Функциональное программирование
Лекция 2. Рекурсия и редукция

Денис Николаевич Москвин

Computer Science Center
Новосибирск

01.02.2020

План лекции

План лекции

1. Теорема о неподвижной точке
2. Редексы и нормальная форма
3. Теорема Чёрча-Россера
4. Стратегии редукции

Термовые уравнения

Схема β-преобразования

$$(\lambda\, n.\, M)\, N \;=\; [n \mapsto N]\, M$$

даёт возможность решать простейшие уравнения на термы.

Пример

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

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

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

Схема β-преобразования

$$(\lambda n.\, M)\, N \;=\; [n \mapsto N]\, M$$

даёт возможность решать простейшие уравнения на термы.

Пример

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

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

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:
$\forall \mathrm{F} \in \Lambda \quad \exists \mathrm{X} \in \Lambda \qquad \lambda \vdash \mathrm{F}\,\mathrm{X} = \mathrm{X}$

Доказательство

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

Теорема

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

Доказательство

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

Y-комбинатор позволяет ввести рекурсию в λ-исчисление.

Пример

Факториал рекурсивно:

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

Переписываем в виде

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

Отсюда видно, что $\mathtt{fac}$ — неподвижная точка для вспомогательной функции $\mathtt{fac}'$:

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

Работа $\mathbf{Y}$-комбинатора

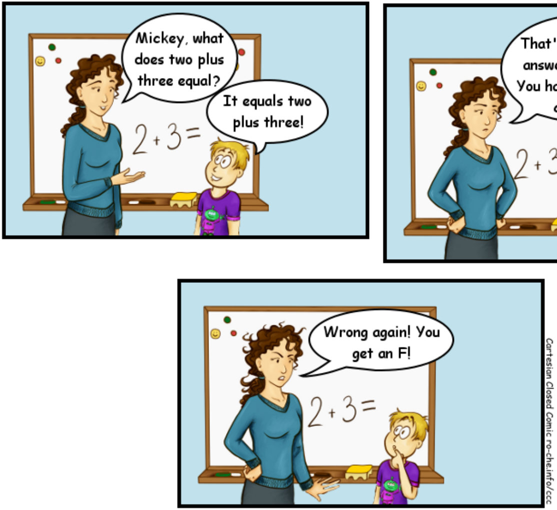
Как работает $\mathtt{fac} \equiv \mathbf{Y}\,\mathtt{fac}'$?

Пример

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

План лекции

Равенство и редукция



Sad fact: many math teachers do not know the difference between equality and reduction.

- Мы строили λ-исчисление как теорию о равенстве термов.
- Эффективный способ доказывать равенство — сократить все редексы:

Докажем, что $\mathbf{K}\ \mathbf{I}\ =\ \mathbf{I}\ \mathbf{I}\ \mathbf{K}_*$

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

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

- А как доказывать неравенство? Например, $\mathbf{K} \neq \mathbf{K}_*$?

Асимметрия β-преобразования

- Процесс вычисления по схеме β-преобразования

$$(\lambda x.\, M)\, N \;=_\beta\; [x \mapsto N]\, M$$

  носит односторонний характер: термы при конверсиях «упрощаются».
- Для исследования подобного вычислительного аспекта вводят ***отношение 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).
- Может ли шаг редукции увеличить число редексов в выражении?

Определение

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

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

Определения

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

$$(\lambda x.\, M)\, N \quad \to_\beta \quad [x \mapsto N]\, M$$

$$\begin{array}{lcl} M \to_\beta N & \Rightarrow & Z\, M \to_\beta Z\, N \\ M \to_\beta N & \Rightarrow & M\, Z \to_\beta N\, Z \\ M \to_\beta N & \Rightarrow & \lambda x.\, M \to_\beta \lambda x.\, N \end{array}$$

По определению $\to_\beta$ это β-правило плюс совместимость.

