ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

D. Basin and P. Müller

Formal Methods and Functional Programming

Exercise Sheet 8: Big Step Semantics

Submission deadline: April 28th, 2009

Assignment 1

(a) Intuitively, $e[y \mapsto e']$ means replacing each occurrence of the variable y in the arithmetic expression e with the arithmetic expression e'. Formally, we define the substitution $e[y \mapsto e']$ as

$$e[y \mapsto e'] = \begin{cases} n & \text{if } e \text{ is the integer } n, \\ e' & \text{if } e \text{ is the variable } y, \\ x & \text{if } e \text{ is the variable } x \neq y, \text{ and} \\ \big(e_1[y \mapsto e'] \; op \; e_2[y \mapsto e']\big) & \text{if } e \text{ is the arithmetic expression } (e_1 \; op \; e_2), \\ & \quad \text{where } op \text{ is an arithmetic operator.} \end{cases}$$

Prove that for all arithmetic expressions e, e', all variables y, and all states σ,

$$\mathcal{A}[\![e[y \mapsto e']]\!]\sigma = \mathcal{A}[\![e]\!]\big(\sigma[y \mapsto \mathcal{A}[\![e']\!]\sigma]\big) \,.$$

What is the intuition of this equality?

(b) Define a substitution function $b[y \mapsto e]$ for Boolean expressions that replaces all occurrences of the variable y in the Boolean expressions b with the arithmetic expression e. Prove that your definition satisfies the equality

$$\mathcal{B}[\![b[y \mapsto e]]\!]\sigma = \mathcal{B}[\![b]\!]\big(\sigma[y \mapsto \mathcal{A}[\![e]\!]\sigma]\big) \,,$$

for all Boolean expressions b, all arithmetic expressions e, all variables y, and all states σ.

Assignment 2

Let s be the following statement:

```
y := 0;
while x>0 do
  y := y + x;
  x := x - 2
end
```

(a) What function is implemented by the **IMP** program s when the variable x initially stores a positive integer?

(b) Let σ be a state with $\sigma(\mathtt{x}) = 3$. Prove that there is a state σ' with $\sigma'(\mathtt{y}) = 4$ such that $\langle s, \sigma\rangle \to \sigma'$.

Assignment 3

In the lecture, you have seen the proof of the direction from left to right of the following claim:

$$\langle \texttt{while } b \texttt{ do } s \texttt{ end}, \sigma\rangle \to \sigma' \Leftrightarrow \langle \texttt{if } b \texttt{ then } s\texttt{; while } b \texttt{ do } s \texttt{ end else skip end}, \sigma\rangle \to \sigma'$$

Prove the direction from right to left of the claim.

Assignment 4

Consider the extension of **IMP** with the construct

$$\texttt{repeat } s \texttt{ until } b$$

where s is a statement and b a Boolean expression.

Give deduction rules for the natural semantics that capture the semantics of this loop construct.

Assignment 5

In this assignment you will write a simple interpreter for **IMP** programs. You will use the programming language Haskell. A skeleton of the **IMP** interpreter as a literate Haskell file is available at the course web page. The skeleton file contains the data types for arithmetic expressions, Boolean expressions, and statements for representing **IMP** programs in Haskell. Moreover, the skeleton file contains some auxiliary functions (e.g., evaluating arithmetic and Boolean expressions, parsing **IMP** programs, and input/output).

Download the skeleton file and complete the definition of the function

```
transNS :: Config -> Config
```

The place where you should insert your code in the skeleton file is marked by the word `TODO`. The function `transNS` should encode the rules of the transition relation from the lecture for the natural semantics. Feel free to extend **IMP**, e.g., with local variables. Test you program on the **IMP** programs that are also available from the course web page.

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

Thai Son Hoang	thai.hoang@inf.ethz.ch
Felix Klaedtke	felixkl@inf.ethz.ch
Mohammad Torabi Dashti	mohammad.torabi@inf.ethz.ch

Assignment 6

(a) Define an ordering $\prec$ over the plane of integers (i.e., $\mathbb{Z} \times \mathbb{Z}$) that is well founded.

(b) Let V be the set of all finite subsets of natural numbers. Convince yourself that the proper-subset relation $\subsetneq$ is well founded over V. State the corresponding induction principle over V for $\subsetneq$. What are the base cases and what needs to be proved in the step cases?

1. Prove by induction that every $S \in V$ has $2^{|S|}$ subsets, where $|S|$ denotes the cardinality of S.
2. Prove by induction that for every $S \in V$, the set $\bigcup_{T \subsetneq S} T$ is finite. Why does your proof fail when S ranges over the elements in $V' = V \cup \{\bigcup_{U \subsetneq \mathbb{N}} U\}$ instead of V?