Technische Universität Wien
Compilers and Languages
Prof. J. Knoop

A-1040 Wien
Argentinierstr. 8
Tel.: 01-58801-18510

"Analysis and Verification (185.276, VU 2.0, ECTS 3.0)" SS 2019

Assignment 2 13 March 2019

Exercise 1 : (4 Points)

Prove by structural induction (on the inductive structure of arithmetic expressions) the Substitution Lemma 1.7.3 of Chapter 1.7:

Lemma 1.7.3 (*Substitution Lemma for* $[\![\ ]\!]_A$)

$$\forall\, a, a' \in \mathbf{Aexpr}.\ \forall\, \sigma \in \Sigma.\ [\![\, a[a'/x]\,]\!]_A(\sigma) = [\![\, a\,]\!]_A(\sigma[[\![\, a'\,]\!]_A(\sigma)/x])$$

Exercise 2 : (4+4 Points)

Let $\sigma \in \Sigma$ be a state with $\sigma(x) = \mathbf{13}$ and $\sigma(y) = \mathbf{5}$. Prove using the

1. structurally operational
2. natural

semantics of WHILE that the program

$$z := 0;\ \mathbf{while}\ y \leq x\ \mathbf{do}\ z := z + 1;\ x := x - y\ \mathbf{od}$$

applied to σ terminates regularly in state σ' with $\sigma' = \sigma[\mathbf{2}/z][\mathbf{5}/y][\mathbf{3}/x]$.

Exercise 3 : (4 Points)

Let $\pi_1, \pi_2 \in \mathbf{Prg}$ be two WHILE-programs, and let $\sigma, \sigma' \in \Sigma$ be two states.
Investigate the validity of the below implication (proof or counterexample):

$$\langle \pi_1; \pi_2, \sigma\rangle \ggg^* \langle \pi_2, \sigma'\rangle\ \Rightarrow\ \exists\, k \in \mathbb{N}_0.\ \langle \pi_1, \sigma\rangle \ggg^k \sigma'$$

Exercise 4 : (4+4 Points)

Extend the programming language WHILE by the loop statement

`repeat` π `until b end`

Provide a

1. SOS rule [rep_{sos}]
2. NS rule [rep_{ns}]

such that the statement gets its "usual" semantics without relying on or exploiting the existence of the while statement in WHILE for defining the two rules.

Submission: Wednesday, 20 March 2019, before the lecture.