# Weakest Precondition and Verification Conditions Generation

José Proença & David Pereira & Eduardo Tovar
{pro,drp,emt}@isep.ipp.pt

Formal Verification of Critical Applications – 2021/2022

## To do

Practice the calculation of weakest preconditions and of verification conditions for simple imperative programs.

## What to submit

There is nothing to submit. This is just a set of exercises for the students to practice and receive feedback during classes.

## Evaluating Hoare Triples

**Ex-1)** For each of the triples presented below, calculate the corresponding weakest precondition and show if the previously defined precondition implies the weakest one that you have produced. Moreover, for the cases of the triples involving while loops, please provide the adequate loop invariant.

1. $\{i = 5\}\, a := i + 2\, \{(a = 7) \wedge (i = 5)\}$
2. $\{i = 5\}\, a := i + 2\, \{(a = 7) \wedge (i > 0)\}$
3. $\{(i = 5) \wedge (a = 3)\}\, a := i + 2;\, \{a = 7\}$
4. $\{a = 7\}\, i := i + 2;\, \{a = 7\}$
5. $\{i = a - 1\}\, i := i + 2;\, \{i = a + 1\}$
6. $\{True\}\, a := i + 2;\, \{a = i + 2\}$
7. $\{a > b\}\, m := 1; n := a - b;\, \{m \times n > 0\}$
8. $\{s = 2^i\}\, i := i + 1;\, s := s * 2\, \{s = 2^i\}$
9. $\{True\}$ **if** $(i < j)$ **then** $min := i$ **else** $min := j \{(min \leq i) \wedge (min \leq j)\}$
10. $\{(i > 0) \wedge (j > 0)\}$ **if** $(i < j)$ **then** $min := i$ **else** $min := j\, \{min > 0\}$
11. $\{s = 2^i\}$ **while** $i < n\, \{?\}$ **do** $i := i + 1; s := s * 2\, \{s = 2^i\}$
12. $\{(s = 2^i) \wedge (i \leq n)\}$ **while** $i < n\, \{?\}$ **do** $i := i + 1; s := s * 2\, \{s = 2^n\}$

**Ex-2)** For each of the Hoare triples presented in the previous exercise, apply the two VC generation algorithms introduced in the classes.

## Solutions to selected exercises

### Weakest Precondition Generation (Ex-1)

**Exercise 1)** $\{i = 5\}\, a := i + 2;\, \{(a = 7) \wedge (i = 5)\}$

For calculating the weakest precondition we will use the **wprec** function. It is straightforward to compute it:

$$\begin{aligned} \textbf{wprec}(a := i + 2, (a = 7) \wedge (i = 5)) = \\ ((a = 7) \wedge (i = 5))[a \mapsto i + 2] = \\ (i + 2 = 7) \wedge (i = 5) \end{aligned}$$

Next, we show that the defined precondition $i = 5$ logically implies the weakest precondition generated; that is: $i = 5 \rightarrow (i + 2 = 7) \wedge (i = 5)$, which is easy to prove that it is valid.

**Digression on Hoare Logic rules:** We start by recalling the first Hoare logic rule for assignments:

$$\frac{}{\{Q[x \mapsto E]\}\, x := e\, \{Q\}}$$

This problem with this rule is the fact that it is too rigid: it cannot be applied directly to the post condition $(a = 7) \wedge (i = 5)$, since $((a = 7) \wedge (i = 5))[a \mapsto i+2]$ reduces to $((i+2 = 7) \wedge (i = 5))$ which does not match $i = 5$, that is, the defined precondition. For solving this issue, the solution consists in first applying the consequence rule

$$\frac{\{P'\}\, C\, \{Q'\}}{\{P\}\, C\, \{Q\}}\text{, if } P \rightarrow P' \text{ and } Q' \rightarrow Q$$

We can use this rule to match the $Q[x \mapsto E]$ that we need so that we can apply the assignment rule. The proof tree presented below shows exactly how that is done.

$$\begin{array}{l}\text{(Assignment)}\ \dfrac{}{\{(i + 2 = 7) \wedge (i = 5)\}\, a := i + 2\, \{(a = 7) \wedge (i = 5)\}} \\ \text{(Consequence)}\ \dfrac{\qquad\qquad}{\{i = 5\}\, a := i + 2\, \{(a = 7) \wedge (i = 5)\}}\end{array} \text{ if } i = 5 \rightarrow (i + 2 = 7) \wedge (i = 5)$$

In the proof tree above, the side condition $i = 5 \rightarrow (i + 2 = 7) \wedge (i = 5)$ is captured by the formula scheme $P \rightarrow P'$ of the consequence rule. Also, we know already that this implication is valid.
Recall also that we have introduced an updated version of Hoare logic's assignment axiom, that is more flexible by not requiring the assertion $Q[x \mapsto E]$ as precondition, by generating a side condition similar to the one generated when we applied the consequence rule. The rule is

$$\frac{}{\{P\}\, x := e\, \{Q\}}\text{, if } P \rightarrow Q[x \mapsto E]$$

Applying this rule goes as follows:

$$\text{(Assignment)}\ \frac{}{\{i = 5\}\, a := i + 2;\, \{(a = 7) \wedge (i = 5)\}} \text{ if } i = 5 \rightarrow (i + 2 = 7) \wedge (i = 5)$$

**Exercise 9)** $\{True\}$ **if** $i < j$ **then** $min := i$ **else** $min := j\{(min \leq i) \wedge (min \leq j)\}$

We preceed as we did in the previous explanation:

$$\textbf{wprec}(\textbf{if}\, i < j\, \textbf{then}\, min := i\, \textbf{else}\, min := j, (min \leq i) \wedge (min \leq j)) =$$
$$(i < j \rightarrow \textbf{wprec}(min := i, (min \leq i) \wedge (min \leq j)) \wedge (\neg(i < j) \rightarrow \textbf{wprec}(min := j, (min \leq i) \wedge (min \leq j)) =$$
$$(i < j \rightarrow ((min \leq i) \wedge (min \leq j))[min \mapsto i] \wedge (\neg(i < j) \rightarrow ((min \leq i) \wedge (min \leq j))[min \mapsto j] =$$
$$(i < j \rightarrow ((i \leq i) \wedge i \leq j)) \wedge (\neg(i < j) \rightarrow ((j \leq i) \wedge (j \leq j))).$$

We now inspect both of them to check if they are correct. Note that here we can ignore the precondition since $True$ does not alter the results of these two parts of the weakest precondition generated. For the left part of the generated formular, we know from the hypothesis that $i < j$, meaning that $i \leq j$ also holds ($i \leq i$ holds trivially). For the right side, we know that $\neg(i < j)$ is the same as $j \leq i$ and hence the conclusion is also correct (note that $j \leq j$ also holds trivially).