

ΕΡΓΑΣΤΗΡΙΟ II

ΣΥΝΘΕΤΟΙ ΟΡΟΙ – ΣΥΝΘΕΤΕΣ ΕΡΩΤΗΣΕΙΣ

ΜΕΡΟΣ I

ΘΕΩΡΙΑ

Όπως είδαμε μέχρι τώρα, τα ορίσματα ενός κατηγορήματος μπορεί να είναι άτομα, αριθμοί ή μεταβλητές. Τα άτομα, οι αριθμοί και οι μεταβλητές ονομάζονται **απλοί όροι**. Εκτός από τους απλούς όρους είναι δυνατόν στην Prolog να συνδυάσουμε σχετικές μεταξύ τους πληροφορίες και να κατασκευάσουμε πιο πολύπλοκες συντακτικές δομές. Οι δομές αυτές ονομάζονται **σύνθετοι όροι** και έχουν τη μορφή: $\phi(\tau_1, \tau_2, \dots, \tau_k)$, όπου ϕ ονομάζεται **συναρτησιακό σύμβολο** τάξης k . Τα ορίσματα $\tau_1, \dots, \tau_k$ είναι και αυτά απλοί ή σύνθετοι όροι. Οι σύνθετοι όροι βοηθούν στην οργάνωση των δεδομένων σε ένα πρόγραμμα, γιατί επιτρέπουν να χειριστούμε σαν μια απλή οντότητα μια ομάδα από σχετικές μεταξύ τους πληροφορίες.

ΜΕΡΟΣ II

Γεγονός : Στοιχεία Υπαλλήλων, όπου τα ορίσματα είναι σύνθετοι όροι

ΣΤΟΙΧΕΙΑ ΥΠΑΛΛΗΛΩΝ						
ΟΝΟΜΑΤΕΠΩΝΥΜΟ		ΔΙΕΥΘΥΝΣΗ			ΜΙΣΘΟΣ	
Όνομα	Επώνυμο	Οδός	Αριθμός	Πόλη	Βασικός €	Κρατήσεις %
Δημήτρης	Δημητρίου	Αθηνάς	80	Δράμα	1300	20
Μαρία	Θεοδοσιάδου	Βενιζέλου	4	Δράμα	1860	30
Γεωργία	Χασιρτζή	Φωκά	127	Καβάλα	1300	20

Κωδικοποιήστε σε ένα πρόγραμμα τον παραπάνω πίνακα που περιέχει μισθολογικά στοιχεία υπαλλήλων μιας εταιρείας. Η κωδικοποίηση να γίνει χρησιμοποιώντας ένα γεγονός της μορφής :

```
data (onomateponymo (onoma (dimitris), eponymo (dimitriou)),  
      dieuth (odos (athinas), arithmos (80), poli (drama)),  
      misthos (basikos (1300), kratiseis (20))).
```

Σημείωση : Ανοίξτε το **Σημειωματάριο** (notepad) και γράψτε τα γεγονότα. Στη συνέχεια σώστε το αρχείο με όνομα **“exercise2.pl”**.

ΜΕΡΟΣ III

ΕΡΩΤΗΣΕΙΣ

1. Τρέξτε το πρόγραμμα **SWI-Prolog**.
2. Επιλέξτε **File → Consult**
3. Βρείτε και ανοίξτε το αρχείο **exercise2.pl**
4. Εκτελέστε την εντολή **“listing.”** για να δείτε το πρόγραμμα που γράψατε μέσα από το **SWI-Prolog**.

5. Διατυπώστε τις εξής ερωτήσεις:

α. Ποιοι υπάλληλοι μένουν στη Δράμα ;

```
?-data(onomateponymo(onoma(O), eponymo(E)), dieuth(_,_, poli(drama)), _).
```

β. Ποια είναι τα στοιχεία της διεύθυνσης της υπαλλήλου με επώνυμο Χασιρτζή;

```
?- data(onomateponymo(_, eponymo(xasirtzh)), D, _).
```

γ. Ποια είναι τα επώνυμα των υπαλλήλων που έχουν κρατήσεις 20% ;

```
?-
```

δ. Σε ποιες πόλεις κατοικούν οι υπάλληλοι ;

```
?- data(_, dieuth(_,_, poli(P)), _).
```

ε. Τι βασικό μισθό παίρνει κάθε υπάλληλος ;

```
?-
```

στ. Ποιοι υπάλληλοι παίρνουν τον ίδιο βασικό μισθό ;

```
?-data(Y1,_, misthos(basikos(X), _)), data(Y2,_, misthos(basikos(X), _)), Y1=Y2.
```

