

3η Σειρά Εργαστηριακών Ασκήσεων

Οι απαντήσεις θα πρέπει να υποβληθούν με **turnin**, το αργότερο μέχρι την **Παρασκευή 22 Απριλίου 2016, ώρα 20:00**.

- Για μεγαλύτερη ευκολία στην εκπόνηση αυτής και της επόμενης εργαστηριακής άσκησης σας συνιστώ:
 1. να δημιουργήσετε έναν κατάλογο Prolog κάτω από το home directory σας.
 2. να χρησιμοποιήτε αυτόν τον κατάλογο για αποθήκευση των αρχείων με τα προγράμματα Prolog που θα γράψετε.
 3. να μεταβαίνετε σε αυτόν τον κατάλογο πριν εκτελέσετε τον διερμηνέα της Prolog.
- Αν θέλετε να χρησιμοποιήσετε τη Prolog στο δικό σας υπολογιστή, μπορείτε να κατεβάσετε το διερμηνέα GNU Prolog από το σύνδεσμο

<http://www.gprolog.org/>

Συνοπτικές οδηγίες για τη χρήση του GNU Prolog υπάρχουν στις σημειώσεις.

- Πριν ξεκινήσετε να γράφετε τα προγράμματα που ζητούνται στις παρακάτω ασκήσεις, θα ήταν χρήσιμο να γράψετε σε ένα αρχείο ορισμένα από τα κατηγορήματα των σημειώσεων, να φορτώσετε το αρχείο στον διερμηνέα της Prolog και να κάνετε ερωτήσεις χρησιμοποιώντας τα κατηγορήματα αυτά, έτσι ώστε να εξοικειωθείτε με την γλώσσα Prolog και το διερμηνέα της.
- Για τη συγγραφή των προγραμμάτων θα πρέπει να χρησιμοποιήσετε το αρχείο πρότυπο Lab3.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 τους παραπάνω κανόνες με ένα κατάλληλο σύνολο προτάσεων που να ορίζει το κάθε κατηγορήμα. **Δεν θα πρέπει να τροποποιήσετε το όνομα κανενός κατηγορήματος ούτε το πλήθος των ορισμάτων του.**
- Οι ασκήσεις 1 και 2 θεωρούν ως δεδομένο τον ορισμό κάποιων κατηγορημάτων. Τα κατηγορήματα αυτά ορίζονται με ένα πλήθος γεγονότων τα οποία συμπεριλαμβάνονται στο αρχείο πρότυπο Lab3.pro. **Δεν θα πρέπει να σβήσετε ούτε να τροποποιήσετε τα γεγονότα αυτά.**

- Μπορείτε να χρησιμοποιήσετε 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.
- 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.
- 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 Lab3.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.
- Για υποβολή με turnin γράψτε (υποθέτοντας ότι έχετε γράψει το πρόγραμμα στο αρχείο Lab3.pro):

turnin Prolog-1@myy401 Lab3.pro

Ασκηση 1.

Μία διακρατική υπηρεσία για την πάταξη της τρομοκρατίας συλλέγει πληροφορίες για ένα πλήθος υπόπτων για τρομοκρατικές ενέργειες και καταγράφει γεγονότα τα οποία ενδεχομένως σχετίζονται με τρομοκρατικές δράσεις.

Τα δεδομένα αυτά καταγράφονται ως γεγονότα σε γλώσσα Prolog. Για λόγους ευκολίας, οι ημερομηνίες παριστάνονται χρησιμοποιώντας φυσικούς αριθμούς ξεκινώντας από κάποια αρχική ημερομηνία.

Μεταξύ άλλων, το πρόγραμμα Prolog περιέχει γεγονότα που ορίζουν τα παρακάτω κατηγορήματα:

- $at(X,A,B,C)$: ο ύποπτος X βρισκόταν στην πόλη C από την ημέρα A ως και την ημέρα B .
- $event(E,C,D)$: το γεγονός E συνέβη στην πόλη C την ημέρα D .
- $country(C,S)$: η πόλη C βρίσκεται στη χώρα S .
- $dif(T1,T2)$: ο όρος $T1$ είναι διαφορετικός από τον $T2$ (προϋποθέτει ότι κανένας από τους όρους $T1$ και $T2$ δεν είναι μεταβλητή).

