Chris Dalziel - Software Testing 2024/5 Requirements

1 Functional Requirements

The functional requirements are largely outlined in the EPL Assignment 3 specification. A summary is provided below, giving the broad strokes for each component of the system.

If at any stage of the process the interpreter finds that there is no matching rule, an error should be thrown. Errors can occur at any step of the process as discusses in the main document.

1.1 Parsing

The parser should take as input a string of text, and produce an abstract syntax representation of the program that text represents. The syntax is outlined in Figure 1. When the parser encounters a token which does not match what is expected, it should throw an error indicating the nature of the failure.

1.2 Type Checking

The bidirectional type-system involves both type checking and type inference, and contains rules for subtyping relations. It takes an abstract syntax tree and verifies that the program is well-typed.

The inference rules are entirely defined in Figure 3, and the checking rules in Figure 4. When an incorrect or mismatching type appears an error should be thrown, unless there's a valid subtyping relation between that type and the required one.

The rules for subtyping are (largely) outlined in Figure 2. Those which are not follow the generic subtyping rules defined in the EPL course materials.

1.3 Substitution

Capture avoiding substitution needs performed to the abstract syntax tree before desugaring and evaluation. Substitution rules are defined in Figure 5.

Substitution requires a way to generate fresh variables as well as the ability to label existing variables to differentiate them (e.g. x in two contexts becoming x_0 and x_1 respectively).

1.4 Desugaring

Certain language features are "syntactic sugar", and must be converted to their generic representations before evaluation. Desugaring takes a type-checked abstract syntax tree and converts it into a semantically equivalent syntax tree with the syntactic sugar removed. The desugaring rules are provided in Figure 6.

1.5 Evaluation

Evaluation takes the abstract syntax tree that has been created and passed through the other components and gets the result of running the program. Evaluation rules are provided in Figure 7.

2 Robustness Requirements

The process should be extremely robust, as each component tests for all cases and throws an error otherwise. The system is entirely deterministic which makes behaviour predictable, as such we should be able to ensure that it reacts correctly to inputs.

Correct behaviour involves both producing valid syntax trees, type-checking, evaluating, etc. correctly and also failing to do any of the above via an error. An interpreter which type-checks incorrectly is just as bad as one which doesn't type-check at all.

We require that the interpreter doesn't fail a single test, as such a failure indicates that it doesn't comply with the language specifications and therefore cannot be robust.

A Figures

$$
\begin{array}{rcll}
\text{Values} \quad v & ::= & n \in \mathbb{N} \mid b \in \mathbb{B} \mid s \in \texttt{string} & \text{Constants} \\
& \mid & x \mid \backslash x.e \mid \mathsf{rec} f(x).e & \text{Functions} \\
& \mid & (v_1, v_2) & \text{Pairs} \\
& \mid & \mathsf{unit} & \text{Unit value} \\
& \mid & \langle \ell_1 = v_1, \ldots, \ell_n = v_n \rangle & \text{Records} \\
& \mid & \mathsf{select}\, \ell\, v & \text{Variants} \\
& \mid & \{\!| v_1, \ldots, v_n |\!\} & \text{Multisets} \\
\\
\text{Expressions} \quad e & ::= & n \in \mathbb{N} \mid e_1 + e_2 \mid e_1 - e_2 \mid e_1 * e_2 & \text{Numbers} \\
& \mid & b \in \mathbb{B} \mid \mathsf{if}\, e\, \mathsf{then}\, e_1\, \mathsf{else}\, e_2 \mid e_1 == e_2 \mid e_1 < e_2 & \text{Booleans} \\
& \mid & s \in \texttt{string} \mid \mathsf{length}(e_1) \mid \mathsf{index}(e_1, e_2) \mid \mathsf{concat}(e_1, e_2) & \text{Strings} \\
& \mid & x \mid \mathsf{let}\, x = e_1\, \mathsf{in}\, e_2 & \text{Binding} \\
& \mid & \backslash x.e \mid \mathsf{rec} f(x).e \mid e_1\, e_2 & \text{Functions} \\
& \mid & (e_1, e_2) \mid \mathsf{fst}(e) \mid \mathsf{snd}(e) & \text{Pairs} \\
& \mid & \mathsf{unit} & \text{Unit} \\
& \mid & \langle \ell_1 = e_1, \ldots, \ell_n = e_n \rangle \mid e.\ell & \text{Records} \\
& \mid & \mathsf{select}\, \ell\, e \mid \mathsf{case}\, e\, \mathsf{of}\, \{\ell_1\, x_1 \to e_1, \ldots, \ell_n\, x_n \to e_n\} & \text{Variants} \\
& \mid & \{\!| e_1, \ldots, e_n |\!\} \mid \mathsf{when}(e_1, e_2) \mid \mathsf{count}(e_1, e_2) & \\
& \mid & \mathsf{sum}(e_1, e_2) \mid \mathsf{diff}(e_1, e_2) \mid \mathsf{flatMap}(e_1, e_2) & \text{Multisets} \\
& \mid & e : \tau & \text{Annotations} \\
& \mid & \mathsf{let}\, (x, y) = e_1\, \mathsf{in}\, e_2 \mid \mathsf{sig}\, f : \tau\, \mathsf{let\ fun}\, f(x) = e_1\, \mathsf{in}\, e_2 & \\
& \mid & \mathsf{sig}\, f : \tau\, \mathsf{let\ rec}\, f(x) = e_1\, \mathsf{in}\, e_2 & \\
& \mid & \mathsf{let}\, \langle \ell_1 = x_1, \ldots, \ell_n = x_n \rangle = e_1\, \mathsf{in}\, e_2 & \\
& \mid & \{\!| e \mid p_1, \ldots, p_n |\!\} \mid \{\!| e |\!\} & \text{Syntactic Sugar} \\
\\
\text{Comprehension Items} \quad p & ::= & x \leftarrow e & \text{Binding} \\
& \mid & e & \text{Guard} \\
& \mid & \mathsf{let}\, x = e & \text{Let} \\
\text{Types} \quad \tau & ::= & \texttt{int} \mid \texttt{bool} \mid \texttt{string} \mid \texttt{unit} \mid \tau_1 \texttt{->} \tau_2 \mid \tau_1 * \tau_2 & \\
& \mid & \langle \ell_1 : \tau_1, \ldots, \ell_n : \tau_n \rangle \mid [\ell_1 : \tau_1, \ldots, \ell_n : \tau_n] \mid \{\!| \tau |\!\} & \\
\\
\text{Contexts} \quad \Gamma & ::= & \cdot \mid \Gamma, x : \tau &
\end{array}
$$

