SE 504 (Formal Methods and Models)
Spring 2025
HW #2: Skip, Assignment, and Selection
Due: 3:00pm, Friday, Feb. 21

Some of the problems call for you to show proofs that involve steps whose justifications rely upon elementary properties of numbers. You are not expected to supply detailed proofs of such properties. For example, if in your proof you have assumed, say, $k > 4$, then feel free to replace $k \geq 0$ by **true** using the obvious theorem $k > 4 \Rightarrow k \geq 0$ as the justification.

For the first two problems, compute the weakest precondition. Recall that

$$[\text{wp.(x := E).}Q \;\equiv\; Q(x := E) \wedge def.E]$$

but that we ignore the second conjunct when it is obvious that E is well-defined in any state in which the variables mentioned in it have values.

1. $\text{wp.}(x, y := x - 2 * y, y + x).(x > y)$

2. $\text{wp.}(x := x - 2).((x + 3) \cdot (x - 5) \geq 0)$
(Note that $a \cdot b \;\geq\; 0 \;\;\equiv\;\; a = 0 \;\vee\; b = 0 \;\vee\; (a > 0 \;\equiv\; b > 0)$.)

The following batch of problems ask you to prove a given Hoare Triple. Recall the Hoare Triple Laws for the `skip` command and assignment commands are

$$\{\text{P}\}\ \text{skip}\ \{\text{Q}\} \;\equiv\; [P \Rightarrow Q]$$

$$\{P\}\ x := E\ \{Q\} \;\equiv\; [P \;\Rightarrow\; Q(x := E) \wedge \texttt{isDef}.E]$$

Make sure to use the Gries and Schneider format. In most cases, you will probably want to use the *Assume the Antecedant* approach when proving an implication.

3. $\{k > 3 \;\wedge\; r \neq 7\}\ \text{skip}\ \{k > 1 \;\vee\; k < -2\}$.

4. $\{y \leq 2\}\ x, y := y + 1, (2 * y) - 5\ \{x > y\}$

5. $\{z \Rightarrow x\}\ x, y := x \wedge z, x \vee y\ \{x \wedge y \;\equiv\; z\}$

6. $\{P \wedge 0 \leq k < n\}\ k, x := k - 1, x\ \texttt{max}\ f.k\ \{P\}$,
where $P:\;\; x = (\max\, j \mid k < j < n \;:\; f.j)$

Here, `max` is the operator that yields the larger of its two operands. (We write it between its two operands, just like other (binary infix) operators.) Note that this operation lacks an identity element, but it is associative and commutative, so it serves well as a quantifier as long as the quantification's range is not empty. It will help to recall the following variation of the **Split off term** (8.23) rule:

Provided $a < b$ (so that the range is nonempty),

$$(\star i \,|\, a \leq i < b \;:\; P) \;\;=\;\; P(i := a) \;\star\; (\star i \,|\, a + 1 \leq i < b \;:\; P)$$

In each of problems 7-9, calculate an expression E that makes the given Hoare triple valid. Each occurrence of an upper case `C` denotes a *rigid variable* (to use the Gries's and Schneider's terminology), not a program variable. Thus, the expression you give as your final answer for E should not include any occurrences of `C`.

For all these, use the standard technique of showing that $\{P\}\ x := G\ \{Q\}$ is valid by proving $[P \Rightarrow Q(x := G)]$. But in doing that proof, derive an appropriate expression for E. Take advantage of opportunities to make use of the assumption P for the purpose of replacing an expression by another expression assumed to be equal to it.

7. $\{y = x^2\}\ x, y := x - 1, y - E\ \{y = x^2\}$

8. $\{\texttt{C} = m - j\}\ m, j := E, m - j\ \{\texttt{C} = 2m - j\}$

9. $\{\texttt{C} = k \cdot m\ \wedge\ \texttt{isEven}.k\}\ k, m := k\ \texttt{div}\ 2, E\ \{\texttt{C} = k \cdot m\}$

Relevant to the last problem is the theorem $\texttt{isEven}.r\ \equiv\ (r\ \texttt{div}\ 2 = r/2)$. (The `div` operator stands for integer division.)

Recall that if **IF** is the program **if** B **then** S **else** T **fi** , then

$$\{P\}\ \text{IF}\ \{Q\}\quad\equiv\quad\{P \wedge B\}\ S\ \{Q\}\ \wedge\ \{P \wedge \neg B\}\ T\ \{Q\}$$

10. Prove (where p, q, and r are boolean variables)

$\{r\}$ **if** p **then** $r := q$ **else** *skip* **fi** $\{r\quad\equiv\quad p \Rightarrow q\}$

11. Prove
$\{P \wedge 0 \le k < b.length\}$
if $b[k] > 0$ **then** $k, z := k + 1, z + b[k]$
else $k := k + 1$
fi
$\{P\}$

where $P : z = (+i \mid 0 \le i < k\ \wedge\ b[i] > 0 : b[i])$

You may make use of this augmented version of Split-off term (8.23):

Provided $a \le b$ (so that the range is nonempty),

$$(\star i \,|\, a \le i < b + 1 \wedge R\ :\ E)\ =\ (\star i \,|\, a \le i < b \wedge R\ :\ E) \star (\text{if } R(i := b) \text{ then } E(i := b) \text{ else } \iota)$$

where ι is the identity element of $\star$.

The expression "if R then F else G" evaluates to either F or G according to whether R evaluates to true or false, respectively. The equivalent expression in Java has the syntax "R ? F : G".