

IMP

MODULE IMP-SYNTAX

```
SYNTAX  AExp ::= Int
        | Id
        | AExp / AExp [strict( strict())]
        | AExp + AExp [strict( strict())]
        | (AExp) [bracket( bracket())]

SYNTAX  BExp ::= Bool
        | AExp ≤ AExp [seqstrict( seqstrict())]
        | ! BExp [strict( strict())]
        | BExp && BExp [strict( strict(1))]
        | (BExp) [bracket( bracket())]

SYNTAX  Block ::= {}
        | {Smt}

SYNTAX  Smt ::= Block
        | Id = AExp ; [strict( strict(2))]
        | if (BExp)Block else Block [strict( strict(1))]
        | while (BExp)Block
        | Smt Smt

SYNTAX  Pgm ::= int Ids ; Smt

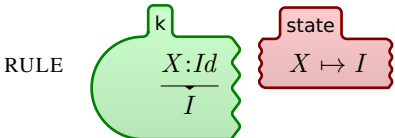
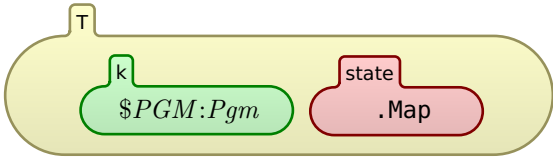
SYNTAX  Ids ::= List{Id, “,”}
```

END MODULE

MODULE IMP

```
SYNTAX  KResult ::= Int
        | Bool
```

CONFIGURATION:



RULE $\frac{I1 \ / \ I2}{I1 \div_{Int} I2}$ requires $I2 \neq_{Int} 0$

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

RULE $\frac{I1 \leq I2}{I1 \leq_{Int} I2}$

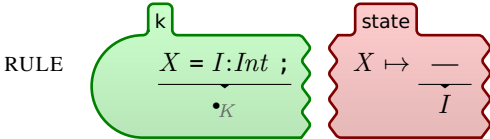
RULE $\frac{! \ T}{\neg_{Bool} T}$

RULE $\frac{\text{true} \ \&\& \ B}{B}$

RULE $\frac{\text{false} \ \&\& \ \text{—}}{\text{false}}$

RULE $\frac{\{\}}{\bullet_K}$

RULE $\frac{\{S\}}{S}$



RULE $\frac{S1:Smt \ S2:Smt}{S1 \curvearrowright S2}$

RULE $\frac{\text{if}(\text{true})S \ \text{else} \ \text{—}}{S}$

RULE $\frac{\text{if}(\text{false})\text{—} \ \text{else} \ S}{S}$

RULE $\frac{\text{while}(B)S}{\text{if}(B)\{S \ \text{while}(B)S\} \ \text{else} \ \{\}}$

RULE

RULE $\frac{\text{int} \ \bullet_{Ids} \ ; \ S}{S}$

END MODULE

[structural(structural())]

[structural(structural())]

[structural(structural())]

[structural(structural())]

[structural(structural())]