

ANNOUNCEMENT

- Assignment 1
 - Will be out today, due Oct 7
 - Use Latex for writing solutions
 - Academic Honesty

LAST LECTURE

- Deductive Verification using constraint solving
- Imp: A simple imperative language
- Specifying correctness

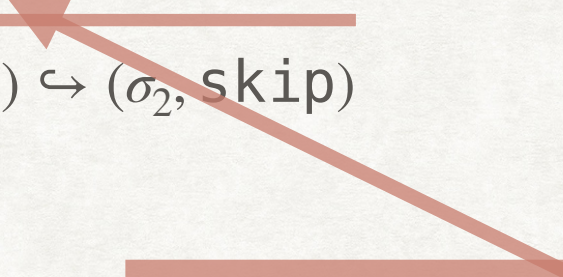
OPERATIONAL SEMANTICS OF IMP

- In order to formally define the verification problem, i.e. 'the program is verified to satisfy its specification', we will first define **Operational Semantics** of Imp.
- The operational semantics formally define how the program state evolves during execution.
- A program state (σ, c) consists of two components:
 - $\sigma : V \rightarrow \mathbb{R}$ is a valuation of program variables
 - c is the rest of the program to be executed
- Let $S = (\mathbb{R}^{|V|} \times \mathcal{P}) \cup \{Error\}$ be the set of all states
 - \mathcal{P} is the set of all Imp programs.
- A transition $(\sigma_1, c_1) \hookrightarrow (\sigma_2, c_2)$ denotes a step taken by the program

TRANSITIONS OF IMP

$$\frac{\sigma_2 = \sigma_1[x \rightarrow \sigma_1(e)]}{(\sigma_1, x := e) \hookrightarrow (\sigma_2, \text{skip})}$$

TRANSITIONS OF IMP

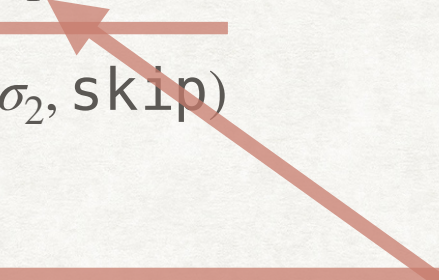
$$\frac{\sigma_2 = \sigma_1[x \rightarrow \sigma_1(e)]}{(\sigma_1, x := e) \hookrightarrow (\sigma_2, \text{skip})}$$


NOTATION ALERT:

$f = g[a \rightarrow b]$ means:

- $f(a) = b$
- $\forall x \in \text{dom}(g). x \neq a \rightarrow f(x) = g(x)$

TRANSITIONS OF IMP

$$\frac{\sigma_2 = \sigma_1[x \rightarrow \sigma_1(e)]}{(\sigma_1, x := e) \hookrightarrow (\sigma_2, \text{skip})}$$


NOTATION ALERT:

For $e \in \text{Exp}(V)$ and $\sigma \in \mathbb{R}^{|V|}$, $\sigma(e)$ denotes the evaluation of e at σ using the standard interpretations of Arithmetic operators.

TRANSITIONS OF IMP

$$\frac{\sigma_2 = \sigma_1[x \rightarrow \sigma_1(e)]}{(\sigma_1, x := e) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-ASSIGN]}$$

TRANSITIONS OF IMP

$$\frac{\sigma_2 = \sigma_1[x \rightarrow \sigma_1(e)]}{(\sigma_1, x := e) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-ASSIGN]}$$

$$\frac{\sigma_2 = \sigma_1[x \rightarrow n] \quad n \in \mathbb{R}}{(\sigma_1, x := \text{havoc}) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-HAVOC]}$$

TRANSITIONS OF IMP

$$\frac{\sigma_2 = \sigma_1[x \rightarrow \sigma_1(e)]}{(\sigma_1, x := e) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-ASSIGN]}$$

$$\frac{\sigma_2 = \sigma_1[x \rightarrow n] \quad n \in \mathbb{R}}{(\sigma_1, x := \text{havoc}) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-HAVOC]}$$

$$\frac{\text{???}}{(\sigma_1, \text{assume}(F)) \hookrightarrow (\sigma_1, \text{skip})} \quad \text{[T-ASSUME]}$$

