

## ЛИСТОК 1: $\lambda$ -исчисление как формальная система

**Упражнение 1.** Расставить скобки и нарисовать дерево для термов:

- |                                    |                                  |
|------------------------------------|----------------------------------|
| (1) $xyz(yx)$                      | (6) $(\lambda xyz . xz(yz))uvw$  |
| (2) $\lambda x . uxy$              | (7) $xx(xxx)x$                   |
| (3) $\lambda u . u(\lambda x . y)$ | (8) $vw(\lambda xy . vx)$        |
| (4) $(\lambda u . vu)zy$           | (9) $(\lambda xy . x)uv$         |
| (5) $ux(yz)(\lambda v . vy)$       | (10) $w(\lambda xyz . xz(yz))uv$ |

**Упражнение 2.** Вычислить подстановку:

- |  |                                 |
|--|---------------------------------|
| (1) $[(uv)/x](\lambda y . x(\lambda w . vwx))$           | (5) $[vw/x](x(\lambda y . yx))$ |
| (2) $[(\lambda y . xy)/x](\lambda y . x(\lambda x . x))$ | (6) $[vw/x](x(\lambda x . yx))$ |
| (3) $[(\lambda y . vy)/x](y(\lambda v . xv))$            | (7) $[ux/x](x(\lambda y . yx))$ |
| (4) $[(uv)/x](\lambda x . zy)$                           | (8) $[uy/x](x(\lambda y . yx))$ |

**Упражнение 3.** Выполнить редукцию к нормальной форме:

- |   |   |
|---|---|
| (1) $(\lambda x . xy)(\lambda u . vu)$        | (6) $(\lambda xyz . xz(yz))((\lambda xy . yx)u)$<br>$((\lambda xy . yx)v)w$ |
| (2) $(\lambda xy . yx)uv$                     | (7) $(\lambda xy . xyy)uv$  |
| (3) $(\lambda x . x(x(yz))x)(\lambda u . uv)$ | (8) $(\lambda xy . yx)(uv)zw$   |
| (4) $(\lambda x . xxy)(\lambda y . yz)$       | (9) $(\lambda x . xx)(\lambda x . xx)$                                      |
| (5) $(\lambda xy . xyy)(\lambda u . uyx)$     | (10) $(\lambda xyz . xz(yz))(\lambda uv . u)$                               |

**Упражнение 4.** Продемонстрировать три стратегии редукции для термов:

- (1)  $(\lambda x . x(x(yz))x)(\lambda u . uv)$
- (2)  $(\lambda xyz . xz(yz))((\lambda xy . yx)u)((\lambda xy . yx)v)w$
- (3)  $(\lambda x . y)((\lambda y . yy)(\lambda y . yy))$
- (4)  $(\lambda y . (\lambda x . x)y)((\lambda u . u)(\lambda v . v))$
- (5)  $(\lambda xy . (\lambda pq . p)yx)((\lambda z . zz)(\lambda z . zz))s$
- (6)  $(\lambda xy . x)t((\lambda x . xx)(\lambda x . xx))$

## Основные определения

**Определение 1.** Пусть задано счётное множество имён переменных  $\mathcal{V} = \{x, y, z, \dots\}$ . Множество  $\lambda$ -термов — это наименьшее множество, удовлетворяющее следующим условиям:

1. Все переменные  $\mathcal{V}$  являются  $\lambda$ -термами.
2. Если  $M$  и  $N$  —  $\lambda$ -термы, то  $(MN)$  — есть  $\lambda$ -терм, называемый *применением* (application).
3. Если  $x$  — переменная, а  $M$  —  $\lambda$ -терм, то  $(\lambda x. M)$  — есть  $\lambda$ -терм, называемый *абстракцией* (abstraction).

**Определение 2.** Множеством свободных переменных  $\lambda$ -терма  $M$  называется множество, обозначаемое  $FV(M)$  и определяемое так:

$$\begin{aligned} FV(x) &= \{x\}, \\ FV(MN) &= FV(M) \cup FV(N), \\ FV(\lambda x. M) &= FV(M) \setminus \{x\}. \end{aligned}$$

**Определение 3.** Операцией *подстановки терма  $N$  в терм  $P$  вместо переменной  $x$* , обозначаемой как  $[N/x]P$ , называется операция преобразования терма, выполняемая по правилам:

- (1)  $[N/x]x = N$ ,
- (2)  $[N/x]y = y$ , если  $x \neq y$ ,
- (3)  $[N/x](PQ) = ([N/x]P)([N/x]Q)$ ,
- (4)  $[N/x](\lambda x. P) = \lambda x. P$ ,
- (5)  $[N/x](\lambda y. P) = \lambda y. P$ , если  $x \notin FV(P)$ ,
- (6)  $[N/x](\lambda y. P) = \lambda y. ([N/x]P)$ , если  $x \in FV(P)$  и  $y \notin FV(N)$ ,
- (7)  $[N/x](\lambda y. P) = \lambda z. ([N/x]([z/y]P))$ , если  $x \in FV(P)$  и  $y \in FV(N)$ ,  
причем  $z \notin FV(NP)$ .

**Определение 4.**  $\beta$ -редукцией называется отношение, которое ставит в соответствие редексу  $(\lambda x. M)N$  терм  $[N/x]M$ . Терм вида  $(\lambda x. M)N$  называется *редексом* (от reducible expression — редуцируемое выражение).  $\lambda$ -терм, не содержащий редексов в качестве подтермов, называется *термом в нормальной форме*.

## Стратегии редукции

- (1) *нормальный порядок*: самый внешний, самый левый редекс.
- (2) *вызов по имени*: самый левый, самый внешний редекс, причём редукция внутри абстракций не производится.
- (3) *вызов по значению*: только самые внешние редексы, причём аргументы каждого такого редекса вычисляются раньше самого редекса.