



ΕΝΟΤΗΤΑ 2.5

Μια μικρή εισαγωγή στην PROLOG

Γιώργος Στάμου

ΜΕΡΟΣ 2: Αναπαράσταση Γνώσης και Συλλογιστική

Programming in Logic



2

Αναπαράσταση Γνώσης με Προτάσεις Horn

Συλλογιστική με ανάλυση SLD
Υλοποίηση αλγόριθμου backward-chaining

Επιπλέον προγραμματιστικά στοιχεία
Ενσωματωμένα κατηγορήματα

Πρώτη αποδοτική υλοποίηση της PROLOG
DHD Warren, Πανεπιστήμιο Εδιμβούργου, 1977

Αρκετές υλοποιήσεις
π.χ. SWI Prolog, <http://www.swi-prolog.org/>

Μέρος 2 – Ενότητα 2

Βασικά στοιχεία PROLOG



3

Γεγονότα

```
father(george,mary) .
```

Κανόνες

```
parent(X,Y):-father(X,Y) .  
parent(X,Y):-mother(X,Y) .
```

Ερωτήματα

```
?-parent(X,Y) .  
X=george,Y=mary;  
no
```

Μέρος 2 – Ενότητα 2

Παράδειγμα (1)



4

Γνώση

```
father(george,mary) .  
father(george,nick) .  
father(peter,marina) .  
mother(helen,mary) .  
mother(helen,nick) .  
mother(ann,marina) .  
parent(X,Y):-father(X,Y) .  
parent(X,Y):-mother(X,Y) .
```

Αποθηκεύεται σε ένα αρχείο, π.χ. family.pl

Φορτώνεται: `?-consult('family.pl') .`

Μέρος 2 – Ενότητα 2

Παράδειγμα (2)



5

Ερωτήματα

- Είναι ο george πατέρας της mary?
`?-father(george,mary) .`
`yes`
- Ποιος (X) έχει πατέρα τον peter?
`?-father(peter,X) .`
`X=marina`
- Ποια είναι τα ζεύγη (X, Y) για τα οποία το X είναι mother του Y?
`?-mother(X,Y) .`
`X=helen,Y=mary;`
`X=helen,Y=nick;`
`X=ann,Y=marina;`
`no`

Μέρος 2 – Ενότητα 2

Σύνταξη και σημασιολογία της PROLOG



6

Βασικά στοιχεία της γλώσσας

Όροι, γεγονότα, κανόνες, ερωτήσεις

Σύνταξη προγράμματος PROLOG

Σύνολο προτάσεων της μορφής:

`A .`

`A: -B1, B2, ..., Bn .`

`?- B1, B2, ..., Bn`

όπου B_i ατομικοί τύποι της μορφής

$p(t_1, t_2, \dots, t_k)$, p κατηγορημα και t_i

όροι (σταθερές, μεταβλητές κλπ)

Σημασιολογία

Δηλωτική ή διαδικαστική (προτάσεις Horn, ανάλυση SLD)

Μέρος 2 – Ενότητα 2

Παράδειγμα (1)



7

Γνώση

```
greek(socrates).
human(turing).
human(socrates).
fallible(X):-human(X).
```

Ερώτημα

```
?-fallible(Y),greek(X).
```

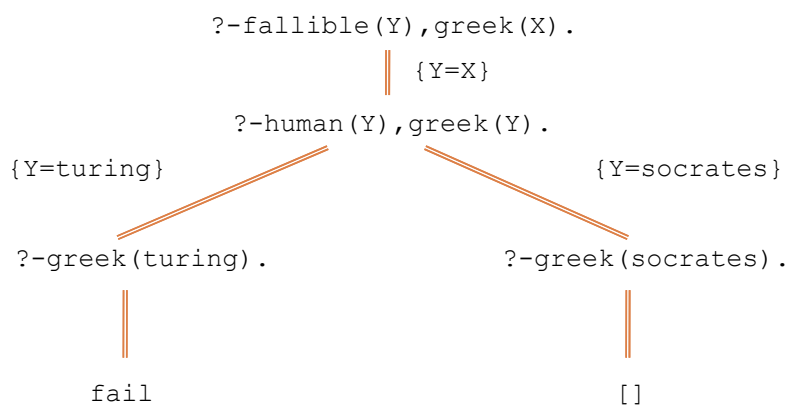
Μέρος 2 – Ενότητα 2

Παράδειγμα (2)



8

Δέντρο υπολογισμού



Μέρος 2 – Ενότητα 2

Αναδρομή (1)



9

Παράδειγμα

```
parent(john, george) .
parent(john, nick) .
parent(jim, bill) .
parent(jim, jack) .
parent(gregory, john) .
parent(gregory, jim) .
parent(bob, gregory) .
parent(joseph, bob) .
```

Μπορούμε να ορίσουμε ένα κατηγορημα για τον πρόγονο?

π.χ. `predecessor(X, Z)`

Μέρος 2 – Ενότητα 2

Αναδρομή (2)



10

Λύση πρώτη

```
predecessor(X, Z) :- parent(X, Z) .
predecessor(X, Z) :- parent(X, Y), parent(Y, Z) .
predecessor(X, Z) :-
    parent(X, Y), parent(Y, W), parent(W, Z) .
...
```

Λύση δεύτερη

```
predecessor(X, Z) :- parent(X, Z) .
predecessor(X, Z) :- parent(X, Y), predecessor(Y, Z)
```

Μέρος 2 – Ενότητα 2

Δομές δεδομένων στην PROLOG



11

Σύνθετες δομές δεδομένων

```
triangle(point(0,0),point(1,4),point(5,12)).
.(a,.(b,.(c,[])))
```

Παραδείγματα λιστών

```
[a b c].
[[a,b],c].
```

Μέρος 2 – Ενότητα 2

Ενσωματωμένα κατηγορήματα (1)



12

Μαθηματικές πράξεις

```
?-X is 3 + 4.
X=7
?-9 is 3 * 3.
yes
```

Διαδικασίες εισόδου-εξόδου

```
?-read(X).
|:hello.
X=hello
?-write(hello),nl,write(world),nl.
hello
world
yes
```

Μέρος 2 – Ενότητα 2

Ενσωματωμένα κατηγορήματα (2)



13

Δυναμική τροποποίηση προγράμματος

```
asserta(X), assertz(X), retract(X)
```

```
father(nick,mary)
```

```
?-father(nick,X).
```

```
X=mary
```

```
?-asserta(father(nick, john)).
```

```
yes
```

```
?-father(nick,X).
```

```
X=john;
```

```
X=mary;
```

```
no
```

Μέρος 2 – Ενότητα 2

Ενσωματωμένα κατηγορήματα (3)



14

Άρνηση ως αποτυχία

```
?-not(member(a,[a,b,c])).
```

```
no
```

```
?-not(member(d,[a,b,c])).
```

```
yes
```

Μέρος 2 – Ενότητα 2