Figure 1: Syntax of Frog

$\boxed{\tau_1 <: \tau_2}$

$$
\frac{\tau \in \mathsf{BaseType}}{\tau <: \tau}
\qquad
\frac{\tau_1 <: \tau_1' \qquad \tau_2 <: \tau_2'}{\tau_1 * \tau_2 <: \tau_1' * \tau_2'}
\qquad
\frac{\tau_1' <: \tau_1 \qquad \tau_2 <: \tau_2'}{\tau_1 \texttt{->} \tau_2 <: \tau_1' \texttt{->} \tau_2'}
$$

$$
\frac{n \geq m \qquad \text{for all } \ell_i', \tau_i' \text{ there exists } \ell_j, \tau_j \text{ with } \ell_i' = \ell_j \text{ and with } \tau_i <: \tau_j'}{\langle \ell_1 = \tau_1, \ldots, \ell_n = \tau_n \rangle <: \langle \ell_1' = \tau_1', \ldots, \ell_m' = \tau_m' \rangle}
$$

$$
\frac{n \leq m \qquad \text{for all } \ell_i, \tau_i \text{ there exists } \ell_j', \tau_j' \text{ with } \ell_i = \ell_j' \text{ and with } \tau_i <: \tau_j'}{[\ell_1 = \tau_1, \ldots, \ell_n = \tau_n] <: [\ell_1' = \tau_1', \ldots, \ell_m' = \tau_m']}
\qquad
\frac{\tau <: \tau'}{\{\!| \tau |\!\} <: \{\!| \tau' |\!\}}
$$

Figure 2: Subtyping rule

$\boxed{\Gamma \vdash e \Rightarrow \tau}$

$$\frac{n \in \mathbb{N}}{\Gamma \vdash n \Rightarrow \texttt{int}} \qquad \frac{\Gamma \vdash e_1 \Leftarrow \texttt{int} \qquad \Gamma \vdash e_2 \Leftarrow \texttt{int}}{\Gamma \vdash e_1 + e_2 \Rightarrow \texttt{int}} \qquad \frac{\Gamma \vdash e_1 \Leftarrow \texttt{int} \qquad \Gamma \vdash e_2 \Leftarrow \texttt{int}}{\Gamma \vdash e_1 - e_2 \Rightarrow \texttt{int}}$$

$$\frac{\Gamma \vdash e_1 \Leftarrow \texttt{int} \qquad \Gamma \vdash e_2 \Leftarrow \texttt{int}}{\Gamma \vdash e_1 * e_2 \Rightarrow \texttt{int}} \qquad \frac{b \in \mathbb{B}}{\Gamma \vdash b \Rightarrow \texttt{bool}} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \tau \qquad \Gamma \vdash e_2 \Leftarrow \tau \qquad \tau \in \textsf{EqType}}{\Gamma \vdash e_1 == e_2 \Rightarrow \texttt{bool}}$$