Για την επεξεργασία των παραπάνω δεδομένων, σας ζητείται να γράψετε κανόνες που να ορίζουν τα παρακάτω κατηγορήματα:

- $q1(C1,C2)$: Στις πόλεις $C1$ και $C2$ έχουν συμβεί καταγεγραμμένα γεγονότα την ίδια ημέρα (το κατηγορήμα να αληθεύει και στην περίπτωση όπου $C1=C2$).
- $q2(X,Y,C)$: Οι ύποπτοι X και Y έχουν βρεθεί την ίδια μέρα στην πόλη C (το κατηγορήμα δεν πρέπει να αληθεύει στην περίπτωση όπου $X=Y$).
- $q3(S)$: Στη χώρα S έχουν συμβεί τρία διαφορετικά γεγονότα σε διάστημα μίας εβδομάδας (ως εβδομάδα θεωρούμε το διάστημα που αποτελείται από οποιεσδήποτε 7 συνεχόμενες ημέρες).
- $q4(X)$: Ο X έχει βρεθεί σε πόλεις δύο διαφορετικών χωρών σε ημέρες κατά τις οποίες συνέβησαν γεγονότα στις πόλεις αυτές.

Για έλεγχο χρησιμοποιήστε τις παρακάτω ερωτήσεις. Οι απαντήσεις προϋποθέτουν ότι τα κατηγορήματα at , $event$, $country$ και dif έχουν οριστεί από τις προτάσεις που περιέχονται στο αρχείο `Lab3.pro`. Για λόγους απόκρυψης πληροφορίας δεν αναγράφονται τα πραγματικά ονόματα των υπόπτων, αλλά χρησιμοποιούνται ψευδώνυμα. Σημειώνεται ότι τα κατηγορήματα που θα ορίσετε θα πρέπει να δουλεύουν σωστά για οποιοδήποτε εναλλακτικό ορισμό των κατηγορημάτων at , $event$ και $country$.

```
| ?- q1('Athens','Athens').
```

```
yes
```

```
| ?- q1('Amsterdam','Hamburg').
```

```
yes
```

```
| ?- q1('London','Paris').
```

```
no
```

```
| ?- setof((C1,C2),(q1(C1,C2),dif(C1,C2)),L).
```

```
L = [('Amsterdam','Hamburg'),('Amsterdam','Milan'),('Athens','Groningen'),  
('Brussels','Frankfurt'),('Frankfurt','Brussels'),('Groningen','Athens'),  
('Hamburg','Amsterdam'),('Milan','Amsterdam'),('Paris','Strasbourg'),  
('Strasbourg','Paris')]
```

```
| ?- q2(tiger,spider,'Berlin').
```

```
yes
```

```
| ?- q2(hawk,spider,'Berlin').
```

```
no
```

```
| ?- setof((X,Y,C),q2(X,Y,C),L).
```

```
L = [(hawk,tiger,'Berlin'),(shark,spider,'Brussels'),(shark,tiger,'Rome'),  
(shark,wolf,'Rome'),(spider,shark,'Brussels'),(spider,tiger,'Berlin'),  
(tiger,hawk,'Berlin'),(tiger,shark,'Rome'),(tiger,spider,'Berlin'),  
(tiger,wolf,'Athens'),(tiger,wolf,'Rome'),(wolf,shark,'Rome'),  
(wolf,tiger,'Athens'),(wolf,tiger,'Rome')]
```

```
| ?- setof(S,q3(S),L).
```

```
L = ['Belgium','France']
```

```
| ?- setof(X,q4(X),L).
```

```
L = [tiger, wolf]
```

Ασκηση 2.

