Lambda Calculus: Natural Numbers - Worksheet

Carlo Allietti
spam@iscienceworld.com

Amateur losing his mind on — August 24, 2022

How do we encode numbers in lambda calculus?

It should be obvious why numbers are important to a formal mathematical/computational system. Numbers in lambda calculus are encoded as **Church numerals**:

$$\begin{aligned} 0 &:= \lambda f.\lambda x.x \\ 1 &:= \lambda f.\lambda x.fx \\ 2 &:= \lambda f.\lambda x.f(fx) \\ 3 &:= \lambda f.\lambda x.f(f(fx)) \\ &\vdots \end{aligned} \tag{1}$$

*note 0 is the same as FALSE.
With this we can define arithmetic operations:

1 Successorship

Successorship is getting the successor of a number, also known as counting by 1.

$$SUCC := \lambda n.\lambda f.\lambda x.f(nfx) \tag{2}$$

So for example, SUCC(2) is:

$$\begin{aligned} SUCC\ 2 = (\lambda n.\lambda f.\lambda x.f(nfx))\lambda f.\lambda x.f(fx) \rightarrow_\alpha \\ (\lambda n.\lambda a.\lambda b.a(nab))\lambda f.\lambda x.f(fx) \rightarrow_\beta \\ \lambda a.\lambda b.a((\lambda f.\lambda x.f(fx))ab) \rightarrow_\beta \\ \lambda a.\lambda b.a(a(ab)) = 3 \end{aligned} \tag{3}$$

2 Addition

Addition is repeated counting.

$$ADD := \lambda m.\lambda n.\lambda f.\lambda x.m\ f(nfx) \tag{4}$$

You can define all these operations in other ways too, like in this case replacing the f with $SUCC$.

3 Multiplication

Multiplication is repeated addition.

$$MULT := \lambda m.\lambda n.\lambda f.m(nf) \tag{5}$$

So when you β-reduce you get, $m(\lambda x.f^n x)$ and then $\lambda x.f^{n^m}x$.

4 Exponentiation

Exponentiation is repeated multiplication.

$$POW := \lambda b.\lambda e.eb \tag{6}$$

This definition is so simple because it ends up causing b to be applied to x, e times, which is like multiplying by b, e times.

5 Predecessor

The predecessor function returns $n-1$ for n, so it is the inverse of successorship. Its definition is much more convoluted.

$$PRED := \lambda n.\lambda f.\lambda x.n(\lambda g.\lambda h.h(gf))(\lambda u.x)(\lambda u.u) \tag{7}$$

If you were to work this lambda term out you would find that it reduces to $\lambda h.h(f^{n-1}x)$.

6 Subtraction

Subtraction is repeated predecessor function.

$$SUB := \lambda m.\lambda n.n\ PRED\ m \tag{8}$$

Keep in mind SUB and PRED only subtract until you reach 0, since negative numbers aren't encoded in these numerals and operations as we have defined them here.

7 Is zero

The ability to go from numbers to booleans is quite important, here is the most fundamental way of converting numbers to booleans:

$$ISZERO := \lambda n.n(\lambda x.FALSE)TRUE \tag{9}$$

ISZERO returns TRUE if 0, and FALSE otherwise.

8 An example

$$ADD\ (MULT\ 2\ 2)\ (SUB\ 5\ 3) \to_\beta ADD\ 4\ 2 \to_\beta 8 \tag{10}$$

Question 1 (Exercise for the reader.)

Find x such that:

$$(\lambda a.\lambda b.a)(ADD\ (POW\ x\ 2)\ (MULT\ 3\ x))(MULT\ 5\ x) \equiv \lambda f.\lambda g.f(f(f(fg))) \tag{11}$$