$$\frac{\Gamma \vdash e_1 \Leftarrow \texttt{int} \qquad \Gamma \vdash e_2 \Leftarrow \texttt{int}}{\Gamma \vdash e_1 < e_2 \Rightarrow \texttt{bool}} \qquad \frac{\Gamma \vdash e \Leftarrow \texttt{bool} \qquad \Gamma \vdash e_1 \Rightarrow \tau \qquad \Gamma \vdash e_2 \Leftarrow \tau}{\Gamma \vdash \textsf{if}\, e\, \textsf{then}\, e_1\, \textsf{else}\, e_2 \Rightarrow \tau} \qquad \frac{s \in \texttt{string}}{\Gamma \vdash s \Rightarrow \texttt{string}}$$

$$\frac{\Gamma \vdash e_1 \Leftarrow \texttt{string}}{\Gamma \vdash \textsf{length}(e_1) \Rightarrow \texttt{int}} \qquad \frac{\Gamma \vdash e_1 \Leftarrow \texttt{string} \qquad \Gamma \vdash e_2 \Leftarrow \texttt{int}}{\Gamma \vdash \textsf{index}(e_1, e_2) \Rightarrow \texttt{string}} \qquad \frac{\Gamma \vdash e_1 \Leftarrow \texttt{string} \qquad \Gamma \vdash e_2 \Leftarrow \texttt{string}}{\Gamma \vdash \textsf{concat}(e_1, e_2) \Rightarrow \texttt{string}}$$

$$\frac{x : \tau \in \Gamma}{\Gamma \vdash x \Rightarrow \tau} \qquad \frac{\Gamma \vdash e \Leftarrow \tau}{\Gamma \vdash e : \tau \Rightarrow \tau} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \tau_1 \qquad \Gamma, x : \tau_1 \vdash e_2 \Rightarrow \tau_2}{\Gamma \vdash \textsf{let}\, x = e_1\, \textsf{in}\, e_2 \Rightarrow \tau_2} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \tau_1 \texttt{->} \tau_2 \qquad \Gamma \vdash e_2 \Leftarrow \tau_1}{\Gamma \vdash e_1\, e_2 \Rightarrow \tau_2}$$

$$\frac{\Gamma \vdash e_1 \Rightarrow \tau_1 \qquad \Gamma \vdash e_2 \Rightarrow \tau_2}{\Gamma \vdash (e_1, e_2) \Rightarrow \tau_1 * \tau_2} \qquad \frac{\Gamma \vdash e \Rightarrow \tau_1 * \tau_2}{\Gamma \vdash \textsf{fst}(e) \Rightarrow \tau_1} \qquad \frac{\Gamma \vdash e \Rightarrow \tau_1 * \tau_2}{\Gamma \vdash \textsf{snd}(e) \Rightarrow \tau_2}$$

$$\frac{}{\Gamma \vdash \textsf{unit} \Rightarrow \texttt{unit}} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \tau_1 \qquad \ldots \qquad \Gamma \vdash e_n \Rightarrow \tau_n}{\Gamma \vdash \langle \ell_1 = e_1, \ldots, \ell_n = e_n \rangle \Rightarrow \langle \ell_1 = \tau_1, \ldots, \ell_n = \tau_n \rangle}$$

$$\frac{\Gamma \vdash e \Rightarrow \langle \ell_1 = \tau_1, \ldots, \ell_n = \tau_n \rangle \qquad 1 \leq j \leq n}{\Gamma \vdash e.\ell_j \Rightarrow \tau_j} \qquad \frac{\Gamma \vdash e \Rightarrow \tau}{\Gamma \vdash \textsf{select}\, \ell\, e \Rightarrow [\ell : \tau]}$$

$$\frac{\Gamma \vdash e \Rightarrow [\ell_1 : \tau_1, \ldots, \ell_n : \tau_n] \qquad \Gamma, x_1 : \tau_1 \vdash e_1 \Rightarrow \tau \qquad \Gamma, x_2 : \tau_2 \vdash e_2 \Leftarrow \tau \qquad \ldots \qquad \Gamma, x_n : \tau_n \vdash e_n \Leftarrow \tau}{\Gamma \vdash \textsf{case}\, e\, \textsf{of}\, \{\ell_1\, x_1 \to e_1, \ldots, \ell_n\, x_n \to e_n\} \Rightarrow \tau}$$

