Name: Galbenu Horia (ghie1544)

Group: 922

<u>Lab code</u>: L3

<u>Problem number and statement</u>: Problem 8

Write a function that reverses a list together with all its sublist elements at any level.

<u>Formal descriptions</u>:

- <u>Mathematical models</u>:

$$\text{reverseAll}(l1\ldots ln): \begin{cases} \text{reverseAux}(ln), \text{reverseAux}(ln\text{-}1), \ldots \text{reverseAux}(l1) \end{cases}$$

$$\text{reverseAux}(elem): \begin{cases} \text{reverseAll}(elem), \text{ if elem is a list} \\ elem, \text{ otherwise} \end{cases}$$

- <u>Meaning of function parameters</u>:

  l1,l2,…,ln – elements of the input list
  elem – either a list or an atom, if list we call the main function on it

<u>Source code</u>:

```
(defun reverseAux (l)
  (cond
    ((listp l) (reverseAll l))
    (t l)))
(defun reverseAll (l)
  (mapcar 'reverseAux (reverse l)))
```

<u>Running examples</u>:

```
(reverseAll '(1 2 (x y z))) -> (reverseAux '(x y z)) -> (reverseAll '(x y z)) -> (z y x)
                            ->  (reverseAux '2) -> 2
                            ->  (reverseAux '1) -> 1
=> ((z y x) 2 1)
```