3.4 Denotational Semantic

We will now expore the semantics of the formal language. To do so, we first define a new context.

Definition 3.1: Variable Context

Let Γ be a type context.

—

$\Delta : \mathcal{V} \nrightarrow \bigcup_{T \in \mathcal{T}} \text{value}_\Gamma(T)$ is called a *variable context.*

The semantics of the type signature was already defined in the last section, as the semantic of a type signature is its type. We therefore define the same concept but now in a denotational style.

Definition 3.2: Type Signature Semantic

Let $T, T' \in \mathcal{T}$, $c, a_0, a \in \mathcal{V}$. Let $t_0, t_1, t_2 \in$ `<type>`, $ltf \in$ `<list-type-fields>` and $lt \in$ `<list-type>`. Let Γ be a type context.

—

$$\begin{aligned}
[\![.]\!]_\Gamma :& \texttt{<list-type-fields>} \to (\mathcal{V} \times \mathcal{T})^* \\
[\![\texttt{""}]\!]_\Gamma =& (\ ) \\
[\![a_0 \quad \texttt{":"} \quad t_0 \quad \texttt{","} \quad ltf]\!]_\Gamma =& ((a_0, T_0), \ldots, (a_n, T_n)) \\
& \text{such that } T_0 = [\![t_0]\!]_\Gamma \\
& \text{and } [\![ltf]\!]_\Gamma = ((a_1, T_1), \ldots, (a_n, T_n)) \\
& \text{where } n \in \mathbb{N} \text{ and } T_i \in \mathcal{T}, a_i \in \mathcal{V} \text{ for all } i \in \mathbb{N}_0^n
\end{aligned}$$

$$\begin{aligned}
[\![.]\!]_\Gamma :& \texttt{<list-type>} \to \mathcal{T}^* \\
[\![\texttt{""}]\!]_\Gamma =& (\ ) \\
[\![t_0 \ \ lt]\!]_\Gamma =& (T_0, \ldots, T_n) \\
& \text{such that } T_0 = [\![t_0]\!]_\Gamma \\
& \text{and } [\![lt]\!]_\Gamma = (T_1, \ldots, T_n) \\
& \text{where } n \in \mathbb{N} \text{ and } T_i \in \mathcal{T} \text{ for all } i \in \mathbb{N}_0^n
\end{aligned}$$

$$
\begin{aligned}
[\![.]\!]_\Gamma :& \texttt{<type>} \to \mathcal{T} \\
[\![\texttt{"Bool"}]\!]_\Gamma =& Bool \\
[\![\texttt{"Int"}]\!]_\Gamma =& Int \\
[\![\texttt{"List"} \quad t]\!]_\Gamma =& List\ T \\
& \text{such that } T = [\![\texttt{t}]\!]_\Gamma \\
& \text{where } T \in \mathcal{T} \\
[\![\texttt{"("} \quad t_1 \quad \texttt{","} \quad t_2 \quad \texttt{")"}]\!]_\Gamma =& (T_1, T_2) \\
& \text{such that } T_1 = [\![t_1]\!]_\Gamma \text{ and } T_2 = [\![t_2]\!]_\Gamma \\
& \text{where } T_1, T_2 \in \mathcal{T} \\
[\![\texttt{"\{"} \quad \mathit{ltf} \quad \texttt{"\}"}]\!]_\Gamma =& \{a_1 : T_1, \dots, a_n : T_n\} \\
& \text{such that } [\![\mathit{ltf}]\!]_\Gamma = ((a_1, T_1), \dots, (a_n, T_n)) \\
& \text{where } n \in \mathbb{N} \text{ and } T_i \in \mathcal{T}, a_i \in \mathcal{V} \text{ for all } i \in \mathbb{N}_0^n \\
[\![t_1 \quad \texttt{"->"} \quad t_2]\!]_\Gamma =& T_1 \to T_2 \\
& \text{such that } [\![t_1]\!]_\Gamma = T_1 \text{ and } [\![t_2]\!]_\Gamma = T_2 \\
[\![c \ \ \mathit{lt}]\!]_\Gamma =& \overline{T}\ T_1 \dots T_n \\
& \text{such that } (c, T) \in \Gamma \\
& \text{and } (T_1, \dots, T_n) = [\![\mathit{lt}]\!]_\Gamma \\
& \text{where } n \in \mathbb{N}, T \in \mathcal{T} \text{ and } T_i \in \mathcal{T} \text{ for all } i \in \mathbb{N}_1^n \\
[\![a]\!]_\Gamma =& \forall b.b
\end{aligned}
$$

An Elm program is nothing more than an expression. Semantics of an expression is therefore the heart piece of this section.

