Функциональное программирование
Лекция 2. Рекурсия и редукция

Денис Николаевич Москвин

СПбАУ РАН, CS Center

19.02.2015

План лекции

План лекции

Термовые уравнения

Схема β-преобразования $(\lambda\, n.\, M)\, N \;=\; M[n := N]$ даёт возможность решать простейшие уравнения на термы.

Пример

Найти F, такой что $\forall\, M, N, L \;\; \lambda \vdash F\, M\, N\, L = M\, L\, (N\, L)$.

$$\begin{aligned} F\, M\, N\, L &= M\, L\, (N\, L) \\ F\, M\, N &= \lambda l.\, M\, l\, (N\, l) \\ F\, M &= \lambda n.\, \lambda l.\, M\, l\, (n\, l) \\ F &= \lambda m\, n\, l.\, m\, l\, (n\, l) \end{aligned}$$

А если уравнение рекурсивное, например, $F\, M = M\, F$?

Термовые уравнения

Схема β-преобразования $(\lambda\, n.\, M)\, N \;=\; M[n := N]$ даёт возможность решать простейшие уравнения на термы.

Пример

Найти F, такой что $\forall\, M, N, L \;\; \lambda \vdash F\, M\, N\, L = M\, L\, (N\, L)$.

$$
\begin{aligned}
F\, M\, N\, L &= M\, L\, (N\, L)\\
F\, M\, N &= \lambda l.\, M\, l\, (N\, l)\\
F\, M &= \lambda n.\, \lambda l.\, M\, l\, (n\, l)\\
F &= \lambda m\, n\, l.\, m\, l\, (n\, l)
\end{aligned}
$$

А если уравнение рекурсивное, например, $F\, M = M\, F$?

Оказывается, имеется универсальный способ решения!

Теоремы о неподвижной точке

Теорема

Для любого λ-терма F существует неподвижная точка:
$\forall F \in \Lambda \quad \exists X \in \Lambda \qquad \lambda \vdash F\,X = X$

Доказательство

Введем $W \equiv \lambda x.\, F\,(x\,x)$ и $X \equiv W\,W$. Тогда
$X \equiv W\,W \equiv (\lambda x.\, F(x\,x))\,W = F\,(W\,W) \equiv F\,X \quad \blacksquare$

Теорема

Существует комбинатор неподвижной точки $\mathbf{Y}$, такой что
$\forall F \quad F(\mathbf{Y}\,F) = \mathbf{Y}\,F.$

Доказательство

Введём $\mathbf{Y} \equiv \lambda f.\, (\lambda x.\, f\,(x\,x))(\lambda x.\, f\,(x\,x))$. Имеем $\mathbf{Y}\,F \equiv$
$(\lambda x.\, F\,(x\,x))(\lambda x.\, F\,(x\,x)) = F\,(\underbrace{(\lambda x.\, F\,(x\,x))(\lambda x.\, F\,(x\,x))}_{\mathbf{Y}\,F}) \equiv F(\mathbf{Y}\,F) \quad \blacksquare$

Y-комбинатор позволяет ввести рекурсию в λ-исчисление.

Пример

Факториал рекурсивно:

$$\mathtt{fac} = \lambda n.\, \mathtt{iif}\,(\mathtt{iszro}\; n)\; 1\; (\mathtt{mult}\; n\,(\mathtt{fac}\,(\mathtt{pred}\; n)))$$

Переписываем в виде

$$\mathtt{fac} = \underbrace{(\lambda f\, n.\, \mathtt{iif}\,(\mathtt{iszro}\; n)\; 1\; (\mathtt{mult}\; n\,(f\,(\mathtt{pred}\; n))))}_{\mathtt{fac}'}\; \mathtt{fac}$$

Отсюда видно, что fac — неподвижная точка для вспомогательной функции fac′:

$$\mathtt{fac} = \mathbf{Y}\; \mathtt{fac}'$$

Как работает $\mathtt{fac} \equiv \mathbf{Y}\,\mathtt{fac}'$?

