Prof. Dr. A. Poetzsch-Heffter
M.Sc. Peter Zeller
Dipl.-Inf. C. Feller

University of Kaiserslautern
Department of Computer Science
Software Technology Group

Exercise Sheet 8: Specification and Verification with Higher-Order Logic (Summer Term 2014)

Exercise 1 Small Step Semantics

In this exercise we return to our small While-language based on the language IMP introduced in the lecture. In particular, we want to create and show properties about an adequate small step semantics for IMP. Download the file `Sheet8_smallstep.thy` from our webpage.

a) Define a small step semantics for IMP within Isabelle/HOL. You should come up with a definition that describes what it means to execute one step as well as a definition that describes the execution of k steps.

Hint: Remember that there are two different kinds of configurations: terminal and non-terminal ones. The theory already contains a datatype `config` for such configurations, which also has an additional nice syntax defined.

b) Prove that terminal configurations are stuck w.r.t the semantics, i.e.

$$\neg \Big(\langle s \rangle \ \rightarrow_1 \ y \Big)$$

c) Show that if you execute at least one step of the program `IF b THEN c ELSE c` and reach a state `y` then there exists a number of steps that you can execute the program `c` and reach the same state.

d) Show the following lemma from the lecture:

$$(\langle c1; c2, s\rangle \rightarrow_n \langle t\rangle) \longrightarrow (\exists m1\ m2\ r.\ n = m1 + m2 \wedge \langle c1, s\rangle \rightarrow_{m1} \langle r\rangle \wedge \langle c2, r\rangle \rightarrow_{m2} \langle t\rangle)$$

e) *(optional)* Consider the program `WHILE 0 <= 0 DO x ::= x + 1`. Show that for every number i you can always find a number k so that the value of `x` is increased by i after executing the program for k steps.