HW 2 Solution – Transition Semantics and Context Semantics

CS 422 – Spring 2016
Revision 1.0

Assigned February 3, 2016
Due February 10, 2016, 8:00pm
Extension 24 hours (10% penalty)

1 Change Log

1.1 Corrected copy-paste mistakes in the statement of problem 2, and in the rules for transition semantics for boolean & and or.

1.0 Initial Release.

2 Turn-In Procedure

Answer the following problems, save you work as a PDF named `hw2-submission.pdf` (either scanned if hand-written, or converted from a program), and submit your PDF by first adding it to your svn repository (`svn add hw2-submission.pdf`) and commit it (`svn commit -m "submitting hw2"`). Your file contain your name, netid and solution. It should be named `hw2-submission.pdf` and committed in your `assignments/hw2` directory.

3 Objectives

The purpose of this HW is to familiarize you with Transition Semantics and Reduction Semantics with Evaluation Contexts (Context Semantics) and their use in giving semantics for simple programming languages.

4 Background

In this assignment you will be giving semantics in two new styles for the first variant of the first simple imperative programming language discussed in class. As background, we will review the syntax and Natural Semantics of each the languages, and the Tranistion Semantics and Context Semantics for the first language.

4.1 Syntax of SIMPL1 and SIMPL2

Recall the syntax of SIMPL1:

$$
\begin{array}{lcl}
I & \in & \textit{Identifiers} \\
N & \in & \textit{Numerals} \\
E & ::= & N \mid I \mid E + E \mid E * E \mid E - E \\
B & ::= & \mathsf{true} \mid \mathsf{false} \mid B \& B \mid B \text{ or } B \mid \text{not } B \mid E < E \mid E = E \\
C & ::= & \mathsf{skip} \mid C; C \mid \{C\} \mid I ::= E \mid \mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi} \mid \mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}
\end{array}
$$

And the syntax of the second language SIMPL2:

$$
\begin{array}{lcl}
I & \in & \textit{Identifiers} \\
N & \in & \textit{Numerals} \\
E & ::= & N \mid I \mid E + E \mid E * E \mid E - E \mid I ::= E \\
B & ::= & \mathsf{true} \mid \mathsf{false} \mid B \& B \mid B \text{ or } B \mid \mathsf{not}\ B \mid E < E \mid E = E \\
C & ::= & \mathsf{skip} \mid C; C \mid \{C\} \mid E \mid \mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi} \mid \mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}
\end{array}
$$

4.2 Natural Semantics of SIMPL1 and SIMPL2

Assuming a set Values of final results of expressions (in this case you can assume integers), and $m,\ m'$: Identifiers $\rightharpoonup$ Values, in class we gave a Natural Semantics for the SIMPL1 as follows:

Constants:

Identifiers: $(I, m) \Downarrow m(I)$ if $m(I)$ exists Numerals are values: $(N, m) \Downarrow N$

Booleans: $(\mathsf{true}, m) \Downarrow \mathsf{true}$ $(\mathsf{false}, m) \Downarrow \mathsf{false}$

Arithmetic Expressions:

$$\frac{(E, m) \Downarrow U \quad (E', m) \Downarrow V \quad U \oplus V = N}{(E \oplus E', m) \Downarrow N} \text{ where } \oplus \in \{+, *, -\} \text{ and } U, V \in \texttt{Values}$$

Arithmetic Relations:

$$\frac{(E, m) \Downarrow U \quad (E', m) \Downarrow V \quad U \sim V = b}{(E \sim E', m) \Downarrow b} \text{ where } \sim \in \{=, <\}$$

Boolean Expressions:

$$\frac{(B, m) \Downarrow \mathsf{false}}{(B \& B', m) \Downarrow \mathsf{false}} \qquad \frac{(B, m) \Downarrow \mathsf{false} \quad (B', m) \Downarrow b}{(B \text{ or } B', m) \Downarrow b} \qquad \frac{(B, m) \Downarrow \mathsf{false}}{(\mathsf{not}\ B, m) \Downarrow \mathsf{true}}$$

$$\frac{(B, m) \Downarrow \mathsf{true} \quad (B', m) \Downarrow b}{(B \& B', m) \Downarrow b} \qquad \frac{(B, m) \Downarrow \mathsf{true}}{(B \text{ or } B', m) \Downarrow \mathsf{true}} \qquad \frac{(B, m) \Downarrow \mathsf{true}}{(\mathsf{not}\ B, m) \Downarrow \mathsf{false}}$$