Пример

$$
\begin{array}{rcl}
\mathtt{fac}\;3 & = & (\mathbf{Y}\,\mathtt{fac}')\;3\\
 & = & \mathtt{fac}'\;(\mathbf{Y}\,\mathtt{fac}')\;3\\
 & = & \mathtt{iif}\;(\mathtt{iszro}\;3)\;1\;(\mathtt{mult}\;3\;((\mathbf{Y}\,\mathtt{fac}')\;(\mathtt{pred}\;3)))\\
 & = & \mathtt{mult}\;3\;((\mathbf{Y}\,\mathtt{fac}')\;2)\\
 & = & \mathtt{mult}\;3\;(\mathtt{fac}'\;(\mathbf{Y}\,\mathtt{fac}')\;2)\\
 & = & \mathtt{mult}\;3\;(\mathtt{mult}\;2\;((\mathbf{Y}\,\mathtt{fac}')\;1))\\
 & = & \mathtt{mult}\;3\;(\mathtt{mult}\;2\;(\mathtt{mult}\;1\;((\mathbf{Y}\,\mathtt{fac}')\;0)))\\
 & = & \mathtt{mult}\;3\;(\mathtt{mult}\;2\;(\mathtt{mult}\;1\;1))\\
 & = & 6
\end{array}
$$

План лекции

Асимметрия β-конверсии

Мы строили λ-исчисление как теорию о равенстве термов.

А как можно было бы доказать неравенство?

Асимметрия β-конверсии

Мы строили λ-исчисление как теорию о равенстве термов.
А как можно было бы доказать неравенство?

Пример

$$\mathbf{K}\,\mathbf{I} \;\equiv\; (\lambda x y.\, x)\,(\lambda z.\, z) \;=\; \lambda y z.\, z$$
$$\mathbf{I}\,\mathbf{I}\,\mathbf{K}_* \;\equiv\; (\lambda x.\, x)\,\mathbf{I}\,\mathbf{K}_* \;=\; \mathbf{I}\,\mathbf{K}_* \;\equiv\; (\lambda x.\, x)\,(\lambda y z.\, z) \;=\; \lambda y z.\, z$$

Видно, что процесс носит односторонний характер: термы при конверсиях «упрощаются». Для исследования подобного вычислительного аспекта вводят понятие ***редукции***:

- $\mathbf{K}\,\mathbf{I} \;\to_\beta\; \mathbf{K}_*$ — редуцируется за один шаг;
- $\mathbf{I}\,\mathbf{I}\,\mathbf{K}_* \;\twoheadrightarrow_\beta\; \mathbf{K}_*$ — редуцируется;
- $\mathbf{K}\,\mathbf{I} \;=_\beta\; \mathbf{I}\,\mathbf{I}\,\mathbf{K}_*$ — конвертируемо (равно).

Редексы

Определение

Терм вида $(\lambda x.\, M)\, N$ называется β-***редексом***.

Определение

Терм $M[x := N]$ называется ***сокращением*** редекса $(\lambda x.\, M)\, N$.

Пример

Терм $\mathbf{I}\ (\mathbf{K}\ \mathbf{I})$ содержит два редекса

$$(\lambda x.\, x)\, ((\lambda y\, z.\, y)\, (\lambda p.\, p))$$
$$(\lambda x.\, x)\, ((\lambda y\, z.\, y)\, (\lambda p.\, p))$$

Может ли сокращение увеличить число редексов?

Понятие редукции

Определение

Бинарное отношение $\mathcal{R}$ над Λ называют ***совместимым*** (с операциями λ-исчисления), если для любых $M, N, Z \in \Lambda$:

$$\begin{aligned} M\,\mathcal{R}\,N \quad \Rightarrow \quad & (Z\,M)\,\mathcal{R}\,(Z\,N), \\ & (M\,Z)\,\mathcal{R}\,(N\,Z), \\ & (\lambda x\,.\,M)\,\mathcal{R}\,(\lambda x\,.\,N). \end{aligned}$$

Определение

Совместимое отношение эквивалентности называют отношением ***конгруэнтности*** над Λ.

Определение

Совместимое, рефлексивное и транзитивное отношение называют отношением ***редукции*** над Λ.

