Semantics and Computability

Compulsory Exercise 2

René Rydhof Hansen

F05-02240

Protecting Memory

- Special memory access (shared memory, memory mapped I/O)
- Example: buffer updates in device driver

```
buf := 42
```

Problems: ???

Protecting Memory

- Special memory access (shared memory, memory mapped I/O)
- Example: buffer updates in device driver
  - Status must be checked before update

```
if (status = true) then

 buf := 42
else
 ...
```

Problems: ???

Protecting Memory

- Special memory access (shared memory, memory mapped I/O)
- Example: buffer updates in device driver
  - Status must be checked before update
  - The log must be updated before the buffer
  - But: it is not mandatory to update the log

```
if (status = true) then
 log := 87;
 buf := 42
else
 ...
```

Problems: ???

Protecting Memory

- Special memory access (shared memory, memory mapped I/O)
- Example: buffer updates in device driver
  - Status must be checked before update
  - The log must be updated before the buffer
  - But: it is not mandatory to update the log
  - Reset can only occur *after* buffer *and* log is updated

```
if (status = true) then
 log := 87;
 buf := 42
else
 ...
```

Problems: ???

Protecting Memory

- Special memory access (shared memory, memory mapped I/O)
- Example: buffer updates in device driver
  - Status must be checked before update
  - The log must be updated before the buffer
  - But: it is not mandatory to update the log
  - Reset can only occur *after* buffer *and* log is updated

```
if (status = true) then
 log := 87;
 buf := 42
else
 ...
```

Problems: tedious, easy to forget, stupid programmers, ...

Solution: Reference Monitor (RM)

- Central control of all memory access
- Enforce safety/security policy
- Problem: how to specify the RM?

Deterministic Finite Automaton (DFA)

$$\mathcal{A} = (\Sigma, Q, q_0, \delta)$$

- Σ: alphabet
- Q: states
- $q_0 \in Q$: start state
- $\delta : Q \times \Sigma \to Q$: transition function

we write $q \xrightarrow{a} q'$ whenever $q' = \delta(q, a)$

DFA for Buffer Updates

- $\Sigma =$
- $Q =$
- Initial state:
- δ

DFA for Buffer Updates

- $\Sigma = \{?\texttt{status}, !\texttt{log}, !\texttt{buf}, !\texttt{reset}\}$
- $Q = \{q_0, q_{?s}, q_{!l}, q_{!b}, q_{!lb}\}$
- Initial state: q_0
- δ

	?status	!log	!buf	!reset
q_0	$q_{?s}$			
$q_{?s}$		$q_{!l}$	$q_{!b}$	
$q_{!b}$				
$q_{!l}$			$q_{!lb}$	
$q_{!lb}$				q_0

How to model this?

- Build DFA into the semantics
- For DFA $\mathcal{A} = (\Sigma, Q, q_0, \delta)$:
  - From $\langle S, s \rangle \rightarrow s'$ to $\langle S, s; q \rangle \rightarrow_{\mathcal{A}} s'; q'$
  - $\Sigma = \{!x | x \in \mathrm{Var}\} \cup \{?x | x \in \mathrm{Var}\}$
  - $Q = ???$
  - $\delta = ???$

Annotated Semantics (AExp)

For $\mathcal{A}$ a DFA define the $\rightarrow_{\mathcal{A}}$ semantics for arithmetic expressions:

$$\langle n, s \quad \rangle \rightarrow_{\mathcal{A}} n$$

$$\langle \mathtt{x}, s \quad \rangle \rightarrow_{\mathcal{A}} s(\mathtt{x}) \qquad \text{if } x \in \mathrm{dom}(s)$$

$$\frac{\langle a_1, s \quad \rangle \rightarrow_{\mathcal{A}} n_1 \qquad \langle a_2, s \quad \rangle \rightarrow_{\mathcal{A}} n_2}{\langle a_1 + a_2, s \quad \rangle \rightarrow_{\mathcal{A}} n_1 + n_2}$$

Annotated Semantics (AExp)

For $\mathcal{A}$ a DFA define the $\rightarrow_{\mathcal{A}}$ semantics for arithmetic expressions:

$$\langle n, s; q \rangle \rightarrow_{\mathcal{A}} n$$

$$\langle \mathtt{x}, s \quad \rangle \rightarrow_{\mathcal{A}} s(\mathtt{x}) \qquad \text{if } x \in \mathrm{dom}(s)$$

$$\frac{\langle a_1, s \quad \rangle \rightarrow_{\mathcal{A}} n_1 \qquad \langle a_2, s \quad \rangle \rightarrow_{\mathcal{A}} n_2}{\langle a_1 + a_2, s \quad \rangle \rightarrow_{\mathcal{A}} n_1 + n_2}$$

Annotated Semantics (AExp)

