

Λογικός Προγραμματισμός

Εργασία 3 P19032 Γεωργιάδης Νικόλαος

Εκφώνηση εργασίας που αντιστοιχεί στο επώνυμο:

[Για φοιτητές με επώνυμο από Α έως Δ]: Υλοποιήστε σε Prolog πρόγραμμα εύρεσης λύσης στο παρακάτω πρόβλημα. Υπάρχουν πέντε διαδοχικά γραφεία, κάθε ένα με διαφορετικό χρώμα. Ο άνθρωπος που εργάζεται σε κάθε γραφείο είναι από διαφορετικό τμήμα, έχει διαφορετικό τύπο υπολογιστή, πίνει διαφορετικό ποτό και έχει διαφορετικό κινητό. Ποιος εργαζόμενος έχει windows laptop; Ποιός εργαζόμενος πίνει νερό;

1. Ο εργαζόμενος από το τμήμα πληροφορικής έχει το κόκκινο γραφείο.
2. Ο εργαζόμενος από το τμήμα οργάνωσης και διοίκησης έχει macbook pro.
3. Ο εργαζόμενος στο πράσινο γραφείο πίνει καφέ.
4. Ο εργαζόμενος από το οικονομικό τμήμα πίνει τσάι.
5. Το πράσινο γραφείο είναι στα δεξιά του μπεζ γραφείου.
6. Ο εργαζόμενος που έχει iphone έχει και macbook air.
7. Ο εργαζόμενος που έχει blackberry έχει το κίτρινο γραφείο.
8. Γάλα πίνει ο εργαζόμενος στο μεσαίο γραφείο.
9. Ο εργαζόμενος από το τμήμα διεθνών σχέσεων έχει το πρώτο γραφείο (από αριστερά).
10. Ο εργαζόμενος που έχει nokia έχει το γραφείο που βρίσκεται δίπλα από το γραφείο του εργαζόμενου με το netbook.
11. Ο εργαζόμενος με το blackberry βρίσκεται δίπλα από τον εργαζόμενο που έχει windows 7 laptop.
12. Ο εργαζόμενος που έχει android πίνει πορτοκαλάδα.
13. Ο εργαζόμενος από το τμήμα προμηθειών έχει ericsson.
14. Ο εργαζόμενος από το τμήμα διεθνών σχέσεων έχει γραφείο που είναι δίπλα στο μπλε γραφείο.

Το αρχείο κώδικα είναι το ακόλουθο:

%We use this list to fill it with the humans. Their characteristics are
%being described in each tuple.

%When L is given, we are making a L-size list recursively. Each
%ΣΣelement location correspond to the location of the human.

humans(0, []) :- !.

humans(L, [(Department, Table, Phone, Pc, Beverage)|T]) :- L1 is L-1,
humans(L1,T).

%We use these, in order to get the L'th element(human) from a list.

%If we want the first human we return the head of the list.

%In any other case, we take the Tail of the list and we are checking

%recursively if the desired human is at the head of the tail.

human(1, [H|_], H) :- !.

human(L, [_|T], Hum) :- L1 is L-1, human(L1, T, Hum).

%%HINTS%%

%The main idea here is that we are iterating lists of tuples to see
%what location does each human may have in the list, and what
%characteristics. We store and check data from each tuple.

%We have plenty types of search/hints:

%1)We know two things about a human(except location) and we are
%trying to find on who these things correspond(location in list), in
%addition to the constraints that are being made during execution.

%2)We know the exact location of a human who has a specific
%characteristic. We tell the program to put him at the correct
%location with the help of human(L, [_|T], Hum) and
%human(1, %[H|_], H).

%3)We know that two humans with different characteristics are next
%to each other. So we tell the program to search and find them
%ONLY next to each other from the list.

%1. Ο εργαζόμενος από το τμήμα πληροφορικής έχει το κόκκινο
%γραφείο.

hint1([(informatics,red,_, _)|_]).

hint1([_|T]) :- hint1(T).

%2. Ο εργαζόμενος από το τμήμα οργάνωσης και διοίκησης έχει
%macbook pro.

hint2([(organization_and_administration,_,_,macbook_pro,_)|_]).

hint2([_|T]) :- hint2(T).

%3. Ο εργαζόμενος στο πράσινο γραφείο πίνει καφέ.

hint3([(_,green,_,_,coffe)|_]).

hint3([_|T]) :- hint3(T).

%4. Ο εργαζόμενος από το οικονομικό τμήμα πίνει τσάι.

hint4([(iconomics,_,_,_,tea)|_]).

hint4([_|T]) :- hint4(T).

%5. Το πράσινο γραφείο είναι στα δεξιά του μπεζ γραφείου.

hint5([(_,beize,_,_,_),(_,green,_,_,_)|_]).

hint5([_|T]) :- hint5(T).

%6. Ο εργαζόμενος που έχει iphone έχει και macbook air.

hint6([(_,_,iphone,macbook_air,_)|_]).

hint6([_|T]) :- hint6(T).

%7. Ο εργαζόμενος που έχει blackberry έχει το κίτρινο γραφείο.

hint7([(_,yellow,blackberry,_,_)|_]).

hint7([_|T]) :- hint7(T).

%8. Γάλα πίνει ο εργαζόμενος στο μεσαίο γραφείο.

hint8(Humans):-

human(3, Humans, (_,_,_,milk)).

