

9 Semantics of Programming Languages (PMS)

Consider the following language with higher-order functions and mutable global references. Here l ranges over mutable location names, that can hold arbitrary values v , n ranges over natural numbers, and x ranges over immutable variable names.

$$e ::= n \mid l \mid x \mid \mathbf{fn} \ x : T \Rightarrow e \mid e \ e' \mid e := e' \mid !e$$

$$v ::= n \mid l \mid \mathbf{fn} \ x : T \Rightarrow e$$

Suppose it has a standard left-to-right call-by-value operational semantics. You need not state this semantics.

The definition of $\Gamma \vdash_{eff} e : T$ below, where an effect eff is a subset of $\{\mathbf{R}, \mathbf{W}\}$, is a flawed attempt to statically compute a sound approximation of the possible dynamic side-effects of expressions. Such an analysis is *sound* if eff contains \mathbf{R} and/or \mathbf{W} whenever there is any execution of $\langle e, s \rangle$, for any store s that is well-typed with respect to Γ , that involves (respectively) reading and/or writing the store.

Function types are annotated with the latent effects that may occur when the function is applied: $T ::= \mathbf{int} \mid T \rightarrow_{eff} T' \mid T \mathbf{ref}$

$$\boxed{\Gamma \vdash_{eff} e : T}$$

$$\frac{}{\Gamma \vdash_{\{\}} n : \mathbf{int}} \quad \text{NUM} \quad \frac{l : T \mathbf{ref} \in \Gamma}{\Gamma \vdash_{\{\mathbf{R}, \mathbf{W}\}} l : T \mathbf{ref}} \quad \text{LOC} \quad \frac{x : T \in \Gamma}{\Gamma \vdash_{\{\}} x : T} \quad \text{VAR}$$

$$\frac{\Gamma, x : T \vdash_{eff} e : T'}{\Gamma \vdash_{\{\}} \mathbf{fn} \ x : T \Rightarrow e : T \rightarrow_{eff} T'} \quad \text{FN} \quad \frac{\Gamma \vdash_{eff} e : T_1 \rightarrow_{eff''} T_2 \quad \Gamma \vdash_{eff'} e' : T_1}{\Gamma \vdash_{eff \cup eff'} e \ e' : T_2} \quad \text{APP}$$

$$\frac{\Gamma \vdash_{eff} e : T \mathbf{ref} \quad \Gamma \vdash_{eff'} e' : T}{\Gamma \vdash_{eff \cup eff' \cup \{\mathbf{W}\}} e := e' : T} \quad \text{ASSIGN} \quad \frac{\Gamma \vdash_{eff} e : T \mathbf{ref}}{\Gamma \vdash_{\{\mathbf{R}\}} !e : T} \quad \text{DEREF}$$

- (a) There are three flaws in the above rules, which make them either not sound or an unnecessarily coarse approximation. Explain each flaw, giving a corrected rule for each and an example that shows the problem (assuming the other flaws are fixed). [15 marks]
- (b) In the system above, functions have to be applied to arguments of exactly the expected type. Define a subtype relation $T <: T'$ and subsumption rule that would let function arguments be used even if they have fewer (latent) effects than those anticipated by the function. [5 marks]