(defun my-fun (list)

(cond

((null list) nil)

; handle unique value

((not (member (car list) (cdr list)))

(cons (car list) (my-fun (cdr list)))

)

; handle non-unique value

(t (my-fun (cdr list)))

)

)

(print (my-fun '(1 1 a a 3 3 3 + + + +)))

Для теста (my-fun '(1 1 a a 3 3 3 + + + + 1 1 1 1)) функция будет работать неправильно. Надо проверять соседние элементы на совпадение

PS

Задание было следующее

Напишите на CLISP программы для работы со списками по заданию. Обязательно использование рекурсии. Ввод всех входных данных должен запрашиваться с клавиатуры в процессе работы программы В программе на языке CLISP не допускается использование: функционалов, а в теле рекурсивной функции - операторов SET и SETQ.

Из каждой группы подряд идущих элементов списка L оставьте только один. Например, для языка CLISP при L=(1 1 a a 3 3 3 + + + +) функция должна вернуть (1 a 3 +).