

Software Engineering Errata for the Course Slides

July 26, 2018

Contents

Lecture 4: Software Project Management (NEW)	1
Slide 35, 'Building Blocks Can Be Arbitrarily Complicated'	1
Lecture 9: Scenarios & Use Cases	1
Slide 46, 'Language of LSC Body: Example'	1
Lecture 10: Live Sequence Charts & RE Wrap-Up	1
Slide 8, 'Loop Condition'	1
Slide 10, 'Example' and Slide 5, 'Language of LSC Body: Example'	1
Lecture 12: Structural Software Modelling II	2
Slide 28, 'More Interesting Example'	2

Lecture 4: Software Project Management (NEW)

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 and 5 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

$$\psi^{\text{Msg}}(q) = \neg \bigvee_{1 \leq i \leq n, \psi \in \text{Msg}(q_i \setminus q)} \psi \wedge \underbrace{\left(\text{strict} \implies \bigwedge_{\psi \in \mathcal{E}_{i_2}^I \cap \text{Msg}(\mathcal{L})} \neg \psi \right)}_{=:\psi_{\text{strict}}(q)}$$

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 10, 'Example' and Slide 5, 'Language of LSC Body: Example'

The loop condition of state q_6 needs to read

$$\neg(G_1^{I_2, I_1} \vee G_7^{I_2, I_1})$$

and the progress condition from q_4 to q_6 needs to read

$$F_7^{I_2, I_3} \wedge \neg G_1^{I_2, I_1} \wedge \neg G_7^{I_2, I_1}$$

Lecture 12: Structural Software Modelling II

Slide 28, 'More Interesting Example'

The studied Proto-OCL formula needs to read:

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