CSC 461 Programming Languages

FALL 2024

Dr. Stephen Krebsbach

Ass #6 - 15 points Due: Tuesday Dec. 17th 11:59 PM

For this assignment I want you to create some basic Lisp functions.

I have created a starting text file for you to use “start.lsp”

You should rename the file when you turn it in using your class id.

So, if class CID is 47 rename the file to “cid47.lsp”

You will see in the file (print “your name”) statement at you should make your name.

There are also some testing setq functions you should not change

I have made header comment blocks for each function for you to use that also help explain the function.

The final function is “Optional” and you do not have to do it, A little more difficult but not much

You should

1. Download the “start.lsp” file
2. Rename using your CID to “cid##.lsp” / ## is your CID for the class
3. Complete and test the functions
4. Drop the text file into the dropbox.
   (the file you drop should load into lisp with out me having to modify)!!!!

I have provided handouts, and we have done some Lisp in class.

You should not use higher level Lisp functions to help such a max or member or …

(if in doubt ask me)

I have included the function headers on the next page. Ask questions if needed!!!

These are the test setq variable used for testing in the file

```
; TEST CASE LISTS

(setq A `((1 2)  3 ( 2 3 (1 9))))

(setq B `(1 2 3 4 5))

(setq C `((1 2) ((3)) 4 (5)))

(setq D `(1 2 (6 3) 4 5))

(setq Empl '( (10 Mary Smith 36 12) (5 Bob Olsen 33 12) (130 Lace Smith 44 16)))
```

```
;****** lookupempl ***************
; returns values related to the Key value of an
; employee tuple defined as  (KEY# Fname Lname Age Dept#)
; Example: employe tuple -->    (10 Mary Smith 36 12)
; A databasee list called Empl will contanin several Employees
; the funtion lookupEmpl will use the Key to return the Emply record list
; of the employee.
; If no employee match return 'NOT FOUND.
; Example:  (lookupempl Empl 10) --> (10 Mary Smith 36 12)
;*******************************

;****** Doubleodd ***************
; Doubleodd: return a list where all the odd numbers have been doubled.
; Example: (doubleodd B) --> (2 2 6 4 10)
; Assume single level list
;*******************************

;****** addAll ******************
; addall: returns the sum of nested list of positive integers
; Example: (addAll C) --> 15
; Return 0 if the list is empty
; Assume that it may be a nested level list
;*******************************

;----- OPTIONAL (do not have to do) ------------------
;****** large_atom ***************
; large_atom : returns the largest value greater than 0
; in a nested list of positive integers
; Example: (large_atom D) --> 6
; Return 0 if the list is empty
; Assume that it may be a nested level list
; DO NOT use higher level funtions like max ..
;
; THIS SHOULD BE THE HARDER ONE :)
;********************************
```