

Formalization of Simple

Pietro Ferrara

November 11, 2010

1 Statements

St ::=	st ₁ := st ₂	(C1)
	t v [:= st ₁]	(C2)
	v	(C3)
	{st ₁ [, st ₂ [, st ₃ [. . .]]}.f	(C4)
	st ₁ [< t ₁ [, t ₂ [. . .]] >]([st ₁ [, st ₂ [. . .]])	(C5)
	new t	(C6)
	c	(C7)
	throw st ₁	(C8)
	skip	(C9)
	cfg	(C10)

where

- t represents the set of type identifiers,
- v represents the set of variable identifiers,
- f represents the set of field identifiers,
- c represents the set of constants (i.e., numbers, strings, ...), and
- cfg represents the set of control flow graphs (see next section).

1.1 An intuition about the semantics

The state of the computation is composed by

- a "normal" state (e.g., an environment - that relates variables to values, and a store - that relates an address and a field name to values, and values are addresses or primitive values like integers),
- a value that represents what is "left" by the previous statement (e.g., the value of the accessed variable or field). For statements (e.g., assignments) that does not return any value, there is a special value called Unit.

2 Control Flow Graph

The cfg blocks are made by list of statements. The edges may have a boolean weight (to represent that the value resulting from the computation of the block has to be true or false in order to traverse that edge), or no weight (to represent something like a goto statement). As sorting edges, each block can have an unweighted edges, or two weighted edges (one with true weight, and the other one with the false weight).

3 Class

3.1 Method declaration

$$\text{Mdec} ::= [\text{mod}_1 [, \text{mod}_2 [\dots]]] \text{ t } m([\text{t}_1 \text{ p}_1 [, \text{t}_2 \text{ p}_2 [\dots]]) = \text{st}$$

where

- mod represents the set of modifiers (e.g., **public**),
- m represents the set of method identifiers,
- t represents the set of type identifiers, and
- p represents the set of parameter identifiers.

3.2 Field declaration

$$\text{Fdec} ::= [\text{mod}_1 [, \text{mod}_2 [\dots]]] \text{ t } f [:= \text{st}]$$

where

- mod represents the set of modifiers (e.g., **public**),
- t represents the set of type identifiers, and
- f represents the set of field identifiers.

3.3 Class declaration

$$\begin{aligned} \text{Cdec} ::= & [\text{mod}_1 [, \text{mod}_2 [\dots]]] \text{ class } c[< \text{t}_1 [, \text{t}_2 [\dots]] >] = \{ \\ & \quad [\text{fdec}_1; [\text{fdec}_2; [\dots]]] \\ & \quad [\text{mdec}_1; [\text{mdec}_2; [\dots]]] \\ & \quad \} \end{aligned}$$

where

- mod represents the set of modifiers (e.g., **public**),
- t represents the set of type identifiers, and
- c represents the set of class identifiers.

4 Package

$$\text{Pdec} ::= \text{package } p \\ [cdec_1 [cdec_2 [\dots]]]$$

where

- p represents the set of package identifiers.