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

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

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

Определение

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

$$
\begin{array}{rcll}
 & & M \twoheadrightarrow_\beta M & \text{(refl)} \\
M \to_\beta N & \Rightarrow & M \twoheadrightarrow_\beta N & \text{(ini)} \\
M \twoheadrightarrow_\beta N, N \twoheadrightarrow_\beta L & \Rightarrow & M \twoheadrightarrow_\beta L & \text{(trans)}
\end{array}
$$

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

Примеры

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

Отношение конвертируемости $=_\beta$

Определение

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

$$\begin{array}{rcll} M \twoheadrightarrow_\beta N & \Rightarrow & M =_\beta N & \text{(ini)} \\ M =_\beta N & \Rightarrow & N =_\beta M & \text{(sym)} \\ M =_\beta N, N =_\beta L & \Rightarrow & M =_\beta L & \text{(trans)} \end{array}$$

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:

$$\begin{tikzcd}
M \arrow[dr] & & & & & & & \\
& \bullet \arrow[dr] & & \bullet \arrow[dl] \arrow[dr] & & & & N \arrow[dl] \\
& & \bullet & & \bullet \arrow[dr] & & \bullet \arrow[dl] & \\
& & & & & \bullet & &
\end{tikzcd}$$

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

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

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

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

Бывают термы, «удлинняющиеся» при редукции.

Пример

$$\begin{array}{rcl} \boldsymbol{\Omega}_3 & \equiv & \boldsymbol{\omega}_3\, \boldsymbol{\omega}_3 \\ & \equiv & (\lambda x.\, x\, x\, x)\, (\lambda x.\, x\, x\, x) \\ & \to_\beta & (\lambda x.\, x\, x\, x)\, (\lambda x.\, x\, x\, x)\, (\lambda x.\, x\, x\, x) \\ & \to_\beta & (\lambda x.\, x\, x\, x)\, (\lambda x.\, x\, x\, x)\, (\lambda x.\, x\, x\, x)\, (\lambda x.\, x\, x\, x) \\ & \to_\beta & \ldots \end{array}$$

С какой скоростью будет расти $\boldsymbol{\Omega}_4 \equiv \boldsymbol{\omega}_4\, \boldsymbol{\omega}_4$?

Нормальная форма (4)

Не все последовательности редукций приводят к β-NF.

Пример

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

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 $\to_\beta$.

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

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

$$G_\beta(\boldsymbol{\Omega}_3) \;=\; ??? \qquad G_\beta((\lambda x.\,\mathbf{I})\,\boldsymbol{\Omega}_3) \;=\; ???$$

Редукционные графы (2)

Не все редукционные графы конечны.

Пример

$$G_\beta\,(\mathbf{\Omega_3}) \;=\; \bullet \longrightarrow \bullet \longrightarrow \bullet \longrightarrow \bullet \longrightarrow \ldots$$

Не все бесконечные редукционные графы не имеют нормальной формы.

Пример

$$G_\beta\,((\lambda x.\,\mathbf{I})\,\mathbf{\Omega_3}) \;=\; \begin{tikzcd} \bullet \arrow[r] \arrow[d] & \bullet \arrow[r] \arrow[dl] & \bullet \arrow[r] \arrow[dll] & \bullet \arrow[r] \arrow[dlll] & \ldots \\ \color{blue}\bullet \end{tikzcd}$$

План лекции

1. Теорема о неподвижной точке
2. Редексы и нормальная форма
3. Теорема Чёрча-Россера
4. Стратегии редукции

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***.

Доказательство теоремы Чёрча-Россера

Доказываем лемму полоски (Strip lemma):

$$
\begin{tikzcd}
 & M \arrow[ld, "\beta"'] \arrow[rdd, two heads, "\beta"] & \\
N_1 \arrow[rdd, dotted, two heads, "\beta"'] & & \\
 & & K \arrow[ld, dotted, two heads, "\beta"] \\
 & L_1 &
\end{tikzcd}
$$

А затем из полосок составляем «ромб».

Термы *подкрашенного* лямбда-исчисления, множество которых мы будем обозначать Λ, определяются индуктивно:

$$\begin{array}{lcr} x \in V & \Rightarrow & x \in \Lambda, \\ M, N \in \Lambda & \Rightarrow & (M\,N) \in \Lambda, \\ M \in \Lambda, x \in V & \Rightarrow & (\lambda x.\,M) \in \Lambda, \\ M, N \in \Lambda, x \in V & \Rightarrow & (\lambda x.\,M)\,N \in \Lambda. \end{array}$$

То есть в редексах (и только в них) некоторые лямбды могут быть покрашены.

