apl. Prof. Dr. Thomas Noll

Kevin van der Pol, Hao Wu

Exercise 1 (AM for While): (1+1+1+1 Points)

a) Write a program in the WHILE programming language computing the greatest common divisor of two positive ,numbers given by variables x and y.

b) Translate the program into intermediate code.

c) Give a run of the program, i.e. the sequence of program states for $x = 4$, $y = 2$.

d) Show that it actually does compute the greatest common divisor using formal semantics.

Exercise 2 (AM: Repeat Until): (1+1 Points)

a) Extend the WHILE language of the lecture with the construct **repeat** c **until** b and specify the corresponding translation function.

b) When modeling the **repeat** c **until** b via the similar construct c; **while** $\neg b$ **do** c; containing a while-loop in a straightforward way, the body c will be translated twice. Can you think of a way to avoid this double translation of c?

Exercise 3 (Compiler Correctness): (5 Points)

Provide the second/missing proof step for theorem of lecture 17, i.e. show that the following lemma holds:
Lemma 1 *For every* $c \in Cmd$, $\sigma, \sigma' \in \Sigma$ *and* $e \in Stk$, $\langle \mathfrak{T}_c[\![c]\!], \epsilon, \sigma \rangle \rhd^* \langle \epsilon, e, \sigma' \rangle$ *implies* $\langle c, \sigma \rangle \to \sigma'$ *and* $e = \epsilon$.

You may use all theorems and lemmata presented in the lecture (except from the one to proof). Additionally you may find the following lemma useful.

Lemma 2 (Decomposition Lemma) *If* $\langle c_1 : c_2, e, s \rangle \rhd^k \langle \epsilon, e'', \sigma'' \rangle$, *then there exists a configuration* $\langle \epsilon, e', \sigma' \rangle$ *and natural numbers* k_1, k_2 *with* $k_1 + k_2 = k$ *such that* $\langle c_1, e, \sigma \rangle \rhd^{k_1} \langle \epsilon, e', \sigma' \rangle$ *and* $\langle c_2, e', \sigma' \rangle \rhd^{k_2} \langle \epsilon, e'', \sigma'' \rangle$.