

Ταυτοποίηση

(Παράδειγμα 1)

(Κεφαλή)

brothers (X, Y) :- parent(Z,X), parent(Z,Y).

↑ ↑
?- brothers(nick, vicky),,

(κλήση)

(Παράδειγμα 2)

a(X, 5, nick) :- b(.....,,

↑ | ↓
?- a(4, 5, Y),,

Εκτέλεση Prolog Προγράμματος

Πρόγραμμα:

B.

B :- B₁, B₂, ..., B_μ.

B₁ :- C₁, ..., C_κ.

B₂ :- D₁, ..., D_λ.

....

B_ρ :- Z₁, ..., Z_ν.

Ερώτηση: ?- B, Γ, Δ,, Ω.

- Η εκτέλεση του προγράμματος ξεκινά από την ερώτηση και συνίσταται στην επανάληψη ενός πεπερασμένου πλήθους βημάτων.
- Σε κάθε βήμα η ερώτηση μετασχηματίζεται σε μια νέα ερώτηση με βάση την εξής διαδικασία:
 - ⇒ Επιλέγεται η πρώτη από αριστερά κλήση της ερώτησης (B)
 - ⇒ Βρίσκεται η πρώτη πρόταση, (διατρέχοντας το πρόγραμμα από πάνω προς τα κάτω), που έχει ίδιο κατηγορημα και τάξη και είναι ταυτοποιήσιμη με την κλήση (B) (π.χ. B_i:- A₁, ..., A_κ.) δίνοντας σαν αποτέλεσμα τον πιο γενικό ταυτοποιητή θ.

⇒ Η νέα ερώτηση προκύπτει αν αντικαταστήσουμε την κλήση με το σώμα του κανόνα που εκτελέστηκε ή αφαιρώντας την αν ταυτοποιήθηκε με γεγονός.

⇒ Στη νέα ερώτηση εφαρμόζουμε τις αντικαταστάσεις που περιλαμβάνει ο θ .

Νέα ερώτηση: $? (A_1, \dots, A_k, \Gamma, \Delta, \dots, \Omega)\theta$

⇒ Η διαδικασία επαναλαμβάνεται εως ότου φτάσουμε σε ερώτηση χωρίς κλήσεις (κενή ερώτηση \square). Τότε έχουμε επιτυχή εκτέλεση του προγράμματος και επιστρέφονται οι τιμές των μεταβλητών.

- Όταν μια κλήση μπορεί να ταυτοποιηθεί με περισσότερες από μια προτάσεις, τότε αναφέρεται ως σημείο επιλογής (choice point) ή σημείο οπισθοδρόμησης (backtracking point)
- Αν κατά την εκτέλεση, η τρέχουσα κλήση:
 - i. δεν βρει πρόταση που ταιριάζει (match), ή
 - ii. αποτύχει στην ταυτοποίηση (unification) με την πρόταση που ταιριάζει,τότε το σύστημα οπισθοδρομεί (backtracks) στην αμέσως προηγούμενη κλήση που είναι σημείο επιλογής ακυρώνοντας ταυτόχρονα τις

αντικαταστάσεις που είχε κάνει από εκείνο το σημείο μέχρι την τρέχουσα κλήση.

- Αν και με την οπισθοδρόμηση δεν φτάσουμε στην κενή ερώτηση, τότε η απάντηση στην αρχική ερώτηση είναι αρνητική (αποτυχία).
- Σε περίπτωση καταφατικής απάντησης (επιτυχίας), το σύστημα οπισθοδρομεί πάλι σε προηγούμενα σημεία επιλογής για να βρεθούν όλες οι εναλλακτικές λύσεις (με το ;)
- Οι τιμές των μεταβλητών της ερώτησης προκύπτουν από τη σύνθεση των ταυτοποιητών κάθε στοιχειώδους υπολογιστικού βήματος $\theta_1 - \theta_2 - \theta_3 \dots - \theta_\omega$.

Άσκηση

1. Τί απάντηση θα πάρετε αν υποβάλλετε την ερώτηση $? - \mathbf{k}(\mathbf{x}, \mathbf{f}(\mathbf{x}))$ στο πρόγραμμα $\mathbf{k}(\mathbf{y}, \mathbf{y})$.

Απάντηση:

$$\mathbf{x} = \mathbf{f}(\mathbf{f}(\mathbf{f}(\dots \mathbf{f}(\dots \mathbf{f}(\dots))))).$$

ή

Error, System Stack Full

ή

No (μόνο αν ο διεκρινευστής της γλώσσας Prolog υποστηρίζει το λεγόμενο "occurs check")

Άσκηση

2. Θεωρείστε το ακόλουθο πρόγραμμα :

f(1 , one).

f(s(1) , two).

f(s(s(1))), three).

$$f(s(s(X)), N) : - f(X, N).$$

Ποιες θα είναι οι απαντήσεις στις ακόλουθες ερωτήσεις :

- α) ?- f(s(1) , A).
- β) ?- f(s(s(1)) , two).
- γ) ?- f(s(s(s(s(1)))) , C).
- δ) ?- f(D , three).

Απάντηση :

- $$\begin{array}{ll} \alpha) \mathbf{A} = \mathbf{two} & \beta) \mathbf{No} \\ \delta) \mathbf{D} = \mathbf{s(s(1))} & ; \\ \mathbf{D} = \mathbf{s(s(s(s(1))))} & ; \\ \mathbf{D} = \mathbf{s(s(s(s(s(s(1))))))} & ; \\ \dots & \end{array}$$