Редукция за один шаг $\to_\beta$

Определение

Бинарное отношение β-***редукции за один шаг*** $\to_\beta$ над Λ:

$$
\begin{array}{rcl}
(\lambda x.\, M)\, N & \to_\beta & M[x := N] \\
M \to_\beta N & \Rightarrow & Z\, M \to_\beta Z\, N \\
M \to_\beta N & \Rightarrow & M\, Z \to_\beta N\, Z \\
M \to_\beta N & \Rightarrow & \lambda x.\, M \to_\beta \lambda x.\, N
\end{array}
$$

По определению $\to_\beta$ является совместимым (с операциями λ-исчисления).

Пример: редуцируем терм $\mathbf{I}\ (\mathbf{K}\ \mathbf{I})$

$$
\begin{array}{rcccl}
(\lambda x.\, x)\, ((\lambda y\, z.\, y)\, (\lambda p.\, p)) & \to_\beta & (\lambda y\, z.\, y)\, (\lambda p.\, p) & \to_\beta & \lambda z\, p.\, p \\
(\lambda x.\, x)\, ((\lambda y\, z.\, y)\, (\lambda p.\, p)) & \to_\beta & (\lambda x.\, x)\, (\lambda z\, p.\, p) & \to_\beta & \lambda z\, p.\, p
\end{array}
$$

Многошаговая редукция $\twoheadrightarrow_\beta$

Определение

Бинарное отношение β-***редукции*** $\twoheadrightarrow_\beta$ над Λ (индуктивно):

$$\begin{array}{ll}
(a) & M \twoheadrightarrow_\beta M \\
(b) & M \to_\beta N \;\Rightarrow\; M \twoheadrightarrow_\beta N \\
(c) & M \twoheadrightarrow_\beta N, N \twoheadrightarrow_\beta L \;\Rightarrow\; M \twoheadrightarrow_\beta L
\end{array}$$

Отношение $\twoheadrightarrow_\beta$ является транзитивным рефлексивным замыканием $\to_\beta$ и, следовательно, отношением редукции.

Примеры

$$\begin{array}{lll}
(\lambda x.\, x)\,((\lambda y\, z.\, y)\,(\lambda p.\, p)) & \twoheadrightarrow_\beta & (\lambda x.\, x)\,((\lambda y\, z.\, y)\,(\lambda p.\, p)) \\
(\lambda x.\, x)\,((\lambda y\, z.\, y)\,(\lambda p.\, p)) & \twoheadrightarrow_\beta & (\lambda y\, z.\, y)\,(\lambda p.\, p) \\
(\lambda x.\, x)\,((\lambda y\, z.\, y)\,(\lambda p.\, p)) & \twoheadrightarrow_\beta & \lambda z\, p.\, p
\end{array}$$

Отношение конвертируемости $=_\beta$

Определение

Бинарное отношение $=_\beta$ над Λ (индуктивно):

$$\begin{array}{ll} (a) & M \twoheadrightarrow_\beta N \;\Rightarrow\; M =_\beta N \\ (b) & M =_\beta N \;\Rightarrow\; N =_\beta M \\ (c) & M =_\beta N, N =_\beta L \;\Rightarrow\; M =_\beta L \end{array}$$

Отношение $=_\beta$ является отношением конгруэнтности.

Утверждение

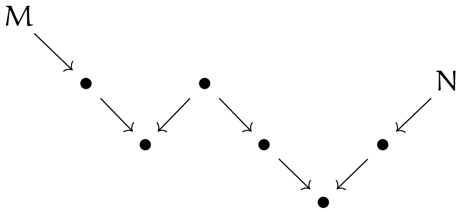
$$M =_\beta N \quad \Leftrightarrow \quad \lambda \vdash M = N.$$

Доказательство

Индукция по определениям. ∎

Отношение конвертируемости $=_\beta$ (интуитивно)

Интуитивно: два терма M и N связаны отношением $=_\beta$, если есть связывающая их цепочка $\to_\beta$-стрелок:



Пример. $\mathbf{K}\,\mathbf{I} \;=_\beta\; \mathbf{I}\,\mathbf{I}\,\mathbf{K}_*$

$$\begin{tikzcd}
 & & & \mathbf{I}\,\mathbf{I}\,\mathbf{K}_* \arrow[dl] \\
\mathbf{K}\,\mathbf{I} \arrow[dr] & & \mathbf{I}\,\mathbf{K}_* \arrow[dl] & \\
 & \mathbf{K}_* & &
\end{tikzcd}$$

Нормальная форма

Определение

λ-терм M находится в β-***нормальной форме*** (β-NF), если в нем нет подтермов, являющихся β-редексами.

