Programming Language Semantics and Compiler Design
—
Midterm Exam of Monday, October 23, 2023

- **Duration**: 1h15.
- 3 double-sided sheets of A4 paper are authorized.
- Any electronic device is forbidden.
- The grading scale is indicative.
- Exercises are **independent**.
- **The care of your submission will be taken into account.**
- It is recommended to read each exercise till the end before answering. **Indicate your group number on your submission.**
- If you don't know how to answer to some question, you may assume the result and proceed with the next question.
- The maximal grade is obtained with 20 points.

In this exam, otherwise stated, we consider the natural operational semantics and the structural operational semantics of **While** as defined in the course. We consider the notion of states in **State** = **Var** $\rightarrow \mathbb{Z}$, where **Var** is the set of variables appearing in the program.

Exercise 1 — Operational Semantics - applying the rules (5 points)

1. **(3 points)** Draw the derivation tree obtained by the execution of $y := y + x; x := y - x; y := y - x$ in state $[x \mapsto 4, y \mapsto 1]$, using natural operational semantics.
2. **(1 point)** Draw the derivation sequence obtained by the same execution using structural operational semantics.
3. **(1 point)** Are there states from which the execution of `while` $x > 0$ `do` $x := x - 1$ `od` is different from the execution of `while` $x \neq 0$ `do` $x := x - 1$ `od`? Explain using an example, showing derivation sequences obtained by structural operational semantics.

Exercise 2 — Natural Operational Semantics - Simultaneous assignment (7 points)

We consider an extension of **While** with simultaneous assignment, extending the syntax of statements as follows:

$$\begin{array}{lcl} S & ::= & \texttt{skip} \mid A \mid S_1; S_2 \mid \texttt{if } b \texttt{ then } S_1 \texttt{ else } S_2 \texttt{ fi} \mid \texttt{while } b \texttt{ do } S_0 \texttt{ od} \\ A & ::= & x := a \mid x := a \,[\!]\, A_0 \end{array}$$

Assignment thus has the general form "$x_1 := a_1 \,[\!]\, \ldots \,[\!]\, x_n := a_n$" ($n > 0$), meaning that:

- the values of $a_1, \ldots, a_n$ are all computed in the same state (the current state) and
- the resulting state is the one obtained by updating each variable x_i with the value of a_i ($1 \leq i \leq n$).

We assume that all x_i's occurring before the assignment symbol in "$x_1 := a_1 \,[\!]\, \ldots \,[\!]\, x_n := a_n$" are distinct.
The axiom of assignment is replaced by the following rule:

$$\frac{(A, \sigma) \rightarrow_A \sigma}{(A, \sigma) \rightarrow \sigma}$$

where $\rightarrow_A$ is an auxilliary transition relation defining the semantics of simultaneous assignment.

1. **(1 point)** Give a rule defining $\rightarrow_A$ for configurations of the form $(x := a, \sigma)$.
2. **(3 points)** Give a rule defining $\rightarrow_A$ for configurations of the form $(x := a \,[\!]\, A, \sigma)$.
3. **(3 points)** Propose instances of $a_1, a_2, \sigma, \sigma_1$, and σ_2 such that:

$$\begin{array}{l} (x_1 := a_1 \,[\!]\, x_2 := a_2, \sigma) \rightarrow \sigma_1, \\ (x_1 := a_1; x_2 := a_2, \sigma) \rightarrow \sigma_2, \\ \text{and } \sigma_1 \neq \sigma_2 \end{array}$$

Draw the instantiated derivation trees of $(x_1 := a_1 \,[\!]\, x_2 := a_2, \sigma) \rightarrow \sigma_1$ and $(x_1 := a_1; x_2 := a_2, \sigma) \rightarrow \sigma_2$.

Exercise 3 — Axiomatic Semantics (8+1 points)

Let S be the following program:

$$y := x; n := 0; \texttt{while } n < x \texttt{ do } y := y * x; n := n + 1 \texttt{ od}$$

Let P_0 be the predicate $x \geq n \land y = x^{n+1}$.

1. **(1 point)** Recall the rule of Hoare logic allowing the proof of the Hoare triple $\{x \geq 0\}\ S\ \{y = x^{x+1}\}$ to be built upon proofs of the subgoals

   $$\{x \geq 0\}\ y := x; n := 0\ \{P_0\}$$

   and

   $$\{P_0\}\ \texttt{while } x > n \texttt{ do } y := y * x; n := n + 1 \texttt{ od}\ \{y = x^{x+1}\}$$

2. **(2 points)** Prove that the Hoare triple $\{x \geq 0\}\ y := x; n := 0\ \{P_0\}$ is valid.
3. **(3 points)** The partial correctness proof of $\{P_0\}\ \texttt{while } x > n \texttt{ do } y := y * x; n := n + 1 \texttt{ od}\ \{y = x^{x+1}\}$ can be derived from the proof of a Hoare triple of the form $\{P_0\}\ \texttt{while } x > n \texttt{ do } y := y * x; n := n + 1 \texttt{ od}\ \{Q_0\}$ given below:

   $$\dfrac{\dfrac{\overline{\{P_2\}\ y := y * x\ \{Q_2\}} \quad \overline{\{P_3\}\ n := n + 1\ \{Q_3\}}}{\{P_1\}\ y := y * x; n := n + 1\ \{Q_1\}}}{\{P_0\}\ \texttt{while } x > n \texttt{ do } y := y * x; n := n + 1 \texttt{ od}\ \{Q_0\}}$$

   Define the predicates $P_1, P_2, P_3, Q_0, Q_1, Q_2$, and Q_3 for this proof to be valid according to the inference rules studied un class.
4. **(1 point)** By which rule can you prove that $\{P_0\}\ \texttt{while } x > n \texttt{ do } y := y * x; n := n + 1 \texttt{ od}\ \{y = x^{x+1}\}$ is valid, given that $\{P_0\}\ \texttt{while } x > n \texttt{ do } y := y * x; n := n + 1 \texttt{ od}\ \{Q_0\}$ is valid. Name the rule and give a justification.
5. **(1 point)** Besides being the precondition of the while loop, what is the predicate P_0 in this proof?
6. **Bonus (1 point)** Justify, without building the proof tree, that the program is totally correct.