < y$ na liczbach naturalnych jest pierwotnie rekurencyjna, o czym świadczy funkcja $\operatorname{sg}(y \dot{-} x)$. Za jej pomocą możemy zdefiniować relacje $=$, $\leq$ i podobne.

Definicja 13. Funkcja Cantora $J(x,y) = \sum_{i<x+y}(i+1)+x = \frac{(x+y)(x+y+1)}{2}+x$ jest bijekcją między $\mathbb{N}^2$ a $\mathbb{N}$ (enumerujemy punkty na kracie idąc po kolejnych przekątnych od dołu w górę). J jest pierwotnie rekurencyjna. Odpowiednie funkcje odwrotne również są pierwotnie rekurencyjne i zadane poprzez $L(z) = \mu x_{\leq z}.[\exists_{y\leq z}\ J(x,y)=z]$ i $R(z) = \mu y_{\leq z}.[\exists_{x\leq z}\ J(x,y)=z]$.

Definicja 14. Możemy kodować ciągi z $\mathbb{N}^k$ w $\mathbb{N}$ poprzez funkcje J^k zadane przez

- $J^1(x) = x$,
- $J^2(x,y) = J(x,y)$,
- $J^{k+1}(x_1,\ldots,x_{k+1}) = J\big(x_1, J^k(x_2,\ldots,x_{k+1})\big)$.

Uwaga. J^k to wielomian stopnia 2^{k-1}.

Funkcja $I_k^p(z) = L\big(R^{(p-1)}(z)\big)$ odzyskuje x_p z kodowania J^k.

Definicja 15. Definiujemy kodowanie ciągu dowolnej długości $\langle x_0, \ldots, x_n\rangle = J^{n+2}(n, x_0, \ldots, x_n)$.

Odkodowania dokonuje tak zwana funkcja β Gödla, która ma typ $\beta : \mathbb{N}^2 \to \mathbb{N}$ i $\beta(a, i) = a_i$, gdy a jest kodem $a_0, \ldots, a_n$ oraz $i \leq n$. Łatwo zobaczyć, że jest ona pierwotnie rekurencyjna.

Jeśli a jest zakodowanym ciągiem elementów $a_0, \ldots, a_n$, to przez $\langle a\rangle_i$ oznaczamy wartość a_i dla $i \leq n$.

Twierdzenie 16. Klasa PRec jest najmniejszą klasą zawierającą funkcje inicjujące (takie jak w definicji PRec) oraz kodowanie i dekodowanie par, zamkniętą na złożenie i iterację (czyli aplikowanie danej funkcji podaną na wejściu liczbę razy).

Dowód. Niech $\mathcal{C}$ będzie tą klasą. Oczywiście $\mathcal{C} \subseteq$ PRec. Ustalmy $f \in$ PRec. Stosując indukcję strukturalną po budowie f pokażemy, że $f \in \mathcal{C}$. Wystarczy pokazać, że jeśli f jest zdefiniowana przez rekursję prostą z funkcji $g, h \in \mathcal{C}$, to $f \in \mathcal{C}$.

Zdefiniujemy funkcję $S(\overline{x}, n) = \langle \overline{x}, n, f(\overline{x}, n)\rangle$ oznaczającą stan obliczeń po n krokach. Będzie nam w tym celu potrzebny iterator $t(S(\overline{x}, n)) = S(\overline{x}, n+1)$. Definiujemy go w następujący sposób

$$t(r) = \big\langle \langle r\rangle_0, \ldots, \langle r\rangle_{k-1}, \langle r\rangle_k + 1, h\big(\langle r\rangle_0, \ldots, \langle r\rangle_{k-1}, \langle r\rangle_k, \langle r\rangle_{k+1}\big)\big\rangle,$$

gdzie k jest długością $\overline{x}$. Widać, że $t \in \mathcal{C}$. Teraz $S(\overline{x}, n) = t^{(n)}(S(\overline{x}, 0))$ i $S(\overline{x}, 0) = \langle \overline{x}, 0, g(\overline{x})\rangle$. Zatem $S \in \mathcal{C}$. Wobec $f(\overline{x}, n) = \langle S(\overline{x}, n)\rangle_{k+1}$ mamy $f \in \mathcal{C}$. □

