Dr. Matthias Heizmann
Tanja Schindler
Dominik Klumpp

Hand in until 10:00 on July 01, 2019
Discussion: July 01, 2019

Tutorial for Program Verification
Exercise Sheet 16

Exercise 1: Strongest Postcondition for the Conditional Statement 2 Points

In the lecture we have seen that the strongest postcondition of a sequential composition or a `while`-loop can be expressed in terms of the strongest postconditions of the sub-statements. In this exercise, you should give a similar formulation for the strongest postcondition of an conditional statement.

Specifically, let st be the statement `if (expr)` $\{\, st_1 \,\}$ `else` $\{\, st_2 \,\}$. Let $S \subseteq S_{V,\mu}$ be a set of states. Express the strongest postcondition $sp(S, st)$ using only the strongest postcondition of st_1, st_2 and of simple statements (`havoc`, assignments or `assume`) for suitable sets of states.

You do not have to prove the correctness of your result.

Exercise 2: Distributivity of sp 4 Points

In this exercise we examine distributivity properties of the strongest postcondition. Let S, S_1, S_2 be arbitrary sets of states, and let st be a statement. Furthermore, let φ_1 and φ_2 be formulas.

For each of the following equalities, either prove its correctness or give a counterexample.

(a) $sp(S_1 \cup S_2, st) = sp(S_1, st) \cup sp(S_2, st)$

(b) $sp(S_1 \cap S_2, st) = sp(S_1, st) \cap sp(S_2, st)$

(c) $sp(S, \texttt{assume } \varphi_1 \vee \varphi_2) = sp(S, \texttt{assume } \varphi_1) \cup sp(S, \texttt{assume } \varphi_2)$

(d) $sp(S, \texttt{assume } \varphi_1 \wedge \varphi_2) = sp(S, \texttt{assume } \varphi_1) \cap sp(S, \texttt{assume } \varphi_2)$

Exercise 3: Strongest Postcondition 2 Points

Consider the following program P.

```
assume x > y;
x := x - y;
havoc z;
assume z > 0;
x := x * z;
```

Compute the strongest postcondition $sp(S, P)$ where S is $\{y > 0\}$.

Exercise 4: Weakest Precondition 3 Points

Analogously to the strongest postcondition we define the weakest precondition for a given set of states and a given statement st as follows.

$$\text{wp}(S, st) = \{s \in S_{V,\mu} \mid \text{ forall } s' \in S_{V,\mu} \ (s, s') \in [\![st]\!] \text{ implies } s' \in S\}$$

Intuitively, the weakest precondition is the set of states such that if we can execute st and st terminates then we are is some state of S.

Let us assume that the set S is given by a formula ψ, i.e., $S = \{\psi\}$. Give a formula φ such that $\text{wp}(S, st) = \{\varphi\}$ for the cases where

(a) st is an assignment statement of the form `x := expr`,

(b) st is an assume statement of the form `assume expr`, and

(c) st is a havoc statement of the form `havoc x`.