

Languages-beta: IMP-3

The PPlanCompS Project

Languages-beta/IMP/IMP-3/IMP-3.cbs*

Language "IMP"

3 Statements and blocks

Syntax $Stmt : stmt ::=$ $block$
| $id = aexp ;$
| $if (bexp) block (else block)?$
| $while (bexp) block$
| $stmt stmt$

Syntax $Block : block ::= \{ stmt? \}$

Rule $\llbracket if (BExp) Block \rrbracket : stmt =$
 $\llbracket if (BExp) Block else \{ \} \rrbracket$

*Suggestions for improvement: plancomps@gmail.com.
Issues: <https://github.com/plancomps/CBS-beta/issues>.

Semantics $\text{execute}[_ : \text{stmt}] : \Rightarrow \text{null-type}$

Rule $\text{execute}[I = AExp ;] =$
 $\text{assign}(\text{bound}(\text{id}[I]),$
 $\text{eval-arith}[AExp])$

Rule $\text{execute}[\text{if } (BExp) \text{ Block}_1 \text{ else Block}_2] =$
 $\text{if-true-else}(\text{eval-bool}[BExp],$
 $\text{execute}[Block_1],$
 $\text{execute}[Block_2])$

Rule $\text{execute}[\text{while } (BExp) \text{ Block}] =$
 $\text{while-true}(\text{eval-bool}[BExp],$
 $\text{execute}[Block])$

Rule $\text{execute}[Stmt_1 Stmt_2] =$
 $\text{sequential}(\text{execute}[Stmt_1],$
 $\text{execute}[Stmt_2])$

Rule $\text{execute}[\{ \}] =$
 null

Rule $\text{execute}[\{ Stmt \}] =$
 $\text{execute}[Stmt]$