Typed Semantics for λ_i

1. Syntax

$$
\begin{array}{llll}
e & ::= & & \text{Expression} \\
 & | & n & \text{Integer Literal} \\
 & | & e_1 + e_2 & \text{Add} \\
 & | & x & \text{Variable} \\
 & | & \lambda x.\, e & \text{Abstraction} \\
 & | & e_1\ e_2 & \text{Application} \\
 & | & e_1 \,,,\, e_2 & \text{Merge} \\
 & | & (e : A) & \text{Annotation}
\end{array}
$$

2. Semantics

$\boxed{v \Rightarrow A}$ Typed Values (Inf)

$$\frac{}{n \Rightarrow \mathsf{Int}}\ \textsc{VI\_Int} \qquad \frac{v_1 \Rightarrow A \qquad v_2 \Rightarrow B}{v_1 \,,,\, v_2 \Rightarrow A \,\&\, B}\ \textsc{VI\_Merge} \qquad \frac{v \Leftarrow A}{(v : A) \Rightarrow A}\ \textsc{VI\_Ann}$$

$\boxed{v \Leftarrow A}$ Typed Values (Chk)

$$\frac{}{\lambda x.\, e \Leftarrow A \to B}\ \textsc{VC\_Abs}$$

$\boxed{e_1 \rightsquigarrow e_2}$ Reduction

$$\frac{e_1 \rightsquigarrow e_3}{e_1 + e_2 \rightsquigarrow e_3 + e_2}\ \textsc{R\_Add1} \qquad \frac{e_1 \rightsquigarrow e_2}{n + e_1 \rightsquigarrow n + e_2}\ \textsc{R\_Add2} \qquad \frac{r = n + m}{n + m \rightsquigarrow r}\ \textsc{R\_Add3}$$

$$\frac{e_1 \rightsquigarrow e_3}{e_1\ e_2 \rightsquigarrow e_3\ e_2}\ \textsc{R\_App1} \qquad \frac{e_1 \rightsquigarrow e_2}{v\ e_1 \rightsquigarrow v\ e_2}\ \textsc{R\_App2} \qquad \frac{v_1 \Rightarrow B \qquad A \neq B \qquad v_1 \rightsquigarrow v_2 \Leftarrow A}{(\lambda x.\, e : A \to C)\ v_1 \rightsquigarrow (\lambda x.\, e : A \to C)\ v_2}\ \textsc{R\_AppSub}$$

$$\frac{v \Leftarrow A}{(\lambda x.\, e : A \to B)\ v \rightsquigarrow e[x \mapsto (v : A)]}\ \textsc{R\_App4} \qquad \frac{v \Rightarrow A}{(\lambda x.\, e : A \to B)\ v \rightsquigarrow e[x \mapsto v]}\ \textsc{R\_App5}$$

$$\frac{e_1 \rightsquigarrow e_3}{e_1 \,,,\, e_2 \rightsquigarrow e_3 \,,,\, e_2}\ \textsc{R\_Merge1} \qquad \frac{e_1 \rightsquigarrow e_2}{v \,,,\, e_1 \rightsquigarrow v \,,,\, e_2}\ \textsc{R\_Merge2} \qquad \frac{e_1 \rightsquigarrow e_2}{(e_1 : A) \rightsquigarrow (e_2 : A)}\ \textsc{R\_Ann1}$$

$$\frac{v_1 \Rightarrow B \qquad A \neq B \qquad v_1 \rightsquigarrow v_2 \Leftarrow A}{(v_1 : A) \rightsquigarrow (v_2 : A)}\ \textsc{R\_AnnSub} \qquad \frac{v \Rightarrow A}{(v : A) \rightsquigarrow v}\ \textsc{R\_Ann3}$$

