HW 10 – Structural Operational and Transition Semantics

CS 421 – Fall 2014
Revision 1.0

Assigned Friday, November 14, 2014
Due Friday, November 21, 2014, 11:59 PM
Extension 48 hours (20% penalty)

1 Change Log

1.1 Corrected the final value of `y`.

1.0 Initial Release.

2 Turn-In Procedure

Answer the problems below, save your work as a PDF (either scanned if handwritten or converted from a program), add the PDF to the subversion repository (`svn add hw10-submission.pdf`)
and commit it (`svn commit -m "submitting hw10"`). Your file should be named `hw10-submission.pdf` and committed in your `assignments/hw10` directory.

3 Objectives and Background

The purpose of this HW is to test your understanding of

- The difference between structural operational semantics and transition semantics.
- How to write rules for transition semantics.

Another purpose of this HW is to provide you with experience answering non-programming written questions of the kind you may experience on the midterms and final.

All problems on the homework will be based on the language discussed in class, which has the following syntax:

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

4 Problems

1. a. (10 points) Using the rules given for natural semantics in class, give a proof that, starting with a memory that maps `x` to 1, `if not(x > 5) then y:= x+1 else y:=y-1 fi` evaluates to a memory where `x` maps to 1 and `y` maps to 2.

Solution:

$$\dfrac{\dfrac{\dfrac{}{(x,\{x\to1\})\Downarrow 1}\,\textit{Identifier} \quad \dfrac{}{(5,\{x\to1\})\Downarrow 5}\,\textit{Constant}}{(x>5,\{x\to1\})\Downarrow \text{false}}\,\textit{Relation}}{(\text{not } (x>5),\{x\to1\})\Downarrow \text{true}}\,\textit{Boolean-not} \qquad \dfrac{\dfrac{\dfrac{}{(x,\{x\to1\})\Downarrow 1}\,\textit{Identifier} \quad \dfrac{}{(1,\{x\to1\})\Downarrow 1}\,\textit{Constant}}{(x+1,\{x\to1\})\Downarrow 2}\,\textit{Arith Exp}}{(y:=x+1,\{x\to1\})\Downarrow \{y\to2,\ x\to1\}}\,\textit{Assignment}$$

$$\dfrac{(\text{not } (x>5),\{x\to1\})\Downarrow \text{true} \qquad (y:=x+1,\{x\to1\})\Downarrow \{y\to2,\ x\to1\}}{(\text{if not } (x>5) \text{ then } y:=x+1 \text{ else } y:=y+1 \text{ fi},\{x\to1\})\Downarrow \{y\to2,\ x\to1\}}\,\textit{If-then-else}$$

b. (5 points) Add the natural semantics (*a.k.a.* big-step semantics) for the `if_then_` command.

Solution:

$$\dfrac{(B,m)\Downarrow \text{true} \qquad (C,m)\Downarrow m'}{(\text{if B then C, m})\Downarrow m'} \qquad \dfrac{(B,m)\Downarrow \text{false}}{(\text{if B then C, m})\Downarrow m}$$

I would also accept

$$\dfrac{(\text{if B then C else skip, m})\Downarrow m'}{(\text{if B then C, m})\Downarrow m'}$$

c. (5 points) Add the transition semantics for the same.

$$\dfrac{}{(\text{if true then C, m})\Rightarrow (C,m)} \qquad \dfrac{}{(\text{if false then C, m})\Rightarrow m}$$

$$\dfrac{(B,m)\Rightarrow (B',m)}{(\text{if B then C, m})\Rightarrow (\text{if B' then C, m})}$$

I would also accept

$$\dfrac{}{(\text{if B then C, m})\Rightarrow (\text{if B then C else skip, m})}$$