

# Mini-while operational semantics (NAT, SOS)

Laure Gonnord

October 10, 2018

## Abstract syntax

Recall the abstract syntax of the course for expressions:

$$\begin{array}{l} e ::= c \quad \text{constant} \\ | x \quad \text{variable} \\ | e + e \quad \text{addition} \\ | e \times e \quad \text{multiplication} \\ | \dots \end{array}$$

and the mini-while language:

$$\begin{array}{l} S(Smt) ::= x := expr \quad \text{assign} \\ | skip \quad \text{do nothing} \\ | S_1; S_2 \quad \text{sequence} \\ | \text{if } b \text{ then } S_1 \text{ else } S_2 \quad \text{test} \\ | \text{while } b \text{ do } S \text{ done} \quad \text{loop} \end{array}$$

## 1 Semantic rules

Notations are in the course, as well as the inductive definitions for  $\mathcal{A}[e]\sigma$  and  $\mathcal{B}[b]\sigma$ . A *configuration* is a tuple (statement, state).

### 1.1 Natural “big steps” semantics (NAT)

Name	Rule
skip	$(\text{skip}, \sigma) \rightarrow \sigma$
assign	$(x := a, \sigma) \rightarrow \sigma[x \mapsto \mathcal{A}[a]\sigma]$
seq	$\frac{(S_1, \sigma) \rightarrow \sigma' \quad (S_2, \sigma') \rightarrow \sigma''}{((S_1; S_2), \sigma) \rightarrow \sigma''}$
iftrue	$\mathcal{B}[b]\sigma = tt : \frac{(S_1, \sigma) \rightarrow \sigma'}{(\text{if } b \text{ then } S_1 \text{ else } S_2, \sigma) \rightarrow \sigma'}$
iffalse	$\mathcal{B}[b]\sigma = ff : \frac{(S_2, \sigma) \rightarrow \sigma'}{(\text{if } b \text{ then } S_1 \text{ else } S_2, \sigma) \rightarrow \sigma'}$
whiletrue	$\mathcal{B}[b]\sigma = tt : \frac{(S, \sigma) \rightarrow \sigma', (\text{while } b \text{ do } S, \sigma') \rightarrow \sigma''}{(\text{while } b \text{ do } S, \sigma) \rightarrow \sigma''}$
whilefalse	$\mathcal{B}[b]\sigma = ff : (\text{while } b \text{ do } S, \sigma) \rightarrow \sigma$

**Proposition 1** (Determinism). *For all  $S$ , for all  $\sigma, \sigma', \sigma''$ , if  $(S, \sigma) \rightarrow \sigma'$  and  $(S, \sigma) \rightarrow \sigma''$  then  $\sigma' = \sigma''$ .*

## 1.2 Structural Operational “small steps” Semantics (SOS)

Name	Rule
skip	$(\text{skip}, \sigma) \Rightarrow \sigma$
assign	$(x := a, \sigma) \Rightarrow \sigma[x \mapsto \mathcal{A}[a]\sigma]$
seq1	$\frac{(S_1, \sigma) \Rightarrow \sigma'}{((S_1; S_2), \sigma) \Rightarrow (S_2, \sigma')}$
seq2	$\frac{(S_1, \sigma) \Rightarrow (S'_1, \sigma')}{((S_1; S_2), \sigma) \Rightarrow (S'_1; S_2, \sigma')}$
iffalse	$\mathcal{B}[b]\sigma = ff : (\text{if } b \text{ then } S_1 \text{ else } S_2, \sigma) \Rightarrow (S_2, \sigma)$
iftrue	$\mathcal{B}[b]\sigma = tt : (\text{if } b \text{ then } S_1 \text{ else } S_2, \sigma) \Rightarrow (S_1, \sigma)$
while	$(\text{while } b \text{ do } S, \sigma) \Rightarrow (\text{if } b \text{ then } (S; \text{while } b \text{ do } S) \text{ else skip}, \sigma)$

## 2 NAT/SOS equivalence

**Theorem 1** (Semantic equivalence).  $(S, \sigma) \rightarrow \sigma' \iff (S, \sigma) \Rightarrow \sigma'$ .

**Proposition 2.** *If  $(S, \sigma) \rightarrow \sigma'$  then  $(S, \sigma) \Rightarrow^* \sigma'$ .*

**Lemma 1** (Extension of seq1 SOS rule). *If  $(S_1, \sigma) \Rightarrow^k \sigma'$  then  $((S_1; S_2), \sigma) \Rightarrow^k (S_2, \sigma')$*

**Proposition 3.** *For all  $S, \sigma, k, \sigma'$ , if  $(S, \sigma) \Rightarrow^k \sigma'$  then  $(S, \sigma) \rightarrow \sigma'$ .*

**Lemma 2.** *If  $(S_1; S_2, \sigma) \Rightarrow^k \sigma''$  then there exists  $\sigma', k_1$  such that  $(S_1, \sigma) \Rightarrow^{k_1} \sigma'$  and  $(S_2, \sigma) \Rightarrow^{k-k_1} \sigma''$*