

LAMBDA

MODULE LAMBDA

SYNTAX $Exp ::= Int$
| $Bool$
| Id
| (Exp) [bracket(bracket())]
| $Exp\ Exp$ [strict(strict())]
| $Exp * Exp$ [strict(strict())]
| Exp / Exp [strict(strict())]
| $Exp + Exp$ [strict(strict())]
| $Exp <= Exp$ [strict(strict())]
| $\text{lambda } Id . Exp$
| $\text{if } Exp \text{ then } Exp \text{ else } Exp$ [strict(strict())]
| $\text{let } Id = Exp \text{ in } Exp$ [strict(strict(2))]
| $\text{letrec } Id\ Id = Exp \text{ in } Exp$
| $\text{mu } Id . Exp$

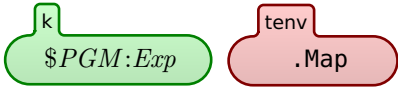
SYNTAX $Type ::= \text{int}$
| bool
| $Type \rightarrow Type$
| $(Type)$ [bracket(bracket())]

SYNTAX $Exp ::= Type$

SYNTAX $Variable ::= Id$

SYNTAX $KResult ::= Type$

CONFIGURATION:



RULE $\frac{I: Int}{\text{int}}$

RULE $\frac{B: Bool}{\text{bool}}$

RULE $\frac{\frac{X: Id}{T}}{X \mapsto T: Type}$

RULE $\frac{T1: Type * T2: Type}{T1 = \text{int} \curvearrowright T2 = \text{int} \curvearrowright \text{int}}$

RULE $\frac{T1: Type / T2: Type}{T1 = \text{int} \curvearrowright T2 = \text{int} \curvearrowright \text{int}}$

RULE $\frac{T1: Type + T2: Type}{T1 = \text{int} \curvearrowright T2 = \text{int} \curvearrowright \text{int}}$

RULE $\frac{T1: Type <= T2: Type}{T1 = \text{int} \curvearrowright T2 = \text{int} \curvearrowright \text{bool}}$

RULE $\frac{\frac{\text{lambda } X: Id . E: Exp}{E \curvearrowright T: Type \rightarrow \square \curvearrowright \text{tenv } (TEnv)}}{TEnv \over TEnv[X <- T]}$

RULE $\frac{T2: Type \curvearrowright T1: Type \rightarrow \square}{T1 \rightarrow T2}$

RULE $\frac{T1: Type\ T2: Type}{T1 = (T2 \rightarrow T: Type) \curvearrowright T}$

RULE $\frac{\text{if } T: Type \text{ then } T1: Type \text{ else } T2: Type}{T = \text{bool} \curvearrowright T1 = T2 \curvearrowright T1}$

SYNTAX $TypeSchema ::= (\text{forall } Set) Type$

SYNTAX $Type ::= MetaVariable$

RULE $\frac{\frac{\text{let } X = T: Type \text{ in } E}{E \curvearrowright \text{tenv } (TEnv)}}{TEnv \over TEnv[X <- (\text{forall } \#metaVariables (T) \rightarrow_{Set} \#metaVariables (\text{tenv } (TEnv))) \#freezeVariables (T, \text{tenv } (TEnv))]}$

RULE $\frac{\frac{X: Id}{\#rename (T, Tvs)}}{X \mapsto (\text{forall } Tvs) T}$

RULE $\frac{\text{letrec } F\ X = E \text{ in } E'}{\text{let } F = \text{mu } F . \text{lambda } X . E \text{ in } E'}$

RULE $\frac{\frac{\text{mu } X: Id . E: Exp}{(T: Type \rightarrow T) \ E \curvearrowright \text{tenv } (TEnv)}}{TEnv \over TEnv[X <- T]}$

SYNTAX $KItem ::= Type = Type$

RULE $\frac{T = T}{\bullet_K}$

SYNTAX $KItem ::= \text{tenv } (Map)$ [klabel(klabel('tenv))]

RULE $\frac{\frac{T: Type \curvearrowright \text{tenv } (TEnv)}{\bullet_K}}{\text{tenv } \over TEnv}$

END MODULE

[macro(macro())]