Commands:

Assignment: $\dfrac{(E, m) \Downarrow V}{(I := E, m) \Downarrow m[I \leftarrow V]}$ where $m[I \leftarrow V](J) = \begin{cases} V & \text{if } J = I \\ m(J) & \text{otherwise} \end{cases}$

Skip: $(\mathsf{skip}, m) \Downarrow m$ Sequencing: $\dfrac{(C, m) \Downarrow m' \quad (C', m') \Downarrow m''}{(C; C', m) \Downarrow m''}$ Block: $\dfrac{(C, m) \Downarrow m'}{(\{C\}, m) \Downarrow m'}$

If-true: $\dfrac{(B, m) \Downarrow \mathsf{true} \quad (C, m) \Downarrow m'}{(\mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C'\ \mathsf{fi}, m) \Downarrow m'}$ If-false: $\dfrac{(B, m) \Downarrow \mathsf{false} \quad (C', m) \Downarrow m'}{(\mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C'\ \mathsf{fi}, m) \Downarrow m'}$

While-false: $\dfrac{(B, m) \Downarrow \mathsf{false}}{(\mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}\ , m) \Downarrow m}$ While-true: $\dfrac{(B, m) \Downarrow \mathsf{true} \ \ (C, m) \Downarrow m' \ \ (\mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}\ , m') \Downarrow m''}{(\mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}\ , m) \Downarrow m''}$

And the Natural Semantics for SIMPL2 is given as follows:
Constants:

Identifiers: $(I, m) \Downarrow (m(I), m)$ if $m(I)$ exists Numerals are values: $(N, m) \Downarrow (N, m)$

Booleans: $(\mathsf{true}, m) \Downarrow (\mathsf{true}, m)$ $(\mathsf{false}, m) \Downarrow (\mathsf{false}, m)$

Arithmetic Operations:

$$\frac{(E, m) \Downarrow (U, m') \quad (E', m') \Downarrow (V, m'') \quad U \oplus V = N}{(E \oplus E', m) \Downarrow (N, m'')}$$ where $\oplus \in \{+, *, -\}$ and $U, V \in \texttt{Values}$

Assignment:

$$\frac{(E, m) \Downarrow (V, m')}{(I := E, m) \Downarrow (V, m'[I \leftarrow V])}$$

Arithmetic Relations:

$$\frac{(E, m) \Downarrow (U, m') \quad (E', m') \Downarrow (V, m'') \quad U \sim V = b}{(E \sim E', m) \Downarrow (b, m'')}$$ where $\sim \in \{=, <\}$

Boolean Expressions:

$$\frac{(B, m) \Downarrow (\mathsf{false}, m')}{(B \& B', m) \Downarrow (\mathsf{false}, m')} \qquad \frac{(B, m) \Downarrow (\mathsf{false}, m') \quad (B', m') \Downarrow (b, m'')}{(B \ \mathsf{or}\ B', m) \Downarrow (b, m'')} \qquad \frac{(B, m) \Downarrow (\mathsf{false}, m')}{(\mathsf{not}\ B, m) \Downarrow (\mathsf{true}, m')}$$

$$\frac{(B, m) \Downarrow (\mathsf{true}, m') \quad (B', m') \Downarrow (b, m'')}{(B \& B', m) \Downarrow b} \qquad \frac{(B, m) \Downarrow (\mathsf{true}, m')}{(B \ \mathsf{or}\ B', m') \Downarrow (\mathsf{true}, m'')} \qquad \frac{(B, m) \Downarrow (\mathsf{true}, m')}{(\mathsf{not}\ B, m) \Downarrow (\mathsf{false}, m')}$$

Commands:

$$\frac{(E, m) \Downarrow (V, m')}{(E, m) \Downarrow m'}$$ Skip: $(\mathsf{skip}, m) \Downarrow m$ Sequencing: $\dfrac{(C, m) \Downarrow m' \quad (C', m') \Downarrow m''}{(C; C', m) \Downarrow m''}$ Block: $\dfrac{(C, m) \Downarrow m'}{(\{C\}, m) \Downarrow m'}$

