

3.18 (7) Построение комбинатора, возвращающего n -е простое число (для нумералов Чёрча).

Замечание: Выражение *Sub* см. в билете 3.15. Построение *Mod* вынесено в отдельный билет, который мы не взяли, но понимать как мы его построили наверное стоит.

Замечание: fn в конце названий некоторых термов это не аргументы - это обозначение того, что это вспомогательная функция (*Modfn*, *NoDivsfn*, *NthPrimefn*)

$$1. Inc = \lambda nfx.f(nfx)$$

$$\begin{aligned} Inc \bar{n} &= (\lambda nfx.f(nfx))\bar{n} = \lambda fx.f(\bar{n}fx) = \\ &= \lambda fx.f(\underbrace{(\lambda gy.g(g(\dots(gy))) \dots)}_{n \text{ раз}})fx) = \lambda fx.f(\underbrace{f(f(\dots(fx))) \dots}_{n \text{ раз}}) = \overline{n+1} \end{aligned}$$

$$2. False = \lambda xy.y; True = \lambda xy.x$$

$$Not = \lambda p.p False True \text{ (если } p, \text{ то выводим } False, \text{ иначе } True)$$

$$And = \lambda pq.pqr \text{ (если } p, \text{ то выводим } q, \text{ иначе - } p)$$

$$3. IsZero = \lambda n.n(\lambda x.False)True$$

$$IsZero \bar{0} = \bar{0}(\lambda x.False)True = (\lambda fx.x)(\lambda x.False)True = True$$

$$IsZero \overline{n+1} = (\lambda fx.f(\dots))(\lambda x.False)True = (\lambda x.False)(\dots) = False$$

$$4. GE = \lambda mn.IsZero(Sub n m) (\geq)$$

$$LT = \lambda mn.Not(GE m n) \text{ (less then, то есть } <)$$

$$IsEqual = \lambda mn.And(GE m n)(GE n m)$$

$$5. Modfn = \lambda fmn.(LT m n)m(f(Sub m n)n)$$

Смысл: если $m < n$, то выводим m , иначе считаем остаток от деления $m - n$ на n

$Mod = Y Modfn$ (Y - комбинатор неподвижной точки) - остаток от деления m на n

$IsDivisible = \lambda nm.IsZero(Mod n m)$ (n кратно m)

6. Выразим терм *IsPrime*. Это индикатор того, что n не делится на числа $2, \dots, n-1$.

Выразим терм "число n не делится на числа $m, \dots, n-1$ "

$$NoDivsfn = \lambda fmn.(IsEqual n m)\bar{1}((IsDivisible n m)False(f(Inc m)))$$

$IsPrime = \lambda n.And(GE n \bar{2})(YNodivsfn\bar{2})$ (добавили условие ≥ 2 , так как наша вспомогательная функция не учитывала 0 и 1)

7. Перейдем к выражению искомого терма. Вспомогательный терм: идем по числам, рассмотрели k , нашли m простых

$$\begin{aligned} NthPrimefn &= \lambda fkm.(IsPrime k)((IsEqual n(Inc m))k(f(Inc k)(Inc m))(f(Inc k)m) \\ &\text{(} n \text{ свободная переменная, которая появится в итоговом терме, делаем } n \text{ как бы гло-} \\ &\text{бальным, чтобы не прокидывать во все уровни рекурсии)} \end{aligned}$$

Смысл: если k - простое, то если нашли все числа, возвращаем k , иначе переходим к следующим m, k и вычисляем ту же функцию. Если k не простое, то просто переходим к следующему k .

$$NthPrime = \lambda n.(YNthPrimefn)\bar{0}\bar{0}$$