## Syntax

$$
\begin{array}{lll}
\text{Literals} & lit & ::= None \mid True \mid False \mid ”a” \mid \ \ldots \mid 1 \mid \ \ldots \\
\text{Expression} & Exp & ::= lit\texttt{@}(\overline{A}) \mid Exp.id \mid f(\overline{Exp}) \mid Exp.f(\overline{Exp}) \mid C[\overline{A}](\overline{Exp}) \\
\text{Typed Expression} & TExp & ::= lit\texttt{@}(\overline{A}) : \tau \mid \ \ldots \\
\text{Assign Op.} & AsgOp & \in \{\texttt{=, +=, -=, *=, /=, \%=, //=}\} \\
\text{Binary Op.} & BinOp & \in \{\texttt{||, \&\&, |, \&, ==, !=, <, >, <=, >=, +, -, *, /, \%, **}\} \\
\text{Statement} & Stm & ::= \textsf{pass} \mid \textsf{return}\ \ Exp \mid Exp\ ; Stm \mid id = Exp\ ; Stm \\
 & & \quad \mid Exp_1\ \ AsgOp\ Exp_2\ ; Stm \mid \textsf{if}\ \ Exp : Stm_1\ ; \textsf{else} : Stm_2\ ; Stm \\
 & & \quad \mid \textsf{try} : Stm\ \ \textsf{except} :\ \ Exp\ ; Stm \mid \textsf{raise}\ \ Exp
\end{array}
$$

## Projection To Python

$$
(Exp) \qquad \llparenthesis lit\texttt{@}(\overline{B}) : \tau \rrparenthesis^A = \begin{cases} lit & \text{if } \ A \in \overline{B} \\ \texttt{Unit.id} & \text{otherwise} \end{cases}
$$

$$
\llparenthesis Exp.id : \tau \rrparenthesis^A = \begin{cases} \llparenthesis Exp \rrparenthesis^A.id & \text{if } \ A \in \text{rolesOf}(Exp.id) \\ \text{absent} & \text{otherwise} \end{cases}
$$

$$
\llparenthesis f(\overline{Exp}) : \tau \rrparenthesis^A = \begin{cases} f(\overline{\llparenthesis Exp \rrparenthesis^A}) & \text{if } \ A \in \text{rolesOf}(f(\overline{Exp})) \\ \texttt{Unit.id}(f(\overline{\llparenthesis Exp \rrparenthesis^A})) & \text{if } \ A \in \text{rolesOf}(\overline{Exp}) \wedge A \notin \text{rolesOf}(f(\overline{Exp})) \\ \texttt{Unit.id}(\overline{\llparenthesis Exp \rrparenthesis^A}) & \text{otherwise} \end{cases}
$$

$$
\llparenthesis Exp.f(\overline{Exp}) : \tau \rrparenthesis^A = \begin{cases} \llparenthesis Exp \rrparenthesis^A.f(\overline{\llparenthesis Exp \rrparenthesis^A}) \\ \qquad \text{if } \ A \in \text{rolesOf}(Exp) \wedge A \in \text{rolesOf}(\overline{Exp}) \\ \qquad\qquad\qquad\qquad \wedge A \in \text{rolesOf}(Exp.f(\overline{Exp})) \\ \texttt{Unit.id}(\llparenthesis Exp \rrparenthesis^A.f(\overline{\llparenthesis Exp \rrparenthesis^A})) \\ \qquad \text{if } \ A \in \text{rolesOf}(Exp) \wedge A \notin \text{rolesOf}(Exp.f(\overline{Exp})) \\ \texttt{Unit.id}(\llparenthesis Exp \rrparenthesis^A, \overline{\llparenthesis Exp \rrparenthesis^A}) \qquad \text{otherwise} \end{cases}
$$

$$
\llparenthesis C[\overline{B}](\overline{Exp}) : \tau \rrparenthesis^A = \begin{cases} \llparenthesis C[\overline{B}] \rrparenthesis^A(\overline{\llparenthesis Exp \rrparenthesis^A}) & A \in \overline{B} \\ \texttt{Unit.id}(\overline{\llparenthesis Exp \rrparenthesis^A}) & \text{otherwise} \end{cases}
$$

$$
\text{rolesOf}(\_ : \tau\texttt{@}(\overline{B})) = \overline{B}
$$

$$
\text{rolesOf}(\overline{Exp}) = \bigcup_i \text{rolesOf}(Exp_i)
$$

$$
\llparenthesis \overline{Exp} \rrparenthesis^A = Exp'_1, Exp'_2, \cdots, Exp'_n \quad \text{where} \quad Exp'_i = \llparenthesis Exp_i \rrparenthesis^A
$$

$$(\mathit{Stm})\quad \llparenthesis \textsf{pass} \rrparenthesis^{A} = \textsf{pass}$$

$$\llparenthesis \textsf{return}\ \mathit{Exp};\ \rrparenthesis^{A} = \textsf{return}\ \llparenthesis \mathit{Exp} \rrparenthesis^{A}$$