For $\mathcal{A}$ a DFA define the $\to_{\mathcal{A}}$ semantics for arithmetic expressions:

$$\langle n, s; q \rangle \to_{\mathcal{A}} n; q$$

$$\langle \mathtt{x}, s \quad \rangle \to_{\mathcal{A}} s(\mathtt{x}) \qquad \text{if } x \in \mathrm{dom}(s)$$

$$\frac{\langle a_1, s \quad \rangle \to_{\mathcal{A}} n_1 \qquad \langle a_2, s \quad \rangle \to_{\mathcal{A}} n_2}{\langle a_1 + a_2, s \quad \rangle \to_{\mathcal{A}} n_1 + n_2}$$

Annotated Semantics (AExp)

For $\mathcal{A}$ a DFA define the $\rightarrow_{\mathcal{A}}$ semantics for arithmetic expressions:

$$\langle n, s; q\rangle \rightarrow_{\mathcal{A}} n; q$$

$$\langle \mathtt{x}, s; q\rangle \rightarrow_{\mathcal{A}} s(\mathtt{x}) \qquad \text{if } x \in \mathrm{dom}(s)$$

$$\frac{\langle a_1, s \quad \rangle \rightarrow_{\mathcal{A}} n_1 \qquad \langle a_2, s \quad \rangle \rightarrow_{\mathcal{A}} n_2}{\langle a_1 + a_2, s \quad \rangle \rightarrow_{\mathcal{A}} n_1 + n_2}$$

Annotated Semantics (AExp)

For $\mathcal{A}$ a DFA define the $\rightarrow_{\mathcal{A}}$ semantics for arithmetic expressions:

$$\langle n, s; q\rangle \rightarrow_{\mathcal{A}} n; q$$

$$\langle \mathtt{x}, s; q\rangle \rightarrow_{\mathcal{A}} s(\mathtt{x}); q' \quad \text{if } x \in \mathrm{dom}(s) \wedge q \xrightarrow{?\mathtt{x}} q'$$

$$\frac{\langle a_1, s \quad\rangle \rightarrow_{\mathcal{A}} n_1 \qquad\qquad \langle a_2, s \quad\rangle \rightarrow_{\mathcal{A}} n_2}{\langle a_1 + a_2, s \quad\rangle \rightarrow_{\mathcal{A}} n_1 + n_2}$$

Annotated Semantics (AExp)

For $\mathcal{A}$ a DFA define the $\rightarrow_{\mathcal{A}}$ semantics for arithmetic expressions:

$$\langle n, s; q \rangle \rightarrow_{\mathcal{A}} n; q$$

$$\langle \mathtt{x}, s; q \rangle \rightarrow_{\mathcal{A}} s(\mathtt{x}); q' \quad \text{if } x \in \mathrm{dom}(s) \wedge \; q \xrightarrow{?\mathtt{x}} q'$$

$$\frac{\langle a_1, s; q \rangle \rightarrow_{\mathcal{A}} n_1; q' \qquad \langle a_2, s \quad \rangle \rightarrow_{\mathcal{A}} n_2}{\langle a_1 + a_2, s; q \rangle \rightarrow_{\mathcal{A}} n_1 + n_2}$$

Annotated Semantics (AExp)

For $\mathcal{A}$ a DFA define the $\rightarrow_{\mathcal{A}}$ semantics for arithmetic expressions:

$$\langle n, s; q\rangle \rightarrow_{\mathcal{A}} n; q$$

$$\langle \mathtt{x}, s; q\rangle \rightarrow_{\mathcal{A}} s(\mathtt{x}); q' \quad \text{if } x \in \mathrm{dom}(s) \wedge \; q \xrightarrow{?\mathtt{x}} q'$$

$$\frac{\langle a_1, s; q\rangle \rightarrow_{\mathcal{A}} n_1; q' \qquad \langle a_2, s; q'\rangle \rightarrow_{\mathcal{A}} n_2; q''}{\langle a_1 + a_2, s; q\rangle \rightarrow_{\mathcal{A}} n_1 + n_2; q''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s \quad \rangle \rightarrow_{\mathcal{A}} s$$

$$\frac{\langle a, s \quad \rangle \rightarrow_{\mathcal{A}} n}{\langle \texttt{x := } a, s \quad \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]}$$

$$\frac{\langle S_1, s \quad \rangle \rightarrow_{\mathcal{A}} s' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s \quad \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s$$

$$\frac{\langle a, s \quad \rangle \rightarrow_{\mathcal{A}} n}{\langle \texttt{x} \;\; \texttt{:=} \;\; a, s \quad \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]}$$

$$\frac{\langle S_1, s \quad \rangle \rightarrow_{\mathcal{A}} s' \qquad\qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s \quad \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s \quad \rangle \rightarrow_{\mathcal{A}} n}{\langle \texttt{x := } a, s \quad \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]}$$

$$\frac{\langle S_1, s \quad \rangle \rightarrow_{\mathcal{A}} s' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s \quad \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s \quad \rangle \rightarrow_{\mathcal{A}} n}{\langle \texttt{x := } a, s; q \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]}$$

