Rachunek lambda

11 marca 2013

Zbiory

Sposób użycia:
$a \in A$ (należenie)

Zbiory

Sposób użycia:
$a \in A$ (należenie)

Tworzenie:
$\{x \mid W(x)\}$ (wycinanie)

Zbiory

Sposób użycia:
$a \in A$ (należenie)

Tworzenie:
$\{x \mid W(x)\}$ (wycinanie)

Ewaluacja:
$$a \in \{x \mid W(x)\} \quad \Leftrightarrow \quad W(a)$$

Zbiory i funkcje

Sposób użycia:

$a \in A$ (należenie) $F(a)$ (aplikacja)

Tworzenie:

$\{x \mid W(x)\}$ (wycinanie)

Ewaluacja:

$a \in \{x \mid W(x)\} \quad \Leftrightarrow \quad W(a)$

Zbiory i funkcje

Sposób użycia:

$a \in A$	(należenie)	$F(a)$	(aplikacja)

Tworzenie:

$\{x \mid W(x)\}$	(wycinanie)	$\lambda x\, W(x)$	(abstrakcja)

Ewaluacja:

$$a \in \{x \mid W(x)\} \quad \Leftrightarrow \quad W(a)$$

Zbiory i funkcje

Sposób użycia:

$a \in A$	(należenie)	$F(a)$	(aplikacja)

Tworzenie:

$\{x \mid W(x)\}$	(wycinanie)	$\lambda x\, W(x)$	(abstrakcja)

Ewaluacja:

$a \in \{x \mid W(x)\} \quad \Leftrightarrow \quad W(a)$ $\qquad (\lambda x\, W(x))(a) = W(a)$

Ekstensjonalność (?)

Dla zbiorów (niewątpliwa):

$A = B$ wtedy i tylko wtedy, gdy $\forall x\,(x \in A \Leftrightarrow x \in B)$

[1] Gdy F nie zawiera x.

Ekstensjonalność (?)

Dla zbiorów (niewątpliwa):

$A = B$ wtedy i tylko wtedy, gdy $\forall x\, (x \in A \Leftrightarrow x \in B)$

$$A = \{x \mid x \in A\}$$

[1] Gdy F nie zawiera x.

Ekstensjonalność (?)

Dla zbiorów (niewątpliwa):

$A = B$ wtedy i tylko wtedy, gdy $\forall x\, (x \in A \Leftrightarrow x \in B)$

$$A = \{x \mid x \in A\}$$

Dla funkcji (wątpliwa):

$F = G$ wtedy i tylko wtedy, gdy $\forall x\, (F(x) = G(x))$

---

[1] Gdy F nie zawiera x.

Ekstensjonalność (?)

Dla zbiorów (niewątpliwa):

$A = B$ wtedy i tylko wtedy, gdy $\forall x\,(x \in A \Leftrightarrow x \in B)$

$$A = \{x \mid x \in A\}$$

Dla funkcji (wątpliwa):

$F = G$ wtedy i tylko wtedy, gdy $\forall x\,(F(x) = G(x))$

$$F = \lambda x\, Fx \quad ([1])$$

[1] Gdy F nie zawiera x.

Ekstensjonalność (?)

Statyczne i dynamiczne rozumienie funkcji:

Ekstensjonalność (?)

Statyczne i dynamiczne rozumienie funkcji:

1. Jako przyporządkowanie, wykres, relację, zbiór par...

Ekstensjonalność (?)

Statyczne i dynamiczne rozumienie funkcji:

1. Jako przyporządkowanie, wykres, relację, zbiór par...
2. Jako regułę, przekształcenie, algorytm, metodę...

Ekstensjonalność (?)

Statyczne i dynamiczne rozumienie funkcji:

1. Jako przyporządkowanie, wykres, relację, zbiór par...
2. Jako regułę, przekształcenie, algorytm, metodę...

Teoria zbiorów nie nadaje się do opisu aspektu (2).

Beztypowy rachunek lambda

- Funkcja rozumiana jako działanie.

Beztypowy rachunek lambda

- Funkcja rozumiana jako działanie.
- Przedmiotem działania może być cokolwiek, zatem...

Beztypowy rachunek lambda

- Funkcja rozumiana jako działanie.
- Przedmiotem działania może być cokolwiek, zatem...
- ...funkcja nie ma a priori ograniczonej dziedziny.

Beztypowy rachunek lambda

- Funkcja rozumiana jako działanie.
- Przedmiotem działania może być cokolwiek, zatem...
- ...funkcja nie ma a priori ograniczonej dziedziny.
- Funkcja może być np. aplikowana sama do siebie.

Beztypowy rachunek lambda

- Funkcja rozumiana jako działanie.
- Przedmiotem działania może być cokolwiek, zatem...
- ...funkcja nie ma a priori ograniczonej dziedziny.
- Funkcja może być np. aplikowana sama do siebie.
- Każdemu obiektowi można przypisać działanie, więc...

Beztypowy rachunek lambda

- Funkcja rozumiana jako działanie.
- Przedmiotem działania może być cokolwiek, zatem...
- ...funkcja nie ma a priori ograniczonej dziedziny.
- Funkcja może być np. aplikowana sama do siebie.
- Każdemu obiektowi można przypisać działanie, więc...
- ...nie ma innych obiektów niż funkcje.

Beztypowy rachunek lambda

- Funkcja rozumiana jako działanie.
- Przedmiotem działania może być cokolwiek, zatem...
- ...funkcja nie ma a priori ograniczonej dziedziny.
- Funkcja może być np. aplikowana sama do siebie.
- Każdemu obiektowi można przypisać działanie, więc...
- ...nie ma innych obiektów niż funkcje.

Analogia: Każdy ciąg bitów można zinterpretować
– jako program;
– jako dane.

