**COMP 4161**
NICTA Advanced Course

**Advanced Topics in Software Verification**

Gerwin Klein, June Andronick, Toby Murray, Rafal Kolanski

$$\{P\} \ldots \{Q\}$$

**Slide 1**

## Content



[a] a1 due; [b] a2 due; [c] session break; [d] a3 due

**Slide 2**



# A Crash Course in Semantics

**Slide 3**

## IMP - a small Imperative Language

**Commands:**

| | | | |
|---|---|---|---|
| **datatype** com | = | SKIP | |
| | \| | Assign vname aexp | (_ := _) |
| | \| | Semi com com | (_; _) |
| | \| | Cond bexp com com | (IF _ THEN _ ELSE _) |
| | \| | While bexp com | (WHILE _ DO _ OD) |

**types** vname = string
**types** state = vname ⇒ nat

**types** aexp = state ⇒ nat
**types** bexp = state ⇒ bool

**Slide 4**

## Example Program

**Usual syntax:**

$B := 1;$
WHILE $A \neq 0$ DO
  $B := B * A;$
  $A := A - 1$
OD

**Expressions are functions from state to bool or nat:**

$B := (\lambda\sigma.\ 1);$
WHILE $(\lambda\sigma.\ \sigma\ A \neq 0)$ DO
  $B := (\lambda\sigma.\ \sigma\ B * \sigma\ A);$
  $A := (\lambda\sigma.\ \sigma\ A - 1)$
OD

**Slide 5**

## What does it do?

**So far we have defined:**

➜ **Syntax** of commands and expressions
➜ **State** of programs (function from variables to values)

**Now we need:** the meaning (semantics) of programs

**How to define execution of a program?**

➜ A wide field of its own
➜ Some choices:
- Operational (inductive relations, big step, small step)
- Denotational (programs as functions on states, state transformers)
- Axiomatic (pre-/post conditions, Hoare logic)

**Slide 6**

Copyright NICTA 2013, provided under Creative Commons Attribution License



## Structural Operational Semantics

$$\frac{}{\langle \mathsf{SKIP}, \sigma\rangle \rightarrow \sigma}$$

$$\frac{e\ \sigma = v}{\langle \mathsf{x := e}, \sigma\rangle \rightarrow \sigma[x \mapsto v]}$$

$$\frac{\langle c_1, \sigma\rangle \rightarrow \sigma' \quad \langle c_2, \sigma'\rangle \rightarrow \sigma''}{\langle c_1; c_2, \sigma\rangle \rightarrow \sigma''}$$

$$\frac{b\ \sigma = \mathsf{True} \quad \langle c_1, \sigma\rangle \rightarrow \sigma'}{\langle \mathsf{IF}\ b\ \mathsf{THEN}\ c_1\ \mathsf{ELSE}\ c_2, \sigma\rangle \rightarrow \sigma'}$$

$$\frac{b\ \sigma = \mathsf{False} \quad \langle c_2, \sigma\rangle \rightarrow \sigma'}{\langle \mathsf{IF}\ b\ \mathsf{THEN}\ c_1\ \mathsf{ELSE}\ c_2, \sigma\rangle \rightarrow \sigma'}$$

**Slide 7**

## Structural Operational Semantics

$$\frac{b\ \sigma = \mathsf{False}}{\langle \mathsf{WHILE}\ b\ \mathsf{DO}\ c\ \mathsf{OD}, \sigma\rangle \rightarrow \sigma}$$

$$\frac{b\ \sigma = \mathsf{True} \quad \langle c, \sigma\rangle \rightarrow \sigma' \quad \langle \mathsf{WHILE}\ b\ \mathsf{DO}\ c\ \mathsf{OD}, \sigma'\rangle \rightarrow \sigma''}{\langle \mathsf{WHILE}\ b\ \mathsf{DO}\ c\ \mathsf{OD}, \sigma\rangle \rightarrow \sigma''}$$

**Slide 8**

Copyright NICTA 2013, provided under Creative Commons Attribution License

## Demo: The Definitions in Isabelle

Slide 9

## Demo: Example Proof

Slide 11

## Proofs about Programs

**Now we know:**

- ➜ What programs are: Syntax
- ➜ On what they work: State
- ➜ How they work: Semantics

**So we can prove properties about programs**

**Example:**
Show that example program from slide 5 implements the factorial.

$$\textbf{lemma}\ \langle \text{factorial}, \sigma\rangle \rightarrow \sigma' \Longrightarrow \sigma' B = \text{fac}\ (\sigma A)$$

$$(\text{where}\quad \text{fac}\ 0 = 1,\quad \text{fac}\ (\text{Suc}\ n) = (\text{Suc}\ n) * \text{fac}\ n)$$