If-true: $\dfrac{(B, m) \Downarrow (\mathsf{true}, m') \quad (C, m') \Downarrow m''}{(\mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C'\ \mathsf{fi}, m) \Downarrow m''}$ If-false: $\dfrac{(B, m) \Downarrow (\mathsf{false}, m') \quad (C', m') \Downarrow m''}{(\mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C'\ \mathsf{fi}, m) \Downarrow m''}$

While-false: $\dfrac{(B, m) \Downarrow (\mathsf{false}, m')}{(\mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}\ , m) \Downarrow m'}$ While-true: $\dfrac{(B, m) \Downarrow (\mathsf{true}, m') \ \ (C, m') \Downarrow m'' \ \ (\mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}\ , m'') \Downarrow m'''}{(\mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}\ , m) \Downarrow m'''}$

4.3 Transition Semantics for SIMPL1

Transition Semantics for SIMPL1 is given by the following rules:
Variables and Constants: Constants (numbers, true, false are constants. (N, m), (true, m) and (false, m) are terminal; they do not transition.

Identifiers: $(I, m) \longrightarrow (m(I), m)$

Arithmetic Expressions:

Let U, V be arithmetic values

$$\frac{(E, m) \longrightarrow (E'', m)}{(E \oplus E', m) \longrightarrow (E'' \oplus E', m)} \qquad \frac{(E, m) \longrightarrow (E', m)}{(V \oplus E, m) \longrightarrow (V \oplus E', m)}$$

$$(U \oplus V, m) \longrightarrow (N, m) \quad \text{where } U \oplus V = N$$

Arithmetic Relations:

$$\frac{(E,m) \longrightarrow (E'',m)}{(E \sim E',m) \longrightarrow (E'' \sim E',m)} \qquad \frac{(E,m) \longrightarrow (E',m)}{(V \sim E,m) \longrightarrow (V \sim E',m)}$$

$$(U \sim V,m) \longrightarrow (b,m) \quad \text{where } U \sim V = b$$

Boolean Expressions:

$$(\mathsf{false} \& B,m) \longrightarrow (\mathsf{false},m) \qquad \frac{(B,m) \longrightarrow (B'',m)}{(B \& B',m) \longrightarrow (B'' \& B',m)} \qquad (\mathsf{true\ or}\ B,m) \longrightarrow (\mathsf{true},m) \qquad \frac{(B,m) \longrightarrow (B'',m)}{(B\ \mathsf{or}\ B',m) \longrightarrow (B''\ \mathsf{or}\ B',m)}$$

$$(\mathsf{true} \& B,m) \longrightarrow (B,m) \qquad (\mathsf{false\ or}\ B,m) \longrightarrow (B,m)$$

$$(\mathsf{not\ true},m) \longrightarrow (\mathsf{false},m) \qquad \frac{(B,m) \longrightarrow (B',m)}{(\mathsf{not}\ B,m) \longrightarrow (\mathsf{not}\ B',m)}$$

$$(\mathsf{not\ false},m) \longrightarrow (\mathsf{true},m)$$

Commands:

$$\text{Skip: } (\mathsf{skip},m) \longrightarrow m \qquad \text{Block: } (\{C\},m) \longrightarrow (C,m)$$

$$\text{Assignment: } \frac{(E,m) \longrightarrow (E',m)}{(I := E,m) \longrightarrow (I := E',m)} \qquad (I := V,m) \longrightarrow m[I \leftarrow V]$$

$$\text{Sequencing: } \frac{(C,m) \longrightarrow (C'',m')}{(C;C',m) \longrightarrow (C'';C',m')} \qquad \frac{(C,m) \longrightarrow m'}{(C;C',m) \longrightarrow (C',m')}$$

If Then Else Command:

$$(\mathsf{if\ true\ then}\ C\ \mathsf{else}\ C'\ \mathsf{fi},m) \longrightarrow (C,m) \qquad (\mathsf{if\ false\ then}\ C\ \mathsf{else}\ C'\ \mathsf{fi},m) \longrightarrow (C',m)$$

$$\frac{(B,m) \longrightarrow (B',m)}{(\mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C'\ \mathsf{fi},m) \longrightarrow (\mathsf{if}\ B'\ \mathsf{then}\ C\ \mathsf{else}\ C'\ \mathsf{fi},m)}$$

