



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικό και Καποδιστριακό
Πανεπιστήμιο Αθηνών

Λογικός Προγραμματισμός

Ασκήσεις

Παναγιώτης Σταματόπουλος

Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Περιεχόμενα

1.	Ασκήσεις "Λογικού Προγραμματισμού" Ακαδημαϊκού Έτους 2005-06.....	3
1.1	Άσκηση 1 (0.3 μονάδες)	3
1.2	Άσκηση 2 (0.4 μονάδες)	3
1.3	Άσκηση 3 (0.4 μονάδες)	4
1.4	Άσκηση 4 (0.5 μονάδες)	4
1.5	Άσκηση 5 (0.3 μονάδες)	8
1.6	Άσκηση 6 (0.6 μονάδες)	8

1. Ασκήσεις "Λογικού Προγραμματισμού" Ακαδημαϊκού Έτους 2005-06

1.1 Άσκηση 1 (0.3 μονάδες)

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 κατηγορημα `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 καθόλου αριθμητική, δηλαδή δεν θα περιλαμβάνει κανένα από τα σύμβολα `is`, `<`, `>`, `>=`, `=<`, `+`, `-`, `*`, αλλά ούτε και τα `name`, `=..`, `arg`, `functor`. Ένα παράδειγμα εκτέλεσης του προγράμματος είναι το εξής:

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

yes

1.2 Άσκηση 2 (0.4 μονάδες)

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. Ορίστε ένα κατηγορημα `remote/3` 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. 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).
```

```
L = [1,2,3,2,4,2]
```

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

```
L = []
```

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

```
L = [1,2,1,3,1,3,1,3,5,6,5,6,3,6,3,6]
```

1.3 Άσκηση 3 (0.4 μονάδες)

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
```

1.4 Άσκηση 4 (0.5 μονάδες)

Έστω ότι δίνεται η παρακάτω Prolog βάση δεδομένων:

```
borders(belgium, france).  
borders(belgium, germany).  
borders(belgium, luxembourg).  
borders(belgium, netherlands).  
borders(denmark, germany).  
borders(france, belgium).  
borders(france, germany).  
borders(france, italy).  
borders(france, luxembourg).  
borders(france, spain).  
borders(germany, belgium).  
borders(germany, denmark).  
borders(germany, france).  
borders(germany, luxembourg).  
borders(germany, netherlands).  
borders(great_britain, ireland).
```

borders(ireland, great_britain).
borders(italy, france).
borders(luxembourg, belgium).
borders(luxembourg, france).
borders(luxembourg, germany).
borders(netherlands, belgium).
borders(netherlands, germany).
borders(portugal, spain).
borders(spain, france).
borders(spain, portugal).

flows(achelous, greece).
flows(aliakmon, greece).
flows(danube, germany).
flows(ebro, spain).
flows(elbe, germany).
flows(garonne, france).
flows(garonne, spain).
flows(loire, france).
flows(maas, belgium).
flows(maas, france).
flows(maas, netherlands).
flows(po, italy).
flows(rhine, france).
flows(rhine, germany).
flows(rhine, netherlands).
flows(rhone, france).
flows(seine, france).
flows(tagus, portugal).
flows(tagus, spain).

```
flows(thames, great_britain).
flows(tiber, italy).

belongs(brussels, belgium).
belongs(copenhagen, denmark).
belongs(lyon, france).
belongs(marseille, france).
belongs(paris, france).
belongs(berlin, germany).
belongs(hamburg, germany).
belongs(munich, germany).
belongs(birmingham, great_britain).
belongs(glasgow, great_britain).
belongs(london, great_britain).
belongs(athens, greece).
belongs(salonika, greece).
belongs(dublin, ireland).
belongs(milan, italy).
belongs(naples, italy).
belongs(rome, italy).
belongs(luxembourg, luxembourg).
belongs(amsterdam, netherlands).
belongs(hague, netherlands).
belongs(rotterdam, netherlands).
belongs(lisbon, portugal).
belongs(madrid, spain).
belongs(barcelona, spain).
```

Τα κατηγορήματα `borders/2`, `flows/2` και `belongs/2` έχουν την προφανή σημασία. Ορίστε ένα κατηγορήμα `answer/2` που να είναι σε θέση να βρίσκει τις κατάλληλες απαντήσεις σε ερωτήσεις που είναι διατυπωμένες σε απλή φυσική Ελληνική γλώσσα¹. Παραδείγματα:

```
?- answer('poies xwres synoreuoun me thn Ispania;',L).
```

```
L = [france,portugal]
```

```
?- answer('me poies xwres synoreuei h Gallia;',L).
```

```
L = [belgium,germany,italy,luxembourg,spain]
```

```
?- answer('poia potamia diarreoun thn Ellada;',L).
```

```
L = [achelous,aliakmon]
```

```
?- answer('poies xwres diarreei o Rhnos;',L).
```

```
L = [france,germany,netherlands]
```

```
?- answer('poies poleis briskontai sthn Ollandia;',L).
```

```
L = [amsterdam,hague,rotterdam]
```

```
?- answer('se poia xwra brisketai to Milano;',L).
```

```
L = [italy]
```

```
?- answer('me poies xwres tis opoies diarreei o Tagos  
synoreuei h Gallia;',L).
```

```
L = [spain]
```

```
?- answer('poies xwres oi opoies synoreuoun me to Louxembourgo  
synoreuoun me thn Italia;',L).
```