Składnia: lambda-wyrażenia

Lambda-wyrażenia:

- Zmienne x, y, z, ...
- Aplikacje (MN);
- Abstrakcje $(\lambda x\, M)$.

Składnia: lambda-wyrażenia

Lambda-wyrażenia:

- Zmienne x, y, z, ...
- Aplikacje (MN);
- Abstrakcje $(\lambda x\, M)$.

Konwencje:

- Opuszczamy zewnętrzne nawiasy;
- Aplikacja wiąże w lewo: MNP oznacza $(MN)P$
- Skrót z kropką: $\lambda x_1 \ldots x_n.M$ oznacza $\lambda x_1(\ldots(\lambda x_n M)\cdots)$.

○

Przykłady

$$\mathsf{I} \; = \; \lambda x.x$$

Przykłady

$$
\begin{aligned}
\mathsf{I} &= \lambda x.x \\
\mathsf{K} &= \lambda xy.x
\end{aligned}
$$

Przykłady

$$\begin{aligned} \mathsf{I} &= \lambda x.x \\ \mathsf{K} &= \lambda xy.x \\ \mathsf{S} &= \lambda xyz.xz(yz) \end{aligned}$$

Przykłady

$$
\begin{aligned}
\mathsf{I} &= \lambda x.x \\
\mathsf{K} &= \lambda xy.x \\
\mathsf{S} &= \lambda xyz.xz(yz) \\
\mathsf{2} &= \lambda fx.f(fx)
\end{aligned}
$$

Przykłady

$$\begin{aligned}
\mathsf{I} &= \lambda x.x \\
\mathsf{K} &= \lambda xy.x \\
\mathsf{S} &= \lambda xyz.xz(yz) \\
\mathsf{2} &= \lambda fx.f(fx) \\
\omega &= \lambda x.xx
\end{aligned}$$

Przykłady

$$
\begin{aligned}
\mathsf{I} &= \lambda x.x \\
\mathsf{K} &= \lambda xy.x \\
\mathsf{S} &= \lambda xyz.xz(yz) \\
\mathsf{2} &= \lambda fx.f(fx) \\
\omega &= \lambda x.xx \\
\Omega &= \omega\omega
\end{aligned}
$$

Przykłady

$$
\begin{aligned}
\mathsf{I} &= \lambda x.x \\
\mathsf{K} &= \lambda xy.x \\
\mathsf{S} &= \lambda xyz.xz(yz) \\
\mathsf{2} &= \lambda fx.f(fx) \\
\omega &= \lambda x.xx \\
\Omega &= \omega\omega \\
\mathsf{Y} &= \lambda f((\lambda x.f(xx))(\lambda x.f(xx))
\end{aligned}
$$

Zmienne *wolne* (globalne)

$$\mathrm{FV}(x) = \{x\};$$

$$\mathrm{FV}(MN) = \mathrm{FV}(M) \cup \mathrm{FV}(N);$$

$$\mathrm{FV}(\lambda x\, M) = \mathrm{FV}(M) - \{x\}.$$

Zmienne *wolne* (globalne)

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

Na przykład:

$$\mathrm{FV}(\lambda x\, x) = \varnothing;$$

Zmienne *wolne* (globalne)

$$\mathrm{FV}(x) = \{x\};$$

$$\mathrm{FV}(MN) = \mathrm{FV}(M) \cup \mathrm{FV}(N);$$

$$\mathrm{FV}(\lambda x\, M) = \mathrm{FV}(M) - \{x\}.$$

Na przykład:

$$\mathrm{FV}(\lambda x\, x) = \varnothing;$$

$$\mathrm{FV}(\lambda x.\, xy) = \{y\};$$

Zmienne *wolne* (globalne)

$$\mathsf{FV}(x) = \{x\};$$

$$\mathsf{FV}(MN) = \mathsf{FV}(M) \cup \mathsf{FV}(N);$$

$$\mathsf{FV}(\lambda x\, M) = \mathsf{FV}(M) - \{x\}.$$

Na przykład:

$$\mathsf{FV}(\lambda x\, x) = \varnothing;$$

$$\mathsf{FV}(\lambda x.\, xy) = \{y\};$$

$$\mathsf{FV}((\lambda x.\, xy)(\lambda y.\, xy)) = \{x, y\}.$$

Alfa-konwersja

Wyrażenia $\lambda x.\, xy$ i $\lambda z.\, zy$ oznaczają tę samą operację („zaaplikuj dany argument do y”).

Alfa-konwersja

Wyrażenia $\lambda x.\, xy$ i $\lambda z.\, zy$ oznaczają tę samą operację („zaaplikuj dany argument do y”).

Należy je uważać za identyczne.

Alfa-konwersja

Wyrażenia $\lambda x.\, xy$ i $\lambda z.\, zy$ oznaczają tę samą operację („zaaplikuj dany argument do y").

Należy je uważać za identyczne.

Alfa-konwersja: Wyrażenia różniące się tylko wyborem zmiennych związanych utożamiamy.

Lambda-termy to klasy abstrakcji tego utożsamienia.

Alfa-konwersja

Wyrażenia $\lambda x.\, xy$ i $\lambda z.\, zy$ oznaczają tę samą operację („zaaplikuj dany argument do y”).

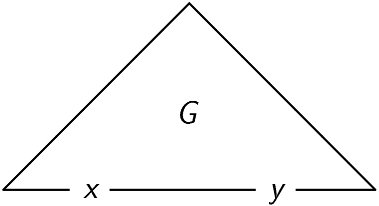
Należy je uważać za identyczne.

Alfa-konwersja: Wyrażenia różniące się tylko wyborem zmiennych związanych utożamiamy.

Lambda-termy to klasy abstrakcji tego utożsamienia.