Slide 10

## Too tedious

**Induction needed for each loop**

**Is there something easier?**

Slide 12

Copyright NICTA 2013, provided under Creative Commons Attribution License

Copyright NICTA 2013, provided under Creative Commons Attribution License

## Floyd/Hoare

NICTA

**Idea:** describe meaning of program by pre/post conditions

**Examples:**

$\{\text{True}\}\quad x := 2 \quad \{x = 2\}$

$\{y = 2\}\quad x := 21 * y \quad \{x = 42\}$

$\{x = n\}\quad \text{IF } y < 0 \text{ THEN } x := x + y \text{ ELSE } x := x - y \quad \{x = n - |y|\}$

$\{A = n\}\quad \text{factorial} \quad \{B = \text{fac } n\}$

**Proofs:** have rules that directly work on such triples

**Slide 13**

## Meaning of a Hoare-Triple

NICTA

$$\{P\}\quad c \quad \{Q\}$$

**What are the assertions $P$ and $Q$?**

➜ Here: again functions from state to bool (shallow embedding of assertions)

➜ Other choice: syntax and semantics for assertions (deep embedding)

**What does $\{P\}\ c\ \{Q\}$ mean?**

**Partial Correctness:**

$\models \{P\}\ c\ \{Q\} \quad \equiv \quad \forall \sigma\ \sigma'.\ P\ \sigma \wedge \langle c, \sigma\rangle \rightarrow \sigma' \longrightarrow Q\ \sigma'$

**Total Correctness:**

$\models \{P\}\ c\ \{Q\} \quad \equiv \quad (\forall \sigma\ \sigma'.\ P\ \sigma \wedge \langle c, \sigma\rangle \rightarrow \sigma' \longrightarrow Q\ \sigma') \wedge$
$(\forall \sigma.\ P\ \sigma \longrightarrow \exists \sigma'.\ \langle c, \sigma\rangle \rightarrow \sigma')$

This lecture: partial correctness only (easier)

**Slide 14**

Copyright NICTA 2013, provided under Creative Commons Attribution License



## Hoare Rules

NICTA

$$\frac{}{\{P\}\quad \text{SKIP} \quad \{P\}} \qquad \frac{}{\{P[x \mapsto e]\}\quad x := e \quad \{P\}}$$

$$\frac{\{P\}\ c_1\ \{R\} \quad \{R\}\ c_2\ \{Q\}}{\{P\}\quad c_1; c_2 \quad \{Q\}}$$

$$\frac{\{P \wedge b\}\ c_1\ \{Q\} \quad \{P \wedge \neg b\}\ c_2\ \{Q\}}{\{P\}\quad \text{IF } b \text{ THEN } c_1 \text{ ELSE } c_2 \quad \{Q\}}$$

$$\frac{\{P \wedge b\}\ c\ \{P\} \quad P \wedge \neg b \Longrightarrow Q}{\{P\}\quad \text{WHILE } b \text{ DO } c \text{ OD} \quad \{Q\}}$$

$$\frac{P \Longrightarrow P' \quad \{P'\}\ c\ \{Q'\} \quad Q' \Longrightarrow Q}{\{P\}\quad c \quad \{Q\}}$$

**Slide 15**

## Hoare Rules

NICTA

$$\frac{}{\vdash \{P\}\quad \text{SKIP} \quad \{P\}} \qquad \frac{}{\vdash \{\lambda\sigma.\ P\ (\sigma(x := e\ \sigma))\}\quad x := e \quad \{P\}}$$

$$\frac{\vdash \{P\}\ c_1\ \{R\} \quad \vdash \{R\}\ c_2\ \{Q\}}{\vdash \{P\}\quad c_1; c_2 \quad \{Q\}}$$

$$\frac{\vdash \{\lambda\sigma.\ P\ \sigma \wedge b\ \sigma\}\ c_1\ \{R\} \quad \vdash \{\lambda\sigma.\ P\ \sigma \wedge \neg b\ \sigma\}\ c_2\ \{Q\}}{\vdash \{P\}\quad \text{IF } b \text{ THEN } c_1 \text{ ELSE } c_2 \quad \{Q\}}$$

$$\frac{\vdash \{\lambda\sigma.\ P\ \sigma \wedge b\ \sigma\}\ c\ \{P\} \quad \bigwedge\sigma.\ P\ \sigma \wedge \neg b\ \sigma \Longrightarrow Q\ \sigma}{\vdash \{P\}\quad \text{WHILE } b \text{ DO } c \text{ OD} \quad \{Q\}}$$

$$\frac{\bigwedge\sigma.\ P\ \sigma \Longrightarrow P'\ \sigma \quad \vdash \{P'\}\ c\ \{Q'\} \quad \bigwedge\sigma.\ Q'\ \sigma \Longrightarrow Q\ \sigma}{\vdash \{P\}\quad c \quad \{Q\}}$$

**Slide 16**

Copyright NICTA 2013, provided under Creative Commons Attribution License

## Are the Rules Correct?

**Soundness:** $\vdash \{P\}\ c\ \{Q\} \Longrightarrow \models \{P\}\ c\ \{Q\}$

**Proof:** by rule induction on $\vdash \{P\}\ c\ \{Q\}$

**Demo:** Hoare Logic in Isabelle

**Slide 17**