



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 2010-11	3
1.1	Άσκηση 1.....	3
1.2	Άσκηση 2.....	3
1.3	Άσκηση 3.....	4

1. Α' Ομάδα Ασκήσεων "Λογικού Προγραμματισμού" Ακαδημαϊκού Έτους 2010-11

1.1 Άσκηση 1

Το φίδι της Πληροφορικής έχει στο σώμα του ένα επαναλαμβανόμενο σχήμα (όχι, κατ' ανάγκη ακέραιο αριθμό φορών), το οποίο αποτελείται από σχέδια καθένα από τα οποία μπορεί να παρασταθεί από ένα χαρακτήρα. Επίσης, είναι γνωστό ότι το φίδι της Πληροφορικής αρέσκεται να ξεκουράζεται σε μία "φιδοειδή" διευθέτηση καταλαμβάνοντας το χώρο ενός ορθογωνίου παραλληλογράμμου δεδομένων διαστάσεων (σε μονάδα μέτρησης το μέγεθος του σχεδίου/χαρακτήρα που συνθέτει τα επαναλαμβανόμενα σχήματα). Γράψτε ένα κατηγορήμα `di_snake/3`, το οποίο όταν καλείται σαν `di_snake(Pattern, Width, Height)` να εκτυπώνει τη στάση ξεκούρασης του φιδιού της Πληροφορικής, με επαναλαμβανόμενο σχήμα τους χαρακτήρες της λίστας `Pattern` και πλάτος (αντίστοιχα, ύψος) του ορθογωνίου που προαναφέρθηκε το μήκος της λίστας `Width` (αντίστοιχα, `Height`). Το πρόγραμμά σας δεν πρέπει να χρησιμοποιήσει ενσωματωμένα κατηγορήματα της `Prolog`, εκτός από τα απολύτως απαραίτητα για την έξοδο του αποτελέσματος, δηλαδή τα `write/1` και `nl/0`. Ένα παράδειγμα εκτέλεσης του προγράμματος είναι το εξής:

```
?- di_snake([a,b,c,d,e,f,g],
            ['_','_','_','_','_','_','_','_','_','_'],
            ['_','_','_','_','_']).
abcdefgabcd
agfedcbagfe
bcdefgabcde
bagfedcbagf
cdefgabcdef
cbagfedcbag

?- di_snake([0,1,2,3,4,5,6,7,8,9],
            ['_','_','_','_','_','_','_','_','_','_','_','_','_','_','_','_'],
            ['_','_','_','_','_','_','_','_','_','_','_','_','_','_','_','_']).
01234567890123456789012345678
76543210987654321098765432109
89012345678901234567890123456
54321098765432109876543210987
67890123456789012345678901234
32109876543210987654321098765
45678901234567890123456789012
10987654321098765432109876543
23456789012345678901234567890
98765432109876543210987654321
01234567890123456789012345678
76543210987654321098765432109
89012345678901234567890123456
54321098765432109876543210987
67890123456789012345678901234
32109876543210987654321098765
45678901234567890123456789012
10987654321098765432109876543
23456789012345678901234567890
```

Παραδοτέο για την άσκηση είναι ένα πηγαίο αρχείο `Prolog`.

1.2 Άσκηση 2

Φανταστείτε ότι σε ένα δωμάτιο έχετε έναν αριθμό τηλεοράσεων, όπου κάθε μία είναι δυνατόν να συντονιστεί σε ένα από N διαθέσιμα κανάλια, έστω τα $1, 2, \dots, N$. Υποθέστε ότι αρχικά οι τηλεοράσεις είναι συντονισμένες σε αυθαίρετα κανάλια και ότι θέλετε να τις συντονίσετε όλες στο ίδιο κανάλι. Έχετε στη διάθεσή σας ένα τηλεχειριστήριο του οποίου η μοναδική λειτουργία είναι να μπορείτε να συντονίσετε οποιαδήποτε τηλεόραση στο επόμενο κανάλι από αυτό που είναι συντονισμένη (το επόμενο κανάλι από το N είναι το 1), αλλά δεν μπορείτε να κάνετε αυτήν την ενέργεια δύο συνεχόμενες φορές στην ίδια τηλεόραση, αν δεν μεσολαβήσει η αλλαγή καναλιού τουλάχιστον κάποιας άλλης. Επίσης, δεν είναι δυνατόν σε μία τηλεόραση να γίνουν N ή περισσότερες αλλαγές καναλιού, δηλαδή, δεν μπορεί να επιστρέψει στο αρχικό κανάλι της. Ορίστε ένα κατηγορημα `remote/4`, το οποίο, όταν καλείται με πρώτο όρισμα μία λίστα που περιέχει τα κανάλια στα οποία είναι συντονισμένες αρχικά οι τηλεοράσεις μας και με δεύτερο όρισμα το πλήθος των διαθέσιμων καναλιών, να επιστρέφει στο τρίτο όρισμα μία λίστα με τους αύξοντες αριθμούς των τηλεοράσεων στις οποίες όταν εφαρμοσθεί διαδοχικά η λειτουργία του τηλεχειριστηρίου, να έχουμε σαν αποτέλεσμα όλες οι τηλεοράσεις τελικά να είναι συντονισμένες στο ίδιο κανάλι, ο αριθμός του οποίου να επιστρέφεται στο τέταρτο όρισμα του κατηγορηματος που θα γράψετε. Μας είναι αδιάφορο ποιο θα είναι το τελικό κανάλι συντονισμού, αλλά μας ενδιαφέρει να κάνουμε τον ελάχιστο αριθμό αλλαγών σε κανάλια. Κάποια παραδείγματα εκτέλεσης θα μπορούσαν να ήταν τα εξής:

```
?- remote([3,1,3,3],5,L,C).
L = [1,2,3,2,4,2]
C = 4
```

```
?- remote([2,2],3,L,C).
L = []
C = 2
```

```
?- remote([2,5,1,6,4,2,6],8,L,C).
L = [1,2,1,3,1,3,1,3,5,6,3,6,3,6,5,6]
C = 6
```

```
?- remote([1,2,3,4,3,2,1,2,3,4,3,2,1,2,3,4,3,2,1,2,3,4],
          5,L,C).
L = [1,2,1,4,6,4,7,4,7,4,8,10,12,10,13,10,13,10,14,16,18,16,
     19,16,22,16,22,19,22,20,22]
C = 3
```

Παραδοτέο για την άσκηση είναι ένα πηγαίο αρχείο Prolog.

1.3 Άσκηση 3

Να ορισθεί ένα κατηγορημα `differentiate/2` το οποίο να δέχεται στο πρώτο όρισμά του μία αλγεβρική έκφραση που μπορεί να περιλαμβάνει γινόμενα, αθροίσματα και διαφορές μεταξύ πολυωνύμων μίας μεταβλητής (έστω x) και να βρίσκει την παράγωγο αυτής της έκφρασης, επιστρέφοντας το αποτέλεσμα στο δεύτερο όρισμα σαν το απλούστερο δυνατό πολυώνυμο. Για παράδειγμα:

```
?- differentiate((2*x^2-3*x+2)*(3*x-4)-(4*x^2+18*x-7),P).
P = 18 * x ^ 2 - 42 * x
```

```
?- differentiate((2*x-3*x^3)-(3*x-x^2+1)*(-x-1)*2+x,P).
P = -15 * x ^ 2 + 8 * x + 11
```

Παραδοτέο για την άσκηση είναι ένα πηγαίο αρχείο Prolog.

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.