Определение

λ-терм M имеет β-нормальную форму, если для некоторого N выполняется $M =_\beta N$ и N находится в β-NF.

Примеры

- Терм $\lambda x\, y.\, x\, (\lambda z.\, z\, x)\, y$ находится в β-нормальной форме.
- Терм $(\lambda x.\, x\, x)\, y$ не находится в β-нормальной форме, но имеет в качестве β-nf терм $y\, y$.

Утверждение

Не все термы имеют β-нормальную форму.

Пример

$$
\begin{array}{rcl}
\boldsymbol{\Omega} & \equiv & \boldsymbol{\omega}\,\boldsymbol{\omega} \\
 & \equiv & (\lambda x.\, x\, x)\,(\lambda x.\, x\, x) \\
 & \to_\beta & (\lambda x.\, x\, x)\,(\lambda x.\, x\, x) \\
 & \to_\beta & \ldots
\end{array}
$$

Это пока не доказательство! Может быть существует терм N в β-NF, такой что $\boldsymbol{\Omega} =_\beta N$, например, так

$$
\begin{tikzcd}
 & M \arrow[dl, two heads] \arrow[dr, two heads] & \\
\boldsymbol{\Omega} & & N
\end{tikzcd}
$$

Нормальная форма (3)

Бывают термы, «удлинняющиеся» при редукции.

Пример

$$\begin{array}{rcl}
\Omega_3 & \equiv & \omega_3\,\omega_3 \\
 & \equiv & (\lambda x.\, x\,x\,x)\,(\lambda x.\, x\,x\,x) \\
 & \to_\beta & (\lambda x.\, x\,x\,x)\,(\lambda x.\, x\,x\,x)\,(\lambda x.\, x\,x\,x) \\
 & \to_\beta & (\lambda x.\, x\,x\,x)\,(\lambda x.\, x\,x\,x)\,(\lambda x.\, x\,x\,x)\,(\lambda x.\, x\,x\,x) \\
 & \to_\beta & \ldots
\end{array}$$

С какой скоростью будет расти $\Omega_4 \equiv \omega_4\,\omega_4$?

Нормальная форма (4)

Не все последовательности редукций приводят к β-NF.

Пример

$$
\begin{aligned}
\mathbf{K}\,\mathbf{I}\,\mathbf{\Omega} &\equiv \mathbf{K}\,\mathbf{I}\,((\lambda x.\,x\,x)\,(\lambda x.\,x\,x)) \\
&\to_\beta \mathbf{K}\,\mathbf{I}\,((\lambda x.\,x\,x)\,(\lambda x.\,x\,x)) \\
&\to_\beta \ldots
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{K}\,\mathbf{I}\,\mathbf{\Omega} &\equiv (\lambda x\,y.\,x)\,\mathbf{I}\,\mathbf{\Omega} \\
&\to_\beta (\lambda y.\,\mathbf{I})\,\mathbf{\Omega} \\
&\to_\beta \mathbf{I}
\end{aligned}
$$

(синим отмечен сокращаемый редекс)

Редукционные графы (1)

Определение

Редукционный граф терма $M \in \Lambda$ (обозначаемый $G_\beta(M)$) — это ориентированный мультиграф с вершинами в $\{N \mid M \twoheadrightarrow_\beta N\}$ и дугами $\to_\beta$.