Μία χώρα αποτελείται από ένα σύμπλεγμα νησιών στα οποία υπάρχουν διασκορπισμένες κάποιες πόλεις. Οι πόλεις αυτές βρίσκονται περίπου στο ίδιο γεωγραφικό πλάτος και μπορούμε να τις διατάξουμε έτσι ώστε οι πόλεις που βρίσκονται πιο ανατολικά να προηγούνται στη διάταξη. Για απλούστευση, οι πόλεις είναι αριθμημένες με φυσικούς αριθμούς έτσι ώστε οι αριθμοί των πόλεων να αυξάνονται όσο κινούμαστε από τα ανατολικά προς τα δυτικά.

Η συγκοινωνία ανάμεσα στις πόλεις γίνεται με αεροπλάνα, πλοία και τρένα. Το κάθε ένα από τα μέσα συνδέει κάποια ζεύγη πόλεων και οι διαθέσιμες συνδέσεις περιγράφονται από τα παρακάτω κατηγορήματα:

- $\text{plane}(A,B)$: η πόλη A συνδέεται με την πόλη B με αεροπλάνο.
- $\text{boat}(A,B)$: η πόλη A συνδέεται με την πόλη B με πλοίο.
- $\text{train}(A,B)$: η πόλη A συνδέεται με την πόλη B με τρένο.

Όλες οι συνδέσεις που περιγράφονται από τα παραπάνω κατηγορήματα είναι και προς τις δύο κατευθύνσεις, και σε όλες τις καταχωρήσεις η πόλη A βρίσκεται ανατολικότερα από την πόλη B.

Έστω ότι τα κατηγορήματα plane , boat και train έχουν οριστεί από ένα πλήθος γεγονότων. Γράψτε κανόνες που να ορίζουν τα παρακάτω κατηγορήματα:

- $e(A,B)$: μπορούμε να μεταβούμε από την πόλη A στην πόλη B κινούμενοι συνεχώς από τα ανατολικά προς τα δυτικά μέσω οποιουδήποτε πλήθους από ενδιαμέσες πόλεις. Δεν υπάρχει κανένας περιορισμός για το μέσο που θα χρησιμοποιήσουμε για τη μετάβαση από τη μία πόλη στην άλλη. Οι A και B θα πρέπει να είναι διαφορετικές πόλεις.
- $q(A,B)$: μπορούμε να μεταβούμε από την πόλη A στην πόλη B κινούμενοι συνεχώς από τα ανατολικά προς τα δυτικά μέσω οποιουδήποτε πλήθους από ενδιαμέσες πόλεις. Υπάρχει ο περιορισμός ότι θα πρέπει να χρησιμοποιήσουμε το πολύ μία φορά αεροπλάνο ή πλοίο. Οι A και B θα πρέπει να είναι διαφορετικές πόλεις.
- $p(A,B,N)$: μπορούμε να μεταβούμε από την πόλη A στην πόλη B χρησιμοποιώντας ακριβώς N συνδέσεις. Οι μεταβάσεις από την μία πόλη στην άλλη μπορεί να γίνονται με οποιοδήποτε μέσο και προς οποιαδήποτε κατεύθυνση. Οι A και B ενδέχεται να ταυτίζονται.

Για έλεγχο χρησιμοποιήστε τις παρακάτω ερωτήσεις. Οι απαντήσεις προϋποθέτουν ότι τα κατηγορήματα plane , boat και train έχουν οριστεί από τα γεγονότα που περιέχονται στο αρχείο `Lab3.pro`. Σημειώνεται ωστόσο ότι τα κατηγορήματα που θα ορίσετε θα πρέπει να δουλεύουν σωστά για οποιοδήποτε εναλλακτικό ορισμό των κατηγορημάτων plane , boat και train .

| ?- e(4,13).

yes

| ?- e(5,14).

no

| ?- setof(B,e(1,B),L).

L = [3,4,5,6,7,8,9,10,11,12,13,15]

| ?- setof(A,e(A,10),L).

L = [1,2,3,4,5,6,7,9]

| ?- q(2,13).

yes

| ?- q(1,13).

no

| ?- setof(B,q(1,B),L).

L = [3,4,6,7,8,9]

| ?- setof(A,q(A,13),L).

L = [2,3,5,6,7,9,10,11,12]

| ?- p(5,3,3).

yes

| ?- p(2,14,4).

no

| ?- setof(B,p(12,B,3),L).

