CSE505 Homework#2

Chenglong Wang

1 Problem 1: Small-step Semantics

In the following context, we use symbol H for Heap, e for expressions, i for int values, x for variables, and l for lists of integers.

Heap we define the heap H as a map from variables to an pair of type ($\mathsf{Int} \times (\mathsf{list\ Int})$). i.e. the heap H has the following type:

$$H : \mathsf{Var} \to \mathsf{Int} \times (\mathsf{list\ Int})$$

Eval The function $Eval(H, e)$ is the evaluation function for an expression, as defined in the lecture, $Eval$ takes the current heap H and an expression e, the result is either a boolean or an integer depending on the type of e.

In the following rules, $H \mid s \to H' \mid s'$ can be pronounced as "Evaluate one step on the statement s under the current heap H will update the heap to H' and the statement is reduced to s'".

$$\frac{}{H \mid \mathsf{Seq\ Nop}\ s \to H \mid s}\ \text{(Step-Nop)} \qquad \frac{Eval(H, e) = n \qquad H\ x = (i, l)}{H \mid \mathsf{Assign}\ x\ e \to [x \mapsto (i, l)]H \mid \mathsf{Nop}}\ \text{(Step-Assign)}$$

$$\frac{H \mid s_1 \to H_1 \mid s_1'}{H \mid \mathsf{Seq}\ s_1\ s_2 \to H_1 \mid \mathsf{Seq}\ s_1'\ s_2}\ \text{(Step-Seq)} \qquad \frac{Eval(H, e) = \texttt{false}}{H \mid \mathsf{Cond}\ e\ s \to H \mid \mathsf{Nop}}\ \text{(Step-Cond-F)}$$

$$\frac{Eval(H, e) = \texttt{true}}{H \mid \mathsf{Cond}\ e\ s \to H \mid s}\ \text{(Step-Cond-T)} \qquad \frac{Eval(H, e) = \texttt{false}}{H \mid \mathsf{While}\ e\ s \to H \mid \mathsf{Nop}}\ \text{(Step-While-F)}$$

$$\frac{Eval(H, e) = \texttt{true}}{H \mid \mathsf{While}\ e\ s \to H \mid \mathsf{Seq}\ s\ (\mathsf{While}\ e\ s)}\ \text{(Step-While-T)}$$

$$\frac{H\ x = (i, l)}{H \mid \mathsf{PushVar}\ x \to [x \mapsto (i, i :: l)]H \mid \mathsf{Nop}}\ \text{(Step-PushVar)}$$

$$\frac{H\ x = (i, [\,])}{H \mid \mathsf{PopVar}\ x \to \mathsf{Nop}}\ \text{(Step-PopVar-E)} \qquad \frac{H\ x = (i, i' :: l)}{H \mid \mathsf{PopVar}\ x \to [x \mapsto (i', l)]H \mid \mathsf{Nop}}\ \text{(Step-PopVar)}$$

2 Problem 2

Before proving the theorem, we first define the function $\mathsf{nPushVar} : \mathsf{Var} \to \mathsf{Stmt} \to Int$.

$$\begin{aligned}
&\mathsf{nPushVar}\ x\ \mathsf{Nop} := 0 \\
&\mathsf{nPushVar}\ x\ (\mathsf{Assign}\ y\ e) := 0 \\
&\mathsf{nPushVar}\ x\ (\mathsf{Seq}\ s_1\ s_2) := \mathsf{nPushVar}\ x\ s_1 + \mathsf{nPushVar}\ x\ s_2 \\
&\mathsf{nPushVar}\ x\ (\mathsf{Cond}\ e\ s) := \mathsf{nPushVar}\ x\ s \\
&\mathsf{nPushVar}\ x\ (\mathsf{While}\ e\ s) := \mathsf{nPushVar}\ x\ s \\
&\mathsf{nPushVar}\ x\ (\mathsf{PushVar}\ y) := \texttt{if}(x \equiv y)\ \texttt{then}\ 1\ \texttt{else}\ 0 \\
&\mathsf{nPushVar}\ x\ (\mathsf{PopVar}\ y) := 0
\end{aligned}$$

Lemma 1. *If a statement s has no While loops, starting from the heap H and $H \mid s \rightarrow H' \mid s'$, then for all x,* $\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s' \leq \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s$*.*

Proof. Proof by induction on s.

- **Case** $s = \mathsf{Nop}$: In this case, no reduction of s can be derived, thus the property hold.
- **Case** $s = \mathsf{Seq}\ s_1\ s_2$: In this case, as $H \mid s \rightarrow H' \mid s'$, by the rule Step-Seq, there exists s_1' s.t.

$$H \mid s_1 \rightarrow H' \mid s_1' \tag{1}$$

  and

$$s' = \mathsf{Seq}\ s_1'\ s_2 \tag{2}$$

  Thus, by induction and (1), we have the following property:

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s_1' \leq \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s_1 \tag{3}$$

  According to the definition of $\mathsf{nPushVar}\ x\ s$, we have that

$$\mathsf{nPushVar}\ x\ (\mathsf{Seq}\ s_1\ s_2) = \mathsf{nPushVar}\ x\ s_1 + \mathsf{nPushVar}\ x\ s_2 \tag{4}$$

  and

$$\mathsf{nPushVar}\ x\ (\mathsf{Seq}\ s_1'\ s_2) = \mathsf{nPushVar}\ x\ s_1' + \mathsf{nPushVar}\ x\ s_2 \tag{5}$$

