

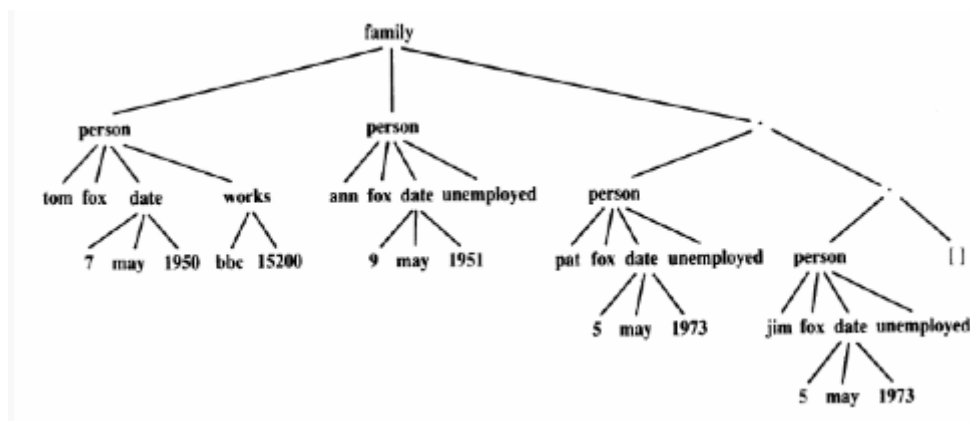
Λογικός Προγραμματισμός

Εργασία 1 P19032 Γεωργιάδης Νικόλαος

Γενική εκφώνηση εργασίας:

Έστω ότι ένα δημαρχείο έχει μία βάση γνώσης σε Prolog στην οποία αποθηκεύει πληροφορίες για τις οικογένειες που είναι εγγεγραμμένες στο δημοτολόγιό του. Το παρακάτω σχήμα δείχνει πως μπορούν να δομηθούν οι πληροφορίες για κάθε οικογένεια. Κάθε οικογένεια αποθηκεύεται σαν μία δήλωση. Προσέξτε ότι επειδή ο αριθμός των παιδιών δεν είναι ο ίδιος για κάθε οικογένεια χρησιμοποιείται λίστα που μπορεί να χειριστεί οποιονδήποτε αριθμό παιδιών. (Κάποια ερωτήματα απαιτούν χρήση αριθμητικών τελεστών που παρουσιάζονται στη σελίδα 214 των διαφανειών.)

Να αναπαρασταθεί η οικογένεια καθώς και 1-2 οικογένειες ακόμη (δική σας επιλογής) χρησιμοποιώντας το κατηγορημα family/3 και κατάλληλους σύνθετους όρους.



Εκφώνηση εργασίας που αντιστοιχεί στο επώνυμο:

[Για φοιτητές με επώνυμο από Α έως Ε] Ανακτήστε όλες τις γυναίκες που έχουν τουλάχιστον τρία παιδιά. Ανακτήστε τα ονόματα των οικογενειών με τουλάχιστον τρία παιδιά.

Στο αρχείο «ergasia1.pl» βρίσκεται ο ζητούμενος κώδικας της εκφώνησης.

Έχουμε δημιουργήσει 4 οικογένειες. Η κάθε οικογένεια έχει από 2 έως και 4 παιδιά. Το κατηγορήμα family εκφράζει την οικογένεια και έχει τα αντίστοιχα 3 ορίσματα: πατέρας, μητέρα, παιδιά(σε λίστα).

Το κάθε άτομο αποτελεί επίσης ένα κατηγορήμα(person) και περιέχει και αυτό με τη σειρά του ορίσματα, τους εξής σύνθετους όρους: όνομα, επίθετο, φύλλο, ημερομηνία γέννησης και θέση εργασίας.

Το κατηγορήμα children που είναι το τρίτο όρισμα του κατηγορήματος family, περιέχει σαν μοναδικό όρισμα μία λίστα με τα παιδιά της οικογένειας. Το κάθε παιδί της λίστας αποτελεί ένα person του οποίου η δομή επεξηγήθηκε προηγουμένως.

Να σημειωθεί, πως στο τέλος του αρχείου έχουμε προσθέσει μερικές δηλώσεις παραπάνω που είναι απαραίτητες για την απάντηση του δεύτερου μισού της εκφώνησης.

Ως παράδειγμα, ας δούμε πως συντάσσεται η εξής οικογένεια:

Όνομα πατέρα: harry jones

Φύλο: male

Ημερομηνία γέννησης: 4 march 1980

Θέση εργασίας: centrica

Όνομα μητέρας: mary jones

Φύλο: female

Ημερομηνία γέννησης: 17 december 1976

Θέση εργασίας: Vodafone

Όνομα παιδιού: ken jones

Φύλο: male

Ημερομηνία γέννησης: 6 may 2003

Θέση εργασίας: unemployed

Όνομα παιδιού: lori jones

Φύλο: female

Ημερομηνία γέννησης: 17 november 2009

Θέση εργασίας: unemployed

```

family(
  person(
    firstname(harry),
    lastname(jones),
    gender(male),
    dob(04,march,1980),
    works(centrica)
  ),
  person(
    firstname(mary),
    lastname(jones),
    gender(female),
    dob(17,december,1976),
    works(vodafone)
  ),
  children([
    person(
      firstname(ken),
      lastname(jones),
      gender(male),
      dob(6,may,2003),
      unemployed
    ),
    person(
      firstname(lori),
      lastname(jones),
      gender(female),
      dob(17,november,2009),
      unemployed
    )
  ])
).

