ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

D. Basin and P. Müller

Formal Methods and Functional Programming

Exercise Sheet 11: Small Step Semantics

Submission deadline: May 14th, 2011

Please submit your solution before **10:15am** on the submission date specified above. Solutions can be submitted via e-mail or by using the boxes in front of **CAB F 51.1**. Make sure that the first page (and preferably each sheet) always contains your name, the exercise sheet number as well as your tutor's name and the weekday (Tuesday or Wednesday) of your exercise group. Don't forget to staple your pages if you submit more than one page.

Assignment 1: Implementing SOS

In this assignment you will extend the simple IMP interpreter with the structural operational semantics. Download the skeleton file `simpi_skeleton2.lhs` from the course web page and implement the function

```
transSOS :: Config -> Config
```

that encodes the rules presented in the lecture for the structural operational semantics. The places where you have to insert your code in the skeleton file are marked by TODO. Compare your implementation of transSOS with the function transNS that implements the rules for the natural semantics.

Please mail your solution of this assignment to your tutor. The email addresses of the tutors are:

Alex Summers	`alexander.summers@inf.ethz.ch`
Yannis Kassios	`ioannis.kassios@inf.ethz.ch`
Malte Schwerhoff	`malte.schwerhoff@inf.ethz.ch`

Assignment 2: SOS derivation sequence

Consider the following **IMP** statement s:

```
while n # 0 do
   (a := a+n;
    b := b*n);
    n := n-1
end
```

Let σ be a state such that $\sigma(\mathtt{a}) = 0$, $\sigma(\mathtt{b}) = 1$, and $\sigma(\mathtt{n}) = 2$. Prove using the structural operational semantics that there is a state σ' with $\sigma'(\mathtt{a}) = 3$, $\sigma'(\mathtt{b}) = 2$, and $\sigma'(\mathtt{n}) = 0$ such that $\langle s, \sigma\rangle \to_1^* \sigma'$. Hint: provide the complete derivation sequence. You have not to provide the derivation tree for each individual transition.

Assignment 3: Composing executions

Let s_1 and s_2 be statements, σ and σ' states, and k a positive integer. Prove that if $\langle s_1, \sigma\rangle \to_1^k \sigma'$ then $\langle s_1; s_2, \sigma\rangle \to_1^k \langle s_2, \sigma'\rangle$.

Assignment 4: Semantic equivalence of NS and SOS

Recall the equivalence theorem on p. 154 of the lecture slides. In order to prove it, two equivalence lemmas have to be proved:

(L1) $\forall \sigma, \sigma' \in \mathtt{State}, s \in \mathtt{Stm} \cdot \langle s, \sigma\rangle \to \sigma' \Rightarrow \langle s, \sigma\rangle \to_1^* \sigma'$

(L2) $\forall \sigma, \sigma' \in \mathtt{State}, s \in \mathtt{Stm}, k \in \mathbb{N} \cdot \langle s, \sigma\rangle \to_1^k \sigma' \Rightarrow \langle s, \sigma\rangle \to \sigma'$

Prove both lemmas. You may use the result of assignment 3 as well as the lemma on p. 132 of the lecture slides.

Assignment 5 - Headache of the week: Transactions

Consider the IMP extension

`revert` s `if` b

where statement s is executed, but the effects of s are reverted if boolean expression b holds in the post-state of s. This construct can be seen as a very simple form of transaction management as used by databases.

Hint I: You might want to consider changing the definition of states.
Hint II: Keep in mind that `revert` can be nested.