

7η Σειρά Εργαστηριακών Ασκήσεων

Οι εργαστηριακές ασκήσεις είναι **ατομικές**. Οι απαντήσεις θα πρέπει να υποβληθούν με **turnin**, το αργότερο μέχρι την **Παρασκευή 21 Μαΐου 2021, ώρα 16:45**. Πριν ξεκινήσετε να γράφεται τα προγράμματα που ζητούνται στις ασκήσεις της σειράς αυτής, **διαβάστε πολύ προσεκτικά τις αναλυτικές οδηγίες** που ακολουθούν.

Οδηγίες

- Για τη συγγραφή των προγραμμάτων επιτρέπεται να χρησιμοποιήσετε προκαθορισμένα κατηγορήματα και προκαθορισμένους τελεστές **μόνο εφόσον αναφέρονται στις σημειώσεις του μαθήματος**.
- Για τη συγγραφή των προγραμμάτων θα πρέπει να χρησιμοποιήσετε το αρχείο πρότυπο Lab7.pro (που υπάρχει στην ιστοσελίδα του μαθήματος), στο οποίο για κάθε κατηγορήμα που ζητείτε να ορίσετε στις παρακάτω ασκήσεις, υπάρχει ένας κανόνας ο οποίος το ορίζει έτσι ώστε να επιστρέφει πάντα την απάντηση no. Για να απαντήσετε στις ασκήσεις αντικαταστήστε καθέναν από τους παραπάνω κανόνες με ένα κατάλληλο σύνολο προτάσεων που να ορίζει το αντίστοιχο κατηγορήμα. **Δεν θα πρέπει να τροποποιήσετε το όνομα κανενός κατηγορήματος ούτε το πλήθος των ορισμάτων του**.
- Μπορείτε να ορίσετε όσα βοηθητικά κατηγορήματα θέλετε, τα οποία θα χρησιμοποιούνται για τον ορισμό των κατηγορημάτων που σας ζητείται να υλοποιήσετε. Σε καμία περίπτωση δεν θα πρέπει να προσθέσετε άλλα ορίσματα στα κατηγορήματα που σας ζητούνται.
- **Αν χρησιμοποιήσετε προκαθορισμένα κατηγορήματα ή τελεστές που δεν αναφέρονται στις σημειώσεις του μαθήματος, η αντίστοιχη άσκηση δεν θα βαθμολογηθεί.**
- Ο έλεγχος της ορθότητας των απαντήσεων θα γίνει με ημι-αυτόματο τρόπο. Σε καμία περίπτωση δεν θα πρέπει ο βαθμολογητής να χρειάζεται να κάνει παρεμβάσεις στο αρχείο που θα υποβάλετε. Συνεπώς θα πρέπει να λάβετε υπόψη τα παρακάτω:
 1. Κάθε ένα από τα κατηγορήματα που σας ζητείται να υλοποιήσετε θα πρέπει να έχει το συγκεκριμένο όνομα και το συγκεκριμένο πλήθος ορισμάτων που περιγράφεται στην εκφώνηση της αντίστοιχης άσκησης και που υπάρχει στο αρχείο πρότυπο Lab7.pro. **Αν σε κάποια άσκηση το όνομα ή το πλήθος των ορισμάτων δεν συμφωνεί με αυτόν που δίνεται στην εκφώνηση, η άσκηση δεν θα βαθμολογηθεί.**
 2. Το αρχείο που θα παραδώσετε δεν θα πρέπει να περιέχει συντακτικά λάθη. Αν υπάρχουν τμήματα κώδικα που περιέχουν συντακτικά λάθη, τότε θα πρέπει να τα διορθώσετε ή να τα αφαιρέσετε πριν από την παράδοση. **Αν το αρχείο που θα υποβάλετε περιέχει συντακτικά λάθη, τότε ολόκληρη η εργαστηριακή άσκηση θα μηδενιστεί.**

