# Exam 2 Software Testing & Verification
# Th. 29th June 2017, 8:30–10:30, EDUC-ALFA

Lecturer: Wishnu Prasetya

You are allowed to bring along Appendix-A of the LN.

## Part I [4pt $(8 \times 0.5)$]

For each question, choose *one* correct answer.

1. The specification $\{*\ Q\ *\}\ S\ \{*\ \textbf{true}\ *\}$ is known to be valid under **total correctness**. Which of the following conclusions is correct?

   (a) $S$ will terminate when executed in any state.

   (b) If the implication $P \Rightarrow Q$ is valid, then $S$ will terminate when executed in any state satisfying $P$.

   (c) The specification $\{*\ Q\ *\}\ S\ \{*\ Q \Rightarrow R\ *\}$ is also valid under partial correctness.

   (d) The specification $\{*\ P \Rightarrow Q\ *\}\ S\ \{*\ \textbf{true}\ *\}$ is also valid under total correctness.

2. Which of the following statements about weakest pre-condition is correct?

   (a) $\{*\ \textbf{wp}\ S\ Q\ *\}\ \ S\ \ \{*\ Q\ *\}$ is always a valid specification.

   (b) $\{*\ P \Rightarrow (\textbf{wp}\ S\ Q)\ *\}\ \ S\ \ \{*\ Q\ *\}$ is always a valid specification.

   (c) $\{*\ Q\ *\}\ \ S\ \ \{*\ \textbf{wp}\ S\ Q\ *\}$ is always a valid specification.

   (d) $\{*\ P\ *\}\ \ S\ \ \{*\ Q\ *\}$ is valid if and only if the predicate $\textbf{wp}\ S\ (P \Rightarrow Q)$ is valid.

3. What is the **weakest** pre-condition of the following statement with respect to the given post-condition?

   $$\{*\ ?\ *\}\ \ \{\ \textbf{if}\ \mathtt{x{=}y}\ \textbf{then}\ \mathtt{x := x{+}1}\ \textbf{else skip}\ \}\ ;\ \mathtt{x := x{+}y}\quad \{*\ \mathtt{x = y}\ *\}$$

   (a) $((\mathtt{x{=}y}) \wedge \mathtt{x{=}{-}1})\ \vee\ ((\mathtt{x{\neq}y}) \wedge \mathtt{x{=}0})$

   (b) $\mathtt{x{=}{-}1} \wedge \mathtt{y{=}{-}1}$

   (c) $(\mathtt{x{=}y})\ \rightarrow\ \mathtt{x{+}1{+}x{+}y{=}y}\ |\ \mathtt{x{+}y{=}y}$

   (d) $(\mathtt{x{=}y}\ \Rightarrow\ \mathtt{x{+}1{+}y{=}y})\ \vee\ (\mathtt{x{\neq}\ y}\ \Rightarrow\ \mathtt{x{+}y{=}y})$

4. What is the **weakest** pre-condition of the following statement with respect to the given post-condition?

$$\{*\ ?\ *\}\quad \mathtt{a}[\mathtt{k}] := \mathtt{a}[\mathtt{0}]{+}\mathtt{a}[\mathtt{k}] \quad \{*\ \mathtt{a}[\mathtt{0}]{=}\mathtt{a}[\mathtt{k}]\ *\}$$

(a) $\mathtt{a}[\mathtt{0}] \;=\; \mathtt{a}[\mathtt{0}] + \mathtt{a}[\mathtt{k}]$

(b) $\mathtt{a}(\mathtt{0}\ \textbf{repby}\ \mathtt{a}[\mathtt{0}]{+}\mathtt{a}[\mathtt{k}])[\mathtt{0}] \;=\; \mathtt{a}[\mathtt{0}] + \mathtt{a}[\mathtt{k}]$

(c) $\mathtt{a}(\mathtt{k}\ \textbf{repby}\ \mathtt{a}[\mathtt{0}]{+}\mathtt{a}[\mathtt{k}])[\mathtt{0}] \;=\; \mathtt{a}[\mathtt{0}] + \mathtt{a}[\mathtt{k}]$

(d) $\mathtt{a}[\mathtt{0}]\ \textbf{repby}\ \mathtt{a}[\mathtt{0}]{+}\mathtt{a}[\mathtt{k}] \;=\; \mathtt{a}[\mathtt{k}]\ \textbf{repby}\ \mathtt{a}[\mathtt{0}]{+}\mathtt{a}[\mathtt{k}]$