$$G_\beta\,(\mathbf{I}\,(\mathbf{I}\,x)) \;=\; \begin{tikzcd} \bullet \arrow[r, bend left] \arrow[r, bend right] & \bullet \arrow[r] & \color{blue}\bullet \end{tikzcd} \qquad G_\beta\,(\mathbf{\Omega}) \;=\; \begin{tikzcd} \bullet \arrow[loop above] \end{tikzcd}$$

$$G_\beta\,((\lambda x.\,\mathbf{I})\,\mathbf{\Omega}) \;=\; \begin{tikzcd} \bullet \arrow[loop above] \arrow[r] & \color{blue}\bullet \end{tikzcd} \qquad G_\beta\,(\mathbf{K}\,\mathbf{I}\,\mathbf{\Omega}) \;=\; \begin{tikzcd} \bullet \arrow[loop above] \arrow[r] & \bullet \arrow[loop above] \arrow[r] & \color{blue}\bullet \end{tikzcd}$$

$$G_\beta\,(\mathbf{\Omega_3}) \;=\; ??? \qquad G_\beta\,((\lambda x.\,\mathbf{I})\,\mathbf{\Omega_3}) \;=\; ???$$

Редукционные графы (2)

Не все редукционные графы конечны.

Пример

$$G_\beta\,(\boldsymbol{\Omega}_3) \;=\; \bullet \longrightarrow \bullet \longrightarrow \bullet \longrightarrow \bullet \longrightarrow \dots$$

Не все бесконечные редукционные графы не имеют нормальной формы.

Пример

$$\begin{tikzcd} G_\beta\,((\lambda x.\,\mathbf{I})\,\boldsymbol{\Omega}_3) \;=\; & \bullet \arrow[r] \arrow[d] & \bullet \arrow[r] \arrow[dl] & \bullet \arrow[r] \arrow[dll] & \bullet \arrow[r] \arrow[dlll] & \dots \\ & \color{blue}\bullet & & & & \end{tikzcd}$$

План лекции

Теорема Чёрча-Россера

Теорема [Чёрч-Россер]

Если $M \twoheadrightarrow_\beta N$, $M \twoheadrightarrow_\beta K$, то существует L, такой что $N \twoheadrightarrow_\beta L$ и $K \twoheadrightarrow_\beta L$.

- Иначе говоря, β-редукция обладает ***свойством ромба***:

$$\begin{tikzcd} & M \arrow[dl, two heads] \arrow[dr, two heads] & \\ N \arrow[dr, two heads, dotted] & & K \arrow[dl, two heads, dotted] \\ & L & \end{tikzcd}$$

- Иногда используют термин ***конфлюентность***.

Следствия теоремы Чёрча-Россера (1)

Теорема о существовании общего редукта

Если $M =_\beta N$, то существует L, такой что, $M \twoheadrightarrow_\beta L$ и $N \twoheadrightarrow_\beta L$.

Доказательство (индукция по генерации $=_\beta$)

- $M =_\beta N$, поскольку $M \twoheadrightarrow_\beta N$. Возьмем $L \equiv N$.
- $M =_\beta N$, поскольку $N =_\beta M$. По гипотезе индукции имеется общий β-редукт L_1 для N, M. Возьмем $L \equiv L_1$.
- $M =_\beta N$, поскольку $M =_\beta N'$, $N' =_\beta N$. Тогда

$$\begin{tikzcd}
M \arrow[dr, ->>] & (IH) & N' \arrow[dl, ->>] \arrow[dr, ->>] & (IH) & N \arrow[dl, ->>] \\
 & L_1 \arrow[dr, dotted, ->>] & CR & L_2 \arrow[dl, dotted, ->>] & \\
 & & L & &
\end{tikzcd}$$

$\blacksquare$

Теорема [Редуцируемость к NF]

Если M имеет N в качестве β-NF, то $M \twoheadrightarrow_\beta N$.

Теперь мы можем доказать отсутствие NF у $\boldsymbol{\Omega}$. Иначе выполнялось бы

$$\boldsymbol{\Omega} \twoheadrightarrow_\beta N, \qquad N \text{ является } \beta\text{-NF}.$$

Но $\boldsymbol{\Omega}$ редуцируется лишь к себе и не является β-NF.

Теорема [Единственность NF]

λ-терм имеет не более одной β-NF.

Теперь мы можем доказывать «неравенства», например

$$\lambda \nvdash \mathtt{tru} = \mathtt{fls}.$$

Иначе было бы $\mathtt{tru} =_\beta \mathtt{fls}$, но это две разные NF, что противоречит единственности.