$$\frac{\langle S_1, s \quad \rangle \rightarrow_{\mathcal{A}} s' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s \quad \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q \rangle \rightarrow_{\mathcal{A}} n}{\langle \texttt{x := } a, s; q \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]}$$

$$\frac{\langle S_1, s \quad \rangle \rightarrow_{\mathcal{A}} s' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s \quad \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q\rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q\rangle \rightarrow_{\mathcal{A}} n; q'}{\langle \texttt{x := } a, s; q\rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]}$$

$$\frac{\langle S_1, s \quad \rangle \rightarrow_{\mathcal{A}} s' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s \quad \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q \rangle \rightarrow_{\mathcal{A}} n; q'}{\langle \texttt{x := } a, s; q \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]; q''} \quad \text{if } q' \xrightarrow{!\texttt{x}} q''$$

$$\frac{\langle S_1, s \quad \rangle \rightarrow_{\mathcal{A}} s' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s \quad \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q \rangle \rightarrow_{\mathcal{A}} n; q'}{\langle \texttt{x := } a, s; q \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]; q''} \quad \text{if } q' \xrightarrow{!\texttt{x}} q''$$

$$\frac{\langle S_1, s \quad \rangle \rightarrow_{\mathcal{A}} s' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s; q \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q \rangle \rightarrow_{\mathcal{A}} n; q'}{\langle \texttt{x := } a, s; q \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]; q''} \quad \text{if } q' \xrightarrow{!\texttt{x}} q''$$

$$\frac{\langle S_1, s; q \rangle \rightarrow_{\mathcal{A}} s' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s; q \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q \rangle \rightarrow_{\mathcal{A}} n; q'}{\langle \texttt{x := } a, s; q \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]; q''} \quad \text{if } q' \xrightarrow{!\texttt{x}} q''$$

$$\frac{\langle S_1, s; q \rangle \rightarrow_{\mathcal{A}} s'; q' \qquad \langle S_2, s' \quad \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s; q \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q \rangle \rightarrow_{\mathcal{A}} n; q'}{\langle \texttt{x := } a, s; q \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]; q''} \quad \text{if } q' \xrightarrow{!\texttt{x}} q''$$

$$\frac{\langle S_1, s; q \rangle \rightarrow_{\mathcal{A}} s'; q' \qquad \langle S_2, s'; q' \rangle \rightarrow_{\mathcal{A}} s''}{\langle S_1; S_2, s; q \rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q\rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q\rangle \rightarrow_{\mathcal{A}} n; q'}{\langle \texttt{x := } a, s; q\rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]; q''} \quad \text{if } q' \xrightarrow{!\texttt{x}} q''$$

$$\frac{\langle S_1, s; q\rangle \rightarrow_{\mathcal{A}} s'; q' \qquad \langle S_2, s'; q'\rangle \rightarrow_{\mathcal{A}} s''; q''}{\langle S_1; S_2, s; q\rangle \rightarrow_{\mathcal{A}} s''}$$

Annotated Semantics (Stmt)

Using the same DFA ($\mathcal{A}$) define the $\rightarrow_{\mathcal{A}}$ semantics for statements:

$$\langle \texttt{skip}, s; q \rangle \rightarrow_{\mathcal{A}} s; q$$

$$\frac{\langle a, s; q \rangle \rightarrow_{\mathcal{A}} n; q'}{\langle \texttt{x := } a, s; q \rangle \rightarrow_{\mathcal{A}} s[\texttt{x} \mapsto n]; q''} \quad \text{if } q' \xrightarrow{!\texttt{x}} q''$$

$$\frac{\langle S_1, s; q \rangle \rightarrow_{\mathcal{A}} s'; q' \qquad \langle S_2, s'; q' \rangle \rightarrow_{\mathcal{A}} s''; q''}{\langle S_1; S_2, s; q \rangle \rightarrow_{\mathcal{A}} s''; q''}$$

Information Security: High Watermark

- All variables classified as either *high security* (H) or *low security* (L) with $\mathsf{L} \sqsubseteq \mathsf{H}$
  - *level* $: \mathrm{Var} \rightarrow \{\mathsf{L}, \mathsf{H}\}$
  - Assume *level*$(\mathtt{l}) = \mathsf{L}$ and *level*$(\mathtt{h}) = \mathsf{H}$
- If a H variable is *read* no L variable must be *written*

```
h := 42;
l := h
```

```
l := 42;
h := l
```

```
h := 1;
if h then
 l := 1
else
 l := 0
```

Exercise(s)

1. Finish the annotated semantics:
   - Finish the $\rightarrow_{\mathcal{A}}$ semantics for AExp
   - Define the $\rightarrow_{\mathcal{A}}$ semantics for BExp
   - Finish the $\rightarrow_{\mathcal{A}}$ semantics for Stmt
2. Specify DFA for “high watermark” security
3. Implement the annotated semantics in SML
   - Can you re-use “old” semantics for AExp and BExp?
   - Can you parameterise your implementation on $\mathcal{A}$?