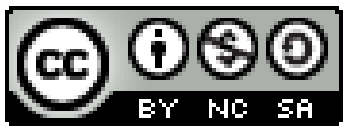
ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΙΚΑ
ΜΑΘΗΜΑΤΑ

Συστήματα Γνώσης

Πρακτικό Κομμάτι Μαθήματος
Περιορισμοί στις Συνθήκες Κανόνων

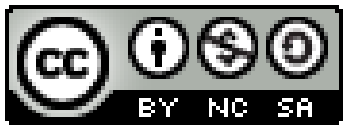
Νίκος Βασιλειάδης, Αναπλ. Καθηγητής
Τμήμα Πληροφορικής



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

Λογική

Συνθήκες Κανόνων με Περιορισμούς

- Εκτός από την απλή ταυτοποίηση συνθηκών με γεγονότα, μπορούμε να εισάγουμε και **περιορισμούς** στις συνθήκες χρησιμοποιώντας πιο εκφραστικές δομές.
  - χρήση συγκεκριμένων **συνδετικών**
  - εισαγωγή **συνθηκών** υπό μορφή συναρτήσεων (test conditions)

Χρήση Συνδετικών *(connectives)*

Λογική Πράξη	Συνδετικό
λογική άρνηση	~
λογική διάζευξη (ή)	|
λογική σύζευξη (και)	&

Χρήση Συνδετικών *(connectives)*

```
(defrule days
   (month Jan day ~mon)
   (hour 12|13)
      =>
   (printout t "Day is not Monday, but it is
   noon" crlf))
```

- `(month Jan day ~mon)` ταυτοποιείται με γεγονότα της μορφής `(month Jan day <symbol>)`
  - Το `<symbol>` **δεν πρέπει** να είναι `mon`
- `(hour 12|13)` ταυτοποιείται **είτε** με `(hour 12)` **ή** με `(hour 13)`

Χρήση Συνδετικών *(connectives)*

- Το συνδετικό "**&**" (and) χρησιμοποιείται συνήθως σε συνδυασμό με άλλους περιορισμούς

```
(defrule days
   (month Jan day ?day&~mon)
   (hour 12|13)
=>
   (printout t "Day is " ?day "and not
  Monday, but it is noon time!" crlf))
```

- Η συνθήκη **?day&~mon** δηλώνει:
  - Στη θέση αυτή πρέπει να υπάρχει κάποιο σύμβολο ***?day*** ***και*** το σύμβολο αυτό να ***μην είναι*** το ***mon***

Παράδειγμα Χρήσης Συνδετικών

- Έστω ότι υπάρχουν τα παρακάτω γεγονότα:

```
(deffacts elements
   (element a)
   (element b)
   (element c)
)
```

- Θέλουμε να τυπώσουμε στην οθόνη όλους τους διαφορετικούς συνδυασμούς ανά δύο

Παράδειγμα Χρήσης Συνδετικών

```
(defrule cartesian
  (element ?a)
  (element ?b)
=>
  (printout t "Elements: " ?a " "
  ?b crlf)
)
```

- Τυπώνει και τους μη-επιθυμητούς συνδυασμούς

```
(Elements: a a)
(Elements: b b)
(Elements: c c)
```

Παράδειγμα Χρήσης Συνδετικών

- Πρέπει να εισάγουμε περιορισμό στη συνθήκη

```
(defrule cartesian
  (element ?a)
  (element ?b&~?a)
=>
  (printout t "Elements: " ?a " "
  ?b crlf)
)
```

- Ο συνδυασμός των συνδετικών επιβάλλει στην τιμή της μεταβλητής **?b** να είναι διαφορετική από εκείνη της **?a**.

Λογικές Συναρτήσεις στις Συνθήκες Κανόνων

```
(defrule emerg2
   (fire drill)
 =>
   (assert (evacuate building)))
```

```
(defrule emerg1
   (emerg type fire)
 =>
   (assert (evacuate building)))
```

```
(defrule emerg
   (or (emerg type fire) (fire drill))
 =>
   (assert (evacuate building)))
```

Η χρήση του `not`

- Αν εμφανίζονται μεταβλητές μέσα σε κάποιο **`(not ...)`** τότε δε γίνεται ανάθεση τιμών σε αυτές

```
(defrule wrong-rule
  (not (element ?b))
  =>
  (printout t "not element" ?b
  crlf) )
```

Η χρήση του test

```
(defrule cartesian
  (element ?a)
  (element ?b)
  (test (neq ?a ?b))
  =>
  (printout t "Elements: " ?a " "
  ?b crlf)
)
```