$$\frac{\Gamma \vdash e_1 \Rightarrow \tau \qquad \Gamma \vdash e_2 \Leftarrow \tau \qquad \cdots \qquad \Gamma \vdash e_n \Leftarrow \tau}{\Gamma \vdash \{\!|e_1, \ldots, e_n|\!\} \Rightarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \{\!|\tau_1|\!\} \qquad \Gamma \vdash e_2 \Rightarrow \tau_1 \texttt{->} \{\!|\tau_2|\!\}}{\Gamma \vdash \textsf{flatMap}(e_1, e_2) \Rightarrow \{\!|\tau_2|\!\}}$$

$$\frac{\Gamma \vdash e_1 \Leftarrow \texttt{bool} \qquad \Gamma \vdash e_2 \Rightarrow \{\!|\tau|\!\}}{\Gamma \vdash \textsf{when}(e_1, e_2) \Rightarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \{\!|\tau|\!\} \qquad \Gamma \vdash e_2 \Leftarrow \{\!|\tau|\!\}}{\Gamma \vdash \textsf{sum}(e_1, e_2) \Rightarrow \{\!|\tau|\!\}}$$

$$\frac{\Gamma \vdash e_1 \Rightarrow \{\!|\tau|\!\} \qquad \Gamma \vdash e_2 \Leftarrow \{\!|\tau|\!\} \qquad \tau \in \textsf{EqType}}{\Gamma \vdash \textsf{diff}(e_1, e_2) \Rightarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \{\!|\tau|\!\} \qquad \Gamma \vdash e_2 \Leftarrow \tau \qquad \tau \in \textsf{EqType}}{\Gamma \vdash \textsf{count}(e_1, e_2) \Rightarrow \texttt{int}}$$

$$\frac{\Gamma \vdash e \Rightarrow \tau}{\Gamma \vdash \{\!|e \mid \, |\!\} \Rightarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e' \Leftarrow \texttt{bool} \qquad \Gamma \vdash \{\!|e \mid p_1 \ldots, p_n|\!\} \Rightarrow \{\!|\tau|\!\}}{\Gamma \vdash \{\!|e \mid e', p_1 \ldots, p_n|\!\} \Rightarrow \{\!|\tau|\!\}}$$

$$\frac{\Gamma \vdash e' \Rightarrow \{\!|\tau'|\!\} \qquad \Gamma, x : \tau' \vdash \{\!|e \mid p_1, \ldots, p_n|\!\} \Rightarrow \{\!|\tau|\!\}}{\Gamma \vdash \{\!|e \mid x \leftarrow e', p_1, \ldots, p_n|\!\} \Rightarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e' \Rightarrow \tau' \qquad \Gamma, x : \tau' \vdash \{\!|e \mid p_1 \ldots, p_n|\!\} \Rightarrow \{\!|\tau|\!\}}{\Gamma \vdash \{\!|e \mid \textsf{let}\, x = e', p_1 \ldots, p_n|\!\} \Rightarrow \{\!|\tau|\!\}}$$

$$\frac{\begin{array}{c}\Gamma, x : \tau_1 \vdash e_1 \Leftarrow \tau_2 \\ \Gamma, f : \tau_1 \texttt{->} \tau_2 \vdash e_2 \Rightarrow \tau\end{array}}{\Gamma \vdash \textsf{sig}\, f : \tau_1 \texttt{->} \tau_2\, \textsf{let fun}\, f(x) = e_1\, \textsf{in}\, e_2 \Rightarrow \tau} \qquad \frac{\begin{array}{c}\Gamma, f : \tau_1 \texttt{->} \tau_2, x : \tau_1 \vdash e_1 \Leftarrow \tau_2 \\ \Gamma, f : \tau_1 \texttt{->} \tau_2 \vdash e_2 \Rightarrow \tau\end{array}}{\Gamma \vdash \textsf{sig}\, f : \tau_1 \texttt{->} \tau_2\, \textsf{let rec}\, f(x) = e_1\, \textsf{in}\, e_2 \Rightarrow \tau}$$

$$\frac{\Gamma \vdash e_1 \Rightarrow \tau_1 * \tau_2 \qquad \Gamma, x : \tau_1, y : \tau_2 \vdash e_2 \Rightarrow \tau}{\Gamma \vdash \textsf{let}\, (x, y) = e_1\, \textsf{in}\, e_2 \Rightarrow \tau}$$

$$\frac{\Gamma \vdash e_1 \Rightarrow \langle \ell_1 = \tau_1, \ldots, \ell_n = \tau_n \rangle \qquad \Gamma, x_1 : \tau_1, \ldots, x_n : \tau_n \vdash e_2 \Rightarrow \tau}{\Gamma \vdash \textsf{let}\, \langle \ell_1 = x_1, \ldots, \ell_n = x_n \rangle = e_1\, \textsf{in}\, e_2 \Rightarrow \tau}$$

Figure 3: Inference rules of Frog

$\boxed{\Gamma \vdash e \Leftarrow \tau}$

$$\frac{\Gamma \vdash e \Leftarrow \texttt{bool} \qquad \Gamma \vdash e_1 \Leftarrow \tau \qquad \Gamma \vdash e_2 \Leftarrow \tau}{\Gamma \vdash \textsf{if}\ e\ \textsf{then}\ e_1\ \textsf{else}\ e_2 \Leftarrow \tau} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \tau_1 \qquad \Gamma, x : \tau_1 \vdash e_2 \Leftarrow \tau_2}{\Gamma \vdash \textsf{let}\ x = e_1\ \textsf{in}\ e_2 \Leftarrow \tau_2} \qquad \frac{\Gamma, x : \tau_1 \vdash e \Leftarrow \tau_2}{\Gamma \vdash \backslash x.e \Leftarrow \tau_1 \texttt{->} \tau_2}$$

$$\frac{\Gamma, f : \tau_1 \texttt{->} \tau_2, x : \tau_1 \vdash e \Leftarrow \tau_2}{\Gamma \vdash \textsf{rec} f(x).e \Leftarrow \tau_1 \texttt{->} \tau_2} \qquad \frac{\Gamma \vdash e_1 \Leftarrow \tau_1 \qquad \Gamma \vdash e_2 \Leftarrow \tau_2}{\Gamma \vdash (e_1, e_2) \Leftarrow \tau_1 * \tau_2}$$

$$\frac{m \leq n \qquad \Gamma \vdash e_1 \Leftarrow \tau_1 \qquad \ldots \qquad \Gamma \vdash e_m \Leftarrow \tau_m \qquad \Gamma \vdash e_{m+1} \Rightarrow \tau_{m+1} \qquad \ldots \qquad \Gamma \vdash e_n \Rightarrow \tau_n}{\Gamma \vdash \langle \ell_1 = e_1, \ldots, \ell_n = e_n \rangle \Leftarrow \langle \ell_1 = \tau_1, \ldots, \ell_m = \tau_m \rangle}$$

$$\frac{\Gamma \vdash e \Leftarrow \langle \ell_j = \tau_j \rangle}{\Gamma \vdash e.\ell_j \Leftarrow \tau_j} \qquad \frac{\Gamma \vdash e \Leftarrow \tau_j \qquad 1 \leq j \leq n}{\Gamma \vdash \textsf{select}\ \ell_j\, e \Leftarrow [\ell_1 : \tau_1, \ldots, \ell_n : \tau_n]}$$

$$\frac{\Gamma \vdash e \Rightarrow [\ell_1 : \tau_1, \ldots, \ell_n : \tau_n] \qquad \Gamma, x_1 : \tau_1 \vdash e_1 \Leftarrow \tau \qquad \ldots \qquad \Gamma, x_n : \tau_n \vdash e_n \Leftarrow \tau}{\Gamma \vdash \textsf{case}\ e\ \textsf{of}\ \{\ell_1\, x_1 \to e_1, \ldots, \ell_n\, x_n \to e_n\} \Leftarrow \tau}$$

$$\frac{\Gamma \vdash e_1 \Leftarrow \tau \qquad \cdots \qquad \Gamma \vdash e_n \Leftarrow \tau}{\Gamma \vdash \{\!|e_1 \ldots, e_n|\!\} \Leftarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \{\!|\tau_1|\!\} \qquad \Gamma \vdash e_2 \Leftarrow \tau_1 \texttt{->} \{\!|\tau_2|\!\}}{\Gamma \vdash \textsf{flatMap}(e_1, e_2) \Leftarrow \{\!|\tau_2|\!\}}$$

$$\frac{\Gamma \vdash e_1 \Leftarrow \texttt{bool} \qquad \Gamma \vdash e_2 \Leftarrow \{\!|\tau|\!\}}{\Gamma \vdash \textsf{when}(e_1, e_2) \Leftarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e_1 \Leftarrow \{\!|\tau|\!\} \qquad \Gamma \vdash e_2 \Leftarrow \{\!|\tau|\!\}}{\Gamma \vdash \textsf{sum}(e_1, e_2) \Leftarrow \{\!|\tau|\!\}}$$