%9. Ο εργαζόμενος από το τμήμα διεθνών σχέσεων έχει το πρώτο %γραφείο (από αριστερά).

hint9(Humans):-

human(1, Humans, (international_relations,_,_,_)).

%10. Ο εργαζόμενος που έχει nokia έχει το γραφείο που βρίσκεται δίπλα από το γραφείο του εργαζόμενου με το netbook.

hint10([(_,_,nokia,_,_),(_,_,_,netbook,_)|_]).

hint10([((_,_,_,netbook,_),(_,_,nokia,_,_)|_]).

hint10([_|T]) :- hint10(T).

%11. Ο εργαζόμενος με το blackberry βρίσκεται δίπλα από τον εργαζόμενο που έχει windows 7 laptop.

hint11([(_,_,blackberry,_,_),(_,_,_,windows_7,_)|_]).

hint11([((_,_,_,windows_7,_),(_,_,blackberry,_,_)|_]).

hint11([_|T]) :- hint11(T).

%12. Ο εργαζόμενος που έχει android πίνει πορτοκαλάδα.

hint12([(_,_,android,_,_,orange_juice)|_]).

hint12([_|T]) :- hint12(T).

%13. Ο εργαζόμενος από το τμήμα προμηθειών έχει ericsson.

hint13([(supplies,_,ericson,_,_)|_]).

hint13([_|T]) :- hint13(T).

%14. Ο εργαζόμενος από το τμήμα διεθνών σχέσεων έχει γραφείο
%που είναι δίπλα στο μπλε γραφείο.

hint14([(blue,_,_,_), (international_relations,_,_,_) | _]).

hint14([(international_relations,_,_,_), (blue,_,_,_) | _]).

hint14([_ | T]) :- hint14(T).

%%QUESTIONS%%

%When the whole execution of the hints finishes, we have a
%structured list with the desired values missing from the humans.

%Then, we add the missing values(they are obviously the answers)

% Ποιός εργαζόμενος πίνει νερό;

question1([(_,_,_,water) | _]).

question1([_ | T]) :- question1(T).

% Ποιος εργαζόμενος έχει windows xp laptop;

question2([(_,_,windows_xp,_) | _]).

question2([_ | T]) :- question2(T).

%%%%%%%%%%%%%%SOLUTION%%%%%%%%%%%%%%

solution(Humans) :-

humans(5, Humans), %making recursively the starting list (length=5)

hint1(Humans), %executing all the hints

hint2(Humans),

hint3(Humans),

hint4(Humans),

hint5(Humans),

hint6(Humans),

hint7(Humans),

hint8(Humans),

hint9(Humans),

hint10(Humans),

hint11(Humans),

hint12(Humans),

hint13(Humans),

hint14(Humans),

question1(Humans), %at the end, all the constraints are met,

question2(Humans). %and we can find the answers

Κατά την εκτέλεση του προγράμματος, όταν εκτελούμε το κατηγορήμα solutions(Humans) παίρνουμε το ακόλουθο αποτέλεσμα:

```
Humans = [(international_relations, yellow, blackberry, netbook, water),  
(iconomics, blue, nokia, windows_7, tea), (informatics, red, iphone,  
macbook_air, milk), (organization_and_administration, beize, android,  
macbook_pro, orange_juice), (supplies, green, ericson, windows_xp, coffe)]
```

Όπως παρατηρούμε νερό πίνει ο εργαζόμενος από το τμήμα διεθνών σχέσεων και windows_xp έχει ο εργαζόμενος από το τμήμα προμηθειών.

Ακολουθεί μία σύντομη περιγραφή του κώδικα(υπάρχουν σχόλια με εκτενέστερη επεξήγηση).

Ο κώδικας έχει τέσσερα κομμάτια. Το πρώτο κομμάτι αποτελεί 2 βασικές δηλώσεις λιστών, για τον έλεγχο του περιεχομένου της λίστας Humans που δημιουργείται κατά την εκτέλεση. Η δήλωση humans αρχειοθετεί την λίστα Humans αναδρομικά με μεταβλητές τιμές κατά την κλήση. Η δήλωση human προσθέτει στοιχεία σε ένα συγκεκριμένο tuple που θα της πούμε μέσα στη λίστα.

Το δεύτερο κομμάτι περιέχει τα Hints τις εκφώνησης, τα οποία είναι απαραίτητα καθώς αποτελούν τους κανόνες μας. Ανάλογα τον κανόνα, η λίστα Humans αρχειοθετείται ανάλογα κατά την εκτέλεση. Υπάρχουν 3 τύποι κανόνων/hints (βλ. σχόλια).

Το τρίτο κομμάτι περιέχει τους κανόνες που καλούνται στο τέλος(question1, question2), και είναι υπεύθυνοι για την συμπλήρωση των τελευταίων χαρακτηριστικών των tuples που λείπουν(αυτά που μας δείχνουν ποιος πίνει νερό και ποιος έχει τα windows_xp).

Τέλος, στο τέταρτο κομμάτι βρίσκεται το κατηγορημα solution που παίρνει σαν όρισμα την λίστα Humans. Η λίστα αρχειοθετείται με τις αρχικές τις τιμές στο σώμα του solution και περνάει σαν όρισμα σε όλα τα hints/questions για να δομηθεί σωστά.