



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικό και Καποδιστριακό
Πανεπιστήμιο Αθηνών

Λογικός Προγραμματισμός

Ασκήσεις

Παναγιώτης Σταματόπουλος

Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Περιεχόμενα

1.	Α' Ομάδα Ασκήσεων "Λογικού Προγραμματισμού" Ακαδημαϊκού Έτους 2007-08.....	3
1.1	Άσκηση.....	3
1.2	Άσκηση 2.....	3
1.3	Άσκηση 3.....	6
1.4	Άσκηση 4.....	7

1. Α' Ομάδα Ασκήσεων "Λογικού Προγραμματισμού" Ακαδημαϊκού Έτους 2007-08

1.1 Άσκηση

Ο ήρωας της μικρής μας ιστορίας διηγείται: "Παντρεύτηκα μία χήρα που είχε μία κόρη. Ο πατέρας μου μας επισκεπτόταν συχνά, γνωρίστηκε με την κόρη της γυναίκας μου, και παντρεύτηκαν. Μήπως είμαι παππούς του εαυτού μου;" Γράψτε πρόγραμμα Prolog και διατυπώστε, επίσης σε Prolog, την κατάλληλη ερώτηση που θα απαντά στο θεμελιώδες υπαρξιακό ερώτημα του φίλου μας. Στο πρόγραμμά σας να διατυπώσετε τη βασική γνώση του κόσμου που δόθηκε προηγουμένως, καθώς και κοινή γενική γνώση που ισχύει στην καθημερινή ζωή. Σ' αυτή την κοινή γνώση, περιλάβετε και ότι το παιδί της/του συζύγου κάποιου/-ας είναι και παιδί του/της ίδιου/-ας (παρότι μπορεί και να μην είναι βιολογικό του/της παιδί). Επίσης, σχεδιάστε και το δέντρο ανάλυσης που προκύπτει κατά τον υπολογισμό της απάντησης στην ερώτηση που πρέπει να υποβληθεί, με βάση το πρόγραμμα που γράψατε, ώστε να αποδειχθεί το ζητούμενο. Για να γίνει σαφής η σύνδεση του δέντρου ανάλυσης που θα σχεδιάσετε με τις προτάσεις του προγράμματός σας, αριθμήστε τις τελευταίες, μέσω σχολίων, στο πρόγραμμά σας.

Παραδοτέο για την άσκηση είναι ένα αρχείο zip, στο οποίο θα περιλαμβάνεται το πηγαίο αρχείο του προγράμματος Prolog που θα γράψετε καθώς και ένα αρχείο εικόνας με το δέντρο ανάλυσης.

1.2 Άσκηση 2

Ο Πολωνός μαθηματικός Waclaw Sierpinski επινόησε και πρότεινε πολλά όμορφα μορφοκλάσματα (fractals). Για παράδειγμα, το χαλί του Sierpinski μπορεί να σχεδιασθεί με επαναλαμβανόμενες εφαρμογές των εξής δύο κανόνων επαναγραφής:

```
      000                XXX
0 --> 000            X --> X0X
      000                XXX
```

Το 0 θεωρήστε ότι παριστάνει τον κενό χαρακτήρα. Αρχίζοντας από ένα X, και εφαρμόζοντας τους παραπάνω κανόνες για δεδομένο πλήθος επαναλήψεων, παίρνουμε διάφορες εκδόσεις του χαλιού. Θεωρώντας ότι οι παραπάνω κανόνες επαναγραφής είναι διατυπωμένοι σαν ένα σύνολο από γεγονότα Prolog της μορφής

```
rewrite(' ', [[' ', ' ', ' '], [' ', ' ', ' '], [' ', ' ', ' ']]).
rewrite('X', [['X', 'X', 'X'], ['X', ' ', 'X'], ['X', 'X', 'X']]).
```

ορίστε σε Prolog ένα κατηγορημα `carpet/1`, το οποίο όταν καλείται με όρισμα το πλήθος εφαρμογών των κανόνων, αρχίζοντας από ένα 'X', να εκτυπώνει το τελικό αποτέλεσμα. Κάποια παραδείγματα εκτέλεσης είναι τα εξής:

```
?- carpet(1).
XXX
X X
```

XXX

?- carpet(2).