¹ Θα βοηθούσε ιδιαίτερα αν χρησιμοποιούσατε τη δυνατότητα που προσφέρουν διάφορα συστήματα Prolog (LPA Prolog, ECLiPS^e, κλπ.) για την περιγραφή γλωσσών (φυσικών και άλλων) μέσω μίας γραμματικής οριστικών προτάσεων (definite clause grammar). Ανατρέξτε στο on-line help ή ζητήστε μου τα εγχειρίδια για περισσότερες λεπτομέρειες.

```
L = [france]
```

```
?- answer('poia xwra sthn opoia brisketai h Glaskwbh  
diarreei o Tameshs;',L).
```

```
L = [great_britain]
```

```
?- answer('poio potami to opoio diarreei to Belgio  
diarreei thn Ollandia;',L).
```

```
L = [maas]
```

1.5 Άσκηση² 5 (0.3 μονάδες)

Το πρόβλημα του χρωματισμού γράφου συνίσταται στην εύρεση του ελάχιστου αριθμού χρωμάτων που πρέπει να χρησιμοποιήσουμε για να χρωματίσουμε τους κόμβους δεδομένου γράφου, έτσι ώστε να μην υπάρχει ζευγάρι γειτονικών κόμβων που να έχουν το ίδιο χρώμα. Επιλύστε το πρόβλημα αυτό για γράφους με N κόμβους και πυκνότητα D , που θα κατασκευάζετε μέσω του `create_graph(N,D,G)` το οποίο θα βρείτε στο αρχείο <http://www.di.uoa.gr/~takis/graph.pl>

1.6 Άσκηση³ 6 (0.6 μονάδες)

Το πρόβλημα της πρόβας ορχήστρας συνίσταται στην εύρεση της σειράς με την οποία πρέπει να γίνει η πρόβα των διαφόρων μερών ενός κονσέρτου που πρόκειται να δώσει μία ορχήστρα, έτσι ώστε να ελαχιστοποιηθεί ο συνολικός χρόνος αναμονής των μουσικών της ορχήστρας. Κάθε μουσικός συμμετέχει σε συγκεκριμένα μέρη του κονσέρτου και αφού αποφασισθεί η σειρά με την οποία θα γίνει η πρόβα των μερών, ο μουσικός οφείλει να είναι παρών όταν αρχίζει η πρόβα του πρώτου μέρους στο οποίο συμμετέχει και αποδεσμεύεται όταν τελειώσει η πρόβα του τελευταίου μέρους στο οποίο συμμετέχει. Οι μουσικοί αμείβονται ανάλογα με τον χρόνο που τους ζητείται να είναι παρόντες στην πρόβα, πράγμα που σημαίνει ότι ενδιαφερόμαστε να ελαχιστοποιήσουμε, όσο είναι δυνατόν, τα χρονικά διαστήματα που οι μουσικοί είναι παρόντες, αλλά γίνονται πρόβες μερών στα οποία δεν συμμετέχουν. Επίσης, για κάθε μέρος του κονσέρτου, είναι γνωστή η διάρκειά του. Δείτε ένα παράδειγμα στον παρακάτω πίνακα:

Μέρος	1	2	3	4	5	6	7	8	9
Μουσικός 1	1	1	0	1	0	1	1	0	1
Μουσικός 2	1	1	0	1	1	1	0	1	0
Μουσικός 3	1	1	0	0	0	0	1	1	0
Μουσικός 4	1	0	0	0	1	1	0	0	1
Μουσικός 5	0	0	1	0	1	1	1	1	0
Διάρκεια	2	4	1	3	3	2	5	7	6

Στο παράδειγμα αυτό, το κονσέρτο αποτελείται από 9 μέρη και συμμετέχουν σ' αυτό 5 μουσικοί. Η διάρκεια κάθε μέρους φαίνεται στην τελευταία γραμμή του πίνακα. Τα 1 και 0 στον πίνακα δείχνουν αν ο κάθε μουσικός συμμετέχει ή όχι στο αντίστοιχο μέρος του κονσέρτου. Αν οι πρόβες των μερών γινότουσαν με τη σειρά του πίνακα, τότε οι χρόνοι αναμονής των μουσικών θα ήταν 11 (=1+3+7), 6 (=1+5), 9 (=1+3+3+2), 20 (=4+1+3+5+7) και 3, αντίστοιχα, δίνοντας ένα συνολικό χρόνο αναμονής για τους μουσικούς ίσο με 49 (=11+6+9+20+3). Υπάρχουν όμως και καλύτερες λύσεις

² Να αντιμετωπισθεί με τη βοήθεια της τεχνολογίας του λογικού προγραμματισμού με περιορισμούς, π.χ. στο περιβάλλον της γλώσσας ECLiPS^e (με χρήση είτε της παλιότερης βιβλιοθήκης `fd` είτε της νεότερης `ic`).

³ Όπως και η Άσκηση 5.

απ' αυτήν για τη σειρά των μερών στην πρόβα. Ορίστε ένα κατηγορημα `rehearsal/2`, το οποίο όταν καλείται σαν `rehearsal(Sequence, WaitTime)` να επιστρέφει στο `Sequence` τη βέλτιστη σειρά πρόβας των μερών ενός κονσέρτου και στο `WaitTime` τον συνολικό χρόνο αναμονής των μουσικών γι' αυτή τη βέλτιστη σειρά. Τα δεδομένα του προβλήματος να δίνονται υπό τη μορφή δύο γεγονότων `players/1` και `durations/1`, όπως φαίνεται στο http://www.di.uoa.gr/~takis/rehearsal_data.pl.

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.

