

2006 Paper 6 Question 11

Semantics of Programming Languages

Let L be the language with syntax below, a call-by-value left-to-right operational semantics, and the standard simple type system.

$$\begin{aligned} e &::= n \mid \mathbf{fn} \ x:T \Rightarrow e \mid e_1 \ e_2 \mid x \\ T &::= \mathbf{int} \mid T_1 \rightarrow T_2 \\ n &\in \mathbb{Z} \end{aligned}$$

- (a) L is not Turing-complete: there are computable functions over the integers that are not expressible as closed L expressions of type $\mathbf{int} \rightarrow \mathbf{int}$. Why not? [2 marks]
- (b) Define a modest extension L' of L that is Turing-complete. Give the additional syntactic forms (for expressions and types), describe their operational semantics informally, and state their precise typing rules. [10 marks]
- (c) Assuming that f is an L' expression of type $\mathbf{int} \rightarrow \mathbf{int}$, give an expression e of type \mathbf{int} that computes the smallest x such that $f \ x$ is zero. Explain how this can be used to prove completeness. [4 marks]
- (d) Discuss whether Turing-completeness is a necessary or sufficient property for a good programming language. [4 marks]