While Command:

$$(\mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}\ ,m) \longrightarrow (\mathsf{if}\ B\ \mathsf{then}\ C;\mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od}\ \ \mathsf{else\ skip\ fi},m)$$

4.4 Reduction Semantics with Evaluation Contexts (Context Semantics) for SIMPL1

4.4.1 Syntax

Arithmetic Expressions:

$$\begin{array}{rcl} I & \in & \textit{Identifiers} \\ N & \in & \textit{Numerals} \\ E & ::= & N \mid I \mid E + E \mid E * E \mid E - E \mid EECxt[E] \\ EECxt & ::= & \Box_{\mathbf{E}} \mid EECxt + E \mid N + EECxt \mid EECxt * E \mid N * EECxt \mid EECxt - E \mid N - EECxt \end{array}$$

Boolean Expressions

$$\begin{array}{rcl} B & ::= & \mathsf{true} \mid \mathsf{false} \mid B \& B \mid B\ \mathsf{or}\ B \mid \mathsf{not}\ B \mid E < E \mid E = E \mid BECxt[E] \mid BBCxt[B] \\ BECxt & ::= & \mid BECxt\ \&\ B \mid BECxt\ \mathsf{or}\ B \mid \mathsf{not}\ BECxt \mid EECxt < E \mid N < EECxt \mid EECxt = E \mid N = EECxt \\ BBCxt & ::= & \Box_{\mathbf{B}} \mid BBCxt\ \&\ B \mid BBCxt\ \mathsf{or}\ B \mid \mathsf{not}\ BBCxt \end{array}$$

Commands

$$\begin{array}{rcl} C & ::= & \mathsf{skip} \mid C; C \mid I ::= E \mid \mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi} \\ & \mid & \{C\} \mid \mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od} \mid CECxt[E] \mid CBCxt[B] \mid CCCxt[C] \\ CECxt & ::= & CECxt; C \mid I := EECxt \mid \mathsf{if}\ BECxt\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi} \\ CBCxt & ::= & CBCxt; C \mid \mathsf{if}\ BBCxt\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi} \\ CCCxt & ::= & \Box_{\mathbf{C}} \mid CCCxt; C \end{array}$$

4.4.2 Equational Rules:

Arithmetic Contexts:

$$\Box_{\mathbf{E}}[E] = E$$
$$(EECxt \oplus E)[E'] = EECxt[E'] \oplus E$$
$$(N \oplus EECxt)[E] = N \oplus (EECxt[E])$$

Substitution in Boolean Contexts:

$$(EECxt \sim E)[E'] = EECxt[E'] \sim E$$
$$(N \sim EECxt)[E] = N \sim (EECxt[E])$$
$$(BECxt \& B)[E] = BECxt[E] \& B$$
$$(BECxt\ \mathsf{or}\ B)[E] = BECxt[E]\ \mathsf{or}\ B$$
$$(\mathsf{not}\ BECxt)[E] = \mathsf{not}\ (BECxt[E])$$

$$\Box_{\mathbf{B}}[B] = B$$
$$(BBCxt \& B)[B'] = BBCxt[B'] \& B$$
$$(EECxt\ \mathsf{or}\ B)[B'] = BBCxt[B']\ \mathsf{or}\ B$$
$$(\mathsf{not}\ BBCxt)[B'] = \mathsf{not}\ (BBCxt[B])$$

Substitution in Command Contexts:

$$(CECxt; C)[E] = CECxt[E]; C$$
$$(I := EECxt)[E] = I := (EECxt[E])$$

$$(\mathsf{if}\ BECxt\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi})[E] = \mathsf{if}\ (BECxt[E])\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi}$$

$$(CBCxt; C)[B] = CBCxt[B]; C$$
$$(\mathsf{if}\ BBCxt\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi})[B] = \mathsf{if}\ BBCxt[B]\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi}$$

$$\Box_{\mathbf{C}}[C] = C$$
$$(CCCxt; C)[C'] = CCCxt[C']; C$$

4.4.3 Transitions

Arithmetic Expressions

$$(I, m) \longrightarrow (m(I), m)$$
$$(N \oplus N', m) \longrightarrow (N'', m) \quad \text{if } N \oplus N' = N''$$