Παράδειγμα χρήσης Λογικών Συνθηκών

- Έστω ότι απαιτείται να επιλεγεί το πιο ακριβό από μια λίστα βιβλίων:

```
(deffacts prices
  (book A price 34)
  (book B price 20)
  (book C price 68)
)
```

Παράδειγμα χρήσης Λογικών Συνθηκών

```
(defrule select-exp2
  (book ?Book price ?price)
  (not (and
       (book ?Book2&~?Book price
  ?price2)
      (test (> ?price2 ?price)) ) )
  =>
  (printout t "The Book with the
  highest price is:" ?Book crlf)    )
```

- Εάν υπάρχει βιβλίο **?Book** με τιμή **?price** και δεν υπάρχει **άλλο** βιβλίο **?Book2** το οποίο να έχει τιμή **?price2** (μεγαλύτερη από την **?price**), τότε τύπωσε το βιβλίο **?Book**

Παράδειγμα χρήσης Λογικών Συνθηκών

```
(defrule select-exp2
  (book ?Book price ?price)
  (not
     (book ?Book2&~?Book
        price ?price2&:(> ?price2
 ?price))
  )
  =>
  (printout t "The Book with the
 highest price is:" ?Book crlf)
     )
```

Έλεγχος Τιμής ως Αποτέλεσμα Συνάρτησης

- Πολλές φορές θέλουμε να συγκρίνουμε την τιμή μιας παραμέτρου ενός γεγονότος αν ισούται με μία σταθερά, αλλά δεν ξέρουμε την σταθερά όταν δημιουργούμε τον κανόνα
  - Προκύπτει at run-time ως αποτέλεσμα μιας συνάρτησης
- Στην περίπτωση αυτή χρησιμοποιούμε στην θέση της σταθεράς την έκφραση `=(...)`

Παράδειγμα

- Δώσε μου όλα τα βιβλία που κοστίζουν ακριβώς όσο 2 άλλα βιβλία!!
- Π.χ.

```
(deffacts prices
   (book A price 34)
   (book B price 20)
   (book C price 68)
   (book D price 54)
)
```

D=A+B

C=A+A

Παράδειγμα

```
(defrule double-price
  (book ?B1 price ?P1)
  (book ?B2 price ?P2)
  (book ?B3 price =(+ ?P1 ?P2))
=>
  (printout t "Book " ?B3 " has
  the sum of the prices of books
  " ?B1 " and " ?B2 crlf))
```

Αποτελέσματα

```
CLIPS> (reset)
==> f-0     (initial-fact)
==> f-1     (book A price 34)
==> f-2     (book B price 20)
==> f-3     (book C price 68)
==> f-4     (book D price 54)
CLIPS> (run)
Book C has the sum of the prices of books A and A
Book D has the sum of the prices of books A and B
Book D has the sum of the prices of books B and A
The Book with the highest price is: C
```

Παραλλαγές

- Αν δεν θέλω δυο φορές το ίδιο βιβλίο

```
(defrule double-price
  (book ?B1 price ?P1)
  (book ?B2&~?B1 price ?P2)
  (book ?B3 price =(+ ?P1 ?P2))
=>
  (printout t "Book " ?B3 " has the
  sum of the prices of books " ?B1
  " and " ?B2 crlf))
```

Αποτελέσματα (1)

```
CLIPS> (reset)
==> f-0     (initial-fact)
==> f-1     (book A price 34)
==> f-2     (book B price 20)
==> f-3     (book C price 68)
==> f-4     (book D price 54)
CLIPS> (run)
Book D has the sum of the prices of books A and
  B
Book D has the sum of the prices of books B and
  A
The Book with the highest price is: C
```

Παραλλαγές (2)

- Αν δεν θέλω δυο φορές το συνδυασμό Α-Β, Β-Α

```
(defrule double-price
  (book ?B1 price ?P1)
  (book ?B2&:(> (str-compare ?B2 ?B1) 0)
  price ?P2)
  (book ?B3 price =(+ ?P1 ?P2))
=>
  (printout t "Book " ?B3 " has the
  sum of the prices of books " ?B1
  " and " ?B2 crlf))
```

Αποτελέσματα (3)

```
CLIPS> (reset)
==> f-0     (initial-fact)
==> f-1     (book A price 34)
==> f-2     (book B price 20)
==> f-3     (book C price 68)
==> f-4     (book D price 54)
CLIPS> (run)
Book D has the sum of the prices of
  books A and B
The Book with the highest price is: C
```

ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΟΙΚΤΑ
ΑΚΑΔΗΜΑΙΚΑ
ΜΑΘΗΜΑΤΑ

Τέλος Ενότητας

Επεξεργασία: Εμμανουήλ Ρήγας

Θεσσαλονίκη, 17/3/2014

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης