

Ασκήσεις Φυλλάδιο 4

Λίστες και ενσωματωμένα κατηγορήματα

1. α) Γράψτε ένα κατηγορήμα Prolog που πετυχαίνει αν μια λίστα αποτελείται από 2 ίδια μέρη:

```
?- symmetric([1,2,3,4,1,2,3,4]).
yes.
```

β) Αναπτύξτε ένα κατηγορήμα το οποίο ελέγχει αν το πρώτο όρισμα (λίστα) είναι επίθεμα (suffix) του δεύτερου ορίσματος (λίστα) :

```
?-end_sublist([e,r,t],[s,f,j,u,e,r,t]).
yes.
```

γ) Αναπτύξτε ένα κατηγορήμα το οποίο πετυχαίνει όταν η λίστα του πρώτου ορίσματος εμφανίζεται δύο φορές μέσα στη λίστα του δεύτερου ορίσματος:

```
?- twice_sublist([3,4],[1,2,3,4,5,3,4,6]).
yes.
```

δ) Υλοποιείστε ένα Prolog κατηγορήμα που βρίσκει το τελευταίο στοιχείο μιας λίστας:

```
?- last_element([3,4,5,1,6],X).
X=6
```

2. Υλοποιείστε ένα κατηγορήμα που βρίσκει το γράμμα το οποίο λείπει από μια λέξη (ορθογραφικό λάθος). Για παράδειγμα:

```
?- missing_letter([p,r,l,o,g],X,W).
X=o
W = [p,r,o,l,o,g]
```

Υποθέστε ότι έχετε μια βάση με όλες τις λέξεις της Αγγλικής:

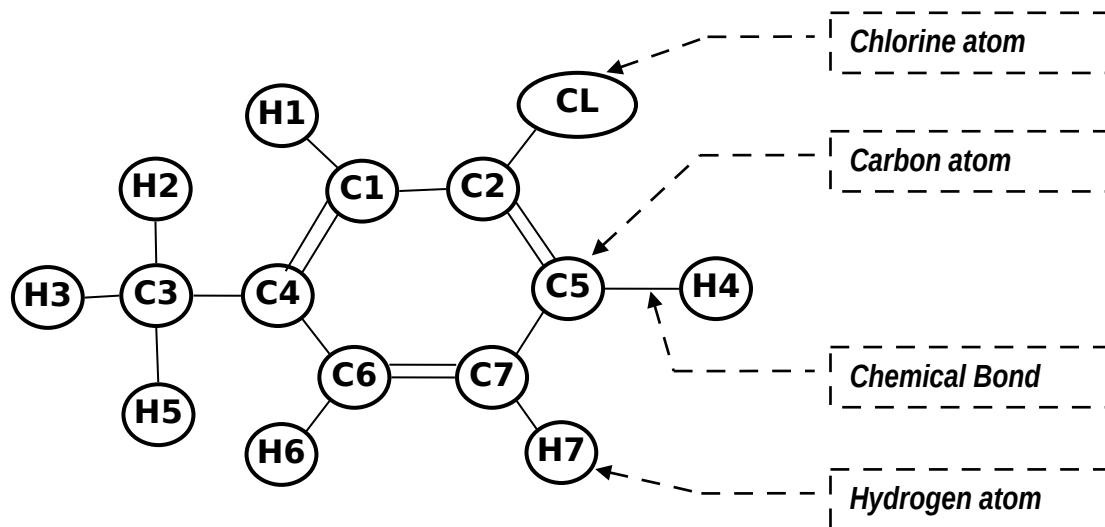
```
word([p,r,o,l,o,g]).
word([m,a,t,h,s]).
...
```

3. Ορίστε το κατηγορήμα **reverse_alt/2** (**reverse_alt(List1, Lis2)**) το οποίο πετυχαίνει όταν η **List2** είναι η ανάστροφη λίστα της **List1**, χρησιμοποιώντας το ενσωματωμένο κατηγορήμα **append/3**.

