Листок 2: λ-исчисление как язык программирования

Упражнение 1. Представить в виде λ-терма и выполнить редукцию к нормальной форме:

(1) $\otimes$ $2 + 3$

(2) if (`is_zero?` 1) then 1 else 2

(3) $\otimes$ `fst` $(1, 3)$

(4) $2 *$ (if `false` then 1 else 0)

Упражнение 2. Определить λ-терм, реализующий функцию `xor` (без использования `and`, `or`, `not`). Вычислить терм: `false xor true`.

Упражнение 3. Определить λ-терм, реализующий операцию $\leqslant$. Вычислить терм $1 \leqslant 2$.

Упражнение 4. Определить λ-терм, реализующий операцию возведения в степень. Вычислить терм: 3 `exp` 2.

Упражнение 5. Определить λ-терм, реализующий операцию умножения через операцию сложения. (*Указание*: используйте частичное применение функции сложения. Под функцией сложения можно понимать терм `plus` $= \lambda m\,.\,\lambda n\,.\,\ldots$, получающийся из определения $m + n = \ldots$)

Комбинаторы неподвижной точки

Упражнение 6. Определить λ-терм `sum`, вычисляющий сумму чисел от 1 до n. Вычислить терм: `sum` 3.

Упражнение 7. Найти терм F, такой что

$$Fxy = FxyF.$$

Упражнение 8. Определить λ-терм `div`, реализующий операцию целочисленного деления. Вычислить терм: 4 `div` 3.

Упражнение 9 ($\otimes$)**.** Определить λ-терм `gcd`, вычисляющий наибольший общий делитель двух чисел. Вычислить терм: 6 `gcd` 8.

Упражнение 10. Показать, что терм

$$\mathbf{Y}_{\text{Turing}} = AA, \text{ где } A = \lambda ux\,.\,x(uux),$$

является комбинатором неподвижной точки.

Основные определения

$\texttt{true} = \lambda x\,.\,\lambda y\,.\,x;\quad \texttt{false} = \lambda x\,.\,\lambda y\,.\,y.$

$$\begin{array}{rcl} 0 & = & \lambda x\,.\,\lambda y\,.\,y \\ 1 & = & \lambda x\,.\,\lambda y\,.\,xy \\ 2 & = & \lambda x\,.\,\lambda y\,.\,x(xy) \\ 3 & = & \lambda x\,.\,\lambda y\,.\,x(x(xy)) \\ & \cdots & \end{array}$$

$$\text{if } C \text{ then } E_1 \text{ else } E_2 = CE_1E_2.$$

$$\texttt{is\_zero?} = \lambda n\,.\,n\,(\lambda x\,.\,\texttt{false})\,\texttt{true}$$

$$\begin{array}{lcl} \texttt{not}\ P & = & \text{if } P \text{ then } false \text{ else } true, \\ P\ \texttt{and}\ Q & = & \text{if } P \text{ then } Q \text{ else } false, \\ P\ \texttt{or}\ Q & = & \text{if } P \text{ then } true \text{ else } Q. \end{array}$$

$$\texttt{succ} = \lambda n\,.\,\lambda x\,.\,\lambda y\,.\,x(nxy)$$

$$(E_1, E_2) = \lambda z\,.\,z\ E_1\ E_2.$$

$$\begin{array}{lcl} m+n & = & \lambda x\,.\,\lambda y\,.\,mx(nxy); \\ m*n & = & \lambda x\,.\,\lambda y\,.\,m(nx)y. \end{array}$$

$$\begin{array}{lcl} \texttt{fst}\ P & = & P\ \texttt{true}; \\ \texttt{snd}\ P & = & P\ \texttt{false}. \end{array}$$

$$\mathbf{Y}_{\mathrm{CR}} = \lambda f\,.\,AA, \text{ где } A = \lambda x\,.\,f(xx).$$