5. Which of the following proofs is correct (according the the proof system of the LN)? Read the steps carefully.

   (a) `PROOF`
   `[A1:]` $(\forall x : P\ x : Q\ x \lor R\ x)$
   `[A2:]` $P\ a$
   `[G:]` $P\ a \land Q\ a$
   `1.` { $\forall$-elimination on A1 using A2 } $Q\ a \lor R\ a$
   `2.` { $\lor$ elimination on 1 } $Q\ a$
   `3.` { conjunction of A2 and 2 } $P\ a \land Q\ a$
   `END`

   (b) `PROOF`
   `[A1:]` $(\exists x : P\ x : Q\ x)$
   `[G:]` $(\forall x :: P\ x \land Q\ x)$
   `1.` { $\exists$-elimination on A1 } [`SOME` x] $P\ x \land Q\ x$
   `2.` { $\forall$-introduction on 1 } $(\forall x :: P\ x \land Q\ x)$
   `END`

   (c) `PROOF`
   `[A1:]` $\neg(\forall x :: P\ x)$
   `[A2:]` $\neg P\ a \Rightarrow Q\ a$
   `[G:]` $Q\ a$
   `1.` { $\forall$-elimination on A1 } $\neg P\ a$
   `2.` { Modus Ponens on A2 using 1 } $Q\ a$
   `END`

   (d) `PROOF`
   `[A1:]` $\neg(\exists x :: P\ x)$
   `[A2:]` $P\ a \lor Q\ a$
   `[G:]` $Q\ a$
   `1.` { rewrite A1 with negate-$\exists$ } $(\forall x :: \neg P\ x)$
   `2.` { $\forall$-elimination on 1 } $\neg P\ a$
   `3.` { rewriting A2 with 2 } $\textbf{false} \lor Q\ a$
   `4.` { simplifying 3 } $Q\ a$
   `END`

6. For integers $m$ and $x$, let $m|x$ mean that $m$ is a divisor of $x$. This means that there exists another integer $k$ such that $x = km$. Consider the following loop; all variables are of type `int`:

$$\{* \ \mathtt{x}{=}\mathtt{1024} \ \wedge \ \mathtt{y}{>}\mathtt{0} \ \wedge \ \mathtt{m}|\mathtt{x} \ \wedge \ \mathtt{m}|\mathtt{y} \ *\}$$

$$\textbf{while } \mathtt{x}{>}\mathtt{y} \textbf{ do } \mathtt{x} := \mathtt{x}{-}\mathtt{y} \ ;$$

$$\{* \ \mathtt{m}|\mathtt{x} \ *\}$$

Which of the predicates below is a consistent invariant of the loop, and enough to prove that the above specification is valid?

(a) $\mathtt{y} \leq \mathtt{x} \leq \mathtt{1024}$

(b) $\mathtt{m}|\mathtt{x} \ \Rightarrow \ \mathtt{m}|\mathtt{y}$

(c) $\mathtt{y}{>}\mathtt{0} \ \wedge \ \mathtt{m}|\mathtt{x}$

(d) $\mathtt{y}{>}\mathtt{0} \ \wedge \ \mathtt{m}|\mathtt{x} \ \wedge \ \mathtt{m}|\mathtt{y}$

7. Consider the following program, with the given specification.

$$\{* \ \mathtt{x}{=}\mathtt{0} \ \wedge \ \mathtt{i}{=}\mathtt{0} \ \wedge \ (\forall \mathtt{k} :: \mathtt{a}[\mathtt{k}] > \mathtt{0}) \ *\}$$

$$\textbf{while } \mathtt{x}{<}\mathtt{100} \textbf{ do}\{ \ \mathtt{x} := \mathtt{x} + \mathtt{a}[\mathtt{i}] \ ; \ \mathtt{i}{:=}\mathtt{i}{+}\mathtt{1} \ \}$$

$$\{* \ \textbf{true} \ *\}$$

Which pair of invariant $I$ and termination metric $m$ is consistent and good enough to prove that the program above terminates?

