Types

Contrast the advantages and disadvantages of explicit and implicit typing. [4 marks]

Consider an ML-style language with types and type schemes

\[ \tau ::= \alpha \mid \text{bool} \mid \tau \rightarrow \tau \mid \tau \text{ list} \]
\[ \sigma ::= \forall A(\tau) \]

Give the typing rules for variables, function abstraction, function application, and let-binding. Make the form of the typing judgement clear. [5 marks]

Give terms \(M_1\) and \(M_2\) such that \(N_1\) is typable (in the empty context) and \(N_2\) is not, where

\[ N_1 \overset{\text{def}}{=} \text{let } \text{val } f = M_1 \text{ in } M_2 \text{ end} \]
\[ N_2 \overset{\text{def}}{=} (\text{fn } f \Rightarrow M_2) M_1 \]

Give all uses of \(\sigma \succ \tau\) required in a typing derivation for \(N_1\); prove that there does not exist a typing derivation for \(N_2\). [7 marks]

What is a principal type scheme? Give the principal type scheme for \(N_1\), or explain informally why it does not have one. [4 marks]