Łatwiej powiedzieć, niż zrobić. . .

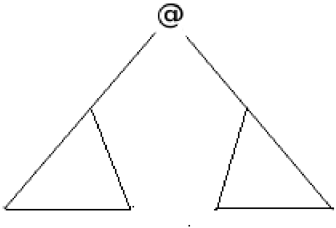
Termy jako grafy:



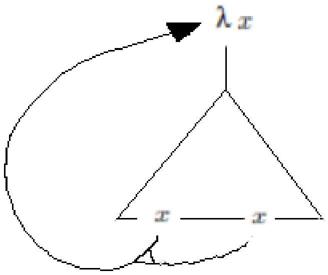
- Jeden wierzchołek początkowy;
- Zmienne wolne jako wierzchołki końcowe (liście).

Termy jako grafy: zmienna

Termy jako grafy: aplikacja

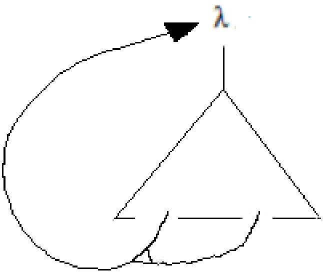


Termy jako grafy: abstrakcja

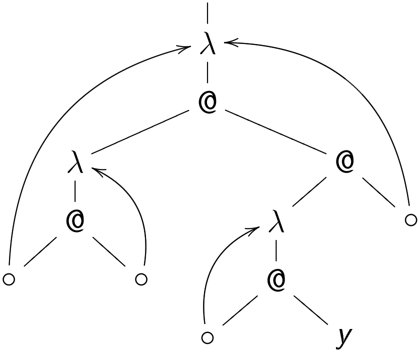


Termy jako grafy: abstrakcja

Zmienne związane sa niepotrzebne.



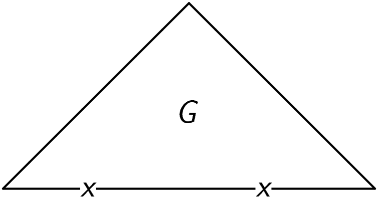
Przykład



To jest graf termu $\lambda x.(\lambda y.\, xy)((\lambda z.\, zy)x)$

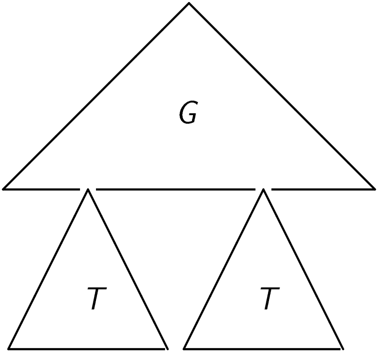
Podstawienie $G[x := T]$

Podstawienie termu T do termu G w miejsce wolnych wystąpień zmiennej x.



Podstawienie $G[x := T]$

Podstawienie termu T do termu G w miejsce wolnych wystąpień zmiennej x.



◦

Podstawienie

- $x[x := N] = N;$

Podstawienie

- $x[x := N] = N$;
- $y[x := N] = y$,

Podstawienie

- $x[x := N] = N$;
- $y[x := N] = y$,
  gdy y jest zmienną różną od x;

Podstawienie

- $x[x := N] = N$;
- $y[x := N] = y$,
  gdy y jest zmienną różną od x;
- $(PQ)[x := N] = P[x := N]Q[x := N]$;

Podstawienie

- $x[x := N] = N$;
- $y[x := N] = y$,
  gdy y jest zmienną różną od x;
- $(PQ)[x := N] = P[x := N]Q[x := N]$;
- $(\lambda y\, P)[x := N] = \lambda y.P[x := N]$,

Podstawienie

- $x[x := N] = N$;
- $y[x := N] = y$,
  gdy y jest zmienną różną od x;
- $(PQ)[x := N] = P[x := N]Q[x := N]$;
- $(\lambda y\, P)[x := N] = \lambda y.P[x := N]$,
  gdy $y \neq x$ oraz $y \notin \mathrm{FV}(N)$.

Podstawienie

- $x[x := N] = N$;
- $y[x := N] = y$,
  gdy y jest zmienną różną od x;
- $(PQ)[x := N] = P[x := N]Q[x := N]$;
- $(\lambda y\, P)[x := N] = \lambda y.P[x := N]$,
  gdy $y \neq x$ oraz $y \notin \mathrm{FV}(N)$.

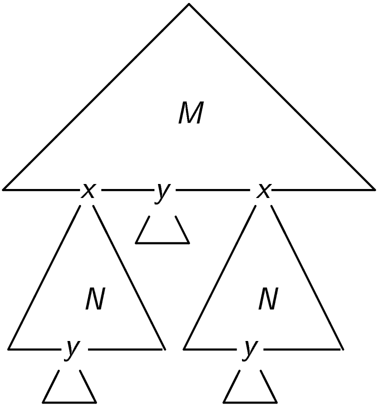
Wykonanie podstawienia na konkretnej reprezentacji termu może wymagać wymiany zmiennych:

$(\lambda y\, P)[x := N] = \lambda z\, P[y := z][x := N]$, gdzie z jest „nowe".

Lemat o podstawieniu

$$M[x := N][y := R] = M[y := R][x := N[y := R]]$$

$$\text{gdy } x \notin \mathrm{FV}(R) \text{ lub } y \notin \mathrm{FV}(M)$$



Beta-redukcja

Najmniejsza relacja $\rightarrow_\beta$, spełniająca warunki:

- $(\lambda x P)Q \rightarrow_\beta P[x := Q]$;
- jeśli $M \rightarrow_\beta M'$, to:
  $MN \rightarrow_\beta M'N$, $NM \rightarrow_\beta NM'$ oraz $\lambda x M \rightarrow_\beta \lambda x M'$.