$$\frac{\Gamma \vdash e_1 \Leftarrow \{\!|\tau|\!\} \qquad \Gamma \vdash e_2 \Leftarrow \{\!|\tau|\!\} \qquad \tau \in \textsf{EqType}}{\Gamma \vdash \textsf{diff}(e_1, e_2) \Leftarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e \Leftarrow \tau}{\Gamma \vdash \{\!|e \mid |\!\} \Leftarrow \{\!|\tau|\!\}}$$

$$\frac{\Gamma \vdash e' \Leftarrow \texttt{bool} \qquad \Gamma \vdash \{\!|e \mid p_1 \ldots, p_n|\!\} \Leftarrow \{\!|\tau|\!\}}{\Gamma \vdash \{\!|e \mid e', p_1 \ldots, p_n|\!\} \Leftarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma \vdash e' \Rightarrow \{\!|\tau'|\!\} \qquad \Gamma, x : \tau' \vdash \{\!|e \mid p_1 \ldots, p_n|\!\} \Leftarrow \{\!|\tau|\!\}}{\Gamma \vdash \{\!|e \mid x \leftarrow e', p_1 \ldots, p_n|\!\} \Leftarrow \{\!|\tau|\!\}}$$

$$\frac{\Gamma \vdash e' \Rightarrow \tau' \qquad \Gamma, x : \tau' \vdash \{\!|e \mid p_1 \ldots, p_n|\!\} \Leftarrow \{\!|\tau|\!\}}{\Gamma \vdash \{\!|e \mid \textsf{let}\ x = e', p_1 \ldots, p_n|\!\} \Leftarrow \{\!|\tau|\!\}} \qquad \frac{\Gamma, x : \tau_1 \vdash e_1 \Leftarrow \tau_2 \qquad \Gamma, f : \tau_1 \texttt{->} \tau_2 \vdash e_2 \Leftarrow \tau}{\Gamma \vdash \textsf{sig}\ f : \tau_1 \texttt{->} \tau_2\ \textsf{let fun}\ f(x) = e_1\ \textsf{in}\ e_2 \Leftarrow \tau}$$

$$\frac{\begin{array}{c}\Gamma, f : \tau_1 \texttt{->} \tau_2, x : \tau_1 \vdash e_1 \Leftarrow \tau_2 \\ \Gamma, f : \tau_1 \texttt{->} \tau_2 \vdash e_2 \Leftarrow \tau\end{array}}{\Gamma \vdash \textsf{sig}\ f : \tau_1 \texttt{->} \tau_2\ \textsf{let rec}\ f(x) = e_1\ \textsf{in}\ e_2 \Leftarrow \tau} \qquad \frac{\Gamma \vdash e_1 \Rightarrow \tau_1 * \tau_2 \qquad \Gamma, x : \tau_1, y : \tau_2 \vdash e_2 \Leftarrow \tau}{\Gamma \vdash \textsf{let}\ (x, y) = e_1\ \textsf{in}\ e_2 \Leftarrow \tau}$$

$$\frac{\Gamma \vdash e_1 \Rightarrow \langle \ell_1 = \tau_1, \ldots, \ell_n = \tau_n \rangle \qquad \Gamma, x_1 : \tau_1, \ldots, x_n : \tau_n \vdash e_2 \Leftarrow \tau}{\Gamma \vdash \textsf{let}\ \langle \ell_1 = x_1, \ldots, \ell_n = x_n \rangle = e_1\ \textsf{in}\ e_2 \Leftarrow \tau} \qquad \frac{\Gamma \vdash e \Rightarrow \tau \qquad \tau <: \tau'}{\Gamma \vdash e \Leftarrow \tau'}$$

