

Languages-beta: SL-4-Definitions *

The PPlanCompS Project

SL-4-Definitions.cbs | PLAIN | PRETTY

OUTLINE

4 Function definitions

Parameter lists

Language “SL”

4 Function definitions

Syntax $FunDef : \text{fun-def} ::= \text{'function' id '(' id-list? ')'} \text{block}$

Semantics $\text{declare}[\![FunDef^* : \text{fun-def}^*]\!] : \Rightarrow \text{environments}$

Rule $\text{declare}[\![\text{'function' } Id \text{'(' } IdList? \text{'')} \text{Block}]\!] =$

$\{id[\![Id]\!] \mapsto$
 function closure
 scope(
 initialise-local-variables,
 sequential(
 pass-list[\![IdList?]\!],
 handle-return(exec[\![Block]\!]))))\}

Rule $\text{declare}[\![]\!] = \text{map}(\)$

Rule $\text{declare}[\![FunDef \text{ } FunDef^+]\!] =$
 collateral($\text{declare}[\![FunDef]\!], \text{declare}[\![FunDef^+]\!])$

Parameter lists

Syntax $IdList : \text{id-list} ::= \text{id '(' , ' id-list)'}?$

Semantics $\text{pass-list}[\![IdList : \text{id-list?}]\!] : \text{lists}(\text{values}) \Rightarrow \text{null-type}$

Rule $\text{pass-list}[\![]\!] = \text{null-value}$

Rule $\text{pass-list}[\![Id]\!] =$
 local-variable-initialise($id[\![Id]\!]$, checked head given)

Rule $\text{pass-list}[\![Id \text{'(' } IdList]\!] =$
 sequential(
 local-variable-initialise($id[\![Id]\!]$, checked head given),
 give($\text{checked tail given}$, $\text{pass-list}[\![IdList]\!])$)

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