XXXXXXXXXX

X XX XX X

XXXXXXXXXX

XXX XXX

X X X X

XXX XXX

XXXXXXXXXX

X XX XX X

XXXXXXXXXX

?- carpet(3).

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

X XX XX XX XX XX XX XX XX X

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXX XXXXXX XXXXXX XXX

X X X XX X X XX X X X

XXX XXXXXX XXXXXX XXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

X XX XX XX XX XX XX XX XX X

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXX XXXXXXXXXXX

X XX XX X X XX XX X

XXXXXXXXXXX XXXXXXXXXXX

XXX XXX XXX XXX

X X X X X X X X

XXX XXX XXX XXX

XXXXXXXXXXX XXXXXXXXXXX

```

X XX XX X          X XX XX X
XXXXXXXXXX          XXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X XX XX XX XX XX XX XX XX X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXX   XXXXXX   XXXXXX   XXX
X X   X XX X   X XX X   X X
XXX   XXXXXX   XXXXXX   XXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X XX XX XX XX XX XX XX XX X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

Το πρόγραμμά σας πρέπει να είναι γενικό και να εκτυπώνει το ζητούμενο για δεδομένους κανόνες επαναγραφής. Τα σύμβολα που επαναγράφονται μέσω των κανόνων πρέπει να είναι τουλάχιστον δύο, τα 'x' και ' ', αλλά δεν απαγορεύεται να υπάρχουν και περισσότερα. Κάθε σύμβολο επαναγράφεται σε ένα, συνήθως τετραγωνικό, αλλά όχι απαραίτητα, διδιάστατο πίνακα συμβόλων (με αναπαράσταση στην Prolog ως λίστα λιστών), για καθένα από τα οποία πρέπει να έχει ορισθεί ακριβώς ένας κανόνας επαναγραφής. Η εφαρμογή των κανόνων αρχίζει πάντα από το σύμβολο 'X'. Σαν ένα δεύτερο παράδειγμα, αν οι κανόνες επαναγραφής ήταν οι

```

rewrite(' ', [[' ', ' ', ' '], [' ', ' ', ' '], [' ', ' ', ' ']]).
rewrite('X', [['X', 'X', 'X'], [' ', 'X', ' '], ['X', 'X', 'X']]).

```

τότε θα έπρεπε να πάρουμε:

```

?- carpet(3).
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
 X  X  X  X  X  X  X  X  X
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
   XXX      XXX      XXX
   X        X        X
   XXX      XXX      XXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
 X  X  X  X  X  X  X  X  X

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXX

X X X

XXXXXXXXXX

XXX

X

XXX

XXXXXXXXXX

X X X

XXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

X X X X X X X X X

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXX XXX XXX

X X X

XXX XXX XXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

X X X X X X X X X

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Παραδοτέο για την άσκηση είναι ένα αρχείο zip, στο οποίο θα περιλαμβάνεται το αρχείο με τον κώδικα του πηγαίου προγράμματος Prolog που θα γράψετε, μέσα στο οποίο δεν θα πρέπει να περιέχονται τα γεγονότα `rewrite/2`. Μπορείτε, αν θέλετε, να συμπεριλάβετε μέσα στο zip αρχείο που θα παραδώσετε και διάφορα αρχεία Prolog με γεγονότα `rewrite/2`, που ορίζουν κανόνες επαναγραφής για τη δημιουργία ενδιαφερόντων μορφοκλασμάτων.

1.3 Άσκηση 3

Θεωρώντας τις συμβολοσειρές με στοιχεία μικρά γράμματα του λατινικού αλφαβήτου (από `a` έως `z`), θέλουμε να έχουμε μία μέθοδο συμπίεσης τέτοιων συμβολοσειρών. Η συμπίεση συνίσταται στην αναγνώριση επαναλαμβανόμενων μορφών στη συμβολοσειρά και στην αντικατάστασή τους με μία πιο συμπαγή μορφή. Για παράδειγμα, αν σε μία συμβολοσειρά έχουμε 7 επαναλήψεις του `p`, δηλαδή `ppppppp`, αυτό το τμήμα της συμβολοσειράς μπορεί να αντικατασταθεί από το `p7`. Αν η επαναλαμβανόμενη μορφή έχει μήκος μεγαλύτερο από ένα χαρακτήρα, πρέπει να χρησιμοποιήσουμε παρενθέσεις. Δηλαδή, το `ababab` πρέπει να γίνει `(ab)3`. Φυσικά, θα θέλαμε η συμπίεση να είναι όσο πιο πλήρης γίνεται, υπό την έννοια ότι το αποτέλεσμα στο οποίο καταλήγουμε να μην μπορεί να συμπιεσθεί περισσότερο. Δηλαδή, το `xcaabaabaabccdadadcaabaabaabcccadady` θα πρέπει να μετασχηματισθεί σε