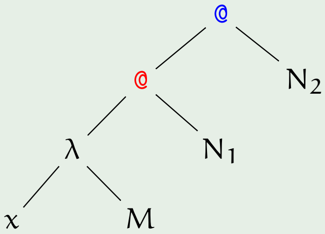
План лекции

Стратегии редукции

- Как мы можем редуцировать терм?
  - Переменная: v — редукция завершена.
  - Абстракция: $\lambda x.\, M$ — редуцируем M.
  - Аппликация: $M\, N$. Все варианты отсюда.
- Разбираем аппликацию до не-аппликации (обычно влево)
  - $(\ldots((v\, N_1)\, N_2)\ \ldots\ N_k)$ — редуцируем отдельно все N_i (обычно слева направо).
  - $(\ldots(((\lambda x.\, M)\, N_1)\, N_2)\ \ldots\ N_k)$. Все варианты отсюда.
- ***Нормальная стратегия:*** сокращаем редекс $(\lambda x.\, M)\, N_1$.
- ***Аппликативная стратегия:*** редуцируем отдельно все N_i (обычно слева направо) до нормальной формы N_i', затем сокращаем редекс $(\lambda x.\, M)\, N_1'$.

Синтаксические деревья для λ-термов

Пример: дерево для терма $((\lambda x.\, M)\, N_1)\, N_2$



- Узлы @ задают аппликацию, «узлы» λ — абстракцию.
- Узлы @ могут задавать редекс (@) или нет (@).
- В первом случае при поиске редекса – кандидата на сокращение есть три варианта (нашли, влево, вправо), во втором — два (влево, вправо).

Аппликативная структура терма

Теорема

Лямбда-терм может иметь одну из двух форм:

$$
\begin{array}{rcl}
\lambda \overrightarrow{x}.\, y\, \overrightarrow{N} & \equiv & \lambda x_1 \ldots x_n.\, y\, N_1 \ldots N_k, \quad n \geqslant 0, k \geqslant 0 \\
\lambda \overrightarrow{x}.\, (\lambda z.\, M)\, \overrightarrow{N} & \equiv & \lambda x_1 \ldots x_n.\, (\lambda z.\, M)\, N_1 \ldots N_k, \quad n \geqslant 0, k > 0
\end{array}
$$

Определение

Первая форма называется ***головной нормальной формой*** (HNF). Переменная y называется ***головной переменной***, а редекс $(\lambda z.\, M)\, N_1$ — ***головным редексом***.

Переменная y может совпадать с одной из x_i.

Определение

Слабая головная нормальная форма (WHNF) — это HNF или лямбда-абстракция, то есть *не редекс на верхнем уровне*.

Синтаксические категории:

- Нормальные формы $\mathtt{NF} \ ::= \ \lambda x.\, \mathtt{NF} \mid \mathtt{NANF}$
- Нормальные формы (не абстракции) $\mathtt{NANF} \ ::= \ v \mid \mathtt{NANF}\ \mathtt{NF}$
- Не абстракции $\mathtt{NA} \ = \ v \mid P\, Q$

Операционная семантика *нормальной* стратегии:

$$\frac{\mathtt{NA} \ \to \ \mathtt{NA}'}{\mathtt{NA}\ N \ \to \ \mathtt{NA}'\ N} \ (\text{Аппл1}) \qquad \frac{N \ \to \ N'}{\mathtt{NANF}\ N \ \to \ \mathtt{NANF}\ N'} \ (\text{Аппл2})$$

$$\frac{M \ \to \ M'}{\lambda x.\, M \ \to \ \lambda x.\, M'} \ (\text{Абстр}) \qquad (\lambda x.\, M)\, N \ \to \ M[x := N] \ (\text{Редук})$$

Нормальная стратегия всегда сокращает самый левый внешний редекс (leftmost outermost).