TRANSITIONS OF IMP

$$\frac{\sigma_2 = \sigma_1[x \rightarrow \sigma_1(e)]}{(\sigma_1, x := e) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-ASSIGN]}$$

$$\frac{\sigma_2 = \sigma_1[x \rightarrow n] \quad n \in \mathbb{R}}{(\sigma_1, x := \text{havoc}) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-HAVOC]}$$

$$\frac{\sigma_1 \models F}{(\sigma_1, \text{assume}(F)) \hookrightarrow (\sigma_1, \text{skip})} \quad \text{[T-ASSUME]}$$

TRANSITIONS OF IMP

$$\frac{\sigma_2 = \sigma_1[x \rightarrow \sigma_1(e)]}{(\sigma_1, x := e) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-ASSIGN]}$$

$$\frac{\sigma_2 = \sigma_1[x \rightarrow n] \quad n \in \mathbb{R}}{(\sigma_1, x := \text{havoc}) \hookrightarrow (\sigma_2, \text{skip})} \quad \text{[T-HAVOC]}$$

$$\frac{\sigma_1 \models F}{(\sigma_1, \text{assume}(F)) \hookrightarrow (\sigma_1, \text{skip})} \quad \text{[T-ASSUME]}$$

$$\frac{\sigma_1 \models F}{(\sigma_1, \text{assert}(F)) \hookrightarrow (\sigma_1, \text{skip})} \quad \text{[T-ASSERT-TRUE]}$$

$$\frac{\sigma_1 \not\models F}{(\sigma_1, \text{assert}(F)) \hookrightarrow (\text{Error}, \text{skip})} \quad \text{[T-ASSERT-FALSE]}$$

TRANSITIONS OF IMP

[T-SEQ-1]

$$(\sigma_1, C_1) \hookrightarrow (\sigma_2, C'_1)$$

$$(\sigma_1, C_1; C_2) \hookrightarrow (\sigma_2, C'_1; C_2)$$

[T-SEQ-2]

$$(\sigma_1, \text{skip}; C_2) \hookrightarrow (\sigma_1, C_2)$$

TRANSITIONS OF IMP

[T-SEQ-1]

$$(\sigma_1, C_1) \hookrightarrow (\sigma_2, C'_1)$$

$$(\sigma_1, C_1; C_2) \hookrightarrow (\sigma_2, C'_1; C_2)$$

[T-SEQ-2]

$$(\sigma_1, \text{skip}; C_2) \hookrightarrow (\sigma_1, C_2)$$

[T-IF-TRUE]

$$\sigma_1 \models F$$

$$(\sigma_1, \text{if}(F) \text{ then } c_1 \text{ else } c_2) \hookrightarrow (\sigma_1, c_1)$$

[T-IF-FALSE]

$$\sigma_1 \not\models F$$

$$(\sigma_1, \text{if}(F) \text{ then } c_1 \text{ else } c_2) \hookrightarrow (\sigma_1, c_2)$$

TRANSITIONS OF IMP

[T-SEQ-1]

$$(\sigma_1, C_1) \hookrightarrow (\sigma_2, C'_1)$$

$$(\sigma_1, C_1; C_2) \hookrightarrow (\sigma_2, C'_1; C_2)$$

[T-SEQ-2]

$$(\sigma_1, \text{skip}; C_2) \hookrightarrow (\sigma_1, C_2)$$

[T-IF-TRUE]

$$\sigma_1 \models F$$

$$(\sigma_1, \text{if}(F) \text{ then } c_1 \text{ else } c_2) \hookrightarrow (\sigma_1, c_1)$$

[T-IF-FALSE]

$$\sigma_1 \not\models F$$

$$(\sigma_1, \text{if}(F) \text{ then } c_1 \text{ else } c_2) \hookrightarrow (\sigma_1, c_2)$$

$$\sigma_1 \models F$$

$$(\sigma_1, \text{while}(F) \text{ do } c) \hookrightarrow (\sigma_1, c; \text{while}(F) \text{ do } c)$$