$$\llparenthesis \mathit{Exp}; \mathit{Stm} \rrparenthesis^{A} = \begin{cases} \textsf{match}\ \llparenthesis \mathit{Exp} \rrparenthesis^{A} : & \text{if}\ \ \mathit{Exp} = \mathit{Exp}.f(\overline{\mathit{Exp}}) : \texttt{Enum@}A \quad \text{and} \\ \qquad \textsf{case}\ \mathit{id} : \llparenthesis \mathit{Stm} \rrparenthesis^{A}; & \qquad \text{Name}(f) = \texttt{Select} \\ \qquad \textsf{case}\ \_\_ : \text{assert False}; & \\ \llparenthesis \mathit{Exp} \rrparenthesis^{A}; \llparenthesis \mathit{Stm} \rrparenthesis^{A} & \text{otherwise} \end{cases}$$

$$\llparenthesis \mathit{id} : \mathit{TE} = \mathit{Exp}\ ; \mathit{Stm} \rrparenthesis^{A} = \begin{cases} \mathit{id} = \llparenthesis \mathit{Exp} \rrparenthesis^{A}; \llparenthesis \mathit{Stm} \rrparenthesis^{A} & \text{if}\ \ A \in \text{rolesOf}(\mathit{TE}) \\ \llparenthesis \mathit{Exp} \rrparenthesis^{A}; \llparenthesis \mathit{Stm} \rrparenthesis^{A} & \text{otherwise} \end{cases}$$

$$\llparenthesis \mathit{Exp}_1\ \mathit{AsgOp}\ \mathit{Exp}_2\ ; \mathit{Stm} \rrparenthesis^{A} = \llparenthesis \mathit{Exp}_1 \rrparenthesis^{A} \mathit{AsgOp}\ \llparenthesis \mathit{Exp}_2 \rrparenthesis^{A}\ ; \llparenthesis \mathit{Stm} \rrparenthesis^{A}$$

$$\llparenthesis \textsf{if}\ \mathit{Exp} : \mathit{Stm}_1\ ; \textsf{else} : \mathit{Stm}_2\ ; \mathit{Stm} \rrparenthesis^{A} =$$

$$\begin{cases} \textsf{if}\ \llparenthesis \mathit{Exp} \rrparenthesis^{A} : \llparenthesis \mathit{Stm}_1 \rrparenthesis^{A}\ ; \textsf{else} : \llparenthesis \mathit{Stm}_2 \rrparenthesis^{A}\ ; \llparenthesis \mathit{Stm} \rrparenthesis^{A} & \text{if}\ \ \text{rolesOf}(\mathit{Exp}) = A \\ \llparenthesis \mathit{Exp} \rrparenthesis^{A}\ ; \llbracket \llparenthesis \mathit{Stm}_1 \rrparenthesis^{A} \rrbracket \sqcup \llbracket \llparenthesis \mathit{Stm}_2 \rrparenthesis^{A} \rrbracket\ ; \llparenthesis \mathit{Stm} \rrparenthesis^{A} & \text{otherwise} \end{cases}$$

$$\llparenthesis \textsf{try} : \mathit{Stm}\ ;\ \textsf{except}\ \mathit{Exp}\ \ : \mathit{Stm}\ ; \mathit{Stm} \rrparenthesis^{A} =$$

$$\begin{cases} \textsf{try} : \llparenthesis \mathit{Stm} \rrparenthesis^{A}\ ;\ \overline{\textsf{except}\ \llparenthesis \mathit{Exp} \rrparenthesis^{A} : \llparenthesis \mathit{Stm} \rrparenthesis^{A}}\ ; \llparenthesis \mathit{Stm} \rrparenthesis^{A} & \text{if}\ \ A \in \text{rolesOf}(\mathit{Exp}) \end{cases}$$

## Merging

Statement

$$\textsf{return}\ \mathit{Exp} \sqcup \textsf{return}\ \mathit{Exp}' = \textsf{return}\ \mathit{Exp} \sqcup \mathit{Exp}'$$

$$(\mathit{Exp}_1\ \mathit{AsgOp}\ \mathit{Exp}_2; \mathit{Stm}) \sqcup (\mathit{Exp}_1'\ \mathit{AsgOp}\ \mathit{Exp}_2'; \mathit{Stm}')$$
$$= (\mathit{Exp}_1 \sqcup \mathit{Exp}_1')\ \mathit{AsgOp}\ (\mathit{Exp}_2 \sqcup \mathit{Exp}_2'); (\mathit{Stm} \sqcup \mathit{Stm}')$$

$$(\mathit{Exp}; \mathit{Stm}) \sqcup (\mathit{Exp}'; \mathit{Stm}') = (\mathit{Exp} \sqcup \mathit{Exp}'); (\mathit{Stm} \sqcup \mathit{Stm}')$$

$$(\textsf{if}\ \mathit{Exp} : \mathit{Stm}_1\ ; \textsf{else} : \mathit{Stm}_2\ ; \mathit{Stm}) \sqcup (\textsf{if}\ \mathit{Exp}' : \mathit{Stm}_1'\ ; \textsf{else} : \mathit{Stm}_2'\ ; \mathit{Stm}')$$
$$= \textsf{if}\ (\mathit{Exp} \sqcup \mathit{Exp}') : (\mathit{Stm}_1 \sqcup \mathit{Stm}_1')\ ; \textsf{else} : (\mathit{Stm}_2 \sqcup \mathit{Stm}_2')\ ; (\mathit{Stm} \sqcup \mathit{Stm}')$$