Booleans

$$(N \sim N', m) \longrightarrow (b, m) \qquad \text{if } N \sim N' = b$$

$$(\mathsf{true} \& B, m) \longrightarrow (B, m) \qquad (\mathsf{false} \& B, m) \longrightarrow (\mathsf{false}, m)$$
$$(\mathsf{true\ or}\ B, m) \longrightarrow (\mathsf{true}, m) \qquad (\mathsf{false\ or}\ B, m) \longrightarrow (B, m)$$
$$(\mathsf{not\ true}, m) \longrightarrow (\mathsf{false}, m) \qquad (\mathsf{not\ false}, m) \longrightarrow (\mathsf{true}, m)$$

Commands

$$(I := V, m) \longrightarrow (\mathsf{skip}, m[I \leftarrow V])$$

$$(\mathsf{skip} : C, m) \longrightarrow (C, m)$$

$$(\{C\}, m) \longrightarrow (C, m)$$

$$(\mathsf{if\ true\ then}\ C\ \mathsf{else}\ C'\ \mathsf{fi}, m) \longrightarrow (C, m)$$
$$(\mathsf{if\ false\ then}\ C\ \mathsf{else}\ C'\ \mathsf{fi}, m) \longrightarrow (C', m)$$

$$(\mathsf{while}\ B\ \mathsf{do}\ C, m) \longrightarrow (\mathsf{if}\ B\ \mathsf{then}\ C; \mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{else\ skip\ fi}, m)$$

5 Problems

1. Give a Transition Semantics for SIMPL2 that is consistent with its Natural Semantics, and at the same level of computation steps as given in the Transition Semantics for SIMPL1. You may either give the whole description, or describe the changes necessary to the Transition Semantics for SIMPL1 to create the Transition Semantics for SIMPL2. If you choose to describe how to change SIMPL1 semantics, your description must be clear as to what changes are done where.

   Solution: Transition Semantics for SIMPL2 is given by the following rules:

   Variables and Constants: Constants (numbers, true, false are constants. (N, m), (true, m) and (false, m) are terminal; they do not transition.

   $$\text{Identifiers:} \quad (I, m) \longrightarrow (m(I), m)$$

   Arithmetic Expressions:

   Let U, V be arithmetic values

   $$\frac{(E, m) \longrightarrow (E'', m')}{(E \oplus E', m) \longrightarrow (E'' \oplus E', m')} \qquad \frac{(E, m) \longrightarrow (E', m')}{(V \oplus E, m) \longrightarrow (V \oplus E', m')}$$

   $$(U \oplus V, m) \longrightarrow (N, m) \quad \text{where } U \oplus V = N$$

   Assignment:

   $$\frac{(E, m) \longrightarrow (E', m')}{(I := E, m) \longrightarrow (I := E', m')} \qquad (I := V, m) \longrightarrow (V, m[I \leftarrow V])$$

   Arithmetic Relations:

   $$\frac{(E, m) \longrightarrow (E'', m')}{(E \sim E', m) \longrightarrow (E'' \sim E', m')} \qquad \frac{(E, m) \longrightarrow (E', m')}{(V \sim E, m) \longrightarrow (V \sim E', m')}$$

   $$(U \sim V, m) \longrightarrow (b, m) \quad \text{where } U \sim V = b$$

Boolean Expressions:

$$(\text{false}\&B, m) \longrightarrow (\text{false}, m) \qquad \frac{(B, m) \longrightarrow (B'', m')}{(B\&B', m) \longrightarrow (B''\&B', m')} \qquad (\text{true or } B, m) \longrightarrow (\text{true}, m) \qquad \frac{(B, m) \longrightarrow (B'', m')}{(B \text{ or } B', m) \longrightarrow (B'' \text{ or } B', m')}$$
$$(\text{true}\&B, m) \longrightarrow (B, m) \qquad (\text{false or } B, m) \longrightarrow (B, m)$$

$$(\text{not true}, m) \longrightarrow (\text{false}, m) \qquad \frac{(B, m) \longrightarrow (B', m')}{(\text{not } B, m) \longrightarrow (\text{not } B', m')}$$
$$(\text{not false}, m) \longrightarrow (\text{true}, m)$$

Commands:

$$\text{Skip:} \quad (\text{skip}, m) \longrightarrow m \qquad \text{Block:} \quad (\{C\}, m) \longrightarrow (C, m)$$

Rules for Assignment have been deleted from the rules for Commands.

