Homework 4 Solution: Lambda Calculus

Objectives

The objective is to get some training with the basic the reduction of lambda expressions. It is vital to pay attention to the proper parenthetization of your expression, so I encourage you to use enough space to spread your work and avoid simple mistakes. Finally, let me reemphasize that I am well aware of the existence of solutions to all the simple classic problems below online (e.g., wikipedia). Yet, I urge you to develop these expressions on your own to truly understand the mechanics.

Introduction

Recall that a Church numeral is an encoding of an integer in term of a lambda expression. Specifically, the first few numerals are

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

In general $\bar{k}$ contains k nested applications of the process x to the base case value y. Also, recall that the booleans true and false are defined as:

$$\begin{array}{rcl} T &=& \lambda x.\lambda y.x \\ F &=& \lambda x.\lambda y.y \end{array}$$

and act as *selectors* to choose one of their two inputs (true chooses the first and false the second).

Function design

Write a lambda expression to implement

- binary substraction: $a - b$ (you can assume $a \geq b$)

$$bPa$$

- binary multiplication: $a * b$

$$\begin{array}{l} \lambda uvx.u(vx))ab \\ \rightarrow \lambda vx.a(vx)b \\ \rightarrow \lambda x.a(bx) \end{array}$$

- binary exponentiation: a^b.

$$\begin{array}{l} a^b = a * a * \ldots a * a = \lambda z.a(a(a(\ldots(az)))) \\ \rightarrow (\lambda sz.s(s(s(\ldots sz))))a \\ \rightarrow ba \end{array}$$

Negative numbers

Fortunately, once you can represent *natural numbers*, integers are quite easy to represent. The idea is quite straightforward and simply consist of representing an integer by a pair of naturals. Namely, let the integer $z = (a, b)$ with $a \geq 0, b \geq 0$. For instance, the number 20 could be represented by $(20, 0)$. An alternative representation would be $(25, 5)$. Indeed, the number 20 can be seen as

$$20 = 20 - 0$$
$$20 = 25 - 5$$

Namely, $z = a - b$. The advantage arises for making the number negative. For instance, $-20 = (0, 20) = 0 - 20$. In other words, to negate an integer you only need to swap the two naturals! Thankfully, addition is equally simple as $z_1 + z_2$ where $z_1 = (a, b)$ and $z_2 = (c, d)$ is simply

$$z_1 + z_2 = a - b + c - d = a + c - b - d = a + c - (b + d) = (a + c, b + d)$$

To get a lambda expression implementing addition, it suffices to leverage the pair utilty alongside the addition over naturals (Church Numerals) to get a full fledged solution.

Implement the above ideas for the following functions

1. Lambda expression for unary negation of an integer

   $$\lambda z.\lambda x.x(zF)(zT)$$

   Let z be the pair (a, b), then $a = zT$ and $b = zF$. Here the pair is (b, a), which represents $b - a$, corresponding to $-z$.

2. Lambda expression for successor of an integer

   $$\lambda z.\lambda x.x(S(zT))(zF)$$

   Let z be the pair (a, b), then $a = zT$ and $b = zF$. $S(zT)$ gives us $a + 1$. The resulting pair is $(a + 1, b)$.

3. Lambda expression for predecessor of an integer

   $$\lambda z.\lambda x.x(zT)(S(zF))$$

   Let z be the pair (a, b), then $a = zT$ and $b = zF$. $S(zF)$ gives us $b + 1$. The resulting pair is $(a, b + 1)$.

4. Lambda expression for binary addition of integers

   $$\lambda u.\lambda v.\lambda x.x((uT)S(vT))((uF)S(vF))$$

   Let u and v be the pair (a, b), and (c, d), then $a = uT$, $b = uF$, $c = vT$ and $d = vF$. The resulting pair is $(a + c, b + d)$.

5. Lambda expression for binary subtraction of two integers

   $$\lambda u.\lambda v.\lambda x.x((uT)S(vF))((uF)S(vT))$$

   Let u and v be the pair (a, b), and (c, d), then $a = uT$, $b = uF$, $c = vT$ and $d = vF$. The resulting pair is $(a + d, b + c)$.

6. Lambda expression for binary multiplication of two integers

   $$\lambda u.\lambda v.\lambda x.x((\lambda a.(uT)(vT)a)S(\lambda a.((uF)(vF)a)))((\lambda a.(uF)(vT)a)S(\lambda a.(vT)(vF)a))$$

   Let u and v be the pair (a, b), and (c, d), then $a = uT$, $b = uF$, $c = vT$ and $d = vF$. The resulting pair is $(ac + bd, bc + ad)$.

Naturally, you should first follow the outline above and write the arithmetic specification of the operation before encoding it as a lambda expression.

Beta-Reductions

Solve the following, assuming that you are dealing with Church numerals (naturals).

1. $succ\ \bar{3}$

$$\begin{aligned}&\lambda nab.a(nab)\bar{3}\\ &\to \lambda ab.a(\bar{3}ab)\\ &\to \lambda ab.a(a(a(ab)))\\ &\to \bar{4}\end{aligned}$$

2. $isZero\ \bar{2}$

$$\begin{aligned}&\lambda x.xF\neg F\bar{2}\\ &\to \bar{2}F\neg F\\ &\to F(F\neg)F\\ &\to IF\\ &\to F\end{aligned}$$

3. $add\ \bar{2}\ \bar{3}$

$$\begin{aligned}&\bar{2}S\bar{3}\\ &\to S(S(\bar{3}))\\ &\to S(\bar{4})\\ &\to \bar{5}\end{aligned}$$

4. $sub\ \bar{3}\ \bar{1}$

$$\begin{aligned}&\bar{1}P\bar{3}\\ &\to P\bar{3}\\ &\to \bar{2}\end{aligned}$$

5. $mul\ \bar{2}\ \bar{3}$

$$\begin{aligned}&(\lambda xya.x(ya))\bar{2}\bar{3}\\ &\to \lambda ya.\bar{2}(ya)\bar{3}\\ &\to \lambda a.\bar{2}(\bar{3}a)\\ &\to \lambda a.(\lambda sb.s(sb))(\bar{3}a)\\ &\to \lambda ab.(\bar{3}a)((\bar{3}a)b))\\ &\to \lambda ab.(\bar{3}a)((a(a(a)b))))\\ &\to \lambda ab.a(a(a(a(a(ab)))))\\ &\to \bar{6}\end{aligned}$$