L = [1,2,4,5,7,9,11,13,15]

| ?- setof(A,p(A,2,2),L).

L = [1,2,4,11]

| ?- setof(A,p(A,A,3),L).

L = [1,3,4,5,6,7,9]

Ασκηση 3.

Για τη βαθμολόγηση των διπλωματικών εργασιών σε κάποιο πανεπιστήμιο ακολουθείται η παρακάτω διαδικασία: τρεις εξεταστές βαθμολογούν την εργασία σε κλίμακα 0 έως 100. Ο μεγαλύτερος από τους τρεις βαθμούς πολλαπλασιάζεται με 25, ο μικρότερος πολλαπλασιάζεται με 30 και ο ενδιάμεσος βαθμός πολλαπλασιάζεται με 45. Τα πολλαπλάσια που προκύπτουν αθροίζονται και το άθροισμα τους διαρείται με 100 ώστε να προκύψει ο σταθμισμένος μέσος όρος (το δεκαδικό μέρος της διαίρεσης αποκόπτεται). Αν δύο από τους εξεταστές έχουν δώσει βαθμολογία μικρότερη από 50 και ο σταθμισμένος μέσος όρος είναι μεγαλύτερος ή ίσος του 50, τότε ο τελικός βαθμός της εργασίας είναι 49. Αν δύο από τους εξεταστές έχουν δώσει βαθμολογία μεγαλύτερη ή ίση του 50 και ο σταθμισμένος μέσος όρος είναι μικρότερος του 50, τότε ο τελικός βαθμός της εργασίας είναι 50. Σε κάθε άλλη περίπτωση ο βαθμός της διπλωματικής εργασίας ισούται με τον σταθμισμένο μέσο όρο.

Γράψτε ένα πρόγραμμα σε Prolog για τον υπολογισμό της βαθμολογίας μίας διπλωματικής εργασίας. Συγκεκριμένα ορίστε ένα κατηγορημα $thesis(A,B,C,S)$, το οποίο θα αληθεύει αν η τιμή της S είναι ο βαθμός της διπλωματικής εργασίας όταν οι τιμές των A,B,C είναι οι βαθμοί των τριών εξεταστών.

Αν κάποια από τις τρεις βαθμολογίες έχει αριθμητική τιμή μικρότερη του 0 ή μεγαλύτερη του 100, τότε η επιστρεφόμενη τιμή θα πρέπει να είναι 0. Δεν μας ενδιαφέρει η συμπεριφορά του κατηγορήματος, όταν κάποιο από τα τρία πρώτα ορίσματα δεν είναι ακέραιος αριθμός.

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

| ?- thesis(45,45,45,S).

S = 45

| ?- thesis(63,71,78,S).

S = 70

| ?- thesis(59,50,64,S).

S = 57

| ?- thesis(83,77,70,S).

S = 76

| ?- thesis(44,35,36,S).

S = 37

| ?- thesis(91,100,87,S).

S = 92

| ?- thesis(41,50,48,S).

S = 46

```
| ?- thesis(47,46,62,S).  
S = 49
```

```
| ?- thesis(45,50,51,S).  
S = 50
```

```
| ?- thesis(42,52,41,S).  
S = 44
```

```
| ?- thesis(38,60,57,S).  
S = 52
```

```
| ?- thesis(101,88,77,S).  
S = 0
```


Ασκηση 4.

Ορίζουμε την παρακάτω αναδρομική συνάρτηση $d : \mathbb{N}^3 \rightarrow \mathbb{N}$:

$$d(i, k, n) = \begin{cases} 0 & \text{αν } k = 0 \\ n & \text{αν } k = 1 \\ d(i \bmod k, k, n) & \text{αν } 2 \leq k \leq i \\ n - \frac{f(0,0,n) \cdot (f(0,0,n)+1)}{2} & \text{αν } k = 2 \text{ και } i = 0 \\ f(0, 0, n) - d(0, 2, n) & \text{αν } k = 2 \text{ και } i = 1 \\ d(0, 2, n) & \text{αν } k \geq 3 \text{ και } i = 0 \\ d(i-1, k-1, d(1, 2, n)) & \text{αν } k \geq 3 \text{ και } 1 \leq i < k \end{cases}$$

