CS314 Fall 2015
Assignment 6
Due Tuesday, November 17 hard copy at beginning of class
Section number and Rutgers ID MUST be included.

Problem 1 – Scheme

Write Scheme programs that generate the following lists as output using only cons as the list building operator:

1. ’(a (b c) d ((e f) (g)))
2. ’(((a b c) d) e f) g
3. ’(~ a 3) such that ((car ’(~ a 3)) 5 3)
   evaluates to 2.

Problem 2 – Scheme

Write the following functions on lists in Scheme. The semantics of the functions is decribed through examples.

1. (define flatten
   (lambda (l
            ...
   ))
   ...
   (flatten ’(a ((b) (c d) (((e)))))) => ’(a b c d e)

2. (define rev
   (lambda (l
            ...
   ))
   ...
   (rev ’(a(b)(c d)(((e)))))) => ’(((e))(d c)(b)a)

   Note: Do not use the Scheme build-in function ”reverse”.

3. (define double
   (lambda (l
            ...
   ))
   ...
   (double ’(a((b)(c d)(((e)))))) => ’(a a((b b)(c c d d)(((e e))))))
4. (define delete
   (lambda (atom l)
     ...
   
   ...
   (delete 'c '(a((b)(c d))((e)))))) → (a((b)(d))((e))))
   (delete 'f '(a((b)(c d))((e)))))) → (a((b)(c d))((e))))

Problem 3 – Scheme

Use the map and reduce functions defined as

(define map
  (lambda (f l)
    (if (null? l)
        ()
        (cons (f (car l)) (map f (cdr l)))))

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

to implement functions minSquareVal and maxSquareVal that determine the minimal square value and maximal square value, respectively, of a list of integer numbers. Example

(define minSquareVal
  (lambda (l)
    ...
  ))

...(minSquareVal '(-5 3 -7 10 -11 8 7)) → 9

(define maxSquareVal
  (lambda (l)
    ...
  ))

...(maxSquareVal '(-5 3 -7 10 -11 8 7)) → 121