4. Χρησιμοποιώντας το ενσωματωμένο κατηγορήμα **append/3** ορίστε ένα κατηγορήμα **rotate_left(Pos,List,RotatedList)**, το οποίο πετυχαίνει όταν το όρισμα **RotatedList** είναι η λίστα **List** που έχει υποστεί αριστερή περιστροφή κατά **Pos** θέσεις. Για παράδειγμα:

```
?- rotate_left(1,[1,2,3,4,5],L).
L = [2, 3, 4, 5, 1]
Yes
?- rotate_left(3,[a,b,c,d,e],L).
L = [d, e, a, b, c]
Yes
```

5. Η οργανική ένωση 3-chloro-toluene έχει την ακόλουθη δομή:



Στην παραπάνω εικόνα, $H1, H2 \dots Hn$ είναι άτομα υδρογόνου (hydrogen), $C1, C2, \dots Cn$ άτομα άνθρακα (carbon) και CL το άτομο χλωρίου (Chlorine) και οι γραμμές που τα ενώνουν αναπαριστούν τους χημικούς δεσμούς ανάμεσα στα άτομα. Η ταυτότητα των ατόμων αναπαρίσταται από γεγονότα της μορφής:

```
atom(hydrogen, [h1, h2, h3, h4, h5, h6, h7]) .
atom(carbon, [c1, c2, c3, c4, c5, c6, c7]) .
atom(chlorine, [cl]) .
```

α) Να ορίσετε ένα κατηγορημα **atom_bonds_to/2** το οποίο αναπαριστά τους χημικούς δεσμούς των ατόμων στη παραπάνω ένωση. Για κάθε άτομο το οποίο συμμετέχει στην ένωση θα πρέπει να υπάρχει ένα αντίστοιχο γεγονός, και οι δεσμοί του με τα γειτονικά του άτομα να αναπαριστώνται με μια λίστα.

Note: Να μην κάνετε διαχωρισμό μεταξύ απλών και διπλών χημικών δεσμών.

[40%]

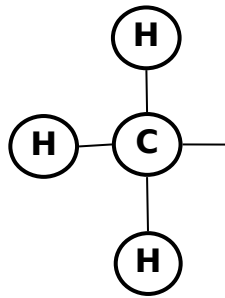
β) Ορίστε τα ακόλουθα κατηγορήματα:

- **carbon/1** πετυχαίνει όταν το όρισμά του είναι ένα άτομο άνθρακα.
- **hydrogen/1** πετυχαίνει όταν το όρισμά του είναι ένα άτομο υδρογόνου
- **bonded(X,Y)** πετυχαίνει όταν ανάμεσα στα ορίσματα του υπάρχει χημικός δεσμός.

Για παράδειγμα:

```
?- carbon(c1) .
yes
?- carbon(h1) .
no
?- hydrogen(h1)
yes
?- bonded(c1, h1) .
yes
?- bonded(c1, c6) .
no
```

γ) Το μεθύλιο (ρίζα μεθυλίου) έχει την ακόλουθη μορφή:



Η ρίζα αναγνωρίζεται μοναδικά από το άτομο άνθρακα καθώς αυτό είναι μοναδικό σε κάθε ρίζα και το οποίο ενώνεται με άλλα άτομα. Ορίστε ένα κατηγορημα **methyl/1**, το οποίο πετυχαίνει όταν το όρισμά του είναι το άτομο άνθρακα της ρίζας μεθυλίου στην χημική ένωση που φαίνεται παραπάνω. Για παράδειγμα:

```
?- methyl(c3) .
yes
?- methyl(c7)
no
```

6. Να υλοποιήσετε ένα κατηγορημα **common_suffix/4** (**common_suffix(L1,L2,Suffix,Pos)**), το οποίο επιστρέφει στο όρισμα **Suffix**, το κοινό επιθέμα των λιστών **L1** και **L2**, με το **Pos** να είναι η θέση της αρχής του κοινού επιθέματος από το τέλος. Κατά την οπισθοδρόμηση θα επιστρέφονται όλα τα κοινά επιθέματα ξεκινώντας με εκείνο που έχει το μεγαλύτερο μήκος. Για παράδειγμα:

```
?- common_suffix([1, b, c], [a, b, c], Suffix, Pos) .
Suffix = [b, c]
Pos = 2
Yes
Suffix = [c]
Pos = 1
Yes
Suffix = []
Pos = 0
Yes

?- common_suffix([a, b, c], [a, b, c], Suffix, Pos) .
Suffix = [a, b, c]
Pos = 3
Yes
Suffix = [b, c]
Pos = 2
Yes
Suffix = [c]
Pos = 1
Yes
Suffix = []
Pos = 0
Yes

?- common_suffix([1, b, 3], [a, b, c], Suffix, Pos) .
Suffix = []
Pos = 0
Yes
```