Beta-redukcja

Najmniejsza relacja $\to_\beta$, spełniająca warunki:

- $(\lambda x P)Q \to_\beta P[x := Q]$;
- jeśli $M \to_\beta M'$, to:
  $MN \to_\beta M'N$, $NM \to_\beta NM'$ oraz $\lambda x M \to_\beta \lambda x M'$.

Term postaci $(\lambda x P)Q$ to *β-redeks*.

Beta-redukcja

Najmniejsza relacja $\to_\beta$, spełniająca warunki:

- $(\lambda x P)Q \to_\beta P[x := Q]$;
- jeśli $M \to_\beta M'$, to:
  $MN \to_\beta M'N$, $NM \to_\beta NM'$ oraz $\lambda x M \to_\beta \lambda x M'$.

Term postaci $(\lambda x P)Q$ to *β-redeks*.

Relacja $\to_\beta$ to zredukowanie jednego dowolnego redeksu.

Relacje pochodne:

Dowolna liczba kroków: $\twoheadrightarrow_\beta$ lub $\longrightarrow_\beta^*$;

Niezerowa liczba kroków: $\longrightarrow_\beta^+$;

Co najwyżej jeden krok: $\longrightarrow_\beta^=$;

Równoważność (beta-konwersja): $=_\beta$.

Przykład: $\mathsf{SKK} =_\beta \mathsf{I}$

$$\mathsf{SKK} = (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x)$$

Przykład: $\mathsf{SKK} =_\beta \mathsf{I}$

$$\mathsf{SKK} = (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x)$$

Przykład: $\mathbf{SKK} =_\beta \mathbf{I}$

$$\begin{aligned}\mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x)\\ &\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x)\end{aligned}$$

Przykład: $\mathsf{SKK} =_\beta \mathsf{I}$

$$\begin{aligned} \mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x) \\ &\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x) \end{aligned}$$

Przykład: $\mathsf{SKK} =_\beta \mathsf{I}$

$$
\begin{aligned}
\mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x) \\
&\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x) \\
&\to_\beta \lambda z.(\lambda xy.x)z((\lambda xy.x)z)
\end{aligned}
$$

Przykład: $\mathsf{SKK} =_\beta \mathsf{I}$

$$
\begin{aligned}
\mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x) \\
&\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x) \\
&\to_\beta \lambda z.(\lambda xy.x)z((\lambda xy.x)z)
\end{aligned}
$$

Przykład: $\mathbf{SKK} =_\beta \mathbf{I}$

$$
\begin{aligned}
\mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x) \\
&\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x) \\
&\to_\beta \lambda z.(\lambda xy.x)z((\lambda xy.x)z) \\
&\to_\beta \lambda z.(\lambda y.z)((\lambda xy.x)z)
\end{aligned}
$$

Przykład: $\mathsf{SKK} =_\beta \mathsf{I}$

$$
\begin{aligned}
\mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x) \\
&\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x) \\
&\to_\beta \lambda z.(\lambda xy.x)z((\lambda xy.x)z) \\
&\to_\beta \lambda z.(\lambda y.z)((\lambda xy.x)z)
\end{aligned}
$$

Przykład: $\mathbf{SKK} =_\beta \mathbf{I}$

$$
\begin{aligned}
\mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x) \\
&\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x) \\
&\to_\beta \lambda z.(\lambda xy.x)z((\lambda xy.x)z) \\
&\to_\beta \lambda z.(\lambda y.z)((\lambda xy.x)z) \\
&\to_\beta \lambda z.(\lambda y.z)(\lambda y.z)
\end{aligned}
$$

Przykład: $\mathsf{SKK} =_\beta \mathsf{I}$

$$
\begin{aligned}
\mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x) \\
&\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x) \\
&\to_\beta \lambda z.(\lambda xy.x)z((\lambda xy.x)z) \\
&\to_\beta \lambda z.(\lambda y.z)((\lambda xy.x)z) \\
&\to_\beta \lambda z.(\lambda y.z)(\lambda y.z)
\end{aligned}
$$

Przykład: $\mathsf{SKK} =_\beta \mathsf{I}$

$$
\begin{aligned}
\mathsf{SKK} &= (\lambda xyz.xz(yz))(\lambda xy.x)(\lambda xy.x) \\
&\to_\beta (\lambda yz.(\lambda xy.x)z(yz))(\lambda xy.x) \\
&\to_\beta \lambda z.(\lambda xy.x)z((\lambda xy.x)z) \\
&\to_\beta \lambda z.(\lambda y.z)((\lambda xy.x)z) \\
&\to_\beta \lambda z.(\lambda y.z)(\lambda y.z) \\
&\to_\beta \lambda z.z = \mathsf{I}
\end{aligned}
$$

∘

Jaja aligatorów

http://worrydream.com/AlligatorEggs/

Rachunek lambda jako teoria równościowa

Termy M i N są beta-równe ($M =_\beta N$) wtedy i tylko wtedy, gdy równość „$M = N$" można udowodnić w systemie:

$$(\beta)\quad (\lambda x\, M)N = M[x := N] \qquad\qquad x = x$$

$$\frac{M = N}{MP = NP} \qquad \frac{M = N}{PM = PN} \qquad (\xi)\ \frac{M = N}{\lambda x\, M = \lambda x\, N}$$

$$\frac{M = N}{N = M} \qquad\qquad \frac{M = N, N = P}{M = P}$$

○

Normalizacja

Postać normalna to term bez redeksów.

Normalizacja

Postać normalna to term bez redeksów.

Nie da się go redukować.

Normalizacja

Postać normalna to term bez redeksów.

Nie da się go redukować.

Term M *ma postać normalną* (*jest normalizowalny*), gdy redukuje się do pewnej postaci normalnej.