$x(c(a2b)3c2(ad)2)2y$. Γράψτε ένα κατηγορημα `compress/2`, το οποίο να συμπιέζει τη συμβολοσειρά που του δίνουμε σαν λίστα στο πρώτο όρισμα και να επιστρέφει το αποτέλεσμα σαν λίστα στο δεύτερο όρισμα. Μερικά παραδείγματα εκτέλεσης είναι τα εξής:

?- `compress([a,b,a,b,c,c,c,d,a,d,a,d],L)`.

`L = ['(,a,b,)',2,c,3,'(,d,a,)',2,d]`

?- `compress([a,b,c,d],L)`.

`L = [a,b,c,d]`

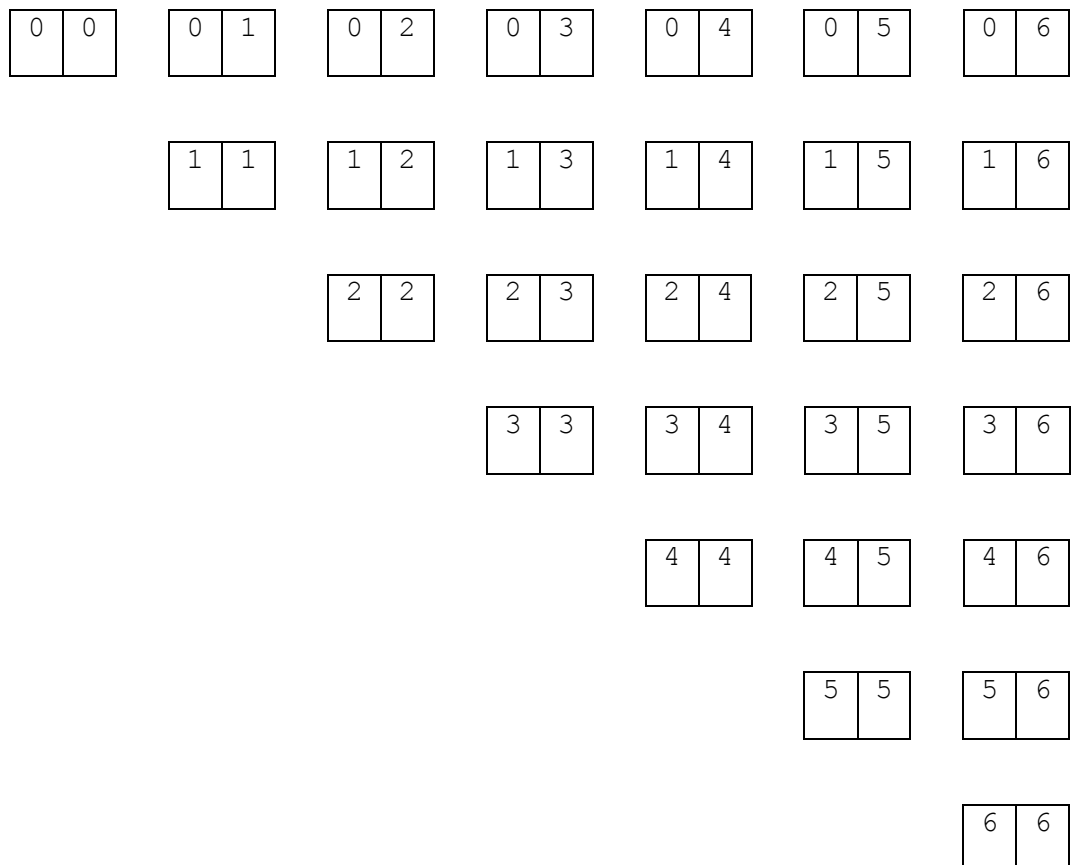
?- `compress([x,c,a,a,b,a,a,b,a,a,b,c,c,a,d,a,d,c,a,a,b,a,a,b, a,a,b,c,c,a,d,a,d,y],L)`.

