

EB4 – Foundations of OOP

1. Subtyping

Exercise 1

- I. Provide a derivation that proves the subtyping judgement $\{x : \mathbb{N}, y : \mathbb{N}, z : \mathbb{N}\} <: \{y : \mathbb{N}\}$. Are there others?
- II. Provide at least two derivations of the subtyping judgement $\{x : \mathbb{N}, y : \mathbb{N}\} <: \{ \}$

Exercise 2

State whether the following are true or false:

- I. $\exists S \forall T T <: S$
- II. $\exists S \forall T S <: T$
- III. $\exists S_1, S_2 \forall T_1, T_2 S_1 \rightarrow S_2 <: T_1 \rightarrow T_2$
- IV. $\exists S_1, S_2 \forall T_1, T_2 T_1 \rightarrow T_2 <: S_1 \rightarrow S_2$

Exercise 3

Exhibit a term that is not typable in the λ -calculus, but is typable in $\lambda_{\leq}^{\rightarrow}$. Consider the type \mathbb{T} and the term xx .

Exercise 4

Determine whether the following are true or false. Prove it, if it is true, otherwise, provide a counterexample.

- I. $T <: S$ iff there exists A such that $S \rightarrow T <: A \rightarrow A$.
- II. $\{x : S, y : T\}$ always has less supertypes than $S \rightarrow T$.
- III. $\{x : S, y : T\}$ never has less supertypes than $S \rightarrow T$.

Exercise 5

In this exercise we work with the types: $\mathbb{B} <: \mathbb{N} <: \mathbb{I} <: \mathbb{F}$, functions and records. We assume \mathbb{F} has operation $+$, that \mathbb{I} has operations pred and suc , and that \mathbb{B} has if , with the usual typing rules:

$$\frac{\Gamma \triangleright M : \mathbb{I}}{\Gamma \triangleright \text{suc}(M) : \mathbb{I}} \text{ (T-Suc)} \qquad \frac{\Gamma \triangleright M : \mathbb{I}}{\Gamma \triangleright \text{pred}(M) : \mathbb{I}} \text{ (T-Pred)}$$

$$\frac{\Gamma \triangleright M : \mathbb{F} \quad \Gamma \triangleright N : \mathbb{F}}{\Gamma \triangleright M + N : \mathbb{F}} \text{ (T-+)} \qquad \frac{\Gamma \triangleright M : \mathbb{B} \quad \Gamma \triangleright N : \sigma \quad \Gamma \triangleright O : \sigma}{\Gamma \triangleright \text{if } M \text{ then } N \text{ else } O : \sigma} \text{ (T-If)}$$

- a) Suppose the typing rule for the function type constructor were contravariant in both argument and result type, in other words:

$$\frac{S <: T \quad U <: V}{T \rightarrow V <: S \rightarrow U} \text{ (S-Arrow')}$$

Show that this is not a good idea by:

- Providing a lambda calculus expression M for which evaluation gets stuck (i.e. does not make sense)
- Show, however, that it is typable.

- b) Suppose the typing rule for the function type constructor were covariant in both argument and result type, in other words:

$$\frac{S <: T \quad U <: V}{S \rightarrow U <: T \rightarrow V} \text{ (S-Arrow")}$$

Show that this too is not a good idea.

Exercise 6

Let σ be a type. Decide which of the types below are related via the subtyping relation $<::$:

- $Ref \ \sigma$
- $Ref \ Ref \ \sigma$
- $Sink \ \sigma$
- $Source \ \sigma$
- $Ref \ Sink \ \sigma$
- $Source \ Ref \ \sigma$
- $Source \ Source \ \sigma$
- $Source \ Sink \ \sigma$