Prof. Dr. Andreas Podelski
Dr. Matthias Heizmann
Christian Schilling

Preference deadline: June 11, 2017
Discussion: June 13, 2017

Tutorial for Program Verification
Exercise Sheet 6

Exercise 1: Loop invariants 1 Point

Consider the following program P.

$$
\begin{array}{l}
\{\mathit{true}\} \\
x := i; \\
y := j; \\
\textbf{while } x \neq 0 \textbf{ do } \{\theta\}\ \{ \\
\quad x := x - 1 \\
\quad y := y - 1 \\
\} \\
\{i = j \rightarrow y = 0\}
\end{array}
$$

(a) Find a suitable loop invariant θ such that $\mathit{true} \models \mathit{wp}(P, i = j \rightarrow y = 0)$ holds.

(b) Give two examples for a loop invariant θ such that $\mathit{true} \models \mathit{wp}(P, i = j \rightarrow y = 0)$ does not hold.

Exercise 2: Guarded commands 2 Points

Consider the following modified version of `Fact` where we added the variable u.

$$
\begin{array}{l}
\{n \geq 0\} \\
u := 1; \\
f := 1; \\
i := 1; \\
\textbf{while } i \leq n \textbf{ do } \{\theta\}\ \{ \\
\quad f := f \cdot i; \\
\quad i := i + 1; \\
\quad u := u + 1; \\
\} \\
\{f = \mathit{fact}(n) \wedge u \geq 1\}
\end{array}
$$

(a) Transform the program (together with its pre-/postcondition) to a guarded command. Use the old θ from the previous exercise sheet:

$$f = \mathit{fact}(i - 1) \wedge 1 \leq i \wedge i \leq n + 1$$

(b) Why will a correctness proof using wp of your guarded command fail?

(c) Modify θ above such that it can be used to show correctness of this program.

Exercise 3: Properties of post 2 Points

We say that *post* distributes over the connective $\odot$ w.r.t. the first argument if the following equation holds.

$$post(\phi_1 \odot \phi_2, \rho) = post(\phi_1, \rho) \odot post(\phi_2, \rho)$$

We say that *post* distributes over the connective $\odot$ w.r.t. the second argument if the following equation holds.

$$post(\phi, \rho_1 \odot \rho_2) = post(\phi, \rho_1) \odot post(\phi, \rho_2)$$

- Determine for $\odot \in \{\wedge, \vee, \rightarrow\}$ if *post* distributes over $\odot$ w.r.t. the first argument or w.r.t. the second argument.
- Determine if the equality $post(\neg\phi, \rho) = \neg post(\phi, \rho)$ holds.

  Determine if the equality $post(\phi, \neg\rho) = \neg post(\phi, \rho)$ holds.

Give a proof for each positive answer, give a counterexample for each negative answer.

Exercise 4: Program representations 1 Point

Consider again the program from Exercise 1 where we encode the postcondition using an **assert** statement and omit the precondition and the loop invariant.

$$\begin{array}{ll}
\ell_0: & x := i; \\
\ell_1: & y := j; \\
\ell_2: & \textbf{while } x \neq 0 \textbf{ do } \{ \\
\ell_3: & \quad\quad x := x - 1 \\
\ell_4: & \quad\quad y := y - 1 \\
\ell_5: & \} \\
\ell_6: & \textbf{assert}(i = j \rightarrow y = 0)
\end{array}$$

(a) State a formal definition of this program in the notation that was introduced in the lecture on Wednesday, May 30, where a program is given as a tuple

$$P = (V, pc, \varphi_{init}, R, \varphi_{err}).$$

(b) Draw the corresponding control flow graph.

Exercise 5: Weakest precondition 2 Points

Let V be a tuple of program variables. Let ϕ be a set of states (i.e., ϕ is a formula whose free variables are in V). Let ρ be a binary relation over program states (i.e., ρ is a formula whose free variables are in $V \cup V'$).

In the lecture we defined the formula $post(\phi, \rho)$ as the image of the set ϕ under the relation ρ.

(a) Define a function *wp* such that the formula $wp(\phi, \rho)$ denotes the largest set of states ψ such that $post(\psi, \rho)$ is a subset of ϕ.

(b) Compute $wp(\phi_i, \rho_i)$ for the following pairs.

$$\begin{array}{ll}
\phi_1 \equiv y \geq 7 & \rho_1 \equiv x < y \wedge x' = x \wedge y' = y \\
\phi_2 \equiv y \geq 7 & \rho_2 \equiv x' = x + y + 3 \wedge y' = y \\
\phi_3 \equiv y \geq 7 \wedge x = 23 & \rho_3 \equiv y' = y
\end{array}$$