Tutorial for Program Verification

Exercise Sheet 15

In this exercise sheet we work with control flow graphs, executions and reachability graphs.
Submit your solution by uploading it as PDF in ILIAS.

Exercise 1: From Programs to CFGs

For each of the programs given below, draw a control-flow graph.

(a) Code of program $P_{\text{pow}}$:

\begin{verbatim}
1 e := 1;
2 z := 0;
3 while (z < y) {
4     e := e * x;
5     z := z + 1;
6 }
\end{verbatim}

(b) Code of program $P_{\text{findmin}}$:

\begin{verbatim}
1 i := lo;
2 min := a[lo, lo];
3 while (i <= hi) {
4     j := lo;
5     while (j <= hi) {
6         if (a[i, j] < min) {
7             min := a[i, j];
8         }
9         j := j + 1;
10     }
11     i := i + 1;
12 }
\end{verbatim}
Exercise 2: Program Configurations

Consider the program $P = (V, \mu, T)$ with $V = \{x, y\}$, $\mu(x) = \mu(y) = \{\text{true}, \text{false}\}$ and $T$ a derivation tree for the statement below on the left. On the right, a CFG for $P$ is shown.

```plaintext
while (x == y) {
    y := x;
    havoc x;
}
```

Draw the reachability graph for this control-flow graph and the precondition-postcondition-pair $(x, x \rightarrow \neg y)$.

Exercise 3: Existence of Program Executions

Recall the following lemma from the lecture slides:

**Lemma** (RelAndExec) Let $st$ be a statement, and let $G = (\text{Loc}, \Delta, \ell_{\text{init}}, \ell_{\text{ex}})$ be a control-flow graph for $st$. Then there exists a program execution $(\ell_0, s_0), \ldots, (\ell_n, s_n)$ with $\ell_0 = \ell_{\text{init}}$ and $\ell_n = \ell_{\text{ex}}$, iff $(s_0, s_n) \in [st]$.

In order to prove this result, we formulated several helper lemmas. For this exercise, prove the following:

**Lemma** (RelAndExec.2) Let $st_1, st_2$ be statements. Assume that for $st_1$ as well as for $st_2$, the lemma RelAndExec holds.

Let now $G = (\text{Loc}, \Delta, \ell_{\text{init}}, \ell_{\text{ex}})$ be a control-flow graph for the sequential composition $st_1st_2$. Then there exists a program execution $(\ell_0, s_0), \ldots, (\ell_n, s_n)$ with $\ell_0 = \ell_{\text{init}}$ and $\ell_n = \ell_{\text{ex}}$, iff $(s_0, s_n) \in [st_1st_2]$.