

# Tutorial

$\parallel$   $s := \underline{x := e} \mid s_1 ; s_2 \mid \text{if}(e) \{ s_1 \} \text{else} \{ s_2 \}$   
 $\mid \text{while}(e) \{ s \} \mid \text{skip}$

$\mid \text{assume}(e) \mid \text{assert}(e)$

$\mid x := \underline{\text{symbol}}$

$\mid x := f(e_1, \dots, e_n) \mid \text{return } e$

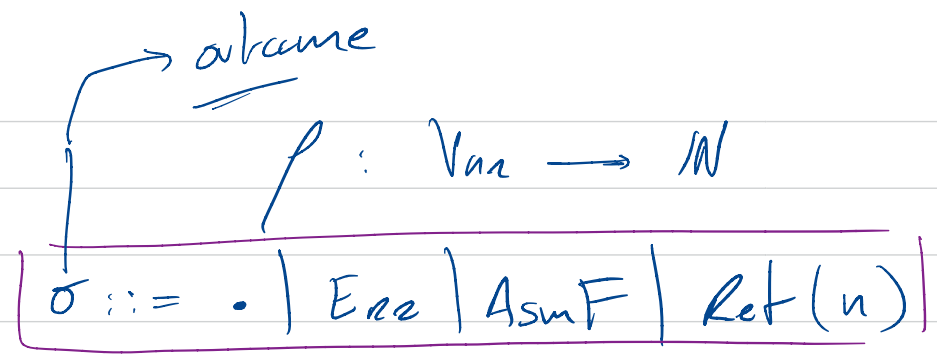
$\parallel e := n \mid x \mid e_1 \oplus e_2 \mid \ominus e$

$\text{func} := \text{function } \text{frame}(x_1, \dots, x_n) \{$   
 $\quad s$   
 $\}$

$\text{prog} := \overline{\text{func}}$

Semantic Check

Big-step  
Small-step



Big-step  $\langle s, \rho \rangle \Downarrow \rho, \sigma$

$\langle \text{skip}, \rho \rangle \Downarrow \rho, \cdot$

$\langle x := e, \rho \rangle \Downarrow \rho[x \mapsto \llbracket e \rrbracket_\rho], \cdot$

$\langle s_1, \rho \rangle \Downarrow \rho_1, \cdot \quad \langle s_2, \rho_1 \rangle \Downarrow \rho_2, \sigma$   
 $\langle s_1; s_2, \rho \rangle \Downarrow \rho_2, \sigma$

$\llbracket e \rrbracket_\rho = 0 \quad \langle s_2, \rho \rangle \Downarrow \rho_2, \sigma$   
 $\langle \text{if}(e) \{ s_1 \} \text{else} \{ s_2 \}, \rho \rangle \Downarrow \rho_2, \sigma$

$\langle \text{if}(e) \{ s; \text{while}(e) \{ s \} \text{else} \{ \text{skip} \} \}, \rho \rangle \Downarrow \rho', \sigma$   
 $\langle \text{while}(e) \{ s \}, \rho \rangle \Downarrow \rho', \sigma$

$\langle s_1, \rho \rangle \Downarrow \rho_1, \sigma \quad \sigma \neq \cdot$   
 $\langle s_1; s_2, \rho \rangle \Downarrow \rho_1, \sigma$

$\llbracket e \rrbracket_\rho = n \neq 0$   
 $\langle \text{assume}(e), \rho \rangle \Downarrow \rho, \cdot$

$\llbracket e \rrbracket_\rho = 0$   
 $\langle \text{assume}(e), \rho \rangle \Downarrow \rho, \text{AsmF}$

$$\frac{\llbracket e \rrbracket_p = n}{\text{return } e, p \Downarrow p, \text{ret}(n)}$$

$$\frac{\llbracket e_i \rrbracket_p = n_i \mid_{i=1}^n \quad \text{params}(f) = [x_1, \dots, x_n] \quad p' = [x_i \mapsto n_i \mid_{i=1}^n] \quad \text{body}(f) = s_b \quad \langle \underline{s_b}, p' \rangle \Downarrow p'', \underline{\text{ret}(n)}}{x := f(e_1, \dots, e_n), p \Downarrow p[x \mapsto n], \bullet}$$

(more cases ...)

def interpret (sto: State) (s: Stmt.t) : Out.t

def f = interpret sto m

match s with

| Seq (s1, s2) =>

|  $\text{Seq}(s_1, s_2) \Rightarrow$

$$\frac{\langle s_1, p \rangle \Downarrow p_1, \bullet \quad \langle s_2, p_1 \rangle \Downarrow p_2, \sigma}{\langle s_1; s_2, p \rangle \Downarrow p_2, \sigma}$$

$$\frac{\langle s_1, p \rangle \Downarrow p_1, \sigma \quad \sigma \neq \bullet}{\langle s_1; s_2, p \rangle \Downarrow p_1, \sigma}$$

let  $\text{out}_1 = f \ s_1$  in  
match  $\text{out}_1$  with

|  $\text{Cut} \Rightarrow f \ s_2$

|  $- \Rightarrow \text{out}_2$