1 Syntax of lambda calculus

Assume a countable set of variable names, denoted by $a, b, c, ..., x, y, z, a_0, a_1, ...$

Definition. A **λ-term** is defined by the following context-free grammar:

$$\begin{aligned} < term > \quad &:= \quad < name > \\ &| \quad (\lambda < name > . < term >) \\ &| \quad (< term >< term >) \end{aligned}$$

Conventions.

1. **Function application** is left-associative, so $(((A_1A_2)A_3)...A_k)$ can be abbreviated as $A_1A_2A_3...A_k$
2. Nested **abstractions** $(\lambda x_1.(\lambda x_2.(...\lambda x_k.A)...))$ can be abbreviated as $\lambda x_1x_2...x_k.A$

Example.
$\lambda xy.FAB$ means $((\lambda x.(\lambda y.F))A)B$

2 Free variables

1. $< name >$ is free in $< name >$
2. $< name >$ is free in $\lambda < name' > . < term >$ if $< name > \neq < name' >$ and $< name >$ is free in $< term >$
3. $< name >$ is free in $< term' >< term'' >$ if $< name >$ is free in $< term' >$ or $< name >$ is free in $< term'' >$

3 Bound variables

1. $< name >$ is bound in $\lambda < name' > . < term >$ if $< name > = < name' >$ or $< name >$ is bound in $< term >$
2. $< name >$ is bound in $< term' >< term'' >$ if $< name >$ is bound in $< term' >$ or $< name >$ is bound in $< term'' >$

Examples. Free and bound occurrences of variables:
$(\lambda x.xy)(\lambda y.y)$

4 Reductions

Denote by $T[x := E]$ the term in which every free occurrence of x is replaced by E.

α-conversion: Bound variables can be renamed, $\lambda x.F \equiv \lambda t.F[x := t]$, provided

that t is not free in F and is not bound by a λ in F whenever it replaces an x.

β-reduction: Applying a function to the argument: $(\lambda x.F)A \equiv F[x := A]$ provided that all free occurrences in A remain free in $F[x := A]$.

Definition. A λ-term is in a normal form if no β-reduction can be applied to it.

Theorem. If a λ-term has a normal form, then the normal form is unique (up to renaming of bound variables).

Computing the normal form: Keep on replacing the leftmost bound variable by the actual argument until no further reduction is possible.
This does not terminate if and only if the initial term has no normal form.

5 Church numerals

The **Church numerals** $C_0, C_1, ...$ are defined as:

$$\begin{array}{lll} C_0 & \equiv & \lambda sz.z \\ C_1 & \equiv & \lambda sz.s(z) \\ C_2 & \equiv & \lambda sz.s(s(z)) \\ ... & ... & ... \end{array}$$

The successor can be defined as:

$$S \equiv \lambda uvw.v(uvw)$$

Exercise. Verify that $SC_n = C_{n+1}$ for every $n \in \mathbb{N}$

Lemma. For every two terms F and A, $C_n F A = F^{(n)} A$

Corollary. Doing addition in λ-calculus: $C_n S C_m = C_{n+m}$

6 Predecessor

Define $True$ and $False$ by:

$$\begin{array}{lll} T & \equiv & \lambda xy.x \\ F & \equiv & \lambda xy.y \end{array}$$

Represent a pair (a, b) by $\lambda z.zab$. Define:

$$\Phi = \lambda pz.z(S(pT))(pT)$$

The **predecessor** is defined as:

$$P \equiv \lambda n.n\Phi(\lambda z.zC_0C_0)F$$