`L = [x,'(,c,'(,a,2,b,)',3,c,2,'(,a,d,)',2,')',2,y]`

Παραδοτέο για την άσκηση είναι ένα αρχείο που περιλαμβάνει τον πηγαίο κώδικα `Prolog` που θα γράψετε.

1.4 Άσκηση 4

Έστω ότι έχουμε στη διάθεσή μας τις παρακάτω 28 πλάκες ντόμινο:



Γράψτε ένα πρόγραμμα Prolog που να μας πληροφορεί πώς πρέπει να τις τοποθετήσουμε σ' ένα πλαίσιο 7x8, έτσι ώστε η τελική διάταξη να είναι η εξής:

3	1	2	6	6	1	2	2
3	4	1	5	3	0	3	6
5	6	6	1	2	4	5	0
5	6	4	1	3	3	0	0
6	1	0	6	3	2	4	0
4	1	5	2	4	3	5	5
4	1	0	2	4	5	2	0

Μπορείτε να κωδικοποιήσετε μέσα στο πρόγραμμά σας τα δεδομένα του προβλήματος, χωρίς η πρόταση αυτή να είναι δεσμευτική, ως εξής:

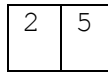
```
dominos([ (0,0), (0,1), (0,2), (0,3), (0,4), (0,5), (0,6),
          (1,1), (1,2), (1,3), (1,4), (1,5), (1,6),
          (2,2), (2,3), (2,4), (2,5), (2,6),
          (3,3), (3,4), (3,5), (3,6),
          (4,4), (4,5), (4,6),
          (5,5), (5,6),
          (6,6) ] ).

frame([[3,1,2,6,6,1,2,2],
       [3,4,1,5,3,0,3,6],
       [5,6,6,1,2,4,5,0],
       [5,6,4,1,3,3,0,0],
       [6,1,0,6,3,2,4,0],
       [4,1,5,2,4,3,5,5],
       [4,1,0,2,4,5,2,0] ] ).
```

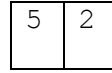
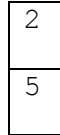
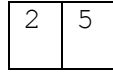
Ως προς τον τρόπο εμφάνισης του αποτελέσματος, μία μη δεσμευτική πρόταση είναι να ορίσετε ένα κατηγορημα `put_dominos/0`, το οποίο όταν καλείται να εκτυπώνει το ζητούμενο. Για παράδειγμα:

```
?- put_dominos.
3-1 2 6 6 1 2-2
  | | | |
3-4 1 5 3 0 3 6
      | |
5 6-6 1 2-4 5 0
|      |
5 6 4 1 3 3-0 0
  | | | |
6 1 0 6 3 2 4 0
|      | | |
4 1-5 2 4 3 5 5
      | |
4-1 0-2 4 5-2 0
```


Σημειώστε ότι ένα ντόμινο μπορεί να τοποθετηθεί στο πλαίσιο με οποιοδήποτε από τους τέσσερις δυνατούς τρόπους. Δηλαδή το ντόμινο



μπορεί να τοποθετηθεί με κάποιον από τους εξής τρόπους:



Παραδοτέο για την άσκηση είναι ένα αρχείο που περιλαμβάνει τον πηγαίο κώδικα Prolog που θα γράψετε.

Σημειώματα

Σημείωμα Ιστορικού Εκδόσεων Έργου

Το παρόν έργο αποτελεί την έκδοση 1.0.

Σημείωμα Αναφοράς

Copyright Εθνικών και Καποδιστριακών Πανεπιστημίων Αθηνών, Παναγιώτης Σταματόπουλος.
«Λογικός Προγραμματισμός, Η γλώσσα προγραμματισμού Prolog». Έκδοση: 1.0. Αθήνα 2015.
Διαθέσιμο από τη δικτυακή διεύθυνση: <http://opencourses.uoa.gr/courses/DI117/>.

Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

Διατήρηση Σημειωμάτων

- Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:
- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων

- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει) μαζί με τους συνοδευόμενους υπερσυνδέσμους.

Σημείωμα Χρήσης Έργων Τρίτων

Το Έργο αυτό κάνει χρήση των ακόλουθων έργων:

Εικόνες/Σχήματα/Διαγράμματα/Φωτογραφίες

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στο πλαίσιο του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Αθηνών**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