(a) invariant: $(\forall \mathtt{k} :: \mathtt{a}[\mathtt{k}] > \mathtt{0})$, termination metric: $\mathtt{100} - \mathtt{i}$

(b) invariant: $(\forall \mathtt{k} :: \mathtt{a}[\mathtt{k}] > \mathtt{0})$, termination metric: $\mathtt{100} - \mathtt{x}$

(c) invariant: $\mathtt{0} \leq \mathtt{i} \leq \mathtt{100}$, termination metric: $\mathtt{100} - \mathtt{i}$

(d) invariant: $\mathtt{x} = \mathtt{SUM}(\mathtt{a}[\mathtt{0..i}])$, termination metric: $\mathtt{x} + \mathtt{i}$

8. Consider the function val defined below. Given a list of digits, e.g. as in val $0\ [3,4,5]$, it will calculate the integer value of the digits if they would be the string "345". In this case the answer is the integer 345. Notice that the function is tail recursive.

$$
\begin{array}{lcl}
\mathsf{val}\ v\ [] & = & v \\
\mathsf{val}\ v\ (x:z) & = & \mathsf{val}\ (10*v+x)\ z
\end{array}
$$

Below is an imperative implementation of the function. The specification is given.

```
{* true *}
v := 0 ; t := s ;
while t≠[ ] do {
   v := 10*v + head(t) ;
   t := tail(t)
}

{* v = val 0 s *}
```

Which of the following is a consistent and good enough invariant to prove the correctness of the above specification?

(a) $v = \mathsf{val}\ 0\ t$

(b) $\mathsf{val}\ v\ t\ =\ \mathsf{val}\ v\ s$

(c) $\mathsf{val}\ v\ t\ =\ \mathsf{val}\ 0\ s$

(d) $v = \mathsf{sum}[s_i*10^i \mid 0 \le i < \mathsf{length}(s)]$

## Part II [6pt]

1. [5 pt] **Loop**

   Consider the following program and its specification. The program checks if the array segment $\texttt{a[0..N)}$ is sorted (in increasing order).

   $\{*\ \ 0 < \texttt{N}\ \ *\}$ // pre-condition

   ```
   p := 1 ;
   k := 1 ;
   ok := true ;
   while k≠N ∧ ok do {
      ok := ok ∧ (a[k−1] < a[k]) ;
      k := k + 1 ;
   } ;
   ```

   $\{*\ \ \texttt{ok} = (\forall \texttt{j} : 0{<}\texttt{j}{<}\texttt{N} : \texttt{a[j}-1\texttt{]} < \texttt{a[j]})\ \ *\}$ // post-condition

   Give a **formal proof** that the program satisfies its specification, under *partial correctness*.

   - Please mention what your chosen invariant is.
   - Every step in your proof should include a justification (the hint/comment part).
   - Steps involving quatifiers should be done in small steps: each step should refer to a proof rule or a theorem in Appendix A. You can additionally use this theorem:

     $\vdash\ P \Rightarrow (P = \mathsf{true})$

   - You don't have to show the wp calculation.

2. [ 0.5 pt] **Program call**

   Consider the following specification of the program P; all parameters are of type Float.

   $\{*\ \texttt{c}{\neq}0\ *\}\ \ \texttt{C} := \texttt{c};\ \texttt{P(a, b, OUT c)}\ \ \{*\ \texttt{ac}^2 > \texttt{bC}\ *\}$

   Consider the following statement, that contains a call to P:

   $\{*\ \texttt{c}{=}1\ *\}\ \ \texttt{P(c, c, c)}\ ;\ \texttt{c} := \texttt{c}^2\ \ \{*\ \texttt{c}{>}1\ *\}$

   Consider the following transformation of the statement above; it is equivalent:

   $\{*\ \texttt{c}{=}1\ *\}$

   ```
   {* (1)  ? *}   @a := c ;
                  @b := c ;
                  @c := c ;
   {* (2)  ? *}   P(@a,@b,@c) ;
   {* (3)  ? *}   c := @c ;
                  c := c²
   ```

   $\{*\ \texttt{c} > 1\ *\}$

Calculate intermediate predicates (1) ... (3) above, such that from the corresponding positions they guarantee the final post-condition `c` $>$ `1`. Use the Black Box reduction rule for program call to calculate (2), and standard `wp` calculation for the others.

Just give the answers; you do not have to show the calculation. The specification is valid; you can use this fact as a check in your own calculation.

3. [ 0.5 pt] **Termination proof**
Consider the following program, with the given specification.

{∗ x∈{0, 1} ∧ y=2 ∗}

**while** y<100 **do**{
    **if** odd(x) **then** { x:=x−1 ; y:=0 }
    **else** y:=y+1
    }

{∗ **true** ∗}

Prove that the above program terminates. Use the proof method from the Lecture Notes. You can skip the proof of the exit condition (EC) since the post-condition (**true**) poses no constraint.