ΜΕΡΟΣ IV

ΚΑΝΟΝΕΣ - ΕΡΩΤΗΣΕΙΣ

1. Δημιουργήστε τη σχέση «ypallilos» με όρισμα το σύνθετο όρο στοιχεία υπαλλήλου (data).

```
ypallilos(data(onomateponymo(onoma(dimitris), eponymo(dimitriou)),  
dieuth(odos(athinas), arithmos(80), poli(drama)), misthos(basikos(1300), krati  
seis(20)))).
```

```
ypallilos(data(onomateponymo(onoma(maria), eponymo(theodosiadou)),  
dieuth(odos(benizelou), arithmos(4), poli(drama)), misthos(basikos(1860),  
kratiseis(30)))).
```

```
ypallilos(data(onomateponymo(onoma(georgia), eponymo(xasirtzh)),  
dieuth(odos(foka), arithmos(127), poli(kavala)), misthos(basikos(1300),  
kratiseis(20)))).
```

2. Να οριστεί η σχέση `katoikei(epwnymo_ypallilou, poli)`.

```
katoikei(E,P):- ypallilos(data(onomateponymo(_, eponymo(E)),  
dieuth(_,_, poli(P)), _)).
```

3. Να οριστεί η σχέση `pairnei_mistho(eponymo_ypallilou, basikos_misthos)`.

4. Να οριστεί η σχέση `idios_misthos(ypallilos1, ypallilos2)`.
`idios_misthos(Y1,Y2):-ypallilos(Y1), ypallilos(Y2), Y1=data(_,_,X), Y2=`
`data(_,_,X), Y1\=Y2.`

Στη συνέχεια διατυπώστε τις εξής ερωτήσεις :

α. Τι βασικό μισθό παίρνει κάθε υπάλληλος ; `?- pairnei_mistho(X,Y) .`

β. Σε ποια πόλη κατοικεί κάθε υπάλληλος ;

?- _____

γ. Ποιοι υπάλληλοι παίρνουν τον ίδιο μισθό; `?- idios_misthos(A,B) .`

ΜΕΡΟΣ V

ΜΕΤΑΒΛΗΤΕΣ ΣΤΗΝ PROLOG

Οι μεταβλητές στην Prolog αντιπροσωπεύουν όρους οι οποίοι δεν είναι γνωστοί κατά τη στιγμή της διατύπωσης της πρότασης. Έτσι διαφέρουν από τις μεταβλητές των κλασικών γλωσσών προγραμματισμού, οι οποίες αντιπροσωπεύουν θέσεις μνήμης που θα δεχθούν κάποια δεδομένα προκαθορισμένου τύπου.

Οι μεταβλητές στην Prolog δεν έχουν τύπο σε αντίθεση με τις αλγοριθμικές γλώσσες προγραμματισμού. Έτσι, μια μεταβλητή στην Prolog μπορεί να πάρει σαν τιμή αριθμό, άτομο, ή σύνθετο όρο χωρίς αυτό να χρειάζεται να δηλωθεί από πριν. Ακόμη πολλές φορές μια μεταβλητή μιας πρότασης παίρνει διαφορετικού «τύπου» τιμές σε διαφορετικές κλήσεις της πρότασης που την περιέχει.

Οι μεταβλητές που έχουν το ίδιο όνομα, αλλά ανήκουν σε διαφορετικούς κανόνες δεν έχουν καμιά σχέση μεταξύ τους. Οι μεταβλητές είναι τοπικές για κάθε πρόταση του προγράμματος.

Σε κάθε χρονική στιγμή της εκτέλεσης ενός προγράμματος, μια μεταβλητή μπορεί να έχει πάρει τιμή ή να μην έχει πάρει τιμή.

Ασκήσεις για εξάσκηση πάνω στην ύλη του εργαστηρίου

- Να ορισθούν μέσω κανόνων οι παρακάτω σχέσεις :
 - `idia_poli(ypallilos1, ypallilos2) .`
- Διατυπώστε με βάση το πρόγραμμα που έχετε γράψει τις εξής ερωτήσεις:
 - Τι βασικό μισθό παίρνει η υπάλληλος με επώνυμο Χασιρτζή ;
 - Ποιοι υπάλληλοι μένουν στην Καβάλα ;
 - Τι βασικό μισθό παίρνει ο υπάλληλος με επώνυμο Δημητρίου ;
 - Ποια είναι η ανάλυση του μισθού των υπαλλήλων του κατοικούν στη Καβάλα ;
 - Ποιοι υπάλληλοι παίρνουν τον ίδιο συνολικά μισθό ;
 - Τι βασικό μισθό παίρνει κάθε υπάλληλος ;
- Κωδικοποιήστε τον προσωπικό σας τηλεφωνικό κατάλογο σε ένα πρόγραμμα Prolog.