3. Rekursja z historią i eliminacja rekursji

2026-10-19

Twierdzenie 17 (Skolem 1923). Klasa PRec jest zamknięta na operację rekursji z historią, to znaczy dla $g : \mathbb{N}^k \to \mathbb{N}$ i $h : \mathbb{N}^{k+2} \to \mathbb{N}$ pierwotnie rekurencyjnych funkcja $f : \mathbb{N}^{k+1} \to \mathbb{N}$ zadana przez

$$\begin{cases} f(\overline{x}, 0) = g(\overline{x}), \\ f(\overline{x}, y+1) = h(\overline{x}, y, \langle f(\overline{x}, 0), \ldots, f(\overline{x}, y)\rangle) \end{cases}$$

jest pierwotnie rekurencyjna.

Dowód. Definiujemy indukcyjnie funkcję $\widehat{f}(\overline{x}, y) = \langle f(\overline{x}, 0), \ldots, f(\overline{x}, y)\rangle$. f odzyskujemy biorąc ostatni element listy. □

Twierdzenie 18 (O eliminacji rekursji prostej, Gödel 1931, Kleene 1936). Klasa Rec to najmniejsza klasa zawierająca dodawanie, mnożenie, rzutowania I_k^i oraz funkcję charakterystyczną równości $\sigma(x, y) = [x = y]$ oraz zamknięta na złożenia i μ-rekursję.

Dowód. Oznaczmy rozważaną klasę przez $\mathcal{C}$. Oczywiście $\mathcal{C} \subseteq$ Rec. Teraz pokażemy Rec $\subseteq \mathcal{C}$. Możemy zbudować funkcję stale równą 0 poprzez $0^n(\overline{x}) = \mu y.\big(I_{n+1}^{n+1}(\overline{x}, y) = 0\big)$. Podobnie mamy $1^n(\overline{x}) = \mu y.(\sigma(0, y) = 0)$. Następnik to $s(x) = I_1^1(x) + 1^1(x)$. Brakuje nam rekursji prostej.

Załóżmy, że $f(\overline{x}, 0) = g(\overline{x})$, $f(\overline{x}, y+1) = h(\overline{x}, y, f(\overline{x}, y))$ i $h, g \in \mathcal{C}$. Pokażemy, że $f \in \mathcal{C}$. W tym celu pokażemy, że funkcja $t(\overline{x}, y)$, będąca historią obliczenia f ze względu na y, jest w $\mathcal{C}$. Wtedy $f(\overline{x}, y) = \beta(t(\overline{x}, y), y)$ i będzie wystarczyło pokazać, że $\beta \in \mathcal{C}$. Zachodzi

$$t(\overline{x}, y) = \mu z.[\beta(z, 0) = g(\overline{x}) \wedge \forall_{i<y}\ (\beta(z, i+1) = h(\overline{x}, i, \beta(z, i)))].$$

μ-rekursja znajduje z będące odpowiednim kodowaniem historii. Musimy pokazać, że β, koniunkcja i kwantyfikator ograniczony są w $\mathcal{C}$. Do tego trzeba pokazać, że możemy stosować μ-rekursję do predykatów (a nie tylko, jak w definicji, przyrównywać funkcje do 0).

Okazuje się, że $\mathcal{C}$ jest zamknięta na operacje logiczne, bo negacja to $\sigma(x, 0)$ a koniunkcja to mnożenie.

Do tego dla predykatu R takiego, że jego funkcja charakterystyczna $f_R \in \mathcal{C}$ mamy $f_{\neg R} \in \mathcal{C}$, więc $\mu y.R(\overline{x}, y) = \mu y.(f_{\neg R}(\overline{x}, y) = 0)$.

Teraz pora na kwantyfikator ograniczony. Zdefiniujmy operator μ^* ograniczony poprzez

$$\mu^* y_{<z}.R(\overline{x}, y) = \begin{cases} \mu y.R(\overline{x}, y), & \exists_{y<z}\ R(\overline{x}, y) \\ z, & \text{wpp} \end{cases}.$$

Jest $\mu^* y_{<z}.R(\overline{x}, y) = \mu y.[R(\overline{x}, y) \vee y = z]$. Teraz mamy $\exists_{y<z}\ R(\overline{x}, y) \equiv [\mu^* y_{<z}.R(\overline{x}, y)] \neq z$ i budujemy $\forall_{y<z}\ R(\overline{x}, y) \equiv \neg(\exists_{y<z}\ \neg R(\overline{x}, y))$.