  By substitution $\mathsf{nPushVar}\ x\ s_1'$ and $\mathsf{nPushVar}\ x\ s_1$ using 4 and 5, we have

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ (\mathsf{Seq}\ s_1'\ s_2) \leq \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ (\mathsf{Seq}\ s_1\ s_2) \tag{6}$$

  Beside, as we have 2 and $s = \mathsf{Seq}\ s_1\ s_2$, replacing them in 6, we have

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s' \leq \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s \tag{7}$$

  and then the case is proved.
- **Case** $s = \mathsf{Cond}\ e\ s_1$. There are two subcases regarding to the value of e evaluated under H.
  - **Sub-Case** $\mathsf{Eval}\ H\ e = \mathtt{false}$: According to the rule Step-Cond-F, after evaluation, $s' = Nop$ and $H' = H$. Thus

$$\mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s' = \mathsf{length}\ (\mathsf{snd}\ (H\ x))$$

    and

$$\mathsf{length}\ (\mathsf{snd}\ (H\ x)) \leq \mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s$$

    So the property holds for this sub-case.
  - **Sub-Case** $\mathsf{Eval}\ H\ e = \mathtt{true}$ According to the rule Step-Cond-T, $s' = s_1$ and $H' = H$. By definition of $\mathsf{nPushVar}$, we have $\mathsf{nPushVar}\ x\ (\mathsf{Cond}\ e\ s_1) = \mathsf{nPushVar}\ x\ s_1$. Thus we have

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s' = \mathsf{length}\ (\mathsf{snd}\ (H\ x))$$

    and

$$\mathsf{length}\ (\mathsf{snd}\ (H\ x)) \leq \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s$$

    And the sub-case is proved.

  With both of the subcase proved, we have the case proved.
- **Case** $s = \mathsf{While}\ e\ s_1$. As we required that there is no While in s, this case does not exist.
- **Case** $s = \mathsf{PushVar}\ y$. There are two subcases based on whether y is x or not.

- **Sub-case:** $y \equiv x$, in this case, $\mathsf{length}(H'\ x) = \mathsf{length}(H\ x) + 1$, while $\mathsf{nPushVar}\ x\ s = \mathsf{nPushVar}\ x\ s' + 1$. So we have

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s' = \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s$$

and the sub-case is proved.

- **Sub-case:** $y \neq x$, in this case, after step-reduction, $\mathsf{snd}\ (H\ x) = \mathsf{snd}\ (H'\ x)$ and the number of $PushVar\ x$ stay unchanged, so the lemma still holds.

Thus in the case the lemma holds.

- **Case** $s = \mathsf{PopVar}\ y$. Similar to the previous case, there are two subcases regarding to the equivalence of x and y.

  - $x \equiv y$, then the number of nPushVar x remain the same but the length of $\mathsf{snd}\ H\ x$ is reduced by 1, which means that

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s' = \mathsf{length}\ (\mathsf{snd}\ (H\ x)) - 1 + \mathsf{nPushVar}\ x\ s$$

  and the lemma holds.

  - $x \neq y$, then both $\mathsf{snd}\ (H\ x) = \mathsf{snd}\ (H'\ x)$ and the number of $\mathsf{PushVar}\ x$ stay unchanged, so the lemma holds.

  Having these two sub-cases proved, the lemma holds in this case.

With all the case proved, the lemma hold for all s and H. □

Lemma 2. *If a statement s with heap H can step to s' with heap status H', then for all x, we have* $\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s' \leq \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s$.

Proof. By induction on the number of derivation steps n.

- $n = 1$, the case is proved according to the Lemma 1.
- $n = k + 1$, by induction, s with H can step k steps to s_1 with H_1, and

$$\mathsf{length}\ (\mathsf{snd}\ (H_1\ x)) + \mathsf{nPushVar}\ x\ s_1 \leq \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s. \tag{8}$$

  As we can step from s_1 with H_1 to s' with H' in one step, according to Lemma 1, we have

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s' \leq \mathsf{length}\ (\mathsf{snd}\ (H_1\ x)) + \mathsf{nPushVar}\ x\ s_1. \tag{9}$$

  By combining 8 with 8, we have

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) + \mathsf{nPushVar}\ x\ s' \leq \mathsf{length}\ (\mathsf{snd}\ (H\ x)) + \mathsf{nPushVar}\ x\ s \tag{10}$$

  So that the case is proved.

Then we have the lemma proved. □

Property 1. *If a statement s has no While loops and from the empty heap s can step to heap h' and statement s', then for all variables x, the length of the stack for x in h' does not exceed the number of* $\mathsf{PushVar}\ x$ *statements in s (the original statement).*

Proof. By applying Lemma 2 with $H = \{\forall\ x.x \mapsto (0, [\,])\}$, we have:

$$\mathsf{length}\ (\mathsf{snd}\ (\ x)) + \mathsf{nPushVar}\ x\ s' \leq 0 + \mathsf{nPushVar}\ x\ s$$

By reduction, we have

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) \leq \mathsf{nPushVar}\ x\ s - \mathsf{nPushVar}\ x\ s'$$

as $\mathsf{nPushVar}\ x\ s'$ is alway greater than 0, we have:

$$\mathsf{length}\ (\mathsf{snd}\ (H'\ x)) \leq \mathsf{nPushVar}\ x\ s$$

So the property is proved. □