$\boxed{e \rightsquigarrow v \Leftarrow A}$ Reduction (Sub)

$$\frac{v \Rightarrow A \to B}{v \rightsquigarrow \lambda y.\, (v\ y : D) \Leftarrow C \to D}\ \textsc{RSub\_Fun} \qquad \frac{}{v \rightsquigarrow (v : A) \,,,\, (v : B) \Leftarrow A \,\&\, B}\ \textsc{RSub\_AndR}$$

$$\frac{\mathsf{ord}\ A \qquad v_1 \Rightarrow B \qquad B \leq A}{v_1 \,,,\, v_2 \rightsquigarrow v_1 \Leftarrow A}\ \textsc{RSub\_AndL1} \qquad \frac{\mathsf{ord}\ A \qquad v_2 \Rightarrow C \qquad C \leq A}{v_1 \,,,\, v_2 \rightsquigarrow v_2 \Leftarrow A}\ \textsc{RSub\_AndL2}$$

Remark 1: R_AppSub and R_AnnSub's condition $A \neq B$ prevents looping (subtyping is reflexive).
Remark 2: If there are no assumptions over v, then assume $\exists A.\, v \Rightarrow A$.
Remark 3: For a closer version of 1-step reduction of sub-rule, check the system below.

3. Metatheory

Lemma 3.1 (Subject reduction (sub)). `[safety_sub]` *If* $\Gamma \vdash v \Rightarrow A$ *and* `<<no parses (char 8): v  > e <***= B >>` *and* $A \leq B$*, then* $\Gamma \vdash e \Leftarrow B$.

Lemma 3.2 (Subject reduction (inf)). `[safety_inf]` *If* $\Gamma \vdash e_1 \Rightarrow A$ *and* $e_1 \rightsquigarrow e_2$*, then* $\Gamma \vdash e_2 \Rightarrow A$.

Lemma 3.3 (Subject reduction (chk)). `[safety_chk]` *If* $\Gamma \vdash e_1 \Leftarrow A$ *and* $e_1 \rightsquigarrow e_2$*, then* $\Gamma \vdash e_2 \Leftarrow A$.

4. Alternative Sub reduction rules

$\boxed{v_1 \rightsquigarrow v_2 \Leftarrow A}$ Reduction (Sub Alt)

$$\frac{v \Rightarrow A \to B}{v \rightsquigarrow \lambda y.\,(v\ y : D) \Leftarrow C \to D}\ \textsc{RSubAlt\_Fun}$$

$$\frac{v \Rightarrow C \qquad \text{ord}\ C \qquad v \rightsquigarrow v_1 \Leftarrow A \qquad v \rightsquigarrow v_2 \Leftarrow B}{v \rightsquigarrow v_1\ ,,\ v_2 \Leftarrow A \,\&\, B}\ \textsc{RSubAlt\_AndR1}$$

$$\frac{v_1 \Rightarrow C \qquad A \neq C \qquad v_1 \rightsquigarrow v_3 \Leftarrow A}{v_1\ ,,\ v_2 \rightsquigarrow v_3\ ,,\ v_2 \Leftarrow A \,\&\, B}\ \textsc{RSubAlt\_AndR2}$$

$$\frac{v_1 \Rightarrow A \qquad v_2 \rightsquigarrow v_3 \Leftarrow B}{v_1\ ,,\ v_2 \rightsquigarrow v_1\ ,,\ v_3 \Leftarrow A \,\&\, B}\ \textsc{RSubAlt\_AndR3} \qquad \frac{\text{ord}\ A \qquad v_1 \Rightarrow B \qquad B \leq A}{v_1\ ,,\ v_2 \rightsquigarrow v_1 \Leftarrow A}\ \textsc{RSubAlt\_AndL1}$$

$$\frac{\text{ord}\ A \qquad v_2 \Rightarrow C \qquad C \leq A}{v_1\ ,,\ v_2 \rightsquigarrow v_2 \Leftarrow A}\ \textsc{RSubAlt\_AndL2}$$

Remark 1: How to get a better RSubAlt_Fun? (i.e. without using annotations)