CSE341 – Programming Languages (Fall 2014)

Homework #1

Reduce Function Definition

```
(define reduce
      (lambda (op l id)
            (if (null? l)
            id
            (op (car l) (reduce op (cdr l) id)))))
```

The reduce function takes a binary operation “op” and applies it right-associatively to a list “l” of an arbitrary number of elements. In other words, this function reduces a list of elements to a single value. If the list empty, reduce returns “id”.

Examples:

```
(reduce + ‘(1 2 3) 0) → 6
(reduce / ‘(24 6 2) 1) → 8
```

In the first example, you should note that:

```
(reduce + ‘(1 2 3) 0)
   → (+ 1 (reduce + ‘(2 3) 0))
   → (+ 1 (+ 2 (reduce + ‘(3) 0)))
   → (+ 1 (+ 2 (+ 3 (reduce + ‘() 0))))
   → (+ 1 (+ 2 (+ 3 0)))
   → (+ 1 (+ 2 3))
   → (+ 1 5)
   → 6
```

In the second example, you should note that:

```
(reduce / ‘(24 6 2) 1)
   → (/ 24 (reduce / ‘(6 2) 1))
   → (/ 24 (/ 6 (reduce / ‘(2) 1)))
   → (/ 24 (/ 6 (/ 2 (reduce / ‘() 1))))
   → (/ 24 (/ 6 (/ 2 1)))
   → (/ 24 (/ 6 2))
   → (/ 24 3)
   → 8
```