Types and Programming Languages, Exercise 2

Philip Wadler

Issued: Monday 28 January 2008. Due: Monday 4 February 2008.

TAPL refers to *Types and Programming Languages* by Benjamin Pierce.

1. Using your implementation of untyped lambda calculus with booleans and naturals from last week, write and test the following:

   (a) Addition, multiplication, and exponentiation on naturals. You will need to use the fixpoint combinator Y or Z to support recursion. Test your work by computing $2 + 3$, 2×3, and 2^3 as naturals.

   (b) Addition, multiplication, and exponentiation on Church numerals. Do not use the fixpoint combinator. Test your work by computing $2 + 3$, 2×3, and 2^3 as Church numerals.

   (c) Functions to convert a Church numeral to the corresponding natural, and vice versa. Test your work by converting 5 from a natural to a Church numeral and back again.

   Do as many additional tests as you see fit. Think about testing corner cases, such as 0 and 1.