LISP Assignment 2

[Note: While submitting your program, name your program as RollNo_Assignment2_Ans1.lsp (For example, 10001004_Assignment2_Ans1.lsp); RollNo_Assignment2_Ans2.lsp; and RollNo_Assignment2_Ans3.lsp].

1. (10 Points) Write a function called my-member that determines if a given element is in a list. The first item in the function call represents the item to be found and the second item is the list to be searched.

(my-member ‘a ‘(b c j a)) => (A)

(my-member ‘(c d) ‘(a b c (a b) (c d) j k)) => ((c d) j k)

(my-member ‘a ‘(b c (a d) e)) => nil

2. (10 Points) Write a function called my-insert that inserts a given element into a list at a given location (starting with position zero). If the entered location is negative then the function places the new element at the beginning of the list. If the entered location is greater than the length of the list, then the function places the new element at the end of the list.

(my-insert ‘a ‘(b c d e) 0) => (A B C D E)

(my-insert ‘a ‘(b c d e) 3) => (B C D A E)

(my-insert ‘a ‘(b c d e) 14) => (B C D E A)

3. (10 Points) Write a function to remove duplicates from a list, the function is to be called my-remv-dups. Note that the function is to retain the last instance of a duplicate element.

(my-remv-dups ‘(a b a c a d)) => (B C A D)

(my-remv-dups ‘(a b (a c) a d)) => (B (A C) A D)

(my-remv-dups ‘(a b (b c) a (b c) d)) => (B A (B C) D)

Due date: Monday, 13 November 2018, 11:59 PM