Definition 3.3: Expression Semantic

Let Γ be a type context and let Δ, Θ be variable contexts. Let $a, a_0, a_1 \in \mathcal{V}$, $e, e_1, e_2, e_3 \in$ `<exp>`. Let $\mathit{lef} \in$ `<list-exp-field>`, $t \in$ `<type>`, $p \in$ `<pattern>`, $\mathit{lc} \in$ `<list-case>`, $b \in$ `<bool>`, $\mathit{nr} \in \mathbb{N}$, $\mathit{le} \in$ `<list-exp>` and $\mathit{mes} \in$ `<maybe-expression-sign>`.

—

$$\llbracket . \rrbracket_{\Gamma,\Delta} : \texttt{<list-exp-field>} \to (\mathcal{V} \times \bigcup_{T \in \mathcal{T}} \text{value}_\Gamma(T))^*$$

$$\begin{aligned}
\llbracket a \quad \texttt{"="} \quad e \rrbracket_{\Gamma,\Delta} =& \{a = s_2\} \\
& \text{such that } s_2 = \llbracket e \rrbracket_{\Gamma,\Delta} \\
& \text{where } s_2 \in \text{value}_\Gamma(T) \text{ for } T \in \mathcal{T} \\
\llbracket a_1 \quad \texttt{"="} \quad e \quad \texttt{","} \quad lef \rrbracket_{\Gamma,\Delta} =& \{a_1 = s_1, \dots, a_n = s_n\} \\
& \text{such that } \{a_1 = s_1\} = \llbracket a \quad \texttt{"="} \quad e \rrbracket_{\Gamma,\Delta} \\
& \text{and } \{a_2 = s_2, \dots, a_n = s_n\} = \llbracket lef \rrbracket_{\Gamma,\Delta} \\
& \text{where } \; n \in \mathbb{N} \text{ and } a_i \in \mathcal{V}, s_i \in \text{value}_\Gamma(T_i) \\
& \text{for } T_i \in \mathcal{T} \text{ for } i \in \mathbb{N}_0^n
\end{aligned}$$

$$\begin{aligned}
\llbracket . \rrbracket :& \texttt{<maybe-exp-sign>} \to (\;) \\
\llbracket \texttt{""} \rrbracket =& (\;) \\
\llbracket a \quad \texttt{":"} \quad t \quad \texttt{";"} \rrbracket =& (\;)
\end{aligned}$$

$$\llbracket . \rrbracket : \texttt{<bool>} \to \text{value}_\varnothing(\mathit{Bool})$$

$$\llbracket b \rrbracket = \begin{cases} \mathit{True} & \text{if } b = \texttt{"True"} \\ \mathit{False} & \text{if } b = \texttt{"False"} \end{cases}$$

$$\begin{aligned}
\llbracket . \rrbracket :& \texttt{<int>} \to \text{value}_\varnothing(\mathit{Int}) \\
\llbracket \texttt{"0"} \rrbracket =& 0 \\
\llbracket \texttt{"-"} \quad nr \rrbracket =& \mathit{Neg}\ \mathit{Succ}^{nr}\ 0 \\
\llbracket nr \rrbracket =& \mathit{Pos}\ \mathit{Succ}^{nr}\ 0
\end{aligned}$$

$$\llbracket . \rrbracket_{\Gamma,\Delta} : \texttt{<list-exp>} \to \bigcup_{T \in \mathcal{T}} \text{value}_\Gamma(T))^*$$

$$\begin{aligned}
\llbracket \texttt{""} \rrbracket_{\Gamma,\Delta} =& \mathit{Empty} \\
\llbracket e \quad \texttt{","} \quad le \rrbracket_{\Gamma,\Delta} =& \mathit{Cons}\ s_1\ s_2 \\
& \text{such that } s_1 = \llbracket e \rrbracket_{\Gamma,\Delta} \wedge s_2 = \llbracket le \rrbracket_{\Gamma,\Delta} \\
& \text{where } \; n \in \mathbb{N} \; \text{ and } \; s_i \in \text{value}_\Gamma(T_i), T_i \in \mathcal{T} \text{ for each } i \in \mathbb{N}_0^n
\end{aligned}$$

