Notes/Corrections/Clarifications for lecture given on 2/03/2011 (Thursday)

Faraz Hussain

February 3, 2011

1 "What is that expression after the function definition?"

Question: While explaining free/bound identifiers, we came across this expression from the handout:

```
{fun {$ N} N end N}
```

The first occurrence of N is a declaration, the second is bound to that declaration and the third is free. However, what does the last N mean?

Answer: I couldn't explain that in class but here's what's happening: look at the expression, its something enclosed in braces, so its actually a function call. The question is: what function is it calling? Well, its calling the (anonymous) function that has just been defined!

This is a use of a so-called "nesting marker" (the $ sign here, see pg. 83 of [1, 2.6.1]). This is an Oz idiom; the meaning is that the value of the expression:

```
fun {$ N} N end
```

is just the value of *whatever would be* at the position of the nesting marker – in this case that's the function name!

As an example, say you had: `fun {Id N} N end` (where Id is already declared). Then you could call this using, for example: `{Browse {Id 5}}`.

Similarly, we can call our anonymous function using:

```
{Browse {fun {$ N} N end 50} }
```

This statement defines and calls an anonymous function in a single line! Try it. (Note that the 50 in the end is the argument being supplied to the function.)

Back to the original code, we have an `N` in position of 50, where `N` is free. In order to run that piece of code we need to declare that particular N. The following code does that:

```
%declares outer N used outside the function definition
%also declares X which will store the result of the function call
local N X in
   N = 50
   X = {fun {$ N}  %%Note: this is both defining an anonymous function and calling it!
          N
       end
       N}
   {Browse X} %print the result
end
```

Above, I separated the calling of the anonymous function from the printing of its result, because I think it makes it clearer.

References

[1] Peter Van Roy and Seif Haridi. *Concepts, Techniques, and Models of Computer Programming*. MIT Press, 2004.