Normalizacja

Postać normalna to term bez redeksów.

Nie da się go redukować.

Term M *ma postać normalną* (*jest normalizowalny*), gdy redukuje się do pewnej postaci normalnej.

Nazywamy ją *postacią normalna* termu M.

Normalizacja

Postać normalna to term bez redeksów.

Nie da się go redukować.

Term M *ma postać normalną* (*jest normalizowalny*), gdy redukuje się do pewnej postaci normalnej.

Nazywamy ją *postacią normalna* termu M.

Term M jest *silnie normalizowalny* ($M \in \mathsf{SN}$), gdy nie istnieje nieskończony ciąg

$$M = M_0 \to_\beta M_1 \to_\beta M_2 \to_\beta \cdots$$

Normalizacja

Postać normalna to term bez redeksów.

Nie da się go redukować.

Term M *ma postać normalną* (*jest normalizowalny*), gdy redukuje się do pewnej postaci normalnej.

Nazywamy ją *postacią normalna* termu M.

Term M jest *silnie normalizowalny* ($M \in \mathrm{SN}$), gdy nie istnieje nieskończony ciąg

$$M = M_0 \to_\beta M_1 \to_\beta M_2 \to_\beta \cdots$$

Inaczej: każdy ciąg redukcji prowadzi do postaci normalnej.

Przykłady

- Term $\mathbf{S} = \lambda xyz.xz(yz)$ jest w postaci normalnej.

Przykłady

- Term $\mathbf{S} = \lambda xyz.xz(yz)$ jest w postaci normalnej.
- Term $\mathbf{SKK}$ jest silnie normalizowalny i ma postać normalną $\mathbf{I}$.

Przykłady

- Term $\mathbf{S} = \lambda xyz.xz(yz)$ jest w postaci normalnej.
- Term $\mathbf{SKK}$ jest silnie normalizowalny i ma postać normalną $\mathbf{I}$.
- Term $\mathbf{\Omega} = (\lambda x.\, xx)(\lambda x.\, xx)$ nie ma postaci normalnej.

Przykłady

- Term $\mathbf{S} = \lambda xyz.xz(yz)$ jest w postaci normalnej.
- Term $\mathbf{SKK}$ jest silnie normalizowalny i ma postać normalną $\mathbf{I}$.
- Term $\mathbf{\Omega} = (\lambda x.\, xx)(\lambda x.\, xx)$ nie ma postaci normalnej.
- Term $(\lambda x.\, y)\mathbf{\Omega}$ ma postać normalną y, ale nie jest silnie normalizowalny.

Wołanie przez nazwę

$$(\lambda x P)Q \to_\beta P[x := Q]$$

Wołanie przez nazwę

$$(\lambda x P)Q \to_\beta P[x := Q]$$

Ewaluacja procedury o parametrze formalnym x i treści P, gdy parametrem aktualnym jest Q:

Wołanie przez nazwę

$$(\lambda x P)Q \to_\beta P[x := Q]$$

Ewaluacja procedury o parametrze formalnym x i treści P, gdy parametrem aktualnym jest Q:

Należy wstawić parametr aktualny do treści procedury,

Wołanie przez nazwę

$$(\lambda x P)Q \to_\beta P[x := Q]$$

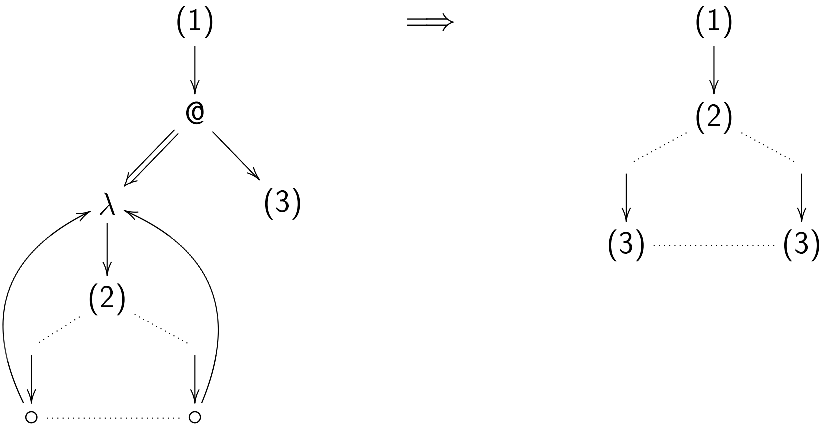
Ewaluacja procedury o parametrze formalnym x i treści P, gdy parametrem aktualnym jest Q:

Należy wstawić parametr aktualny do treści procedury, wymieniając, jeśli trzeba, lokalne identyfikatory na nowe.

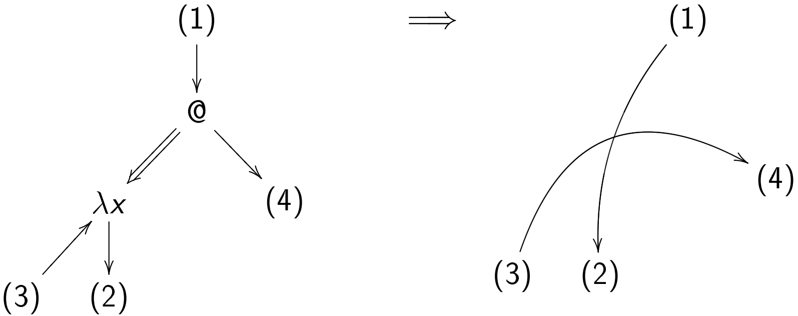
Beta-redeks w grafie

$$
\begin{tikzcd}
 & (1) \arrow[d] & \\
 & @ \arrow[dl, Rightarrow] \arrow[dr] & \\
\lambda \arrow[d] & & (3) \\
(2) & &
\end{tikzcd}
$$

Beta-redukcja w grafie



Beta-redukcja nieco wyidealizowana



Kompozycjonalność

Lemat

(1) Jeśli $M \to_\beta M'$, *to* $M[x := N] \to_\beta M'[x := N]$;

(2) Jeśli $N \to_\beta N'$, *to* $M[x := N] \twoheadrightarrow_\beta M[x := N']$,

Dowód: Indukcja ze względu na długość M. $\square$

Wniosek

Jeśli $M \twoheadrightarrow_\beta M'$ *i* $N \twoheadrightarrow_\beta N'$, *to* $M[x := N] \twoheadrightarrow_\beta M'[x := N']$.