Let $s \in \bigcup_{T \in \mathcal{T}} \text{value}_\Gamma(T)$ for the following function.

$$\llbracket . \rrbracket_{\Gamma,\Delta} : \texttt{<exp>} \to \bigcup_{T \in \mathcal{T}} \text{value}_\Gamma(T)$$

$$\begin{aligned}
\llbracket \texttt{"foldl"} \rrbracket_{\Gamma,\Delta} =& \lambda f.\lambda e_1.\lambda l_1. \begin{cases} e_1 & \text{if } [\;] = l_1 \\ f(e_2, s(f, e_1, l_2)) & \text{if } \mathit{Cons}\ e_2\ l_2 = l_1 \end{cases} \\
& \text{where } \; e_1 \in \text{value}_\Gamma(T_1), e_2 \in \text{value}_\Gamma(T_2) \\
& \text{and } l_1, l_2 \in \text{value}_\Gamma(\mathit{List}\ T_2) \\
& \text{and } f \in \text{value}_\Gamma(T_2 \to T_1 \to T_1) \text{ for } T_1, T_2 \in \mathcal{T}
\end{aligned}$$

$$
\begin{aligned}
[\![\texttt{"(::)"}]\!]_{\Gamma,\Delta} =& \lambda e.\lambda l.Cons\ e\ l \\
& \text{where} \ \ e \in \text{value}_\Gamma(T) \\
& \text{and} \ \ l \in \text{value}_\Gamma(List\ T) \\
& \text{for } T \in \mathcal{T}
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"(+)"}]\!]_{\Gamma,\Delta} =& \lambda n.\lambda m.n + m \\
& \text{where } n, m \in \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"(-)"}]\!]_{\Gamma,\Delta} =& \lambda n.\lambda m.n - m \\
& \text{where } n, m \in \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"(*)"}]\!]_{\Gamma,\Delta} =& \lambda n.\lambda m.n * m \\
& \text{where } n, m \in \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"(//)"}]\!]_{\Gamma,\Delta} =& \lambda n.\lambda m. \begin{cases} \left\lfloor \frac{n}{m} \right\rfloor & \text{if } m \neq 0 \\ 0 & \text{else} \end{cases} \\
& \text{where } n, m \in \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"(<)"}]\!]_{\Gamma,\Delta} =& \lambda n.\lambda m.n < m \\
& \text{where } n, m \in \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"(==)"}]\!]_{\Gamma,\Delta} =& \lambda n.\lambda m.(n = m) \\
& \text{where } n, m \in \mathbb{Z}
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"not"}]\!]_{\Gamma,\Delta} =& \lambda b.\neg b \\
& \text{where } b \in \text{value}_\Gamma(Bool)
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"(\&\&)"}]\!]_{\Gamma,\Delta} =& \lambda b_1.\lambda b_2.b_1 \wedge b_2 \\
& \text{where } b_1, n_2 \in \text{value}_\Gamma(Bool)
\end{aligned}
$$

$$
\begin{aligned}
[\![\texttt{"(||)"}]\!]_{\Gamma,\Delta} =& \lambda b_1.\lambda b_2.b_1 \vee b_2 \\
& \text{where } b_1, n_2 \in \text{value}_\Gamma(Bool)
\end{aligned}
$$

$$
\begin{aligned}
[\![e_1 \quad \texttt{"|>"} \quad e_2]\!]_{\Gamma,\Delta} =& f(s_1) \\
& \text{such that } s' = [\![e_1]\!]_{\Gamma,\Delta} \\
& \text{and } f = [\![e_2]\!]_{\Gamma,\Delta} \\
& \text{where } f \in \text{value}(T_1 \to T_2), s' \in \text{value}(T_1) \text{ for } T_1, T_2 \in \mathcal{T}
\end{aligned}
$$

$$
\begin{aligned}
[\![e_1 \quad \texttt{">>"} \quad e_2]\!]_{\Gamma,\Delta} =& f \circ g \\
& \text{such that } g = [\![e_1]\!]_{\Gamma,\Delta} \\
& \text{and } f = [\![e_2]\!]_{\Gamma,\Delta} \\
& \text{where } g \in \text{value}(T_1 \to T_2), f \in \text{value}(T_2 \to T_3) \text{ for} \\
& T_1, T_2, T_2 \in \mathcal{T}
\end{aligned}
$$

