LAMBDA

MODULE LAMBDA

SYNTAX $Val ::= Id$
      | $\lambda Id.Exp$ [binder]

SYNTAX $Exp ::= Val$
      | $Exp\ Exp$ [strict]
      | (Exp) [bracket]

SYNTAX $Variable ::= Id$

SYNTAX $KResult ::= Val$

RULE $\dfrac{(\lambda X{:}Id.E{:}Exp)\ \ V{:}Val}{E[V\ /\ X]}$

SYNTAX $Val ::= Int$
      | $Bool$

SYNTAX $Exp ::= Exp * Exp$ [strict]
      | $Exp\ /\ Exp$ [strict]
      | $Exp + Exp$ [strict]
      | $Exp <= Exp$ [strict]

RULE $\dfrac{I1{:}Int * I2{:}Int}{I1 *_{Int} I2}$

RULE $\dfrac{I1{:}Int\ /\ I2{:}Int}{I1 \div_{Int} I2}$

RULE $\dfrac{I1{:}Int + I2{:}Int}{I1 +_{Int} I2}$

RULE $\dfrac{I1{:}Int <= I2{:}Int}{I1 \leq_{Int} I2}$

SYNTAX $Exp ::=$ if Exp then Exp else Exp [strict(1)]

RULE $\dfrac{\text{if true then } E \text{ else } \text{---}}{E}$

RULE $\dfrac{\text{if false then } \text{---} \text{ else } E}{E}$

SYNTAX $Exp ::=$ let $Id = Exp$ in Exp

RULE $\dfrac{\text{let } X = E \text{ in } E'{:}Exp}{(\lambda X.E')\ E}$ [macro]

SYNTAX $Exp ::=$ letrec $Id\ Id = Exp$ in Exp
      | $\mu Id.Exp$ [binder]

RULE $\dfrac{\text{letrec } F{:}Id\ \ X = E \text{ in } E'}{\text{let } F = \mu F.\lambda X.E \text{ in } E'}$ [macro]

RULE $\dfrac{\mu X.E}{E[(\mu X.E)\ /\ X]}$

SYNTAX $Exp ::=$ callcc Exp [strict]

SYNTAX $Val ::=$ cc (K) [klabel('cc)]

RULE

$$\dfrac{\text{callcc } V{:}Val \curvearrowright K}{V\ \ \text{cc } (K)}$$
(in k cell)

RULE

$$\dfrac{\text{cc } (K)\ \ V \curvearrowright \text{---}}{V \curvearrowright K}$$
(in k cell)

END MODULE