Figure 4: Checking rules of Frog

$$\begin{array}{rcl}
v[e/x] & = & v \\
\langle \ell_1 = e_1, \ldots, \ell_n = e_n \rangle[e/x] & = & \langle \ell_1 = e_1[e/x], \ldots, \ell_n = e_n[e/x] \rangle \\
(e_0.\ell)[e/x] & = & (e_0[e/x]).\ell \\
(\mathsf{let}\ \langle \ell_1 = x_1, \ldots, \ell_n = x_n \rangle = e_1\ \mathsf{in}\ e_2)[e/x] & = & \\
\multicolumn{3}{r}{\mathsf{let}\ \langle \ell_1 = y_1, \ldots, \ell_n = y_n \rangle = e_1[e/x]\ \mathsf{in}\ (e_2(x_n \leftrightarrow y_n) \cdots (x_1 \leftrightarrow y_1))[e/x]} \\
 & & (y_1, \ldots, y_n \text{ fresh}) \\
(\mathsf{select}\ \ell\ e_1)[e/x] & = & \mathsf{select}\ \ell\ (e_1[e/x]) \\
(\mathsf{case}\ e_0\ \mathsf{of}\ \{\ell_1\ x_1 \to e_1, \ldots, \ell_n\ x_n \to e_n\})[e/x] & = & \\
\multicolumn{3}{r}{\mathsf{case}\ e_0[e/x]\ \mathsf{of}\ \{\ell_1\ y_1 \to e_1(x_1 \leftrightarrow y_1)[e/x], \ldots, \ell_n\ y_n \to e_n(x_n \leftrightarrow y_n)[e/x]\}} \\
 & & (y_1, \ldots, y_n \text{ fresh}) \\
\{\!|e_1, \ldots, e_n|\!\}[e/x] & = & \{\!|e_1[e/x], \ldots, e_n[e/x]|\!\} \\
\mathsf{when}(e_1, e_2)[e/x] & = & \mathsf{when}(e_1[e/x], e_2[e/x]) \\
\mathsf{sum}(e_1, e_2)[e/x] & = & \mathsf{sum}(e_1[e/x], e_2[e/x]) \\
\mathsf{diff}(e_1, e_2)[e/x] & = & \mathsf{diff}(e_1[e/x], e_2[e/x]) \\
\mathsf{count}(e_1, e_2)[e/x] & = & \mathsf{count}(e_1[e/x], e_2[e/x]) \\
\mathsf{flatMap}(e_1, e_2)[e/x] & = & \mathsf{flatMap}(e_1[e/x], e_2[e/x]) \\
\{\!|e_1 \mid p_1, \ldots, p_n|\!\}[e/x] & = & \{\!|e_1[e/x] \mid (p_1, \ldots, p_n)[e/x]|\!\} \\
(x_1 \leftarrow e_1, p_1, \ldots, p_n)[e/x] & = & (y \leftarrow e_1[e/x], (p_1, \ldots, p_n)(x_1 \leftrightarrow y)[e/x]) \\
 & & (y \text{ fresh}) \\
(\mathsf{let}\ x_1 = e_1, p_1, \ldots, p_n)[e/x] & = & (\mathsf{let}\ y = e_1[e/x], (p_1, \ldots, p_n)(x_1 \leftrightarrow y)[e/x]) \\
 & & (y \text{ fresh}) \\
(e_1, p_1, \ldots, p_n)[e/x] & = & (e_1[e/x], (p_1, \ldots, p_n)[e/x])
\end{array}$$

Figure 5: Substitution rules of Frog

$$\begin{array}{rcl}
e : \tau & \longrightarrow & e \\
\mathsf{sig}\ f : \tau\ \mathsf{let\ fun}\ f(x) = e_1\ \mathsf{in}\ e_2 & \longrightarrow & \mathsf{let}\ f = \backslash x.e_1\ \mathsf{in}\ e_2 \\
\mathsf{sig}\ f : \tau\ \mathsf{let\ rec}\ f(x) = e_1\ \mathsf{in}\ e_2 & \longrightarrow & \mathsf{let}\ f = \mathsf{rec} f(x).e_1\ \mathsf{in}\ e_2 \\
\mathsf{let}\ (x, y) = e_1\ \mathsf{in}\ e_2 & \longrightarrow & \mathsf{let}\ p = e_1\ \mathsf{in}\ e_2[\mathsf{fst}(p)/x, \mathsf{snd}(p)/y] \\
 & & \quad (p \notin FV(e_2)) \\
\mathsf{let}\ \langle \ell_1 = x_1, \ldots, \ell_n = x_n \rangle = e_1\ \mathsf{in}\ e_2 & \longrightarrow & \mathsf{let}\ r = e_1\ \mathsf{in}\ e_2[(r.\ell_1)/x_1, \ldots, (r.\ell_n)/x_n] \\
 & & \quad (r \notin FV(e_2)) \\
\{\!|e \mid |\!\} & \longrightarrow & \{\!|e|\!\} \\
\{\!|e \mid x \leftarrow e', p_1, \ldots, p_n|\!\} & \longrightarrow & \mathsf{flatMap}(e', \backslash x.\{\!|e \mid p_1, \ldots, p_n|\!\}) \\
\{\!|e \mid \mathsf{let}\ x = e', p_1, \ldots, p_n|\!\} & \longrightarrow & \mathsf{let}\ x = e'\ \mathsf{in}\ \{\!|e \mid p_1, \ldots, p_n|\!\} \\
\{\!|e \mid e', p_1, \ldots, p_n|\!\} & \longrightarrow & \mathsf{when}(e', \{\!|e \mid p_1, \ldots, p_n|\!\})
\end{array}$$

