

Misprints in Programming Language Design and Implementation

Torben Mogensen

March 12, 2024

This will list significant errors in the text only, but you are welcome to send even minor corrections to `torbenm@di.ku.dk`.

page 109: In line 4, replace “The second `printf` command will print 2” by “The second `printf` command will print 1”. A clearer formulation of the entire paragraph can read

The above uses C-style syntax, but similar code can be written in most block-structured languages. If static scoping is used, the first `printf` command will print 0, since the assignment inside `f` affects the global variable declared in the first line. The second `printf` command will print 1, since only the globally declared `x` (that was updated in `f`) is visible. If dynamic scoping is used, the first `printf` command will print 1, because the declaration of `x` in `main` inside the block just prior to the call to `f` overrides the global declaration, so `f` will update the variable declared in this block. When this block is exited, the global variable is restored, so the second `printf` command will print 2. C, like most ALGOL-derived languages, uses static scoping, so the results would be 0 and 1. Some scripting languages (like Perl and Bash) as well as some extended forms of BASIC that include procedure declarations use dynamic scoping.

page 125: In Fig. 5.11, change the line

```
datatype intIntClosure = G of () | H of (), int)
```

to

```
datatype intIntClosure = G of unit | H of (unit, int)
```

It can be helpful to add the following to the end of Sect. 5.5.1:

`unit` is the type that has only one value, which is `()`.

page 207: Point 3 in the unification algorithm should read:

3. If t is a bound type variable A , find the type u to which A is bound and unify s and u .

page 288, 291, and 292: In Figs. 11.6, 11.8, and 11.9, replace

```
if  $x$  then  $s_1$  else  $s_2$ 
```

by

```
if  $X$  then  $S_1$  else  $S_2$ .
```