

LAMBDA

MODULE LAMBDA

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

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

SYNTAX $KResult ::= Val$

RULE
$$\frac{(\lambda X.E)\ V}{E[V / X]}$$

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

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

RULE
$$\frac{I1 * I2}{I1 *_{Int} I2}$$

RULE
$$\frac{I1 / I2}{I1 \div_{Int} I2}$$

RULE
$$\frac{I1 + I2}{I1 +_{Int} I2}$$

RULE
$$\frac{I1 <= I2}{I1 \leq_{Int} I2}$$

SYNTAX $Exp ::= \text{if } Exp \text{ then } Exp \text{ else } Exp$ [strict(1)]

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

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

SYNTAX $Exp ::= \text{let } Id = Exp \text{ in } Exp$

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

SYNTAX $Exp ::= \text{letrec } Id\ Id = Exp \text{ in } Exp$

SYNTAX $Id ::= \$x$
 | $\$y$

RULE
$$\frac{\text{letrec } F\ X = E \text{ in } E'}{\text{let } F = (\lambda \$x.((\lambda F.\lambda X.E)\ (\lambda \$y.(\$x\ \$x\ \$y))))\ (\lambda \$x.((\lambda F.\lambda X.E)\ (\lambda \$y.(\$x\ \$x\ \$y)))) \text{ in } E'}$$
 [macro]

END MODULE