Errata in *Programming Language Foundations*

Aaron Stump
Computer Science
The University of Iowa

October 11, 2016

Corrections below

Despite many previous readings, attentive readers have found some errors in the book. I correct these below, sometimes underlining changes to the text. Thanks to the following people for reporting bugs: Ryan Brummet, Junyang Chen, Wyatt Kaiser, Tingting Liu, Talal Riaz, John Bodeen.

Chapter 1

1. page 28, second to last case of the proof. The sentence should say "So by the induction hypothesis, we have that $\underline{[\![t_1]\!]\sigma'}$ and $\underline{[\![t_2]\!]\sigma'}$ are both defined and equal to $\underline{[\![t_1]\!]\sigma}$ and $\underline{[\![t_2]\!]\sigma}$, respectively."

2. page 30, Section 1.14.2, problem 2. The theorem cannot be proved without some additional assumption. So the revised problem is:

   2. Let us temporarily define $\sigma \leq \sigma'$ for assignments σ and σ' to mean that for all variables x, if $\sigma(x)$ is defined then so is $\sigma'(x)$ and we have $\sigma(x) \leq \sigma'(x)$. Suppose that t is a term which does not contain the negation or subtraction symbols, <u>and suppose that σ is an assignment where $0 \leq \sigma(x)$ for every variable x.</u> Prove by induction on the structure of t that if $\sigma \leq \sigma'$, then $\underline{0 \leq}[\![t]\!]\sigma \leq [\![t]\!]\sigma'$. (You can use the proof of Theorem 1.11.1 as a guide.)

Chapter 2

1. page 35, top of the page: it says we can "easily define the syntax for all commands except `while`-commands", but it should say "easily define the <u>semantics</u> for all commands except `while`-commands".

2. page 45, proof of Theorem 2.5.4. In the proof, I wrote "$f(c(n)) \sqsubset f(\sqcup c)$". It should say "$f(c(n)) \sqsubseteq f(\sqcup c)$" instead (we are not using the $\sqsubset$ symbol in this chapter).

Chapter 3

1. page 92, start of Section 3.8. It should say "This means that the strongest formula is *False* and the weakest is *True*." (The book has *False* and *True* reversed.)

Chapter 4

1. page 100. The multi-step derivation at the bottom of the page has a bug in the left branch (a proof rule is not being correctly applied). The correct derivation is:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{x := 1, \sigma \rightsquigarrow \sigma[x \mapsto 1]}}{x := 1; y := 2, \sigma \rightsquigarrow y := 2, \sigma[x \mapsto 1]}}{x := 1; y := 2, \sigma \rightsquigarrow^* y := 2, \sigma[x \mapsto 1]} \quad \dfrac{\dfrac{}{y := 2, \sigma[x \mapsto 1] \rightsquigarrow \sigma[x \mapsto 1, y \mapsto 2]}}{y := 2, \sigma[x \mapsto 1] \rightsquigarrow^* \sigma[x \mapsto 1, y \mapsto 2]}}{x := 1; y := 2, \sigma \rightsquigarrow^* \sigma[x \mapsto 1, y \mapsto 2]}$$

2. page 120, part 1 of problem 4.5.1. The final state should be $\{x \mapsto -10, y \mapsto 20, z \mapsto -1\}$, not $\sigma[z \mapsto -1]$.

3. page 120, part 1 of problem 4.5.2. The problem should be asking for a reduction sequence (using the small-step operational semantics), not a single small-step derivation. So the problem can be changed to:

   Write a sequence of reduction steps (with the small-step semantics) which show how to reduce the following command and starting state to the final state $\{x \mapsto 90\}$*. You do not need to give derivations proving your individual small steps.*

   $$\texttt{if}\ x < 100\ \texttt{then}\ x := x * 10\ \texttt{else}\ \texttt{skip},\ \{\, x \mapsto 9\, \}$$

4. page 120, part 2 of problem 4.5.2. The final state should be $\{\, x \mapsto 14,\ y \mapsto 3\}$, not $\{\, x \mapsto 14,\ y \mapsto 1\}$.

5. page 121, part 4 of problem 4.5.2. The problem asks you to find a command c' making a particular small-step judgment is provable, but that judgment mentions c rather than c'. It should mention c'.

Chapter 8

1. page 209, Figure 8.2: we are missing the rule for `skip`, which is:

   $$\dfrac{}{\texttt{skip}, \sigma \rightsquigarrow \sigma}$$

2. page 218, example command using `await`: to get correct behavior, the command should await $y + y' = 0$, rather than $y * y' = 0$. Also, we should initialize z and z' to 1 before the concurrently executing commands begin. So change the definition of $exp_{z,y,n}$ to be

   $$exp_{z,y,n} = (\texttt{while}\ y > 0\ \texttt{do}\ y := y - 1; z := z * n)$$

   and then use this for the command:

   $$\begin{array}{l} y := x; y' := x; z := 1; z' := 1; \\ (\texttt{exp}_{z,y,2} \parallel \texttt{exp}_{z',y',3} \parallel \texttt{await}\ y + y' = 0\ \texttt{then}\ z := z + z') \end{array}$$

3. page 221, Figure 8.8. The third rule in the figure (the one on the right) should have stars on all the arrows:

   $$\dfrac{P \overset{\gamma}{\rightarrow}{}^{*}_{\Delta} P'' \quad P'' \overset{\gamma'}{\rightarrow}{}^{*}_{\Delta} P'}{P \overset{\gamma\gamma'}{\rightarrow}{}^{*}_{\Delta} P'}$$

4. page 223. Several arrows – the ones for multi-step reduction – are missing their stars (as in the issue noted just previously).

5. page 227, part 3 of problem 8.8.2: there is an `od` missing at the end of the displayed statement.

6. page 227, part 1 of problem 8.8.3. The judgment to prove should have $\rightsquigarrow^*$ instead of $\rightsquigarrow$.