

Proof

April 6, 2023

$$\nu = (0 \rightarrow 0) \rightarrow (0 \rightarrow 0)$$

Let's consider t - two-digit function and N, M - two numerals and f as general number function, then their types are:

$$\begin{aligned} t &: \nu \rightarrow \nu \rightarrow \nu \\ N, M &: \nu \\ f &: 0 \rightarrow 0 \end{aligned}$$

Then let's consider normal form of $t' = tNMf : 0 \rightarrow 0$

Important to notice that t is closed.

We will prove that any subterm of t' has one of types from set $S = \{0, 0 \rightarrow 0, \nu\}$.

Declare function for our comfort in usage T which returns type of expression in some context:

$$\forall e - \text{term}, \Gamma - \text{context} : T(\Gamma, e) = \tau, \text{ if } \Gamma \vdash e : \tau$$

For this let's introduce Lemma which will help us in it:

Lemma 0.1. *Consider some expression in normal form e in context Γ with such requirements:*

1. *Let's consider e has type τ then τ must be in set S*
2. *For each free variable $x \in FV(e)$ is true that $T(\Gamma, x) \in S$*

Then each subterm of e also has type which is in set S

Proof. We will prove this by induction on each step considering expression constructed by one of the possible ways.

Consider all possible deconstructions of expression e

1. $e = v_i$ where $v_i \in FV(e)$, it is a base case for induction
2. $e = \lambda x.B$ then $\tau = t_1 \rightarrow t_2$
3. $e = PQ$ then $\exists \alpha. P : \alpha \rightarrow \tau, Q : \alpha$

For case 1 the lemma is explicitly satisfied since e doesn't have any subterm, except itself so lemma is correct

For case 2 consider few options

Option 1:

$$\begin{cases} \tau &= 0 \rightarrow 0 \\ t_1 &= 0 \\ t_2 &= 0 \end{cases}$$

Then $FV(B) = FB(e) \cup \{x\}$ and because of B exists in context $\Gamma' = \Gamma, x : 0$ and has type 0 then by induction case is clear that each B 's subterm has correct type, but because if x is subterm B , then x is subterm of e .

Other option:

$$\begin{cases} \tau &= \nu \\ t_1 &= 0 \rightarrow 0 \\ t_2 &= 0 \rightarrow 0 \end{cases}$$

Same proof as for option 1.

For case 3 let's recall that e in normal form that means P - is not lambda (otherwise it would be reduced) and two options

1. $P \in FV(e)$ than the same variants for type τ as for case 2 with induction step for Q
2. $P = P_2Q_1$

We can repeat choosing option 2 finite number of times because we are in normal form.

Then consider

$$e = PA_1A_2...A_k, \text{ where } P \in FV(e)$$

Again few options:

$$\begin{cases} e &: 0 \\ P &: \nu \\ A_1 &: 0 \rightarrow 0 \\ A_2 &: 0 \end{cases}$$

$$\begin{cases} e &: 0 \\ P &: 0 \rightarrow 0 \\ A_1 &: 0 \end{cases}$$

$$\begin{cases} e &: 0 \rightarrow 0 \\ P &: \nu \\ A_1 &: 0 \rightarrow 0 \end{cases}$$

Again induction

□

Lemma 0.2. *Let t - two-digits closed form function ($t : \nu \rightarrow \nu \rightarrow \nu$), N, M - church numeral stands for some $n, m \geq 0$ and some $f : 0 \rightarrow 0$. Consider normal form of $t' = tNMf$. Then each subterm with type ν either be N or M .*

Proof. Because $t' : 0$ and context for it is $\Gamma_0 = \{N : \nu, M : \nu, f : 0 \rightarrow 0\}$ then lemma 0.1 applies to it.

Let's proof that for each sub-term of t' its context doesn't contain any variables (except for N, M) with type ν .

For induction base let's take Γ_0 .

Context can expand only on proofing type for subterm as $\lambda x.B$. Consider this subterm in some context Γ_k :

$$\Gamma_k \vdash \lambda x.B : \tau \rightarrow \beta \tag{1}$$

Then consider $\Gamma_{k+1} = \Gamma_k \cup \{x : \tau\}$ such that $\Gamma_{k+1} \vdash B : \beta$.

By lemma 0.1 $\tau \rightarrow \beta \in S$ (remind that S stands for $\{0, 0 \rightarrow 0, \nu\}$). Then we have two cases:

$$\begin{cases} \tau &= 0 \rightarrow 0 \\ \beta &= 0 \rightarrow 0 \end{cases}$$

$$\begin{cases} \tau &= 0 \\ \beta &= 0 \end{cases}$$

In both cases expanding of context doesn't add variables with type ν . By induction statement is proofed.

Lets take a look on where we can find such subterm.