Przypomnijmy, że $J(x, y) = \frac{(x+y)^2+3x+y}{2}$. Mamy $J \in \mathcal{C}$, bo dzielenie przez 2 definiujemy przez $\left\lfloor \frac{z}{2} \right\rfloor = \mu y.[2y = z \vee 2y + 1 = z]$ a pozostałe operacje to mnożenie i dodawanie.

Mamy $R(z) = \mu y.[\exists_{x<z+1}\ J(x, y) = z]$ i $L(z) = \mu x.[\exists_{y<z+1}\ J(x, y) = z]$.

Zdefiniujmy funkcję $\rho(b, c, i) = b \pmod{1 + (i + 1)c}$ przez $\mu r.[\exists_{q<b+1}\ b = q(1 + (i + 1)c) + r]$. Dla dowolnego ciągu liczb naturalnych $s_0, \ldots, s_m$ istnieją $b, c \in \mathbb{N}$ takie, że $\rho(b, c, j) = s_j$. Istotnie, jeśli weźmiemy c większe od wszystkich s_j i podzielne przez $m!$, to liczby $c + 1, 2c + 1, \ldots, (m + 1)c + 1$ są parami względnie pierwsze (bo różnica dwóch takich to jc dla $j \leq m$, więc jej dzielniki pierwsze nie dzielą żadnego $jc + 1$). Wtedy odpowiednie b istnieje z chińskiego twierdzenia o resztach.

Definiujemy

$$W(a, i) = \mu w.[\rho(L(w), R(w), 0) = a \wedge \forall_{j<i}\ \rho(L(w), R(w), j + 1) = R(\rho(L(w), R(w), j))].$$

$w = W(a, i)$ jest parą $w = J(b, c)$ taką, że $\rho(b, c, \cdot)$ koduje ciąg $a, R(a), \ldots, R^{(i)}(a)$ i istnieje z powyższych rozważań. Określając $T(a, i) = \rho(L(W(a, i)), R(W(a, i)), i)$ mamy $T(a, i) = R^{(i)}(a)$. Wobec $\beta(a, i) = L\big(R^{i+1}(a)\big) = a_i$ widzimy, że $\beta \in \mathcal{C}$. □

Definicja 19. Wprowadzamy numerację funkcji z klasy Rec. Robimy to w następujący sposób.

- $\langle 0, n\rangle$ jest numerem 0^n,
- $\langle 1\rangle$ jest numerem następnika,
- $\langle 2, n, i\rangle$ jest numerem I^i_n,
- $\langle 3, b_1, \ldots, b_n, a\rangle$ jest numerem funkcji $f(\overline{x}) = g(h_1(\overline{x}), \ldots, h_n(\overline{x}))$ gdzie b_i to numer h_i a a to numer g,
- $\langle 4, a, b\rangle$ to numer funkcji powstałej z rekursji prostej o funkcji bazowej o numerze a i funkcji kroku o numerze b,
- $\langle 5, a\rangle$ to numer funkcji powstałej z μ rekursji zastopowanej do funkcji o numerze g.

Uwaga. Numeracja funkcji nie jest surjektywna.

Definicja 20. Jeśli $f \in$ Rec, to przez e_f oznaczamy najmniejszą liczbę będącą numerem schematu obliczeń dla f.

4. Twierdzenie o formie normalnej rekursji

2026-10-26

Definicja 21. Zdefiniujemy drzewa obliczeń dla funkcji rekurencyjnych. Wierzchołkiem drzewa obliczeń jest funkcja $f \in$ Rec z argumentami $x_1, \ldots, x_n$ i wartością z. Inaczej mówiąc, wierzchołek $\langle e_f, \langle x_1, \ldots, x_n\rangle, z\rangle$ koduje informację $f(x_1, \ldots, x_n) = z$.

Jeśli $f(\overline{x}) = g(h_1(\overline{x}), \ldots, h_n(\overline{x}))$, to w korzeniu mamy wierzchołek $f(\overline{x}) = z$, a jego dzieci to drzewa kolejnych $h_i(\overline{x}) = z_i$ oraz drzewo dla $g(z_1, \ldots, z_k) = z$.

