Proving with Computer Assistance, 2IMF15

Herman Geuvers, TUE

Exercises on Polymorphic type Theory

1. Recall: $\bot := \forall\alpha.\alpha$, $\top := \forall\alpha.\alpha\to\alpha$.

    (a) Verify that in Church $\lambda 2$: $\lambda x{:}\top.x\top x : \top\to\top$.

    (b) Verify that in Curry $\lambda 2$: $\lambda x.xx : \top\to\top$

    (c) Find a type in Curry $\lambda 2$ for $\lambda x.x\,x\,x$

    (d) Find a type in Curry $\lambda 2$ for $\lambda x.(x\,x)(x\,x)$

    (e) Find a type in Curry $\lambda 2$ for $\lambda z.z(\lambda x.x\,x)$

2. Let $x : \top$ and remember that $\top := \forall\alpha{:}*.\alpha\to\alpha$.

    (a) Give a type to the term

    $$\lambda y.x\,y\;x(\lambda z.z\,x\,z)$$

    in $\lambda 2$ à la Curry and give the typing derivation of your result.

    (b) Give a type to the term

    $$\lambda y.x\,y\,(x(\lambda z.z\,z))$$

    in $\lambda 2$ à la Curry. Also give the typing derivation of your result.

3. Define: $\sigma\times\tau := \forall\alpha.(\sigma\to\tau\to\alpha)\to\alpha$,
$\sigma+\tau := \forall\alpha.(\sigma\to\alpha)\to(\tau\to\alpha)\to\alpha$
$\text{Tree}_{A,B} := \forall\alpha.(B\to\alpha)\to(A\to\alpha\to\alpha\to\alpha)\to\alpha$

    (a) Define $\text{inl} : \sigma\to\sigma+\tau$

    (b) Define pairing : $[-,-] : \sigma\to\tau\to\sigma\times\tau$

    (c) Show that the addition function (as defined on the slides) behaves as expected.

    (d) Define leaf : $B\to\text{Tree}_{A,B}$ and join : $\text{Tree}_{A,B}\to\text{Tree}_{A,B}\to A\to\text{Tree}_{A,B}$

    (e) Give the Tree-iteration scheme for $\text{Tree}_{A,B}$ and define $h : \text{Tree}_{A,B}\to\text{Nat}$ that counts the number of leaves of a tree.