Christian-Albrechts-Universität zu Kiel
Institut für Informatik und Praktische Mathematik
Prof. Dr. W.-P. de Roever
Harald Fecher, Martin Steffen

Fool

Sommersemester 2005 — **Serie 6** — 10 June 2004

Thema: Recursive types

Ausgabetermin: 10 June 2004

Abgabe:

Aufgabe 1 (Subtyping recursive types) During the lecture, we had a "simplified" variant of S-Rec on the blackboard:

$$\frac{\Gamma \vdash \mu(X).S \qquad \Gamma \vdash \mu(X).T \qquad \Gamma, X \leq \mathit{Top} \vdash S \leq T}{\Gamma \vdash \mu(X)S \leq \mu(X)T}\ \text{S-Rec}$$

Is this rule "ok"? Prove this, or show that it's not ok.

Aufgabe 2 (*fold* und *unfold*) We tried to type a few simple programs in the *iso-recursive* approach and in the $\mathrm{Ob}_{1\leq\mu}$-calculus. The basic trick was to use the *unfolded* recursive type as type of the self-parameter, i.e, the methods all looked like

$$\varsigma(s : UT).\mathit{body}$$

where UT is the unrolled version of T, when T is the (recursive) type the method occurs in. Inside the body, the uses of s were in the example all first folded, i.e., $\mathit{fold}(T, s)$ was used.

Can you think of an example, where not only *fold*, but also *unfold* is used? I.e., starting from an untyped program, that the correct typing requires the addition of *unfold*?

Aufgabe 3 In [1, Section 9.2], the notions of variance (co-/contra-/in-variance) are re-iterated, this time using the newly introduced concept of type variables.

The text claims, that the subtyping rule for recursive type "determines the variance behavior of recursive types". Verify this claim.

Literatur

[1] Martín Abadi and Luca Cardelli. *A Theory of Objects*. Monographs in Computer Science. Springer, 1996.