

ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

6^ο Εξάμηνο

- Κατηγορήματα για αριθμητική επεξεργασία



Δημοσθένης Σταμάτης

<http://www.iee.ihu.gr/~demos>

Τμήμα Μηχανικών Πληροφορικής & Ηλεκτρονικών Συστημάτων

Αριθμητικές εκφράσεις στην Prolog

Εκφράσεις όπως η $2 + 5$ είναι κανονικοί όροι (terms) στην Prolog
Τι γίνεται όμως όταν κάνουμε μία ερώτηση όπως οι παρακάτω?


?- $2 + 5 = 7$.
 No (false)
?- $7 = 2 + 5$.
 No (false)

Πρόκειται για μυστήριο? Ας το ξεκαθαρίσουμε!




Η Prolog σκόπιμα δεν αποδίδει τιμή σε αριθμητικές εκφράσεις, όπως η παραπάνω, καθώς τις αντιμετωπίζει ως συμβολικές εκφράσεις. Συντακτικά το $2+5$ είναι σύνθετος όρος ενώ το 7 μία σταθερά.


Ενσωματωμένα κατηγορήματα αριθμητικών υπολογισμών

 Όταν απαιτείται να γίνουν αριθμητικοί υπολογισμοί, μπορούν να χρησιμοποιηθούν ενσωματωμένα κατηγορήματα που διαθέτει η Prolog για το σκοπό αυτό.

Το ενσωματωμένο κατηγορήμα **is**:

 Υπολογίζει το σύνθετο όρο που βρίσκεται στα δεξιά του ως μία αριθμητική έκφραση και **ενοποιεί(*)** την τιμή που προκύπτει με την μεταβλητή που βρίσκεται στο αριστερό του μέρος (η μεταβλητή αυτή είναι συνήθως **μη-τοποθετημένη(*)**).

Παράδειγμα:

 ?- X **is** 2 * 3 + 4 * 6.
X = 30

(*) Προσέξτε τους όρους **ενοποιεί** (και όχι αναθέτει) και **μη-τοποθετημένη**

Το ενσωματωμένο κατηγορήμα **is**


Το κατηγορήμα **is** συνήθως χρησιμοποιείται με τον **ενθεματικό (infix)** τρόπο γραφής, του ως τελεστής (operator) και έχει τη γενική μορφή:

 **Variable is Expression.**


Όπου **Expression** οποιαδήποτε αριθμητική παράσταση που μπορεί να υπολογιστεί

Μπορεί όμως να χρησιμοποιηθεί και με την **προθεματική του (prefix)** μορφή, όπως τα άλλα κατηγορήματα της Prolog που έχουμε δει.

 **is(Variable, Expression).**

 Παράδειγμα:
?- **is**(X, 2 * 3 + 4 * 6).
X = 30


Το ενσωματωμένο κατηγορήμα **is**

 Οι μεταβλητές που μπορούν να υπάρχουν στο δεξιό μέρος του **is**, πρέπει οπωσδήποτε κατά την ώρα της εκτέλεσης να είναι τοποθετημένες. Σε αντίθετη περίπτωση, προκύπτει σφάλμα


Παράδειγμα:


 `?- X is 2*3 + 4 * Y.`
 ERROR: is/2: Arguments are not sufficiently instantiated

Ενώ:


 `?- Y=6, X is 2*3 + 4 * Y.`
`X=30`

Το ενσωματωμένο κατηγορήμα **is**

 Το κατηγορήμα **is**, προσομοιώνει κατά κάποιο τρόπο την εντολή αντικατάστασης (τελεστής "=") των κλασσικών γλωσσών προγραμματισμού, όπως οι Java/C++/Python


 Δεν μπορεί όμως η ίδια μεταβλητή να εμφανίζεται και δεξιά και αριστερά του **is**, καθώς στην Prolog μία μεταβλητή, στο πεδίο δράσης της, όταν τοποθετηθεί με μία τιμή, αυτή δεν μπορεί να αλλάξει.

Παράδειγμα :

 `?- X=6, X is 2*3 + 4 * X.`
 No (false)

Επίσης δεν μπορεί να χρησιμοποιηθεί σε καμιά άλλη περίπτωση, παρά μόνο για υπολογισμό αριθμητικών εκφράσεων.

Παράδειγμα :

 `?- CH is 'string1'+ 'string2'.`
 ERROR