```

Παρακάτω εκτελούμε τα κατάλληλα ερωτήματα στον διερμηνευτή SWI για να απαντήσουμε στα ζητούμενα της εκφώνησης:

- Ανακτήστε όλες τις γυναίκες που έχουν τουλάχιστον τρία παιδιά

```
?- family(_person(F, L, gender(female), D, W), children([_, _, _|_])).
F = firstname(crystal),
L = lastname(lee),
D = dob(21, september, 1978),
W = works(tesco) ;
F = firstname(tracy),
L = lastname(young),
D = dob(23, december, 1970),
W = works(aviva) ;
false.

?- ■
```

Μέσω της εντολής:

```
?- family(_person(F, L, gender(female), D, W), children([_, _, _|_])).
```

αναζητούμε τις μητέρες των οικογενειών ανεξαρτήτως πατέρα(για αυτό έχουμε χρησιμοποιήσει την ανώνυμη μεταβλητή “_” σαν όρισμα στον πατέρα) με τουλάχιστον 3 παιδιά. Οι 3 πρώτες γεμισμένες θέσεις πριν την ουρά(όρισμα μετά το “|”) της λίστας children, εξασφαλίζουν ότι στο αποτέλεσμα θα υπάρχουν τουλάχιστον 3 παιδιά(η ουρά μπορεί να είναι και η κενή λίστα) τα οποία δεν μας αφορούν ποια είναι απαραίτητα. Μέσω των μεταβλητών F,L,D και W επιστρέφεται το όνομα, το επίθετο, η ημερομηνία γέννησης και η θέση εργασίας αντίστοιχα των μητέρων.

Αν παρατηρήσουμε το αρχείο «ergasia1.pl», πράγματι η crystal lee και η tracy young έχουν τουλάχιστον 3 παιδιά.

- **Ανακτήστε τα ονόματα των οικογενειών με τουλάχιστον τρία παιδιά.**

Εδώ αναζητάμε όλα τα ονόματα των μελών κάθε οικογένειας που έχουν τουλάχιστον 3 παιδιά. Χρησιμοποιώντας αναδρομή, επιτυγχάνουμε την εύρεση των ονομάτων δοσμένης μίας λίστας με παιδιά.

```
print([]).
```

```
print(R):-
```

```
    [person(CFn, _, _, _) | Rest]=R,  
    write("\n"),  
    write(CFn),  
    print(Rest).
```

Οι παραπάνω δηλώσεις λειτουργούν ως εξής:

Η κάτω δήλωση `print` παίρνει σαν όρισμα τη λίστα `R` με τα παιδιά, εκτυπώνει το όνομα του παιδιού που βρίσκεται στην κεφαλή και μετά συνεχίζει αυτή τη διαδικασία αναδρομικά στην ουρά της λίστας. Όταν πια θα ισχύει `Rest=[]`, θα επιστραφεί `true` λόγω της πρώτης δήλωσης, οπότε σταματάει η αναδρομή.

get_names():-

```
family(person(F1, L, gender(male), _, _), person(F2, L, gender(female), _, _),
children([person(CF1, L, _, _, _), person(CF2, L, _, _, _), person(CF3, L, _, _, _) | Rest])),
write("family: "), write(L),
write("\nfather: "), write(F1),
write("\nmother: "), write(F2),
write("\nchildren:\n"),
write(CF1),
write("\n"),
write(CF2),
write("\n"),
write(CF3),
print(Rest).
```

Η παραπάνω δήλωση είναι αυτή που θα εκτελέσουμε στον διερμηνευτή για να μας εκτυπωθούν τα ονόματα των οικογενειών. Έχουμε τοποθετήσει μεταβλητές στα σημεία των ονομάτων διότι θέλουμε να τις χρησιμοποιήσουμε στην δήλωση write. Επίσης, όπως και στο προηγούμενο ερώτημα έτσι και εδώ, για να εκφράσουμε το ζητούμενο “3 τουλάχιστον παιδιά” είναι αναγκαίο να συμπληρώσουμε 3 εκ αυτών στη λίστα, μαζί με την ουρά(μπορεί να είναι και κενή). Αφού μας εκτυπωθεί το επίθετο της οικογένειας, το όνομα της μητέρας, το όνομα του πατέρα και των 3^{ων} πρώτων παιδιών, ξεκινάει η αναδρομική διαδικασία για τα υπόλοιπα παιδιά στην ουρά της λίστας, όπως εξηγήσαμε παραπάνω.

Ακολουθεί η παρουσίαση της εκτέλεσης:

```
?- get_names().  
family: lastname(lee)  
father: firstname(toby)  
mother: firstname(crystal)  
children:  
  firstname(erik)  
  firstname(ken)  
  firstname(cindy)  
true ;  
family: lastname(young)  
father: firstname(lucas)  
mother: firstname(tracy)  
children:  
  firstname(oscar)  
  firstname(mary)  
  firstname(cindy)  
  firstname(tina)  
true
```