Eindhoven University of Technology

Faculty of Mathematics and Computer Science

Written Exam **Proving with Computer Assistance 2IMF15**
Monday April 8, 2019, 13.30–16.30
The maximum number of points per exercise is indicated in the margin. Maxium 100 points in total.
NB Typing derivations may be given in “flag style” or in “sequent style”.

---

(10) 1. Fill in types for the ? in the following term P and show that it is typable in simple type theory ($\lambda\!\to$) à la Church with type $((\alpha \to \beta \to \beta) \to (\gamma \to \beta \to \gamma) \to \delta) \to \delta$.

$$\lambda f : ?_1 . f\,(\lambda x : ?_2 . \lambda y : ?_3 . y)\,(\lambda v : ?_4 . \lambda w : ?_5 . v)$$

Give a typing derivation.

(10) 2. Give a term M in simple type theory ($\lambda\!\to$) à la Church with type

$$(((\alpha \to \beta) \to \beta) \to \beta) \to \alpha \to \beta.$$

Give a typing derivation that gives the type of M.

(20) 3. Consider the following two terms

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

For each of these terms, compute its principle type in simple type theory ($\lambda\!\to$) à la Curry, if it exists.
Give the end result and show your computation; if the term has no principle type, show how your computation yields ‘fail’.

(10) 4. Let $x : \top$ and remember that $\top := \forall \alpha : * . \alpha \to \alpha$. Give a type to the term

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

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

Continue on the other side

5. In $\lambda 2$ à la Church we define the type of binary trees with natural numbers as leaves as follows.

$$\mathbb{T} \ := \ \forall \alpha{:}*.(\mathsf{N} \to \alpha) \to (\alpha \to \alpha \to \alpha) \to \alpha$$

Remember that $\mathsf{N} := \forall \alpha{:}*.\alpha \to (\alpha \to \alpha) \to \alpha$ is the type of natural numbers.

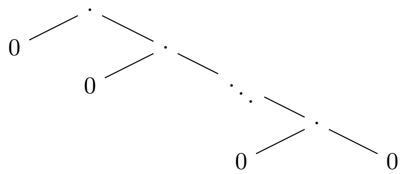
The data type $\mathbb{T}$ has 2 constructors:

$$\begin{array}{lllll} \mathsf{leaf} & : & \mathsf{N} \to \mathbb{T}, & \mathsf{leaf} & := \ \lambda n{:}\mathsf{N}.\lambda \alpha{:}*.\lambda \ell : \mathsf{N} \to \alpha.\lambda j : \alpha \to \alpha \to \alpha.\ell\, n, \\ \mathsf{join} & : & \mathbb{T} \to \mathbb{T} \to \mathbb{T} & \mathsf{join} & := \ \ldots \end{array}$$

(5) (a) Define the constructor $\mathsf{join} : \mathbb{T} \to \mathbb{T} \to \mathbb{T}$.

(8) (b) Define the function $\mathsf{leftmost} : \mathbb{T} \to \mathsf{N}$ that finds the left-most number in a binary tree.

(8) (c) Define the function $F : \mathsf{N} \to \mathbb{T}$ that, given a number n, builds a binary tree with $n + 2$ leaves, all containing 0, of the following shape.



(12) 6. In the system λP, give a term of type $\Pi x, y{:}A.R\, x\, (f\, y) \to R\, x\, (f\, x)$ in the context

$$\begin{array}{lll} \Gamma & := & A : *, R : A \to A \to *, f : A \to A, \quad s : \Pi x, y{:}A.R\, x\, (f\, y) \to R\, (f\, y)\, x, \\ & & t : \Pi x, y, z{:}A.R\, x\, y \to R\, y\, z \to R\, x\, (f\, z). \end{array}$$

That is, solve the following type inhabitation problem in λP:

$$\Gamma \vdash ? : \Pi x, y{:}A.R\, x\, (f\, y) \to R\, x\, (f\, x).$$

Also give a typing derivation in short version (where you don't have to establish the well-formedness of the types themselves).

7. Given $A : *$, $f : A \to A$ and $a : A$, we define, in the Calculus of Constructions λC,

$$Q := \lambda z{:}A.\, \Pi P{:}A \to *.P\, (f\, a) \to (\Pi y{:}A.P\, (f\, y) \to P\, y) \to P\, z$$

(9) (a) Give a term of type

$$Q\, a$$

(8) (b) Give a term of type

$$\Pi z{:}A.Q\, (f\, z) \to Q\, z.$$

NB. In both cases, give a typing derivation in short version (where you don't have to establish the well-formedness of the types themselves).

END