3. Οι ερωτήσεις που δίνονται στο τέλος κάθε άσκησης θα πρέπει να επιστρέφουν απάντηση. Αν κάποιες από τις επιστρεφόμενες απαντήσεις δεν είναι σωστές, αυτό θα ληφθεί υπόψη στη βαθμολογία, ωστόσο η άσκηση θα βαθμολογηθεί κανονικά. **Αν ωστόσο κάποια από τις παραπάνω ερωτήσεις δεν επιστρέφει απάντηση, (π.χ. προκαλείται υπερχείλιση στοίβας, ατέρμονος υπολογισμός ή κάποιο σφάλμα χρόνου εκτέλεσης) τότε ο βαθμός για την υλοποίηση του αντίστοιχου κατηγορήματος θα είναι μηδέν.**
 4. Κατα τη διόρθωση των ασκήσεων οι βαθμολογητές δεν θα κάνουν χρησιμοποιήσουν ερωτήσεις που εμπεριέχουν τα βοηθητικά κατηγορήματα τα οποία ενδεχομένως θα έχετε ορίσει. Η χρήση των βοηθητικών κατηγορημάτων θα πρέπει να γίνεται μέσα από τα κατηγορήματα που σας ζητείται να υλοποιήσετε.
- Μετά το τέλος της εκφώνησης κάθε άσκησης δίνονται παραδείγματα ερωτήσεων με τις αντίστοιχες αναμενόμενες απαντήσεις, που μπορείτε να χρησιμοποιήσετε για έλεγχο της ορθότητας των προγραμμάτων σας.
 - Μπορείτε να συνδέεστε στην ομάδα ‘ΕΡΓΑΣΤΗΡΙΟ στις ΑΡΧΕΣ ΓΛΩΣΣΩΝ ΠΡΟΓΡΑΜΜΑΤΙΜΟΥ’ στο MsTeams τις ώρες του εργαστηρίου (Τρίτη 10-12), για την επίλυση αποριών και προβλημάτων που ενδέχεται να συναντήσετε κατά τη συγγραφή των προγραμμάτων στο πλαίσιο των εργαστηριακών ασκήσεων. Για αντίστοιχα προβλήματα ή απορίες που θα προκύψουν στο διάστημα από την περάτωση του εργαστηρίου μέχρι την υποβολή της εργασίας μπορείτε να επικοινωνήσετε με την κ. Βίκυ Σταμάτη την Παρασκευή 10:00-12:00 είτε μέσω MsTeams (προσωπικό μήνυμα) είτε μέσω ηλεκτρονικού ταχυδρομείου (vstamati@uoi.gr). Θα απαντηθούν μόνο ηλεκτρονικά μηνύματα που έχουν σταλεί από τον ιδρυματικό σας λογαριασμό.
 - Για υποβολή με turnin γράψτε:

turnin Prolog-7@myy401 Lab7.pro

Ασκηση 1.

Σε μία ψηφοφορία για την ανάδειξη του καλύτερου ποδοσφαιριστή της χρονιάς, ψηφίζουν n δημοσιογράφοι και νικητής είναι όποιος συγκεντρώσει περισσότερες από $n/2$ ψήφους. Αν κανένας υποψήφιος δεν συγκεντρώσει περισσότερες από $n/2$ ψήφους τότε δεν υπάρχει νικητής.

Γράψτε ένα πρόγραμμα σε Prolog το οποίο με δεδομένη μία λίστα με τις ψήφους των δημοσιογράφων θα βρίσκει το νικητή της ψηφοφορίας, εφόσον υπάρχει. Συγκεκριμένα, ορίστε ένα κατηγορημα `majority(L,X)`, το οποίο θα αληθεύει αν το `X` εμφανίζεται σε περισσότερες θέσεις από όσο είναι το μισό του μήκους της λίστας `L`.