$\circ$

Własność Churcha-Rossera (CR)

Jeśli $a \twoheadrightarrow b$ i $a \twoheadrightarrow c$, to istnieje takie d, że $b \twoheadrightarrow d$ i $c \twoheadrightarrow d$.

Własność Churcha-Rossera (CR)

Jeśli $a \twoheadrightarrow b$ i $a \twoheadrightarrow c$, to istnieje takie d, że $b \twoheadrightarrow d$ i $c \twoheadrightarrow d$.

$$
\begin{tikzcd}
 & a \arrow[dl, two heads] \arrow[dr, two heads] & \\
b \arrow[dr, two heads, dashed] & & c \arrow[dl, two heads, dashed] \\
 & d &
\end{tikzcd}
$$

Twierdzenie Churcha-Rossera: Beta ma własność CR

Relacja pomocnicza $\xrightarrow{1}$

- $x \xrightarrow{1} x$, gdy x jest zmienną;
- jeśli $M \xrightarrow{1} M'$, to $\lambda x M \xrightarrow{1} \lambda x M'$;
- jeśli $M \xrightarrow{1} M'$ i $N \xrightarrow{1} N'$, to:

  $MN \xrightarrow{1} M'N'$, oraz

  $(\lambda x M)N \xrightarrow{1} M'[x := N']$.

Sens: jednoczesna redukcja kilku redeksów

Twierdzenie Churcha-Rossera: Beta ma własność CR

Relacja pomocnicza $\xrightarrow{1}$

- $x \xrightarrow{1} x$, gdy x jest zmienną;
- jeśli $M \xrightarrow{1} M'$, to $\lambda x M \xrightarrow{1} \lambda x M'$;
- jeśli $M \xrightarrow{1} M'$ i $N \xrightarrow{1} N'$, to:

  $MN \xrightarrow{1} M'N'$, oraz

  $(\lambda x M)N \xrightarrow{1} M'[x := N']$.

Sens: jednoczesna redukcja kilku redeksów już obecnych w termie.

○

Pełne rozwinięcie

Term $M^\bullet$ to *pełne rozwinięcie* termu M.

– $x^\bullet = x$;

– $(\lambda x\, M)^\bullet = \lambda x\, M^\bullet$;

– $(MN)^\bullet = M^\bullet N^\bullet$, gdy M nie jest abstrakcją;

– $((\lambda x\, M)N)^\bullet = M^\bullet[x := N^\bullet]$.

Sens: jednoczesna redukcja wszystkich istniejących redeksów.

Własności relacji $\xrightarrow{1}$

(1) Jeśli $M \xrightarrow{1} M'$, to $\mathsf{FV}(M') \subseteq \mathsf{FV}(M)$.

Własności relacji $\xrightarrow{1}$

(1) Jeśli $M \xrightarrow{1} M'$, to $\mathsf{FV}(M') \subseteq \mathsf{FV}(M)$.

(2) Dla dowolnego M zachodzi $M \xrightarrow{1} M$ oraz $M \xrightarrow{1} M^{\bullet}$.

Własności relacji $\xrightarrow{1}$

(1) Jeśli $M \xrightarrow{1} M'$, to $\mathsf{FV}(M') \subseteq \mathsf{FV}(M)$.

(2) Dla dowolnego M zachodzi $M \xrightarrow{1} M$ oraz $M \xrightarrow{1} M^{\bullet}$.

(3) Jeśli $M \xrightarrow{1} M'$ i $N \xrightarrow{1} N'$, to $M[x := N] \xrightarrow{1} M'[x := N']$.

Własności relacji $\xrightarrow{1}$

(1) Jeśli $M \xrightarrow{1} M'$, to $\mathrm{FV}(M') \subseteq \mathrm{FV}(M)$.

(2) Dla dowolnego M zachodzi $M \xrightarrow{1} M$ oraz $M \xrightarrow{1} M^\bullet$.

(3) Jeśli $M \xrightarrow{1} M'$ i $N \xrightarrow{1} N'$, to $M[x := N] \xrightarrow{1} M'[x := N']$.

(4) Jeśli $M \xrightarrow{1} M'$, to $M' \xrightarrow{1} M^\bullet$.

○

Relacja $\xrightarrow{1}$ ma własność rombu

$$
\begin{tikzcd}
& M \arrow[dl, "1"'] \arrow[dr, "1"] & \\
M' \arrow[dr, dashed, "1"'] & & M'' \arrow[dl, dashed, "1"] \\
& M^{\bullet} &
\end{tikzcd}
$$

Dowód twierdzenia Churcha-Rossera

1) Ponieważ relacja $\xrightarrow{1}$ ma własność rombu, więc tym bardziej jej domknięcie przechodnio-zwrotne $\overset{1}{\twoheadrightarrow}$ ma własność rombu.