$$\text{Expressions:} \quad \frac{(E, m) \longrightarrow (E', m')}{(E, m) \longrightarrow (E', m')} \qquad \text{Value:} \quad (V, m) \longrightarrow m$$

In Expressions, $\longrightarrow$ below line is between command-memory pairs, and above the line it is between expression-memory pairs.

$$\text{Sequencing:} \quad \frac{(C, m) \longrightarrow (C'', m')}{(C; C', m) \longrightarrow (C''; C', m')} \qquad \frac{(C, m) \longrightarrow m'}{(C; C', m) \longrightarrow (C', m')}$$

If Then Else Command:

$$(\text{if true then } C \text{ else } C' \text{ fi}, m) \longrightarrow (C, m) \qquad (\text{if false then } C \text{ else } C' \text{ fi}, m) \longrightarrow (C', m)$$
$$\frac{(B, m) \longrightarrow (B', m')}{(\text{if } B \text{ then } C \text{ else } C' \text{ fi}, m) \longrightarrow (\text{if } B' \text{ then } C \text{ else } C' \text{ fi}, m')}$$

While Command:

$$(\text{while } B \text{ do } C \text{ od }, m) \longrightarrow (\text{if } B \text{ then } C; \text{while } B \text{ do } C \text{ od } \text{ else skip fi}, m)$$

Note: in Expressions, the transition above the line is done as an expression, while the tranisiton below the line is done as a command.

2. Give a Context Semantics for SIMPL2 that is consistent with its Natural Semantics, and at the same level of computation steps as given in the Context Semantics for SIMPL1. You may either give the whole description, or describe the changes necessary to the Context Semantics for SIMPL1 to create the Context Semantics for SIMPL2. If you choose to describe how to change SIMPL1 semantics, your description must be clear as to what changes are done where.

Solution:
Syntax
Arithmetic Expressions:

$$\begin{aligned} I &\in \textit{Identifiers} \\ N &\in \textit{Numerals} \\ E &::= N \mid I \mid E + E \mid E * E \mid E - E \mid I ::= E \mid EECxt[E] \\ EECxt &::= \square_{\mathbf{E}} \mid EECxt + E \mid N + EECxt \mid EECxt * E \mid N * EECxt \mid EECxt - E \mid N - EECxt \mid I ::= EECxt \end{aligned}$$

Boolean Expressions

$$\begin{aligned} B &::= \text{true} \mid \text{false} \mid B\&B \mid B \text{ or } B \mid \text{not } B \mid E < E \mid E = E \mid BECxt[E] \mid BBCxt[B] \\ BECxt &::= \mid BECxt \ \& \ B \mid BECxt \text{ or } B \mid \text{not } BECxt \mid EECxt < E \mid N < EECxt \mid EECxt = E \mid N = EECxt \\ BBCxt &::= \square_{\mathbf{B}} \mid BBCxt \ \& \ B \mid BBCxt \text{ or } B \mid \text{not } BBCxt \end{aligned}$$

Commands

$$
\begin{array}{rcl}
C & ::= & \mathsf{skip} \mid C;C \mid E \mid \mathsf{if}\ B\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi} \\
 & \mid & \{C\} \mid \mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{od} \mid CECxt[E] \mid CBCxt[B] \mid CCCxt[C] \\
CECxt & ::= & CECxt;C \mid EECxt \mid \mathsf{if}\ BECxt\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi} \\
CBCxt & ::= & CBCxt;C \mid \mathsf{if}\ BBCxt\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi} \\
CCCxt & ::= & \Box_{\mathbf{C}} \mid CCCxt;C
\end{array}
$$

The production for assignments has been removed from the syntactic category of Commands C, and the corresponding assignment context has been removed from Command Expression Contexts $CECxt$.

