In this exercise sheet, we work with the strongest postcondition \( sp \) and its dual, the weakest precondition \( wp \). These functions are also known as predicate transformers.

Submit your solution by uploading it as PDF in ILIAS.

**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 \texttt{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 \texttt{if (expr) \{ st}_1 \} \texttt{else \{ st}_2 \}. Let \( S \subseteq S_{V^p} \) 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 (\texttt{havoc}, assignments or \texttt{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 \( \varphi_1 \) and \( \varphi_2 \) be formulas.

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

\begin{enumerate}
  \item \( sp(S_1 \cup S_2, st) = sp(S_1, st) \cup sp(S_2, st) \)
  \item \( sp(S_1 \cap S_2, st) = sp(S_1, st) \cap sp(S_2, st) \)
  \item \( sp(S, \texttt{assume } \varphi_1 \lor \varphi_2) = sp(S, \texttt{assume } \varphi_1) \cup sp(S, \texttt{assume } \varphi_2) \)
  \item \( sp(S, \texttt{assume } \varphi_1 \land \varphi_2) = sp(S, \texttt{assume } \varphi_1) \cap sp(S, \texttt{assume } \varphi_2) \)
\end{enumerate}

**Exercise 3: Strongest Postcondition**  
2 Points  
Consider the following program \( P \).

```
1 assume x > y;
2 x := x - y;
3 havoc z;
4 assume z > 0;
5 x := x * z;
```

Compute the strongest postcondition \( sp(S, P) \) where \( S \) is \( \{ y > 0 \} \).
Exercise 4: Weakest Precondition

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

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

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

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

(a) \( st \) is an assignment statement of the form \( x := \text{expr} \),
(b) \( st \) is an assume statement of the form \( \text{assume expr} \), and
(c) \( st \) is a havoc statement of the form \( \text{havoc } x \).