1 Syntax

$$
\begin{aligned}
\text{Expressions } e \;::=\;& varident \\
&\mid num\ n \mid false \mid true \\
&\mid e_1 e_2 \\
&\mid (e_1, e_2) \\
&\mid \lambda(p)\ .\ e \\
&\mid let\ p\ =\ e_1\ in\ e_2 \\
&\mid letrec\ p\ =\ e_1\ in\ e_2 \\
&\mid if(e_1)\ then\ e_2\ else\ e_3
\end{aligned}
$$

$$
\begin{aligned}
\text{Pattern } p \;::=\;& varident \\
&\mid (p, p)
\end{aligned}
$$

2 Type System

$$
\begin{aligned}
\text{Types } \tau \;::=\;& nat \\
&\mid bool \\
&\mid \tau_1 \to \tau_2 \\
&\mid \tau_1 \times \tau_2 \\
&\mid \alpha
\end{aligned}
$$

$$
\begin{aligned}
\text{Type-Schemes } \sigma \;::=\;& \tau \\
&\mid \forall \alpha . \sigma
\end{aligned}
$$

$$
\begin{aligned}
\text{Context } \Gamma \;::=\;& \cdot \\
&\mid \Gamma : (x : \tau)
\end{aligned}
$$

$$
\text{Typing} \;::=\; \Gamma \vdash e : \sigma
$$

2.1 Rules

$$\Gamma \vdash true : bool \tag{T-True}$$

$$\Gamma \vdash false : bool \tag{T-False}$$

$$\Gamma \vdash num\ n : nat \tag{T-Num}$$

$$\frac{\sigma \geq \tau \qquad \Gamma \vdash varident : \sigma}{\Gamma \vdash varident : \tau} \tag{T-Mono}$$

$$\frac{\Gamma \vdash e_1 : \tau_2 \rightarrow \tau_1 \qquad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash e_1 e_2 : \tau_1} \tag{T-App}$$

$$varident : \sigma \rightarrow \cdot : (varident) : \sigma \tag{T-BuildContext1}$$

$$\frac{p_1 : \sigma_1 \rightarrow \Gamma_1 \qquad p_2 : \sigma_2 \rightarrow \Gamma_2 \qquad \Gamma_3 = \Gamma_1 \bigcup \Gamma_2}{(p_1, p_2) : \sigma_1 \times \sigma_2 \rightarrow \Gamma_3} \tag{T-BuildContext2}$$

$$\frac{p : \tau_2 \rightarrow \Gamma_2 \qquad \Gamma_2 \bigcup \Gamma_1 \vdash e : \tau_1}{\Gamma_1 \vdash \lambda(p).e : \tau_2 \rightarrow \tau_1} \tag{T-Fun}$$

$$\frac{\Gamma \vdash e_1 : bool \qquad \Gamma \vdash e_2 : \tau \qquad \Gamma \vdash e_3 : \tau}{\Gamma \vdash if\ e_1\ then\ e_2\ else\ e_3 : \tau} \tag{T-IfThenElse}$$

$$\frac{\Gamma \vdash e_1 : \tau_1 \qquad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash (e_1, e_2) : \tau_1 \times \tau_2} \tag{T-Pair}$$

$$\frac{\Gamma \vdash e_2 : \tau_2 \quad \sigma = gen(\Gamma, \tau) \quad p : \sigma \rightarrow \Gamma_2 \quad \Gamma \bigcup \Gamma_2 \vdash e_1 : \tau}{\Gamma \vdash let\ p\ =\ e_2\ in\ e_1 : \tau} \tag{T-Let}$$

$$\frac{\Gamma \vdash let\ p\ =\ fix\ (\lambda p.e_2)\ in\ e_1 : \tau}{\Gamma \vdash letrec\ p\ =\ e_2\ in\ e_1 : \tau} \tag{T-Letrec}$$

$$\frac{\Gamma \vdash e : \tau \rightarrow \tau}{\Gamma \vdash fix(e) : \tau} \tag{T-Fix}$$

3 Operational semantics

$$
\begin{aligned}
\text{Values } v ::= &\ numn \mid true \mid false \\
&\mid (v, v) \\
&\mid \lambda p.e
\end{aligned}
$$

$$
\text{Evaluation } ::= e \to e'
$$

3.1 Rules

$$if\ true\ then\ e_1\ else\ e_2 \to e_1 \tag{E-IfTrue}$$

$$if\ false\ then\ e_1\ else\ e_2 \to e_2 \tag{E-IfFalse}$$

$$\frac{e_1 \to e_1'}{if\ e_1\ then\ e_2\ else\ e_3 \to if\ e_1'\ then\ e_2\ else\ e_3} \tag{E-IfThenElse}$$

$$\frac{e_1 \to e_1'}{(e_1, e_2) \to (e_1', e_2)} \tag{E-PairLeft}$$

$$\frac{e_2 \to e_2'}{(e_1, e_2) \to (e_1, e_2')} \tag{E-PairRight}$$

$$\frac{e_1 \to e_1'}{let\ p\ =\ e_1\ in\ e_2 \to let\ x\ =\ e_1'\ in\ e_2} \tag{E-Let}$$

$$let\ x\ =\ v\ in\ e \to [x \mapsto v]e \tag{E-LetV}$$

$$letrec\ p =\ e_1\ in\ e_2 \to let\ p\ = fix(\lambda p.e_1)\ in\ e_2 \tag{E-LetRec}$$

$$\frac{e \to e'}{fix(e) \to fix(e')} \tag{E-Fix}$$

$$fix(\lambda(p.e)) \to [p \mapsto (fix(\lambda(p.e))]e \tag{E-FixRec}$$

$$let\ (p_1, p_2)\ =\ (e_1, e_2)\ in\ e_3 \to let\ p_1\ =\ e_1\ in\ (let\ p_2\ =\ e_2\ in\ e_3) \tag{E-PatternMatch}$$

$$\frac{e_1 \to e_1'}{e_1 e_2 \to e_1' e_2} \tag{E-App1}$$

$$\frac{e_2 \to e_2'}{v\ e_2 \to v\ e_2'} \tag{E-App2}$$

$$(\lambda x.e)\ v \to [x \mapsto v]e \tag{E-Lambda}$$

$$(\lambda(p_1, p_2).e_3)\ (e_1, e_2) \to (\lambda p_1.(\lambda p_2.e_3)\ e_2)\ e_1 \tag{E-MatchLambda}$$