Jeśli f jest zdefiniowane przez rekursję z warunku bazowego g i kroku h, to drzewo $f(\overline{x}, 0)$ to korzeń $f(\overline{x}, 0) = z$ z doczepionym drzewem $g(\overline{x}) = z$, natomiast drzewo $f(\overline{x}, y + 1)$ to korzeń $f(\overline{x}, y + 1) = z$

z doczepionymi drzewami dla $f(\overline{x}, y) = z_1$ i $h(\overline{x}, y, z_1) = z$.

Drzewo dla μ-rekursji $f(\overline{x}) = \mu y.(g(\overline{x}, y) = 0)$ ma korzeń $f(\overline{x}) = z$ z doczepionymi drzewami $g(\overline{x}, i) = t_i$ dla $i = 0, \ldots, z-1$ (gdzie $t_i \neq 0$) oraz $g(\overline{x}, z) = 0$.

Drzewem dla funkcji inicjujących klasę Rec są odpowiednie liście: $f(\overline{x}) = 0$, $S(x) = x + 1$ lub $I_n^i(\overline{x}) = x_i$.

Numerem wierzchołka $v := f(x_1, \ldots, x_n) = z$ jest liczba $\widehat{v} := \langle e_f, \langle x_1, \ldots, x_n\rangle, z\rangle$. Numerem drzewa T o korzeniu v i poddrzewach $T_1, \ldots, T_n$ jest $\widehat{T} = \left\langle \widehat{v}, \widehat{T_1}, \ldots, \widehat{T_n} \right\rangle$.

Twierdzenie 22 (O formie normalnej rekursji, Kleene 1936)**.** Istnieje funkcja pierwotnie rekurencyjna $U : \mathbb{N} \to \mathbb{N}$ oraz dla każdego $n \in \mathbb{N}$ istnieje pierwotnie rekurencyjny predykat $\tau^n \subseteq \mathbb{N}^{n+2}$ takie, że dla każdej funkcji rekurencyjnej $f : \mathbb{N}^n \to \mathbb{N}$ zachodzi

1. $\forall_{x_1,\ldots,x_n}\ \exists_y\ \tau^n(e_f, x_1, \ldots, x_n, y)$,
2. $f(x_1, \ldots, x_n) = U(\mu y.\tau^n(e_f, x_1, \ldots, x_n, y))$.

y ma w tej wypowiedzi sens jako historia obliczenia funkcji f na argumentach $x_1, \ldots, x_n$.

Dowód. Zaczniemy od konstrukcji predykatu $T(y)$, który oznacza, że y jest kodem pewnego drzewa obliczeń. Zauważmy, że jeśli $y = \left\langle \widehat{v}, \widehat{T_1}, \ldots, \widehat{T_n} \right\rangle$, to:

- $\langle y\rangle_1 = \widehat{v} = \langle e_f, \langle x_1, \ldots, x_n\rangle, z\rangle$;
- $\langle y\rangle_{1,1}$ jest numerem funkcji f;
- $\langle y\rangle_{1,2}$ jest numerem zestawu argumentów;
- $\langle y\rangle_{1,3}$ jest wartością $f(x_1, \ldots, x_n)$;
- $\langle y\rangle_{i+1} = \widehat{T_i}$.

Wprowadzamy predykat $A(y)$ oznaczający wstępne warunki na to, że y jest kodem pewnego obliczenia:

$$A(y) \equiv \mathrm{Seg}(y) \wedge \mathrm{Seg}(\langle y\rangle_1) \wedge \mathrm{len}(\langle y\rangle_1) = 3 \wedge \mathrm{Seg}\Big(\langle y\rangle_{1,1}\Big) \wedge \mathrm{Seg}\Big(\langle y\rangle_{1,2}\Big),$$

gdzie $\mathrm{Seg}(y)$ oznacza, że y jest kodowaniem pewnego ciągu a $\mathrm{len}(y)$ jest długością tego ciągu.