Операционная семантика аппликативной стратегии

Синтаксические категории:

- Нормальные формы $\mathtt{NF} ::= \lambda x.\,\mathtt{NF} \mid \mathtt{NANF}$
- Нормальные формы (не абстракции) $\mathtt{NANF} ::= v \mid \mathtt{NANF}\ \mathtt{NF}$
- Не абстракции $\mathtt{NA} = v \mid P\,Q$

Операционная семантика *аппликативной* стратегии:

$$\frac{\mathtt{NA} \to \mathtt{NA}'}{\mathtt{NA}\ N \to \mathtt{NA}'\ N}\ (\text{Аппл1}) \qquad \frac{N \to N'}{\mathtt{NANF}\ N \to \mathtt{NANF}\ N'}\ (\text{Аппл2})$$

$$\frac{M \to M'}{\lambda x.\,M \to \lambda x.\,M'}\ (\text{Абстр}) \qquad (\lambda x.\,M)\,\mathtt{NF} \to M[x := \mathtt{NF}]\ (\text{Редук})$$

$$\frac{N \to N'}{(\lambda x.\,M)\ N \to (\lambda x.\,M)\ N'}\ (\text{Аппл3})$$

Аппликативная стратегия сокращает самый левый внутренний редекс (leftmost innermost).

Теорема о нормализации

Аппликативная vs нормальная стратегии

$$\begin{aligned} \mathbf{K}\,\mathbf{I}\,\boldsymbol{\Omega} &\equiv \mathbf{K}\,\mathbf{I}\,((\lambda x.\,x\,x)\,(\lambda x.\,x\,x)) \to_\beta \ldots \\ \mathbf{K}\,\mathbf{I}\,\boldsymbol{\Omega} &\equiv (\lambda x\,y.\,x)\,\mathbf{I}\,\boldsymbol{\Omega} \twoheadrightarrow_\beta \mathbf{I} \end{aligned}$$

Теорема о нормализации

Если терм M имеет нормальную форму, то последовательное сокращение самого левого внешнего редекса приводит к этой нормальной форме.

- То есть нормальная стратегия гарантированно нормализует нормализуемое.
- Можем доказывать отсутствие NF. Например, $\mathbf{K}\ \boldsymbol{\Omega}\ \mathbf{I}$.

Недостаток нормальной стратегии — возможная неэффективность, достоинство — не считает ничего «лишнего».

Нормальная vs аппликативная стратегии

- Пусть N — «большой» терм

  $$(\lambda x.\, F\, x\, (G\, x)\, x)\, \mathsf{N} \to_\beta F\, \mathsf{N}\, (G\, \mathsf{N})\, \mathsf{N}$$

  В процессе дальнейших редукций редексы в N придётся сокращать три раза.
- Зато в

  $$(\lambda x\, y.\, y)\, \mathsf{N} \to_\beta \lambda y.\, y$$

  нормальная стратегия не вычисляет N ни разу.
- Аппликативная стратегия в обоих примерах вычислит N один раз.

Стратегии редукции и ЯП

- Аппликативная стратегия похожа на стратегию вычислений («энергичную», eager) большинства языков программирования. Сначала вычисляются аргументы, затем происходит применение функции.
- Нормальная стратегия похожа на способ вычисления в «ленивых» (lazy) языках (Haskell, Clean).
- Для решения проблем с эффективностью в «ленивых» языках используют *механизм разделения* (через вычисления в контекстах или через редукцию на графах).

Стратегии редукции и ЯП (2)

- Нет необходимости всегда доводить редукцию до NF. На практике часто ограничиваются WHNF.
- Это позволяет избежать захвата переменной в замкнутом терме (почему?)
- При наличии констант (в расширенных системах) понятие WHNF (и HNF) дополняют частично применёнными константными функциями, например

$$\mathbf{and}\ \mathbf{true}$$

поскольку его можно записать в η-эквивалентном WHNF-виде

$$\lambda x.\, \mathbf{and}\ \mathbf{true}\ x$$

Стратегии редукции и ЯП (3)

- ***Механизм вызова*** — термин, применяемый при исследовании высокоуровневых языков программирования.
- В функциональных языках:
  - «вызов по значению» — аппликативный порядок редукций до WHNF;
  - «вызов по имени» — нормальный порядок редукций до WHNF;
  - «вызов по необходимости» — «вызов по имени» плюс разделение.