Prof. Dr. Andreas Podelski
Matthias Heizmann

1.2.2012
Submission: 7.2.2012
at the beginning of the lecture

Tutorials for Program Verification
Exercise sheet 13

Let ϕ be a formula over V, let ρ be a formula over V and V'. In Exercise 4 of Exercise Sheet 7 you defined a pre-condition function $wp(\phi, \rho)$. The following solution was correct.

$$wp(\phi, \rho) = \forall V'.\rho \to \phi[V'/V]$$

Exercise 1: Connection Between *wp* and *post* 2 points

Prove that the following equivalence holds.

$$post(\psi, \rho) \subseteq \phi \text{ iff } \psi \subseteq wp(\phi, \rho)$$

Exercise 2: Properties of *wp* 5 points

Prove or refute the following propositions

(a) The pre-condition function $wp(\phi, \rho)$ distributes over conjunction in the first argument, i.e., $wp(\psi_1 \wedge \psi_2, \rho) = wp(\psi_1, \rho) \wedge wp(\psi_2, \rho)$

(b) The pre-condition function $wp(\phi, \rho)$ distributes over disjuction in the first argument, i.e., $wp(\psi_1 \vee \psi_2, \rho) = wp(\psi_1, \rho) \vee wp(\psi_2, \rho)$

(c) The pre-condition function $wp(\phi, \rho)$ distributes over conjunction in the second argument, i.e., $wp(\psi, \rho_1 \wedge \rho_2) = wp(\psi, \rho_1) \wedge wp(\psi, \rho_2)$

(d) The pre-condition function $wp(\phi, \rho)$ distributes over disjuction in the second argument, i.e., $wp(\psi, \rho_1 \vee \rho_2) = wp(\psi, \rho_1) \vee wp(\psi, \rho_2)$

(e) Relational composition and iterative application of the pre-condition function wp coincide, i.e., $wp(\psi, \rho_1 \circ \rho_2) = wp(wp(\psi, \rho_2), \rho_1))$

Exercise 3: Quantifier Eliminiation 2 points

In Exercise 4 of Exercise Sheet 7 you showed that if t is a term that does not contain the variable x the formula $\exists x = t \wedge \phi$ is equivalent to the formula $\phi[t/x]$. You used this equivalence to eliminate existential quantifiers.

- State an analogous equivalence that allows you to eliminate an universal quantifier in some formulas.
- Use this equivalence to eliminate the universal quantifier in the formula

$$wp(y = 23, y' = x).$$

Exercise 4: Backward Reachability Analysis 3 points

In the reachability analysis (presented in the lecture on Tue 29.11.2011) we used *post* to define the set states which are reachable from φ_{init} to check if a program is safe.

State an analogous analysis where you check if a program is safe by using *wp* to define the set of states from which φ_{err} is reachable. Use properties proven in the preceeding exercises to explain that your analysis is correct.