Teraz zdefiniujemy predykaty:

- $B(y)$ – y jest kodem liścia dla funkcji inicjującej;
- $C(y)$ – y jest drzewem dla złożenia;
- $D(y)$ – y jest drzewem dla rekursji prostej;
- $E(y)$ – y jest drzewem dla μ-rekursji.

$$\begin{aligned} B(y) \equiv \mathrm{len}(y) = 1 \wedge \Big[&\Big(\mathrm{len}\Big(\langle y\rangle_{1,1}\Big) = 2 \wedge \langle y\rangle_{1,1,1} = 0 \wedge \mathrm{len}\Big(\langle y\rangle_{1,2}\Big) = \langle y\rangle_{1,1,2} \wedge \langle y\rangle_{1,3} = 0\Big) \vee \\ &\Big(\langle y\rangle_{1,1} = \langle 1\rangle \wedge \mathrm{len}\Big(\langle y\rangle_{1,2}\Big) = 1 \wedge \langle y\rangle_{1,3} = \langle y\rangle_{1,2,1} + 1\Big) \vee \\ &\Big(\mathrm{len}\Big(\langle y\rangle_{1,1}\Big) = 3 \wedge \langle y\rangle_{1,1,1} = 2 \wedge \mathrm{len}\Big(\langle y\rangle_{1,2}\Big) = \langle y\rangle_{1,1,2} \wedge \\ &1 \leq \langle y\rangle_{1,1,3} \leq \langle y\rangle_{1,1,2} \wedge \langle y\rangle_{1,3} = \langle y\rangle_{1,2,\langle y\rangle_{1,1,3}}\Big)\Big]. \end{aligned}$$

Pozostałe robimy podobnie. Teraz mamy

$$T(y) \equiv A(y) \wedge [B(y) \vee C(y) \vee D(y) \vee E(y)] \wedge \big[\mathrm{len}(y) > 1 \implies \forall_{2 \leq i \leq \mathrm{len}(y)}\ T(\langle y\rangle_i)\big].$$

Możemy zdefiniować $\tau^n(e_f, x_1, \ldots, x_n, y) \equiv T(y) \wedge \langle y\rangle_{1,1} = e_f \wedge \langle y\rangle_{1,2} = \langle x_1, \ldots, x_n\rangle$ oraz $U(y) = \langle y\rangle_{1,3}$. □

5. Rachunek lambda

2026-11-9

Definicja 23. Termami w rachunku lambda są:

1. zmienne z ustalonego przeliczalnego zbioru C;
2. operacje abstrakcji: jeśli M jest termem a x zmienną, to $\lambda x.M$ jest termem;
3. aplikacje: jeśli M, N to termy, to (MN) jest termem;
4. nic innego nie jest termem.

Notacja. Piszemy $\lambda x_1 x_2 \dots x_n.M$ mając na myśli $\lambda x_1.(\lambda x_2.(\dots(\lambda x_n.M)\dots))$.

Piszemy $MN_1N_2\dots N_k$ mając na myśli $(\dots((MN_1)N_2)\dots N_k)$.

Definicja 24. Zmienne wolne termu M, oznaczane $\mathrm{FV}(M)$, definiujemy następująco:

- $\mathrm{FV}(x) = \{x\}$;
- $\mathrm{FV}(MN) = \mathrm{FV}(M) \cup \mathrm{FV}(N)$;
- $\mathrm{FV}(\lambda x.M) = \mathrm{FV}(M) \setminus \{x\}$.

Term domknięty to taki, który nie ma zmiennych wolnych.

Definicja 25. Niech M, N będą termami a x zmienną. $M[x := N]$ definiujemy jako podstawienie w M za x termu N, to znaczy, że dla zmiennej $y \neq x$ i termów P, Q jest:

- $x[x := N] = N$;
- $y[x := N] = y$;
- $(PQ)[x := N] = P[x := N]Q[x := N]$;
- $(\lambda x.P)[x := N] = \lambda x.P$;
- $(\lambda y.P)[x := N] = \lambda y.P[x := N]$ jeśli $y \notin \mathrm{FV}(N)$ lub $x \notin \mathrm{FV}(P)$;
- $(\lambda y.P)[x := N] = \lambda z.P[y := z][x := N]$ jeśli $y \in \mathrm{FV}(N)$ i $x \in \mathrm{FV}(P)$ a z jest wybrane jako zmienna nie należąca do $\mathrm{FV}(P) \cup \mathrm{FV}(N)$.

