

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΜΗΧ/ΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ

ΤΕΧΝΟΛΟΓΙΕΣ ΕΥΦΥΩΝ ΣΥΣΤΗΜΑΤΩΝ ΚΑΙ ΡΟΜΠΟΤΙΚΗ

1^ο ΣΕΤ-Ασκήσεις LISP

1. Γράψτε μια συνάρτηση `butmid` που να παίρνει σαν είσοδο ένα αριθμό n , μια λίστα και ένα δεύτερο αριθμό m , και να επιστρέφει τη λίστα χωρίς τα πρώτα n και τα τελευταία m στοιχεία της.

π.χ. `(butmid 2 '(a b c d e) 1) ==> (C D)`

2. Θεωρώντας μια a -list (λίστα ζευγών) σαν αναπαράσταση ζευγών του τύπου μεταβλητή-τιμή, γράψτε μια συνάρτηση `return-binding` που παίρνει σαν είσοδο μια μεταβλητή και μια λίστα ζευγών και επιστρέφει την τιμή της μεταβλητής. Εάν δεν υπάρχει τιμή τότε επιστρέφει το άτομο 'unbound'.

π.χ. `(return-binding `y '((x.a) (y.b) (z.c))) ==> B`

`(return-binding `w '((x.a) (y.b) (z.c))) ==> UNBOUND`

3. Γράψτε μια συνάρτηση Lisp (`count-elems`) που παίρνει σαν όρισμα μια λίστα που περιέχει άτομα και άλλες μη κενές λίστες οι οποίες με τη σειρά τους περιέχουν άτομα και άλλες μη κενές λίστες κ.ο.κ. (είναι δηλ. οποιουδήποτε βάθους) και επιστρέφει το πλήθος των ατόμων στη λίστα.

π.χ. `(count-elems `(1 (a ((2 1) b)) 3 (a c))) → 8.`

Γράψτε δύο εκδοχές της συνάρτησης, μια με `if` και μια με `cond`.

4. Θεωρείστε μια a -list σαν αναπαράσταση μιας αντικατάστασης σε μια διαδικασία ενοποίησης, δηλ. σαν μια λίστα ζευγών μεταβλητή-όρος, όπου το 'όρος' μπορεί να είναι οτιδήποτε (σύμβολο, αριθμός, λίστα). Γράψτε μια συνάρτηση `compatible-substitutions` που να παίρνει σαν είσοδο δύο αντικαταστάσεις (a -lists) και ελέγχει αν είναι συμβατές, δηλ. αν κάθε μεταβλητή έχει μόνο ένα ή τον ίδιο όρο και στις δύο αντικαταστάσεις, οπότε επιστρέφει `T`. Αν δεν είναι συμβατές επιστρέφει `NIL`.

π.χ. `(compatible-substitutions 1((x.a) (y.b) (z a b)) '((w.c) (r.b) (x.a))) ==> T`

`(compatible-substitutions '((x.a) (y.b)) '((w.c) (x.d))) ==> NIL`

Γράψτε δύο εκδοχές της συνάρτησης, μια επαναληπτική και μια αναδρομική.