Figure 6: Desugaring rules of Frog

$\boxed{e \Downarrow v}$

$$\frac{v \text{ is a value}}{v \Downarrow v} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{e_1 + e_2 \Downarrow v_1 +_{\mathbb{N}} v_2} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{e_1 - e_2 \Downarrow v_1 -_{\mathbb{N}} v_2} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{e_1 * e_2 \Downarrow v_1 *_{\mathbb{N}} v_2} \qquad \frac{e \Downarrow \mathsf{true} \quad e_1 \Downarrow v}{\mathsf{if}\ e\ \mathsf{then}\ e_1\ \mathsf{else}\ e_2 \Downarrow v}$$

$$\frac{e \Downarrow \mathsf{false} \quad e_2 \Downarrow v}{\mathsf{if}\ e\ \mathsf{then}\ e_1\ \mathsf{else}\ e_2 \Downarrow v} \qquad \frac{e_1 \Downarrow v \quad e_2 \Downarrow v}{e_1 == e_2 \Downarrow \mathsf{true}} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2 \quad (v_1 \neq v_2)}{e_1 == e_2 \Downarrow \mathsf{false}} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{e_1 < e_2 \Downarrow n_1 <_{\mathbb{N}} n_2}$$

$$\frac{e \Downarrow v}{\mathsf{length}(e) \Downarrow \text{length of string } v} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{\mathsf{index}(e_1, e_2) \Downarrow \text{the } v_2\text{-th character of } v_1}$$

$$\frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{\mathsf{concat}(e_1, e_2) \Downarrow \text{concatenation of } v_1 \text{ and } v_2} \qquad \frac{e_1 \Downarrow \backslash x.e \quad e_2 \Downarrow v_1 \quad e[v_1/x] \Downarrow v_2}{e_1\, e_2 \Downarrow v_2}$$

$$\frac{e_1 \Downarrow \mathsf{rec} f(x).e \quad e_2 \Downarrow v_1 \quad e[v_1/x, \mathsf{rec} f(x).e/f] \Downarrow v_2}{e_1\, e_2 \Downarrow v_2} \qquad \frac{e_1 \Downarrow v_1 \quad e_2[v_1/x] \Downarrow v}{\mathsf{let}\ x = e_1\ \mathsf{in}\ e_2 \Downarrow v} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{(e_1, e_2) \Downarrow (v_1, v_2)}$$

$$\frac{e \Downarrow (v_1, v_2)}{\mathsf{fst}(e) \Downarrow v_1} \qquad \frac{e \Downarrow (v_1, v_2)}{\mathsf{snd}(e) \Downarrow v_1} \qquad \frac{e_1 \Downarrow v_1 \quad \cdots \quad e_n \Downarrow v_n}{\langle \ell_1 = e_1, \ldots, \ell_n = e_n \rangle \Downarrow \langle \ell_1 = v_1, \ldots, \ell_n = v_n \rangle} \qquad \frac{e \Downarrow \langle \ell_1 = v_1, \ldots, \ell_n = v_n \rangle}{e.\ell_j \Downarrow v_j}$$

$$\frac{e \Downarrow v}{\mathsf{select}\ \ell\ e \Downarrow \mathsf{select}\ \ell\ v} \qquad \frac{e \Downarrow \mathsf{select}\ \ell_j\ v_j \quad e_j[v_j/x_j] \Downarrow v}{\mathsf{case}\ e\ \mathsf{of}\ \{\ell_1\, x_1 \to e_1, \ldots, \ell_n\, x_n \to e_n\} \Downarrow v} \qquad \frac{e_1 \Downarrow v_1 \quad \cdots \quad e_n \Downarrow v_n}{\{\!| e_1, \ldots, e_n |\!\} \Downarrow \{\!| v_1, \ldots, v_n |\!\}}$$

$$\frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{\mathsf{flatMap}(e_1, e_2) \Downarrow \text{flatMap of the multiset } v_1 \text{ and function } v_2} \qquad \frac{e_1 \Downarrow \mathsf{true} \quad e_2 \Downarrow v}{\mathsf{when}(e_1, e_2) \Downarrow v} \qquad \frac{e_1 \Downarrow \mathsf{false}}{\mathsf{when}(e_1, e_2) \Downarrow \{\!||\!\}}$$

$$\frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{\mathsf{sum}(e_1, e_2) \Downarrow \text{sum of two multisets } v_1, v_2} \qquad \frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{\mathsf{diff}(e_1, e_2) \Downarrow \text{difference of two multisets } v_1, v_2}$$

$$\frac{e_1 \Downarrow v_1 \quad e_2 \Downarrow v_2}{\mathsf{count}(e_1, e_2) \Downarrow \text{count of } v_2 \text{ in the multiset } v_1}$$

Figure 7: Evaluation rules of Frog