Definicja 26. α-konwersją (α-równoważnością) nazywamy najmniejszą zwrotną, przechodnią i symetryczną relację dwuargumentową $=_\alpha$ na termach taką, że $\lambda x.P =_\alpha \lambda y.P[x := y]$ dla każdej zmiennej $y \notin \mathrm{FV}(P)$ oraz jeśli $P =_\alpha P'$, to:

- dla każdej zmiennej x jest $\lambda x.P =_\alpha \lambda x.P'$;
- dla każdego termu Z jest $PZ =_\alpha P'Z$ oraz $ZP =_\alpha ZP'$.

Definicja 27. β-konwersją (β-redukcją) nazywamy najmniejszą relację dwuargumentową $\to_\beta$ na termach taką, że $(\lambda x.P)Q \to_\beta P[x := Q]$ oraz jeśli $P \to_\beta P'$, to:

- dla każdej zmiennej x jest $\lambda x.P \to_\beta \lambda x.P'$;
- dla każdego termu Z jest $PZ \to_\beta P'Z$ oraz $ZP \to_\beta ZP'$.

Term po lewej stronie $\to_\beta$ nazywamy β-redeksem, a po prawej β-kontraktum. Term M jest w β-formie normalnej, jeśli nie ma termu N takiego, że $M \to_\beta N$. Często pomijamy przedrostek β- stosując to nazewnictwo.

$\twoheadrightarrow_\beta$ oznacza zwrotne i przechodnie domknięcie $=_\alpha \cup \to_\beta$.

$=_\beta$ oznacza zwrotne, przechodnie i symetryczne domknięcie $=_\alpha \cup \to_\beta$.

Twierdzenie 28 (Church, Rosser 1936)**.** Dla dowolnych termów M, N, K takich, że $M \twoheadrightarrow_\beta N$ i $M \twoheadrightarrow_\beta K$ istnieje term L taki, że $N \twoheadrightarrow_\beta L$ i $K \twoheadrightarrow_\beta L$.

Dowód. Przez indukcję po M, N, K. □

Twierdzenie 29. Dla dowolnych termów M, N takich, że $M =_\beta N$ istnieje term L taki, że $M \twoheadrightarrow_\beta L$ i $N \twoheadrightarrow_\beta L$.

Dowód. Jeśli $M =_\beta N$, to istnieją termy $K_1, \ldots, K_n$ takie, że $M \twoheadrightarrow_\beta K_1$, $K_2 \twoheadrightarrow_\beta K_1$, $K_2 \twoheadrightarrow_\beta K_3$ i tak dalej aż do N. Przeprowadzimy indukcję po n. Dla $n = 0$ mamy $M \twoheadrightarrow_\beta N$, dla $n = 1$ szukanym termem jest K_1. Dalej zakładamy $n \geq 2$. Z twierdzenia Churcha-Rossera istnieje $K_{1,2,3}$ takie, że $K_1, K_3 \twoheadrightarrow_\beta K_{1,2,3}$ (przy czym może być $K_3 = N$). Mamy $M \twoheadrightarrow_\beta K_{1,2,3}$ i $K_3 \twoheadrightarrow_\beta K_{1,2,3}$. To daje nam krótszy ciąg termów, dla którego mamy tezę. □

Twierdzenie 30 (O jedyności formy normalnej)**.** Niech M, N, K będą termami. Jeśli $M \twoheadrightarrow_\beta K$ i $M \twoheadrightarrow_\beta N$ oraz oba N, K są w formie normalnej, to $N =_\alpha K$.

Dowód. Z twierdzenia Churcha-Rossera istnieje L takie, że $N, K \twoheadrightarrow_\beta L$. Z normalności N, K odpowiedni ciąg przejść składa się tylko z α-konwersji, więc $N =_\alpha K$. □

Przykład. Możemy zdefiniować identyczność $I := \lambda x.x$ oraz rzutowania $\pi_1 := \lambda xy.x$ i $\pi_2 := \lambda xy.y$. Wtedy parę uporządkowaną termów P, Q można zdefiniować aplikując je do termu $\lambda yzx.xyz$, co daje nam $[P, Q] := \lambda x.xPQ$, dla którego zachodzi $[P, Q]\pi_1 \twoheadrightarrow_\beta P$ i $[P, Q]\pi_2 \twoheadrightarrow_\beta Q$.

Przykład. Niech $\omega = \lambda x.xx$. Term $\omega\omega$ nie ma formy normalnej. Rozpisując β-redukcje mamy $\omega\omega =_\beta (\lambda x.xx)\omega =_\beta \omega\omega$.

