CS-257L
Nonimperative Programming: Scheme!

Instructor:
Joel Castellanos
e-mail: joel@unm.edu
Web: http://cs.unm.edu/~joel/
Office: Farris Engineering Center (FEC) room 321

3/7/2008
Homework Due Monday, March 10

- Nothing to hand-in.
- Scheme and The Art of Programming
- Read Chapter 5, and do
  - Exercise 5.3
  - Exercise 5.9
  - Exercise 5.10
- Be prepared for a quiz on this material.
  - Open notes
  - Open book
  - Closed Computer
(define harmonic-sum-it
    (lambda (n)
        (cond
            ((or (not (integer? n)) (< n 1))
                '("Error: argument must be a positive whole number."))
            ((= n 1) 1)
            (else (+ (/ 1 n) (harmonic-sum-it (- n 1))))
        )
    )
)
Scheme: Lexically Scoped

(((lambda (x)
   ((lambda (y)
      (- x y)
   )
   15
) )
20
)
)
Scope of Bindings for Variables

User global environment bindings:
\[ (\text{define} \ variable \ expression) \]

Local bindings:
\[ (\text{let} ( (\text{variable} \ expression)) \ body) \]
\[ (\text{let} ( (\text{var}_1 \ \text{exp}_1) \ (\text{var}_2 \ \text{exp}_2) \ \ldots \ (\text{var}_n \ \text{exp}_n)) \ body) \]
Quiz 4 – What is the output?

```
(define count-parens
  (lambda (ls)
    (display ls) (newline)
    (cond
      ((null? ls) 2)
      (else
        (cond
          ((atom? (car ls))
            (count-parens (cdr ls))
          )
          (else
            (+ (count-parens (car ls))
                (count-parens (cdr ls))
            )
          ))))))

(load "atom.scm")
(count-parens '(((a b) ()) c))
```