Fachbereich Mathematik und Informatik
Prof. Dr. K. Ostermann

Sebastian Erdweg, seba@informatik
Tillmann Rendel, rendel@informatik

November 5, 2009

Programming Languages and Types
Homework Assignment 4

Please hand in your homework by email to `mailto:pllecture@informatik.uni-marburg.de` until October 12. Please submit your solutions in appropriate file formats.

H4.1 Explicit Call-by-reference

Do exercise 14.3.2 from the textbook, that is:
There appears to be a neutral ground between call-by-value and call-by-reference. Consider the following proposed syntax:

```
{with {swap {fun {x}
              {fun {y}
                {with {z x}
                  {seqn {set x y}
                        {set y z}}}}}}}
  {with {a 3}
    {with {b 2}
      {seqn {{swap {ref a}} {ref b}}
            b}}}}
```

The ref notation is an indicator to the interpreter to pass the variables location rather than its value; that is, by using ref a and ref b, the invoker of the procedure indicates his willingness to have his variables be aliased and thus, potentially, be mutated.

1. Modify the interpreter to support the use of ref for procedure arguments.

2. Does this proposal result in a procedural abstraction of the process of swapping the values of two variables? If it does, this would reconcile the design tension between the two invocation techniques: it avoids the difficulty of call-by-value (the inability to write a swap procedure) as well as that of call-by-reference (aliasing of parameters without the callers knowledge). Discuss.

3. Suppose programmers are allowed to apply ref to variables elsewhere in the program. What type should the interpreter use to represent the resulting value? How does this compare to an l-value? Does this introduce the need for additional operators in the language? How does this relate to the & operator in C?