$$\left[\!\left[\begin{array}{l}\texttt{"if"}\ e_1\ \texttt{"then"}\\ e_2\ \texttt{"else"}\ e_3\end{array}\right]\!\right]_{\Gamma,\Delta} = \begin{cases}[\![e_2]\!]_{\Gamma,\Delta} & \text{if } b\\ [\![e_3]\!]_{\Gamma,\Delta} & \text{if } \neg b\end{cases}$$
$$\text{such that } b = [\![e_1]\!]_{\Gamma,\Delta}$$
$$\text{where } b \in \text{value}(\mathit{Bool})$$

$$[\![\texttt{"\{"}\quad lef\quad \texttt{"\}"}]\!]_{\Gamma,\Delta} = [\![lef]\!]_{\Gamma,\Delta}$$

$$[\![\texttt{"\{ \}"}]\!]_{\Gamma,\Delta} = \{\}$$

$$[\![\texttt{"\{"}\quad a\quad \texttt{"|"}\quad lef\quad \texttt{"\}"}]\!]_{\Gamma,\Delta} = \{a_1 = s_1, \ldots, a_m = s_m\}$$
$$\text{such that } \{a_1 = s_1, \ldots, a_n = s_n\} = [\![lef]\!]_{\Gamma,\Delta}$$
$$\text{and } (a, \begin{Bmatrix} a_1 = \_, \ldots, a_n = \_,\\ a_{n+1} = s_{n+1}, \ldots, a_m = s_m\end{Bmatrix}) \in \Delta$$
$$\text{where } n, m \in \mathbb{N} \text{ such that } n \leq m \text{ and } a_i \in \mathcal{V},$$
$$s_i \in \text{value}(T_i), T_i \in \mathcal{T} \text{ for } i \in \mathbb{N}_0^m$$

$$[\![a_0\quad \texttt{"."}\quad a_1]\!]_{\Gamma,\Delta} = s'$$
$$\text{such that } \Delta(a_0) = \{a_1 : s', \ldots\}$$
$$\text{where } s' \in \text{value}(T)\text{for } T \in \mathcal{T}$$

$$\left[\!\left[\begin{array}{l}\texttt{"let"}\ mes\ a\ \texttt{"="}\ e_1\\ \texttt{"in"}\ e_2\end{array}\right]\!\right]_{\Gamma,\Delta} = [\![e_2]\!]_{\Gamma,\Delta\cup\{(a,s')\}}$$
$$\text{such that } s' = [\![e_1]\!]_{\Gamma,\Delta}$$
$$\text{where } s' \in \text{value}(T) for T \in \mathcal{T}$$

$$[\![e_1\ \ e_2]\!]_{\Gamma,\Delta} = s_1(s_2)$$
$$\text{such that } s_1 = [\![e_1]\!]_{\Gamma,\Delta}$$
$$\text{and } s_2 = [\![e_2]\!]_{\Gamma,\Delta}$$
$$\text{where } s_1 \in \text{value}_\Gamma(T_1 \to T_2) \text{ and } s_2 \in \text{value}_\Gamma(T_1)\text{for } T_1, T_2 \in \mathcal{T}$$

$$[\![b]\!]_{\Gamma,\Delta} = [\![b]\!]$$

$$[\![i]\!]_{\Gamma,\Delta} = [\![i]\!]$$

$$[\![\texttt{"["}\quad le\quad \texttt{"]"}]\!]_{\Gamma,\Delta} = [s_1, \ldots, s_n]$$
$$\text{such that } (s_1, \ldots, s_n) = [\![le]\!]_{\Gamma,\Delta}$$
$$\text{where } n \in \mathbb{N} \text{ and } s_i \in \text{value}_\Gamma(T) \text{ for } T \in \mathcal{T}$$

$$[\![\texttt{"("}\quad e_1\quad \texttt{","}\quad e_2\quad \texttt{")"}]\!]_{\Gamma,\Delta} = (s_1, s_2)$$
$$\text{such that } s_1 = [\![e_1]\!]$$
$$\text{and } s_2 = [\![e_2]\!]$$
$$\text{where } s_1 \in \text{value}_\Gamma(T_1) \text{ and } s_2 \in \text{value}_\Gamma(T_1)$$

$$[\![\texttt{"\textbackslash"}\quad a\quad \texttt{"->"}\quad e]\!]_{\Gamma,\Delta} = \lambda b.[\![e]\!]_{\Gamma,\Delta\cup\{(a,b)\}}$$
$$\text{where } b \in \mathcal{V}$$

$$[\![c]\!]_{\Gamma,\Delta} = s \text{ such that } (c,s) \in \Delta$$
$$[\![a]\!]_{\Gamma,\Delta} = s \text{ such that } (a,s) \in \Delta$$

Statements are, semantically speaking, just functions that either map the type- or variable-context.