Dowód twierdzenia Churcha-Rossera

1) Ponieważ relacja $\xrightarrow{1}$ ma własność rombu, więc tym bardziej jej domknięcie przechodnio-zwrotne $\overset{1}{\twoheadrightarrow}$ ma własność rombu.

$$
\begin{tikzcd}
M_1 \arrow[r] \arrow[d] & \cdot \arrow[r] \arrow[d, dotted] & \cdot \arrow[r] \arrow[d, dotted] & M_2 \arrow[d, dotted] \\
\cdot \arrow[r, dotted, no head] \arrow[d] & \cdot \arrow[r, dotted] \arrow[d, dotted] & \cdot \arrow[r, dotted] \arrow[d, dotted] & \cdot \arrow[d, dotted] \\
M_3 \arrow[r, dotted] & \cdot \arrow[r, dotted] & \cdot \arrow[r, dotted] & M_4
\end{tikzcd}
$$

Dowód twierdzenia Churcha-Rossera

1) Ponieważ relacja $\xrightarrow{1}$ ma własność rombu, więc tym bardziej jej domknięcie przechodnio-zwrotne $\overset{1}{\twoheadrightarrow}$ ma własność rombu.

$$
\begin{tikzcd}
M_1 \arrow[r] \arrow[d] & \cdot \arrow[r] \arrow[d, dotted] & \cdot \arrow[r] \arrow[d, dotted] & M_2 \arrow[d, dotted] \\
\cdot \arrow[r, dotted, dash] \arrow[d] & \cdot \arrow[r, dotted] \arrow[d, dotted] & \cdot \arrow[r, dotted] \arrow[d, dotted] & \cdot \arrow[d, dotted] \\
M_3 \arrow[r, dotted] & \cdot \arrow[r, dotted] & \cdot \arrow[r, dotted] & M_4
\end{tikzcd}
$$

2) Ponieważ $\rightarrow_\beta \subseteq \xrightarrow{1} \subseteq \twoheadrightarrow_\beta$, więc $\overset{1}{\twoheadrightarrow}$ i $\twoheadrightarrow_\beta$ są równe.

Dowód twierdzenia Churcha-Rossera

1) Ponieważ relacja $\xrightarrow{1}$ ma własność rombu, więc tym bardziej jej domknięcie przechodnio-zwrotne $\overset{1}{\twoheadrightarrow}$ ma własność rombu.

$$
\begin{tikzcd}
M_1 \arrow[r] \arrow[d] & \cdot \arrow[r] \arrow[d, dotted] & \cdot \arrow[r] \arrow[d, dotted] & M_2 \arrow[d, dotted] \\
\cdot \arrow[r, dotted, no head] \arrow[d] & \cdot \arrow[r, dotted] \arrow[d, dotted] & \cdot \arrow[r, dotted] \arrow[d, dotted] & \cdot \arrow[d, dotted] \\
M_3 \arrow[r, dotted] & \cdot \arrow[r, dotted] & \cdot \arrow[r, dotted] & M_4
\end{tikzcd}
$$

2) Ponieważ $\rightarrow_\beta \subseteq \xrightarrow{1} \subseteq \twoheadrightarrow_\beta$, więc $\overset{1}{\twoheadrightarrow}$ i $\twoheadrightarrow_\beta$ są równe.

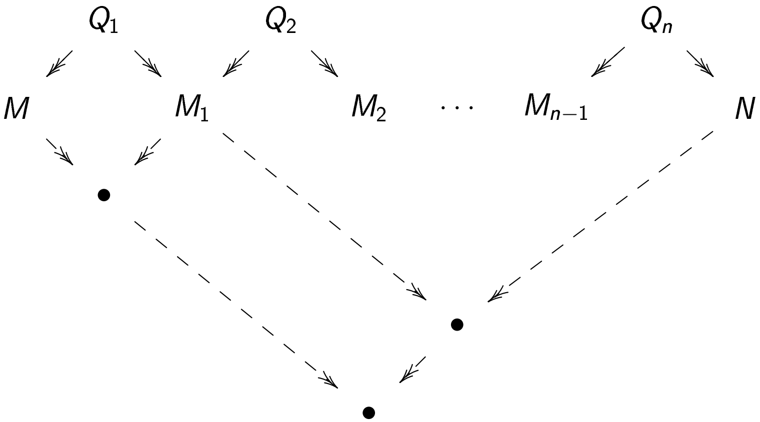
3) Własność rombu dla $\twoheadrightarrow_\beta$ to własność CR dla $\rightarrow_\beta$.

Wnioski z twierdzenia Churcha-Rossera

(1) *Jeśli* $M =_\beta N$, *to* $M \twoheadrightarrow_\beta Q \;_\beta\!\twoheadleftarrow N$, *dla pewnego* Q.

Wnioski z twierdzenia Churcha-Rossera

(1) *Jeśli* $M =_\beta N$, *to* $M \twoheadrightarrow_\beta Q \,{}_\beta\!\twoheadleftarrow N$, *dla pewnego* Q.



Wnioski z twierdzenia Churcha-Rossera

(2) *Każdy term ma co najwyżej jedną postać normalną (i do niej się redukuje).*

Wnioski z twierdzenia Churcha-Rossera

(2) *Każdy term ma co najwyżej jedną postać normalną (i do niej się redukuje).*

Dowód:
Jeśli $M =_\beta N$ i N normalne, to $M \twoheadrightarrow_\beta Q\,{}_\beta\!\twoheadleftarrow N$.
Skoro N normalne, to $N = Q$.
Jeśli M też normalne, to $M = Q = N$.

$\circ$

Wnioski z twierdzenia Churcha-Rossera

(3) *Beta-konwersja jest niesprzeczną teorią równościową*

Wnioski z twierdzenia Churcha-Rossera

(3) *Beta-konwersja jest niesprzeczną teorią równościową*

Dowód: Na przykład $\nvdash x = y$, ponieważ $x \neq_\beta y$.

