Software Engineering

Errata for the Course Slides EN/2018

September 17, 2021

Note: The following ‘repairs’ apply to the slides of 2018 as distributed with the recordings on the ‘EN’ track. Most errors have been fixed with the 2019-slides (except for the one ‘NEW’ one).

Contents

Lecture 4: Software Project Management

Slide 35, ‘Building Blocks Can Be Arbitrarily Complicated’

The first item in the list to the right of the diagram needs to read:

> If a test detected an error in M,

Lecture 9: Scenarios & Use Cases

Slide 46, ‘Language of LSC Body: Example’

See Lecture 10, “Slide 10, ‘Example’ and Slide 5, ‘Language of LSC Body: Example”’ below.

Lecture 10: Live Sequence Charts & RE Wrap-Up

Slide 8, ‘Loop Condition’

The message aspect of the loop condition (first bullet point) needs to read

$$\psi^{\mathsf{Msg}}(q) = \neg \bigvee_{1 \leq i \leq n, \psi \in \mathsf{Msg}(q_i \setminus q)} \psi \wedge \underbrace{\left(\mathit{strict} \implies \bigwedge_{\psi \in \mathcal{E}_{!?}^{\mathcal{I}} \cap \mathsf{Msg}(\mathcal{L})} \neg\psi\right)}_{=:\psi_{\text{strict}}(q)}$$

that is, in the non-strict case, the loop accepts all letters where *none* of the messages of any successor cut is sent or received.

Slide 8, ‘Loop Condition’ (NEW NEW)

In the second bullet point, the last sentence needs to read

- Local invariant $(l_0, \iota_0, \phi, l_1, \iota_1)$ is *active* at cut (!) q
  if and only if $l_0 \preceq l \prec l_1$ for some front location l of cut q.

Short notice note: If the 2021 exam has related tasks, the outcome will not depend on this fix but be the same either way (with the erroneous or the fixed definition).

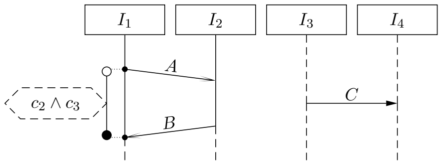
Slide 9, ‘Progress Condition’ (NEW)

The last bullet point needs to read:

- $\psi_\theta^{\mathsf{LocInv},\bullet}(q, q_i) = \bigwedge_{\lambda=(l,\iota,\phi,l',\iota') \in \mathsf{LocInv},\ \Theta(\lambda)=\theta,\ \lambda\ \bullet\text{-active for } (q,q_i)} \phi$

  Local invariant $(l_0, \iota_0, \phi, l_1, \iota_1)$ is *$\bullet$-active* for (q, q_i) if and only if
  - $l_0 \in (q_i \setminus q) \wedge \iota_0 = \bullet$, or
  - $l_0 \in q \wedge l_1 \notin q_i$, or
  - $l_1 \in (q_i \setminus q) \wedge \iota_1 = \bullet$.

The fixed definition in particular treats the following case correctly, where an observed sequence of messages $A_!, C_{!?}, A_?$ needs to consider the local invariant together with $C_{!?}$:



Slide 10, ‘Example’ and Slide 5, ‘Language of LSC Body: Example’

The loop condition of state q_6 needs to read

$$\neg(G_!^{I_2,I_1} \vee G_?^{I_2,I_1})$$

and the progress condition from q_4 to q_6 needs to read

$$F_?^{I_2,I_3} \wedge \neg G_!^{I_2,I_1} \wedge \neg G_?^{I_2,I_1}$$

Slide 40, 'LSC Semantics with Pre-Chart'

Each of the four inner table cells had one '$\wedge$' too much, and the second lines need to read

$$\wedge\, w^1, \dots, w^m \in \mathit{Lang}_{\mathit{fin}}(\mathcal{B}(PC))$$

and

$$\wedge\, w^{k+1}, \dots, w^m \in \mathit{Lang}_{\mathit{fin}}(\mathcal{B}(PC)),$$

respectively.

Meaning: the sub-word consisting of the 1st (or $k+1$-th) up to m-th letter of word w is in the 'finite' language of the pre-chart TBA, i.e., we read the pre-chart TBA as a standard Deterministic Finite Automaton (DFA) with the standard DFA-acceptance criterion of reaching an accepting state with consumption of the last letter of the word.

Lecture 12: Structural Software Modelling II

Slide 28, 'More Interesting Example'

The studied Proto-OCL formula needs to read:

$$\forall\, c \in \mathit{allInstances}_C \bullet x(n(c)) \neq 27$$