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

Preference deadline: May 28, 2017
Discussion: May 30, 2017

Tutorial for Program Verification
Exercise Sheet 5

Exercise 1: Weakest precondition for sequential composition 2 Points

In the lecture we discussed that the weakest precondition of the sequential composition is independent of the way we add parentheses, i.e.,

$$\mathbf{wp}((\mathtt{C}_1 \,;\, \mathtt{C}_2) \,;\, \mathtt{C}_3, \phi) \;\equiv\; \mathbf{wp}(\mathtt{C}_1 \,;\, (\mathtt{C}_2 \,;\, \mathtt{C}_3), \phi)$$

Use the following program and postcondition to exemplarily show this fact, i.e., compute **wp** for both interpretations step by step and compare the results.

$$\begin{aligned} &\mathtt{C}_1 : \textbf{if } x > 0 \textbf{ then } x := 1 \textbf{ else } x := 2 \\ &\mathtt{C}_2 : y := 1 \qquad\qquad\qquad\qquad \phi : x = 3 \\ &\mathtt{C}_3 : x := x + y \end{aligned}$$

Exercise 2: Recursive equation for loop invariants 2 Points

In this exercise we derive a recursive equation for the loop invariant of a while loop. This equation might be useful to guess inductive loop invariants.

Consider the following equivalence of commands.

$$\textbf{while } \mathtt{b} \textbf{ do } \mathtt{C}_0 \qquad \equiv \qquad \textbf{if } \mathtt{b} \textbf{ then } \mathtt{C}_0 \,;\, \textbf{while } \mathtt{b} \textbf{ do } \mathtt{C}_0 \textbf{ else skip}$$

(a) Use the operational semantics of commands ("$\rightsquigarrow$") to show that the preceding equivalence holds, i.e., show that the following equation is valid.

$$[\![\textbf{while } \mathtt{b} \textbf{ do } \mathtt{C}_0]\!] \quad = \quad [\![\textbf{if } \mathtt{b} \textbf{ then } \mathtt{C}_0 \,;\, \textbf{while } \mathtt{b} \textbf{ do } \mathtt{C}_0 \textbf{ else skip}]\!]$$

(b) Use the weakest precondition $\mathbf{wp}(\cdot,\cdot)$ to state a recursive equation for a loop invariant θ of a while loop $\textbf{while } \mathtt{b} \textbf{ do } \mathtt{C}_0$.

Hint: Start computing **wp** for both sides. Finally, the right-hand side of the equation should be a first-order logic formula that contains $\mathtt{b}$, θ, and $\mathbf{wp}(\mathtt{C}_0, \phi)$ for some suitable first-order logic formula ϕ.

Exercise 3: Hoare logic derivation – Multiplication 2 Points

(a) Write down a partial correctness specification (i.e., precondition and postcondition) for a program `C` that multiplies two integers m and n, where m is nonnegative, and stores the result in r.

(b) Write down a program `C` as specified above that only uses addition (but not multiplication). Use the command language introduced in the lecture.

Hint: Using an auxiliary variable may be helpful for the next part of the exercise.

(c) Annotate the while loop of your program with a suitable loop invariant and construct a Hoare logic derivation that proves that your program `C` fulfills your correctness specification.

Exercise 4: Hoare logic derivation – Factorial function 2 Points

Consider the annotated program Fact that was presented in the lecture.

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

Recall that $\mathit{fact}(n)$ denotes the factorial function of n.

In Figure 1 you find a derivation of the given partial correctness specification in the Hoare calculus and the following loop invariant.

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

Collect all side conditions from the strengthening/weakening rule applications (marked with "s/w") and show that they are valid (you can skip trivial proofs). Note that one of the proofs requires a case split.

$$\cfrac{\cfrac{\cfrac{\cfrac{}{\{1=1\wedge n\geq 0\}\, f:=1\,\{f=1\wedge n\geq 0\}}\,\text{asgn}}{\{n\geq 0\}\, f:=1\,\{f=1\wedge n\geq 0\}}\,\text{s/w} \qquad \cfrac{\cfrac{}{\{f=1\wedge 1=1\wedge n\geq 0\}\, i:=1\,\{f=1\wedge i=1\wedge n\geq 0\}}\,\text{asgn}}{\{f=1\wedge n\geq 0\}\, i:=1\,\{f=1\wedge i=1\wedge n\geq 0\}}\,\text{s/w}}{\{n\geq 0\}\, f:=1\ ;\ i:=1\,\{f=1\wedge i=1\wedge n\geq 0\}}\,\text{seq} \qquad (1)}{\{n\geq 0\}\,\texttt{Fact}\,\{f=\mathit{fact}(n)\}}\,\text{seq}$$

Proof tree for (1):

$$\cfrac{\cfrac{\cfrac{(2) \qquad \cfrac{\cfrac{}{\{f=\mathit{fact}(i+1-1)\wedge 1\leq i+1\wedge i+1\leq n+1\}\, i:=i+1\,\{\theta\}}\,\text{asgn}}{\{f=\mathit{fact}(i)\wedge 1\leq i\wedge i\leq n\}\, i:=i+1\,\{\theta\}}\,\text{s/w}}{\{\theta\wedge i\leq n\}\, f:=f\cdot i\ ;\ i:=i+1;\ \{\theta\}}\,\text{seq}}{\{\theta\}\,\textbf{while}\ i\leq n\ \textbf{do}\ \{\theta\}\ \{f:=f\cdot i\ ;\ i:=i+1\}\,\{\theta\wedge\neg(i\leq n)\}}\,\text{whl}}{\{f=1\wedge i=1\wedge n\geq 0\}\,\textbf{while}\ i\leq n\ \textbf{do}\ \{\theta\}\ \{f:=f\cdot i\ ;\ i:=i+1\}\,\{f=\mathit{fact}(n)\}}\,\text{s/w}$$

Proof tree for (2):

$$\cfrac{\cfrac{}{\{f\cdot i=\mathit{fact}(i-1)\cdot i\wedge 1\leq i\wedge i\leq n\}\, f:=f\cdot i\,\{f=\mathit{fact}(i-1)\cdot i\wedge 1\leq i\wedge i\leq n\}}\,\text{asgn}}{\{\theta\wedge i\leq n\}\, f:=f\cdot i\,\{f=\mathit{fact}(i)\wedge 1\leq i\wedge i\leq n\}}\,\text{s/w}$$

Figure 1: Hoare derivation for `Fact` function and $\theta\equiv f=\mathit{fact}(i-1)\wedge 1\leq i\wedge i\leq n+1$. Due to space constraints the proof tree is split into three subtrees and we have not substituted θ. On the web page you can find a full picture of the proof tree.