○

Eta-reduction

The least relation $\to_\eta$, satisfying the conditions:

- $\lambda x.Mx \to_\eta M$, when $x \notin \mathrm{FV}(M)$;
- jeśli $M \to_\eta M'$, to $MN \to_\eta M'N$, $NM \to_\eta NM'$ and $\lambda xM \to_\eta \lambda xM'$.

Eta-reduction

The least relation $\to_\eta$, satisfying the conditions:

- $\lambda x.Mx \to_\eta M$, when $x \notin \mathrm{FV}(M)$;
- jeśli $M \to_\eta M'$, to $MN \to_\eta M'N$, $NM \to_\eta NM'$ and $\lambda xM \to_\eta \lambda xM'$.

Symbol $\to_{\beta\eta}$ stands for the union of relations $\to_\beta$ and $\to_\eta$.

Other definitions and notation are applicable respectively.

Eta-reduction

Fakt

Eta-reduction is strongly normalizing (because terms shrink under eta).

Eta-reduction

Fakt

Eta-reduction is strongly normalizing (because terms shrink under eta).

Fakt

Beta-eta-reduction is Church-Rosser.

Standardization

Standard reduction: Always reduce the leftmost redex.

Theorem:

If M has a β-normal form then the leftmost reduction leads to the normal form.

Standardization

Standard reduction: Always reduce the leftmost redex.

Theorem:

If M has a β-normal form then the leftmost reduction leads to the normal form.

Slogan:
The leftmost *reduction strategy* is normalizing.

○

Head normalization

A term $\lambda\vec{x}.z\vec{R}$ is in *head normal form*.

A term $\lambda\vec{x}.(\lambda y.P)Q\vec{R}$ has a *head redex* $(\lambda y.P)Q$.

Head normalization

A term $\lambda\vec{x}.z\vec{R}$ is in *head normal form*.

A term $\lambda\vec{x}.(\lambda y.P)Q\vec{R}$ has a *head redex* $(\lambda y.P)Q$.

A reduction step of the form

$$M = \lambda\vec{x}.(\lambda y.P)Q\vec{R} \to_\beta \lambda\vec{x}.P[y := Q]\vec{R} = N$$

is called *head reduction*. Write $M \xrightarrow{\mathrm{h}} N$.

Other reductions are *internal*. Write $M \xrightarrow{\mathrm{i}} N$.

Main Lemma

Lemma: If $M \twoheadrightarrow_\beta N$ then $M \stackrel{\mathrm{h}}{\twoheadrightarrow} P \stackrel{\mathrm{i}}{\twoheadrightarrow} N$, for some P.

Main Lemma

Lemma: If $M \twoheadrightarrow_\beta N$ then $M \overset{\mathrm{h}}{\twoheadrightarrow} P \overset{\mathrm{i}}{\twoheadrightarrow} N$, for some P.

Warning: A naive proof attempt fails. This diagram is wrong.

$$
\begin{tikzcd}
 & \cdot \arrow[dr, "h"] & \\
\cdot \arrow[ur, two heads, "i"] \arrow[dr, dashed, "h"'] & & \cdot \\
 & \cdot \arrow[ur, dashed, two heads, "i"'] &
\end{tikzcd}
$$

Curry's fixed point combinator $\mathbf{Y}$

$$\mathbf{Y} = \lambda f((\lambda x.f(xx))(\lambda x.f(xx))$$

Curry's fixed point combinator $\mathbf{Y}$

$$\mathbf{Y} = \lambda f((\lambda x.f(xx))(\lambda x.f(xx))$$

Fact: $\mathbf{Y}F =_\beta F(\mathbf{Y}F)$, for every F.

Curry's fixed point combinator $\mathbf{Y}$

$$\mathbf{Y} = \lambda f((\lambda x.f(xx))(\lambda x.f(xx))$$

Fact: $\mathbf{Y}F =_\beta F(\mathbf{Y}F)$, for every F.

Proof: $\mathbf{Y}F \rightarrow_\beta ((\lambda x.F(xx))(\lambda x.F(xx)) \rightarrow_\beta$

$$F((\lambda x.F(xx))(\lambda x.F(xx))\ {}_\beta\!\leftarrow F(\mathbf{Y}F)$$

$\circ$

$\mathbf{Y}F =_\beta F(\mathbf{Y}F)$

Example: Find an M such that $Mxy =_\beta MxxyM$.

$\mathbf{Y}F =_\beta F(\mathbf{Y}F)$

Example: Find an M such that $Mxy =_\beta MxxyM$.

Solution: No problem, $M = \mathbf{Y}(\lambda m\, \lambda xy.\, mxxym)$.

$\circ$

Turing's fixed-point combinator

$$\mathbf{\Theta} = (\lambda xf.\, f(xxf))(\lambda xf.\, f(xxf))$$

Turing's fixed-point combinator

$$\mathbf{\Theta} = (\lambda xf.\, f(xxf))(\lambda xf.\, f(xxf))$$

Fact: $\mathbf{\Theta} F \twoheadrightarrow_\beta F(\mathbf{\Theta} F)$, all F.

Turing's fixed-point combinator

$$\mathbf{\Theta} = (\lambda xf.\, f(xxf))(\lambda xf.\, f(xxf))$$

Fact: $\mathbf{\Theta} F \twoheadrightarrow_\beta F(\mathbf{\Theta} F)$, all F.

Proof: $\mathbf{\Theta} F = (\lambda xf.\, f(xxf))(\lambda xf.\, f(xxf))F \to_\beta$
$(\lambda f.\, f((\lambda xf.\, f(xxf))(\lambda xf.\, f(xxf))f)F \to_\beta$
$F((\lambda xf.\, f(xxf))(\lambda xf.\, f(xxf))F) = F(\mathbf{\Theta} F)$

$\circ$