Подкрашенные редукции (одношаговые и многошаговые) определяются стандартным образом на базе правил сокращения:

$$\begin{array}{l} (\lambda x.\,M)\,N \to_\beta [x \mapsto N]\,M, \\ (\lambda x.\,M)\,N \to_\beta [x \mapsto N]\,M. \end{array}$$

То есть вычисления игнорируют раскраску, но не стирают ее.

Операции над раскрасками

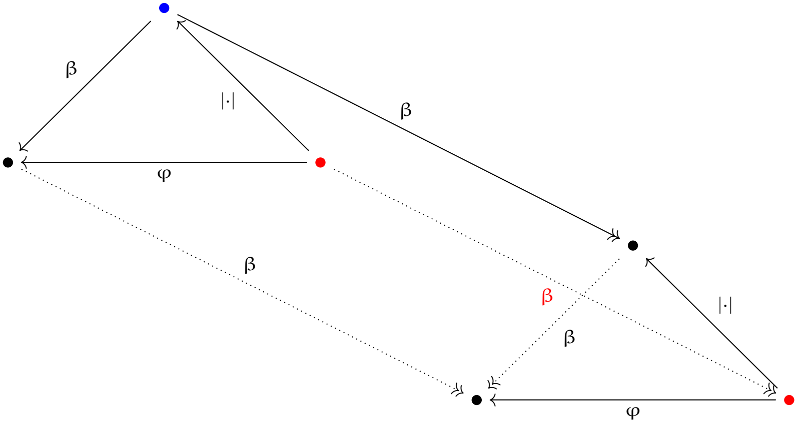
- Вводится операция *стирания подкраски*. Если $M \in \Lambda$, то $|M| \in \Lambda$ получается заменой в M всего красного на черное.
- Задается операция $\varphi : \Lambda \to \Lambda$, заключающаяся в сокращении всех подкрашенных редексов изнутри наружу:

$$
\begin{array}{lcl}
\varphi(x) & = & x, \\
\varphi((\lambda x.\, M)\, N) & = & [x \mapsto \varphi(N)]\, \varphi(M), \\
\varphi(M\, N) & = & \varphi(M)\, \varphi(N), \\
\varphi(\lambda x.\, M) & = & \lambda x.\, \varphi(M).
\end{array}
$$

- Общая идея доказательства: пока жив, покрашенный редекс остается редексом.
- Он может размножится, исчезнуть (по внешним причинам или сократившись), M и N могут измениться, но редекс не может «развалиться».

Доказательство леммы полоски

Три леммы: четырехугольники с красным и нижний треугольник.



Лемма полоски — грань без красного. ■

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

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

■

Теорема [Редуцируемость к NF]

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

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

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

Но $\mathbf{\Omega}$ 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.

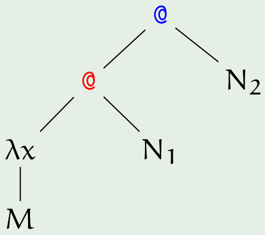
План лекции

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, то есть в M):
  - $(\ldots((v\, N_1)\, N_2)\, \ldots\, N_k)$ — 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 $(\lambda x.\, M)\, N_1'$.
- Иногда аппликативной называют стратегию, когда только N_1 редуцируется до нормальной формы.

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:

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

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*.

Нормальная vs аппликативная стратегии

$$\mathbf{K}\,\mathbf{I}\,\mathbf{\Omega} \;\equiv\; (\lambda x\,y.\,x)\,\mathbf{I}\,\mathbf{\Omega} \twoheadrightarrow_\beta \mathbf{I}$$

$$\mathbf{K}\,\mathbf{I}\,\mathbf{\Omega} \;\equiv\; \mathbf{K}\,\mathbf{I}\,((\lambda x.\,x\,x)\,(\lambda x.\,x\,x)) \to_\beta \ldots$$

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, $\mathbf{K}\,\mathbf{\Omega}\,\mathbf{I}$.

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

Нормальная vs аппликативная стратегии

- Пусть N — «большой» терм

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

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

- Зато в

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

  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 при редуцировании *замкнутого* терма. (почему?)
- При наличии констант (в расширенных системах) понятие WHNF (и HNF) дополняют частично применёнными константными функциями, например

$$\mathbf{and\ true}$$

  поскольку его можно записать в η-эквивалентном WHNF-виде

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

- В Haskell к WHNF относят и конструктор данных, примененный полностью или частично.

Механизмы вызова

- ***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.