όπου η $f : \mathbb{N}^3 \rightarrow \mathbb{N}$ ορίζεται από τον παρακάτω αναδρομικό τύπο:

$$f(x, y, n) = \begin{cases} x - 1 & \text{αν } y > n \\ f(x + 1, y + x + 1, n) & \text{αν } y \leq n \end{cases}$$

Γράψτε ένα πρόγραμμα σε Prolog για τον υπολογισμό της συνάρτησης d . Συγκεκριμένα ορίστε ένα κατηγορήμα $d(I,K,N,D)$, το οποίο θα αληθεύει αν η τιμή της D ισούται με την τιμή της συνάρτησης d με ορίσματα τις τιμές των I,K,N . Δεν μας ενδιαφέρει η συμπεριφορά του κατηγορήματος, όταν κάποιο από τα τρία πρώτα ορίσματα δεν είναι φυσικός αριθμός.

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

```
| ?- d(4,0,132,D) .  
D = 0  
| ?- d(7,1,65,D) .  
D = 65  
| ?- d(1,2,2820,D) .  
D = 29  
| ?- d(0,2,2820,D) .  
D = 45  
| ?- d(0,4,181506,D) .  
D = 3  
| ?- d(3,4,181506,D) .  
D = 5  
| ?- d(9,4,181506,D) .  
D = 4  
| ?- d(42,4,181506,D) .  
D = 2  
| ?- d(1,47,1102,D) .  
D = 4
```

Ασκηση 5.

Μπορούμε να αναπαραστήσουμε ζεύγη φυσικών αριθμών στην Prolog χρησιμοποιώντας το ατομικό όρο 0 και δύο συναρτησιακά σύμβολα με ένα όρισμα, π.χ. τα r και s . Το ζεύγος (m, n) αναπαρίσταται με τον όρο $\underbrace{r(\dots r}_{m}(\underbrace{s(s(\dots s}_{n}(0)\dots))\dots))$. Για παράδειγμα, το ζεύγος $(2, 3)$ αναπαρίσταται με τον όρο $r(r(s(s(s(0))))))$.

Ορίζουμε την πράξη $\oplus : \mathbb{N}^2 \rightarrow \mathbb{N}^2$ έτσι ώστε $(a, b) \oplus (c, d) = (\max(a, c), |b - d|)$.

Γράψτε ένα πρόγραμμα σε Prolog για υλοποίηση της πράξης \oplus . Συγκεκριμένα ορίστε ένα κατηγορήμα `oplus(X,Y,Z)`, το οποίο θα αληθεύει αν X, Y, Z είναι όροι που παριστάνουν ζεύγη σύμφωνα με την αναπαράσταση που περιγράφεται παραπάνω και το ζεύγος που παριστάνεται από τον όρο Z προκύπτει εφαρμόζοντας την πράξη \oplus στα ζεύγη που παριστάνονται από τους όρους X, Y .

Για έλεγχο χρησιμοποιήστε τις παρακάτω τιμές:

```
| ?- oplus(r(s(s(s(s(0))))),r(s(0)),Z).  
Z = r(s(s(s(0))))
```

```
| ?- oplus(r(r(r(r(s(s(0)))))),r(r(r(s(s(s(s(0))))))),Z).  
Z = r(r(r(r(s(s(0))))))
```

```
| ?- oplus(r(r(s(s(s(0))))),r(r(r(r(s(s(s(0))))))),Z).  
Z = r(r(r(r(0))))
```

```
| ?- oplus(r(r(r(0))),s(s(0)),Z).  
Z = r(r(r(s(s(0)))))
```

```
| ?- oplus(0,s(s(0)),Z).  
Z = s(s(0))
```

```
| ?- oplus(r(r(r(0))),0,Z).  
Z = r(r(r(0)))
```

```
| ?- oplus(s(r(0)),r(s(0)),Z).  
no
```