CSE 3302 Lab Assignment 1

Due July 7, 2014

Goal:

Understanding of Scheme and elementary functional programming concepts.

Requirements:

1. a. Write a Scheme function to implement a general `partition` function to extract elements of a list and combine them together to produce an s-exp. The format of an application of the `partition` function will be:

   (`partition` *partition-exp lat*)

   where *partition-exp* is an s-exp whose atoms are non-negative integers and *lat* is an unnested list whose atoms are unrestricted.

   `partition` takes *partition-exp* and substitutes for each of its atoms a sub-list with that many consecutive atoms from *lat*. The nested structure of *partition-exp* must be preserved.

   If *partition-exp* requires more atoms than are present in *lat*, just report the error.

   b. Write a Scheme function to perform `inverse-partition`, i.e. this will take a result from `partition` and produce the orginal *partition-exp*. (The necessary prefix of the original *lat* is easily obtained using `flatten`.)

2. Email your program to `sourabh.bose@mavs.uta.edu` by 12:45 p.m. on July 7.

Getting Started:

1. Don't be concerned about efficiency.

2. Some examples:

```
> (partition '(2 3 4 (2) (0 (1 1)) 1) '(a b c d e f g h i j k l m n o p))
'((a b) (c d e) (f g h i) ((j k)) (() ((l) (m))) (n))
> (partition '(2 3 ((1)(0)(3)) (2) (0 (1 (((3 2))) 1)) 1) '(a b c d e f g h i j k l m n o p q r s t u))
'((a b) (c d e) (((f)) (()) ((g h i))) ((j k)) (() ((l) ((((m n o) (p q)))) (r))) (s))
> (partition '((((1) (2) 3) 4) 5) '(a b c d e f g h i j k l m n o p q r s t u))
'((((((a)) ((b c)) (d e f)) (g h i j)) (k l m n o))
> (partition '((((1) (2) 3) 4) 5) '(a b c d e f g h i j k l m n))
"lat is too short"

> (inverse-partition (partition '(2 3 4 (2) (0 (1 1)) 1)
                                '(a b c d e f g h i j k l m n o p)))
'(2 3 4 (2) (0 (1 1)) 1)
> (inverse-partition (partition '(2 3 ((1)(0)(3)) (2) (0 (1 (((3 2))) 1)) 1)
                                '(a b c d e f g h i j k l m n o p q r s t u)))
'(2 3 ((1) (0) (3)) (2) (0 (1 (((3 2))) 1)) 1)
> (inverse-partition (partition '((((1) (2) 3) 4) 5)
                                '(a b c d e f g h i j k l m n o p q r s t u)))
'((((1) (2) 3) 4) 5)
> (inverse-partition (partition '((((0) (0) 0) 0) 0)
                                '(a b c d e f g h i j k l m n)))
'((((0) (0) 0) 0) 0)
> (inverse-partition (partition '(0 0 0)
                                '(a b c d e f g h i j k l m n)))
'(0 0 0)
```

3. The Ten Commandments and The Five Rules from *The Little Schemer* will lead you to many days of happiness.

4. `set!` will lead to nights of suffering (and loss of points).