Equational Rules:
Arithmetic Contexts:

$$\Box_{\mathbf{E}}[E] = E$$
$$(EECxt \oplus E)[E'] = EECxt[E'] \oplus E \qquad \oplus \in \{+, -.*\}$$
$$(N \oplus EECxt)[E] = N \oplus (EECxt[E])$$
$$(I := EECxt)[E] = I := (EECxt[E])$$

Substitution in Boolean Contexts:

$$(EECxt \sim E)[E'] = EECxt[E'] \sim E \qquad \sim \in \{=, <\}$$
$$(N \sim EECxt)[E] = N \sim (EECxt[E])$$
$$(BECxt \& B)[E] = BECxt[E] \& B$$
$$(BECxt\ \mathsf{or}\ B)[E] = BECxt[E]\ \mathsf{or}\ B$$
$$(\mathsf{not}\ BECxt)[E] = \mathsf{not}\ (BECxt[E])$$

$$\Box_{\mathbf{B}}[B] = B$$
$$(BBCxt \& B)[B'] = BBCxt[B'] \& B$$
$$(EECxt\ \mathsf{or}\ B)[B'] = BBCxt[B']\ \mathsf{or}\ B$$
$$(\mathsf{not}\ BBCxt)[B'] = \mathsf{not}\ (BBCxt[B])$$

Substitution in Command Contexts:

$$(CECxt;C)[E] = CECxt[E];C$$

$EECxt[E] = EECxt[E]$ On the lhs $[\,] : CECxt \to E \to C$, rhs $[\,] : EECxt \to E$

Rule for subtitution in an assignment removed from Command Contexts $CCCxt$.

$$(\mathsf{if}\ BECxt\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi})[E] = \mathsf{if}\ (BECxt[E])\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi}$$

$$(CBCxt;C)[B] = CBCxt[B];C$$
$$(\mathsf{if}\ BBCxt\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi})[B] = \mathsf{if}\ BBCxt[B]\ \mathsf{then}\ C\ \mathsf{else}\ C\ \mathsf{fi}$$

$$\Box_{\mathbf{C}}[C] = C$$
$$(CCCxt;C)[C'] = CCCxt[C'];C$$

Transitions

Arithmetic Expressions

$$(I, m) \longrightarrow (m(I), m)$$
$$(N \oplus N', m) \longrightarrow (N'', m) \quad \text{if } N \oplus N' = N''$$
$$(I := N, m) \longrightarrow (N, m[I \leftarrow N])$$

Booleans

$$(N \sim N', m) \longrightarrow (b, m) \qquad \text{if } N \sim N' = b$$

$$(\mathsf{true} \& B, m) \longrightarrow (B, m) \qquad (\mathsf{false} \& B, m) \longrightarrow (\mathsf{false}, m)$$
$$(\mathsf{true\ or}\ B, m) \longrightarrow (\mathsf{true}, m) \qquad (\mathsf{false\ or}\ B, m) \longrightarrow (B, m)$$
$$(\mathsf{not\ true}, m) \longrightarrow (\mathsf{false}, m) \qquad (\mathsf{not\ false}, m) \longrightarrow (\mathsf{true}, m)$$

Commands

Rule for transitioning an assignment removed from Commands.

$$(\mathsf{skip}; C, m) \longrightarrow (C, m)$$

$$(N, m) \longrightarrow (\mathsf{skip}, m)$$

$$(\{C\}, m) \longrightarrow (C, m)$$

$$(\mathsf{if\ true\ then}\ C\ \mathsf{else}\ C'\ \mathsf{fi}, m) \longrightarrow (C, m)$$
$$(\mathsf{if\ false\ then}\ C\ \mathsf{else}\ C'\ \mathsf{fi}, m) \longrightarrow (C', m)$$

$$(\mathsf{while}\ B\ \mathsf{do}\ C, m) \longrightarrow (\mathsf{if}\ B\ \mathsf{then}\ C; \mathsf{while}\ B\ \mathsf{do}\ C\ \mathsf{else\ skip\ fi}, m)$$

Context Reductions (unchanged from SIMP1)

$$\frac{(E, m) \longrightarrow (E', m)}{(EECxt[E], m) \longrightarrow (EECxt[E'], m)} \qquad \frac{(E, m) \longrightarrow (E', m)}{(BECxt[E], m) \longrightarrow (BECxt[E'], m)} \qquad \frac{(E, m) \longrightarrow (E', m)}{(CECxt[E], m) \longrightarrow (CECxt[E'], m)}$$

$$\frac{(B, m) \longrightarrow (B', m)}{(BBCxt[B], m) \longrightarrow (BBCxt[B'], m)} \qquad \frac{(B, m) \longrightarrow (B', m)}{(CBCxt[B], m) \longrightarrow (CBCxt[B'], m)}$$

$$\frac{(C, m) \longrightarrow (C', m)}{(CCCxt[C], m) \longrightarrow (CCCxt[C'], m)}$$