Twierdzenie 31 (Kleene; o operatorze punktu stałego)**.** Istnieje term domknięty Y zwany operatorem punktu stałego taki, że dla każdego termu M zachodzi $M(YM) =_\beta YM$.

Dowód. Definiujemy $Y = \lambda z.(\lambda x.z(xx))(\lambda x.z(xx))$. Mamy

$$YM =_\beta (\lambda x.M(xx))(\lambda x.M(xx)) =_\beta M((\lambda x.M(xx))(\lambda x.M(xx))) =_\beta M(YM).$$

□

6. Arytmetyka w rachunku lambda

2026-11-16

Definicja 32. Niech n będzie liczbą naturalną. n-ty numeral Churcha to $\underline{n} = \lambda sz.S(\ldots(SZ)\ldots)$, gdzie S jest iterowane n razy (czyli $\underline{0} = \lambda sz.z$). Tak zdefiniowane numerale są w postaci normalnej, z czego wynika, że są parami β-nierównoważne.

Definicja 33. Funkcja $f : \mathbb{N}^k \to \mathbb{N}$ jest reprezentowana przez domknięty λ term F, gdy dla dowolnych $n_1, \ldots, n_k \in \mathbb{N}$ zachodzi $F\underline{n_1}\ldots\underline{n_k} =_\beta \underline{f(n_1, \ldots, n_k)}$.

Przykład. Następnik definiujemy jako $S = \lambda nsx.s(nsx)$. Dodawanie jako $D = \lambda nmsx.ns(msx)$. Mnożenie jako $M = \lambda nmsx.n(\lambda z.msz)x$.

Instrukcję warunkową if $n = 0$ then k else m można opisać jako IF $= \lambda nkmsx.n(\lambda z.msx)(ksx)$.

Przykład. Zdefiniujemy teraz funkcję monus 1 (czyli poprzednik). Idea jest taka: definiujemy funkcję $t(x, y) = (x + 1, x)$. Wtedy $t^{(n)}(0, 0) = (n, n - 1)$. Mamy termy opisujące parę $[\cdot, \cdot]$ i rzutowania π_1, π_2, więc możemy zdefiniować poprzednik jako

$$M_1 = \lambda n.((n(\lambda z.[D\underline{1}(z\pi_1), z\pi_1]))[\underline{0}, \underline{0}])\pi_2.$$

Twierdzenie 34 (Church 1935)**.** Funkcje rekurencyjne mają swoje reprezentacje w λ-rachunku.

Dowód. Mamy $0^n = \lambda x_1 \ldots x_n.\underline{0}$. Następnik już mamy, $S = \lambda nsx.s(nsx)$. Rzutowanie to $I_i^n = \lambda x_1 \ldots x_n.x_i$.

Jeśli $h(x_1, \ldots, x_k) = g(f_1(x_1, \ldots, x_k), \ldots, f_n(x_1, \ldots, x_k))$, to mając termy $G, F_1, \ldots, F_n$ reprezentujące odpowiednie funkcje możemy zdefiniować $H = \lambda x_1 \ldots x_k.G(F_1 x_1 \ldots x_k) \ldots (F_n x_1 \ldots x_k)$.

Jeśli $f(x_1, \ldots, x_k, 0) = g(x_1, \ldots, x_k)$ i $f(x_1, \ldots, x_k, y+1) = h(x_1, \ldots, x_k, y, f(x_1, \ldots, x_k, y))$ oraz mamy odpowiednie termy G, H, to zaczynamy od budowy termu

$$M = \lambda f x_1 \ldots x_k y.\mathrm{IF} y (G x_1 \ldots x_k)(H x_1 \ldots x_k (M_1 y)(f x_1 \ldots x_k (M_1 y)))$$

Reprezentacją f jest YM. Dowodzimy tego korzystając z $YM = M(YM)$ i indukując się po y.

Jeśli $f(x_1, \ldots, x_k) = \mu y.[g(x_1, \ldots, x_k, y) = 0]$ oraz mamy term G, to definiujemy

$$M = \lambda f y x_1 \ldots x_k.\mathrm{IF}(G x_1 \ldots x_k y) y (f(Sy) x_1 \ldots x_k).$$

Reprezentacją f jest $YM\underline{0}$. □