Because t' is in normal form, it can't be left side of any application. Than only case when it is subterm is when it is one of argument of application as in cases 3 in lemma 0.1. There we showed that only type that it could be is $0 \rightarrow 0$. Hence only variables from context can have type ν . Because for each subterm, as we proofed, only N, M has that type, only possibility for subterm to have it is to be N or M .

- TODO add more accurate proof that any lam has type $0 \rightarrow 0$ and concrete
- that only f has type $0 \rightarrow 0$ in any Γ_k

□

Lemma 0.3. *Each subterm of t' from lemma 0.2 with type $0 \rightarrow 0$ is polynomial according to n, m or constant of polynomial. This means that for every s - subterm t' with type $0 \rightarrow 0$ exists $P(n, m)$ - polynomial such that s works same as $f^{P(n, m)}$ or $\lambda y.f^{P(n, m)}z$ ($z \neq y$)*

Proof. Consider possible ways how such subterm s could appear.

1. $s = v_i$ where $v_i \in FV(e)$, $v_i : 0 \rightarrow 0$
2. $s = PQ$ where $P : \nu, Q : 0 \rightarrow 0$
3. $s = \lambda x.B$ where $x : 0, B : 0$

Case 1: From lemma 0.2 only such v_i hence $s = f^1$

Case 2: By induction get that $Q = f^{P_1(n,m)}$ or $Q = \lambda y.f^{P_1(n,m)}z$.

P can be either N or M by same lemma.

Then consider both case for example for $P = N$:

Option 1:

$$s = PQ = (\lambda g.g^n)f^{P_1(n,m)} = f^{P_1(n,m) \cdot n}$$

Option 2:

$$s = PQ = (\lambda g.g^n)(\lambda y.f^{P_1(n,m)}z) = \lambda y.f^{P_1(n,m)}z \quad n \geq 1$$

$$s = PQ = (\lambda g.(\lambda x.x))(\lambda y.f^{P_1(n,m)}z) = \lambda x.x \quad n = 0$$

Case 3:

Because $B : 0$ it is either variable z or result of some function $s_1 : 0 \rightarrow 0$ on value $B_1 : 0$

In first option we get constant polynom lets continue choosing option two until it ends with variable.

Then:

$$s = \lambda y.s_1(s_2(\dots s_k(z) \dots))$$

Again by induction each s_i either $f^{P_i(n,m)}$ or $\lambda y.f^{P_i(n,m)}z_i$

Let's proof that chain of f_i with such condition suits for $f^{P'(n,m)}z'$ (z' can be any variable as y too)

Base 0:

$$f_0(z) = \begin{cases} (\lambda x.f^{P_0(n,m)}x)z = f^{P_0(n,m)}z & \text{if is not constant} \\ (\lambda x.f^{P_0(n,m)}z_0)z = f^{P_0(n,m)}z_0 & \text{otherwise} \end{cases}$$

Base 1:

$$f_1(f_0(z)) = \begin{cases} (\lambda x.f^{P_1(n,m)}x)((\lambda x.f^{P_0(n,m)}x)z) = f^{P_0(n,m)+P_1(n,m)}z & \text{if } f_0, f_1 \text{ are not constants} \\ (\lambda x.f^{P_1(n,m)}z_1)((\lambda x.f^{P_0(n,m)}x)z) = f^{P_1(n,m)}z_1 & \text{if } f_0 \text{ is not constant, } f_1 \text{ is constant} \\ (\lambda x.f^{P_1(n,m)}x)((\lambda x.f^{P_0(n,m)}z_0)z) = f^{P_0(n,m)}z_0 & \text{if } f_0 \text{ is constant, } f_1 \text{ is not constant} \\ (\lambda x.f^{P_1(n,m)}z_1)((\lambda x.f^{P_0(n,m)}z_0)z) = f^{P_1(n,m)}z_1 & \text{if } f_0, f_1 \text{ are constants} \end{cases}$$

Induction:

$$f_n(f_{n-1}(\dots f_0(z))) = \begin{cases} (\lambda x.f^{P_n(n,m)}x)(f^{P'(n,m)}z') = f^{P'(n,m)+P_n(n,m)}z' & \text{if } f_n \text{ is not constant} \\ (\lambda x.f^{P_n(n,m)}z_n)(f^{P'(n,m)}z') = f^{P_n(n,m)}z_n & \text{if } f_n \text{ is constant} \end{cases}$$

What means that s also suits for lemma

□

Lemma 0.4. t' - *polynomial*

Proof. Because t' is subterm for itself and has type $0 \rightarrow 0$ hence t' is polynom or constant polynom. But because it doesn't have any free variable with type 0 it is polynom. \square

That means that all closed functions resulting from polynomials are polynomials.