Definition 3.4: Statement Semantic

Let Γ be a type context. Let $a, a_0 \in \mathcal{V}$, $t \in$ `<type>`, $lsv \in$ `<list-statement-var>`, $lt \in$ `<list-type>`, $lss \in$ `<list-statement-sort>`, $st \in$ `<statement>`, $ls \in$ `<list-statement>`, $mss \in$ `<maybe-statement-sign>` and $mms \in$ `<maybe-main-sign>`. Let $\mathcal{S}$ be the class of all finite sets.

—

$$\begin{aligned}
[\![.]\!] :& \texttt{<list-statement-var>} \to \mathcal{V}^* \\
[\![\texttt{""}]\!] =& (\ ) \\
[\![a_0 \quad lsv]\!] =& (a_0, \dots, a_n) \\
& \text{such that } (a_1, \dots, a_n) = [\![lsv]\!] \\
& \text{where } \ n \in \mathbb{N} \text{ and } a_i \in \mathcal{V} \text{ for } i \in \mathbb{N}_0^n
\end{aligned}$$

$$\begin{aligned}
[\![.]\!] : \texttt{<list-statement>} \to ((\mathcal{V} \nrightarrow \mathcal{T}) \times (\mathcal{V} \nrightarrow \mathcal{S})) \to ((\mathcal{V} \nrightarrow \mathcal{T}) \times (\mathcal{V} \nrightarrow \mathcal{S}))& \\
[\![\texttt{""}]\!] =& id \\
[\![st \quad \texttt{","} \quad ls]\!] =& g \circ f \\
& \text{such that } f = [\![st]\!] \text{ and } g = [\![ls]\!] \\
& \text{where } f, g \in ((\mathcal{V} \nrightarrow \mathcal{T}) \times (\mathcal{V} \nrightarrow \mathcal{S})) \to ((\mathcal{V} \nrightarrow \mathcal{T}) \times (\mathcal{V} \nrightarrow \mathcal{S}))
\end{aligned}$$

$$\begin{aligned}
[\![.]\!] :& \texttt{<maybe-statement-sign>} \to (\ ) \\
[\![\texttt{""}]\!] =& (\ ) \\
[\![a \quad \texttt{":"} \quad t \ \texttt{";"}]\!] =& (\ )
\end{aligned}$$

$$\begin{aligned}
[\![.]\!] : \texttt{<statement>} \to& ((\mathcal{V} \nrightarrow \mathcal{T}) \times (\mathcal{V} \nrightarrow \mathcal{S})) \to ((\mathcal{V} \nrightarrow \mathcal{T}) \times (\mathcal{V} \nrightarrow \mathcal{S})) \\
[\![mss \quad a \quad \texttt{"="} \quad e]\!](\Gamma, \Delta) =& (\Gamma, \Delta \cup \{(a, s')\}) \\
& \text{such that } s' = [\![e]\!]_{\Gamma,\Delta} \\
& \text{where } s' \in \text{value}(T) \text{ for } T \in \mathcal{T} \\
\left[\!\!\left[\begin{matrix}\texttt{"type alias"} \\ c \ lsv \ \texttt{"="} \ t\end{matrix}\right]\!\!\right](\Gamma, \Delta) =& (\Gamma \cup \{(c, T)\}, \Delta) \\
& \text{such that } T = [\![\texttt{t}]\!]_\Gamma
\end{aligned}$$

$$\begin{aligned}
[\![.]\!] :& \texttt{<maybe-main-sign>} \to (\ ) \\
[\![\texttt{""}]\!] =& (\ ) \\
[\![\texttt{"main : "} \quad t \quad \texttt{";"}]\!] =& (\ )
\end{aligned}$$

$$[\![.]\!] : \texttt{<program>} \rightarrow \bigcup_{T \in \mathcal{T}} \text{value}_{\varnothing}(T)$$

$$[\![ls \quad mms \quad \texttt{"main = "} \quad e]\!] = [\![e]\!]_{\Gamma, \Delta}$$

such that $(\Gamma, \Delta) = [\![ls]\!](\varnothing, \varnothing)$

where Γ is a type context and Δ is a variable context.