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

```
| ?- majority([],X).
```

no

```
| ?- majority(['Messi'],X).
```

X = 'Messi'

```
| ?- majority(['Ronaldo', 'Messi'],X).
```

no

```
| ?- majority(['Mbappe', 'Messi', 'Mbappe'],X).
```

X = 'Mbappe'

```
| ?- majority(['Ronaldo', 'Ronaldo', 'Messi', 'Messi', 'Ronaldo',  
'Ronaldo', 'Ronaldo', 'Salah', 'Mbappe'],X).
```

X = 'Ronaldo'

```
| ?- majority(['Ronaldo', 'Messi', 'Messi', 'Dybala', 'Messi',  
'Messi', 'Salah', 'Mbappe'],X).
```

no

```
| ?- majority(['Messi', 'Ronaldo', 'Messi', 'Salah', 'Salah',  
'Salah', 'Salah'],X).
```

X = 'Salah'

```
| ?- majority(['Dybala', 'Dybala', 'Dybala', 'Dybala', 'Messi'],X).
```

X = 'Dybala'

```
| ?- majority(['Messi', 'Messi', 'Messi', 'Ronaldo', 'Haaland', 'Ronaldo',  
'Ronaldo', 'Mbappe', 'Messi', 'Messi', 'Messi'],X).
```

```
X = 'Messi'
```

```
| ?- majority(['Mbappe', 'Messi', 'Mbappe', 'Ronaldo', 'Mbappe',  
'Ronaldo', 'Mbappe', 'Messi', 'Mbappe'],X).
```

```
X = 'Mbappe'
```

```
| ?- majority(['Mbappe', 'Messi', 'Salah', 'Ronaldo', 'Dybala', 'Ronaldo',  
'Mbappe', 'Messi', 'Mbappe', 'Salah'],X).
```

```
no
```

```
| ?- majority(['Ronaldo', 'Haaland', 'Ronaldo', 'Ronaldo', 'Messi', 'Ronaldo',  
'Haaland', 'Messi', 'Ronaldo', 'Ronaldo', 'Messi', 'Messi', 'Messi', 'Messi',  
'Messi', 'Mbappe', 'Messi', 'Mbappe', 'Messi', 'Messi', 'Messi', 'Messi',  
'Mbappe', 'Messi', 'Dybala', 'Messi', 'Dybala', 'Messi', 'Messi', 'Dybala'],X).
```

```
X = 'Messi'
```

Άσκηση 2.

Μπορούμε να προσθέσουμε στο τέλος μίας λίστας L επαναληπτικά το στοιχείο 0 , μέχρι να προκύψει λίστα το μήκος της οποίας είναι δύναμη του 2 . Ονομάζουμε τη λίστα που προκύπτει με αυτή τη διαδικασία 2^n -επέκταση της L . Αν το μήκος της L είναι δύναμη του 2 , τότε η 2^n -επέκτασή της είναι η ίδια η L .

Γράψτε ένα πρόγραμμα σε Prolog το οποίο να υπολογίζει τη 2^n -επέκταση μίας δεδομένης λίστας. Συγκεκριμένα ορίστε ένα κατηγορημα $\text{expand}(L,E)$, το οποίο θα αληθεύει αν L και E είναι λίστες και η E είναι η 2^n -επέκταση της L .

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

```
| ?- expand([],E).
```

```
E = [0]
```

```
| ?- expand([nil],E).
```

```
E = [nil]
```

```
| ?- expand([yes,no],E).
```

```
E = [yes,no]
```

```
| ?- expand([three,two,one],E).
```

```
E = [three,two,one,0]
```

```
| ?- expand([africa, america, asia, europe, oceania],E).
```

```
E = [africa,america,asia,europe,oceania,0,0,0]
```

```
| ?- expand(['.',:',...',':','::','::'],E).
```

```
E = ['.',:',...',':','::','::',0,0]
```

```
| ?- expand([=>,==>,<=,<==,<>,<=>,<==>],E).
```

```
E = [=>,==>,<=,<==,<>,<=>,<==>,0]
```

```
| ?- expand([ooo,ooi,oio,oii,ioo,ioi,iio,iii],E).
```

```
E = [ooo,ooi,oio,oii,ioo,ioi,iio,iii]
```

```
| ?- expand([a1,a2,a3,b1,b2,b3,c1,c2,c3],E).
```

```
E = [a1,a2,a3,b1,b2,b3,c1,c2,c3,0,0,0,0,0,0,0]
```

```
| ?- expand([a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z],E).
```

```
E = [a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z,0,0,0,0,0,0,0]
```