$$\begin{array}{llllll} \textsf{match}\ \mathit{Exp} : & & \textsf{match}\ \mathit{Exp}' : & & \textsf{match}\ \mathit{Exp} \sqcup \mathit{Exp}' : \\ \quad \textsf{case}\ \mathit{id}_a : \mathit{Stm}_a'; & & \quad \textsf{case}\ \mathit{id}_a : \mathit{Stm}_a''; & & \quad \textsf{case}\ \mathit{id}_a : \mathit{Stm}_a' \sqcup \mathit{Stm}_a''; \\ \quad \ldots & & \quad \ldots & & \quad \ldots \\ \quad \textsf{case}\ \mathit{id}_x : \mathit{Stm}_x'; & \sqcup & \quad \textsf{case}\ \mathit{id}_x : \mathit{Stm}_x''; & = & \quad \textsf{case}\ \mathit{id}_x : \mathit{Stm}_x' \sqcup \mathit{Stm}_x''; \\ \quad \textsf{case}\ \mathit{id}_y : \mathit{Stm}_y'; & & & & \quad \textsf{case}\ \mathit{id}_y : \mathit{Stm}_y'; \\ & & \quad \textsf{case}\ \mathit{id}_z : \mathit{Stm}_z'; & & \quad \textsf{case}\ \mathit{id}_z : \mathit{Stm}_z'; \\ \quad \textsf{case}\ \ \_\_ : \mathit{Stm}_{ex}'; & & \quad \textsf{case}\ \ \_\_ : \mathit{Stm}_{ex}''; & & \quad \textsf{case}\ \ \_\_ : \mathit{Stm}_{ex}' \sqcup \mathit{Stm}_{ex}''; \\ \mathit{Stm} & & \mathit{Stm}' & & \mathit{Stm} \sqcup \mathit{Stm}' \end{array}$$

*tryexcept*

Expression

$$f(\overline{\mathit{Exp}}) \sqcup f(\overline{\mathit{Exp}'}) = f(\overline{\mathit{Exp} \sqcup \mathit{Exp}'})$$

$$\mathit{Exp}.f(\overline{\mathit{Exp}}) \sqcup \mathit{Exp}'.f(\overline{\mathit{Exp}'}) = (\mathit{Exp} \sqcup \mathit{Exp}').f(\overline{\mathit{Exp} \sqcup \mathit{Exp}'})$$

## Normaliser

Statements

$$\llbracket \textsf{pass} \rrbracket = \textsf{pass}$$

$$\llbracket \textsf{return}\ Exp \rrbracket = \textsf{return}\ \llbracket Exp \rrbracket$$

$$\textsc{noop}(Exp) = \begin{cases} [blank] & \text{if} \quad Exp \in \{\texttt{Unit.id, None}\} \\ Exp & \text{otherwise} \end{cases}$$

$$\llbracket Exp_1\ AsgOp\ Exp_2; Stm \rrbracket = \begin{cases} \llbracket Exp_1 \rrbracket; \llbracket Stm \rrbracket & \text{if} \quad \textsc{noop}(\llbracket Exp_2 \rrbracket) = [blank] \\ \llbracket Exp_2 \rrbracket; \llbracket Stm \rrbracket & \text{if} \quad \textsc{noop}(\llbracket Exp_1 \rrbracket) = [blank] \\ \llbracket Stm \rrbracket & \text{if} \quad \textsc{noop}(\llbracket Exp_1 \rrbracket, \llbracket Exp_2 \rrbracket) = [blank] \\ \llbracket Exp_1 \rrbracket AsgOp \llbracket Exp_2 \rrbracket; \llbracket Stm \rrbracket & \text{otherwise} \end{cases}$$

$$\llbracket Exp; Stm \rrbracket = \begin{cases} \llbracket Stm \rrbracket & \text{if} \quad \textsc{noop}(\llbracket Exp \rrbracket) = [blank] \\ \llbracket Exp \rrbracket; \llbracket Stm \rrbracket & \text{otherwise} \end{cases}$$

$$\llbracket \textsf{if}\ Exp : Stm_1\ ;\ \textsf{else} : Stm_2\ ; Stm \rrbracket = \textsf{if}\ \llbracket Exp \rrbracket : \llbracket Stm_1 \rrbracket\ ;\ \textsf{else} : \llbracket Stm_2 \rrbracket\ ; \llbracket Stm \rrbracket$$

$$\llbracket \textsf{match}\ Exp : \overline{\textsf{case}\ id : Stm'}; \textsf{case}\ \_ : Stm''; Stm \rrbracket = \textsf{match} \llbracket Exp \rrbracket : \overline{\textsf{case}\ id : \llbracket Stm' \rrbracket};\ \textsf{else} : \llbracket Stm'' \rrbracket; \llbracket Stm \rrbracket$$