[T-WHILE-TRUE]

TRANSITIONS OF IMP

[T-SEQ-1]

$$(\sigma_1, C_1) \hookrightarrow (\sigma_2, C'_1)$$

$$(\sigma_1, C_1; C_2) \hookrightarrow (\sigma_2, C'_1; C_2)$$

[T-SEQ-2]

$$(\sigma_1, \text{skip}; C_2) \hookrightarrow (\sigma_1, C_2)$$

[T-IF-TRUE]

$$\sigma_1 \models F$$

$$(\sigma_1, \text{if}(F) \text{ then } c_1 \text{ else } c_2) \hookrightarrow (\sigma_1, c_1)$$

[T-IF-FALSE]

$$\sigma_1 \not\models F$$

$$(\sigma_1, \text{if}(F) \text{ then } c_1 \text{ else } c_2) \hookrightarrow (\sigma_1, c_2)$$

$$\sigma_1 \models F$$

$$(\sigma_1, \text{while}(F) \text{ do } c) \hookrightarrow (\sigma_1, c; \text{while}(F) \text{ do } c)$$

[T-WHILE-TRUE]

$$\sigma_1 \not\models F$$

$$(\sigma_1, \text{while}(F) \text{ do } c) \hookrightarrow (\sigma_1, \text{skip})$$

[T-WHILE-FALSE]

EXAMPLE

```
assume(i = 0 ∧ n ≥ 0);  
while(i < n) do  
    i := i + 1;  
assert(i = n);
```

$(\{i \mapsto 0, n \mapsto 2\}, \text{assume}(i=0 \wedge n \geq 0); \dots)$

$\hookrightarrow (\{i \mapsto 0, n \mapsto 2\}, \text{skip}; \dots)$

[T-SEQ-1, T-ASSUME]

$\hookrightarrow (\{i \mapsto 0, n \mapsto 2\}, \text{while}(i < n) \text{ do } i := i + 1; \dots)$

[T-SEQ-2]

$\hookrightarrow (\{i \mapsto 0, n \mapsto 2\}, i := i + 1; \text{while}(i < n) \text{ do } i := i + 1; \dots)$ [T-WHILE-TRUE]

$\hookrightarrow (\{i \mapsto 1, n \mapsto 2\}, \text{while}(i < n) \text{ do } i := i + 1; \dots)$ [T-SEQ-1, T-ASSIGN, T-SEQ-2]

$\hookrightarrow (\{i \mapsto 1, n \mapsto 2\}, i := i + 1; \text{while}(i < n) \text{ do } i := i + 1; \dots)$ [T-WHILE-TRUE]

$\hookrightarrow (\{i \mapsto 2, n \mapsto 2\}, \text{while}(i < n) \text{ do } i := i + 1; \dots)$ [T-SEQ-1, T-ASSIGN, T-SEQ-2]

$\hookrightarrow (\{i \mapsto 2, n \mapsto 2\}, \text{assert}(i=n);)$

[T-WHILE-FALSE, T-SEQ-2]

$\hookrightarrow (\{i \mapsto 2, n \mapsto 2\}, \text{skip};)$

[T-ASSERT-TRUE]

REACHABILITY AND VERIFICATION

- Let $T \subseteq S \times S$ be the set of transitions (\hookrightarrow) defined in the previous slides.
 - Is T finite?
 - Is T defined for a specific program c or for any program?
- Given a program c , a sequence of transitions $(\sigma_0, c) \hookrightarrow (\sigma_1, c_1) \dots \hookrightarrow (\sigma_n, c_n)$ is called an **execution** of c .
 - A program state σ is called **reachable** if there exists an execution $(\sigma_0, c) \hookrightarrow \dots \hookrightarrow (\sigma, c_n)$ which ends in the state σ .
- Verification Problem: Is $(Error, c')$ reachable for some c' ?
 - Program c is called **safe** if the error state is not reachable.
 - What about the initial state?