

ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΤΥΠΙΚΕΣ ΜΕΘΟΔΟΥΣ ΑΝΑΛΥΣΗΣ ΣΥΣΤΗΜΑΤΩΝ

- Οι *τυπικές μέθοδοι* παρέχουν ένα πλαίσιο μέσα στο οποίο μπορούμε να προδιαγράψουμε και να εγκυροποιήσουμε ένα σύστημα με συστηματικό τρόπο.
- Όταν γράφουμε τυπικές περιγραφές είναι βολικό να χρησιμοποιούμε κάποιους διαδεδομένους μαθηματικούς συμβολισμούς.

ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΤΥΠΙΚΕΣ ΜΕΘΟΔΟΥΣ ΑΝΑΛΥΣΗΣ ΣΥΣΤΗΜΑΤΩΝ

- Ένας αριθμός τυπικών μεθόδων βασίζεται σε κάποιες απαραίτητες μαθηματικές γνώσεις:

Προτασιακού Λογισμού (propositional calculus) με τον οποίο μπορούμε να συνθέσουμε λογικές εκφράσεις (δηλ. εκφράσεις καθεμία από τις οποίες είναι είτε «αληθής» είτε «ψευδής»).

Κατηγορηματικού Λογισμού (predicate calculus) με τον οποίο μπορούμε να περιγράψουμε ιδιότητες και περιορισμούς χρησιμοποιώντας ποσοτικούς και υπαρξιακούς τελεστές.

Θεωρίας Συνόλων με την οποία περιγράφουμε λειτουργίες πάνω σε συλλογές από αντικείμενα.

ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΤΥΠΙΚΕΣ ΜΕΘΟΔΟΥΣ ΑΝΑΛΥΣΗΣ ΣΥΣΤΗΜΑΤΩΝ

- **Είδη τυπικών μεθόδων**

Μπορούν να ταξινομηθούν με βάση διάφορα κριτήρια

Μέθοδοι που προδιαγράφουν άμεσα τη συμπεριφορά του συστήματος κατασκευάζοντας ένα μοντέλο και πιθανώς χρησιμοποιώντας μαθηματικούς όρους, όπως σύνολα, ακολουθίες, σχέσεις κ.α.

Μέθοδοι που επικεντρώνονται στις ιδιότητες και προδιαγράφουν έμμεσα το σύστημα δηλώνοντας ένα σύνολο ιδιοτήτων - συνήθως με τη μορφή αξιωμάτων - που πρέπει το σύστημα να ικανοποιεί.

ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΤΥΠΙΚΕΣ ΜΕΘΟΔΟΥΣ ΑΝΑΛΥΣΗΣ ΣΥΣΤΗΜΑΤΩΝ

- **Ποια τυπική μέθοδος είναι η καλύτερη;**

Για κάθε διαφορετικό σκοπό υπάρχουν και κάποιες που είναι καταλληλότερες.

Σκοπός του μαθήματος είναι να έχετε μία συνολική άποψη για τις ποιο διαδεδομένες τυπικές μεθόδους και να είστε ικανοί χρήστες μιας εξ' αυτών δηλαδή της *επαλήθευσης με έλεγχο μοντέλων*.

ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΤΥΠΙΚΕΣ ΜΕΘΟΔΟΥΣ ΑΝΑΛΥΣΗΣ ΣΥΣΤΗΜΑΤΩΝ

- Μέθοδοι που βασίζονται σε ένα μοντέλο του συστήματος
μηχανές καταστάσεων → έλεγχος μοντέλων
VDM, Z → προδιαγραφές συστημάτων
Petri Nets, → για ταυτόχρονα - κατανεμημένα
Calculus of Communication συστήματα
Systems, Communicating
Sequential Processes (CSP)

ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΤΥΠΙΚΕΣ ΜΕΘΟΔΟΥΣ ΑΝΑΛΥΣΗΣ ΣΥΣΤΗΜΑΤΩΝ

- Αξιοματικές μέθοδοι που επικεντρώνονται στις ιδιότητες και χρησιμοποιούν προ-συνθήκες και μετά-συνθήκες κατηγορηματικής λογικής για να αποδείξουν κριτήρια ορθότητας (με ημιαυτόματο τρόπο).

τριπλέτες Hoare & —————> επαλήθευση προγραμμάτων

Λογισμός Hoare - Floyd

Weakest Precondition Calculus —————> επαλήθευση προγραμμάτων

OBJ —————> αλγεβρικές περιγραφές με
αφαιρετικούς τύπους
δεδομένων

LOTOS —————> για κατανεμημένα συστήματα

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ I

- Προτασιακός λογισμός είναι μία γλώσσα που μας επιτρέπει *να θεμελιώνουμε ισχυρισμούς με τυπικό τρόπο* για ένα μοντέλο που μπορεί να αναπαριστά μία πιθανή κατάσταση που μελετάμε ως επαγγελματίες πληροφορικοί.
- Θεμελίωση ισχυρισμών για μία πιθανή κατάσταση σημαίνει το να αναπτύσσουμε συμπεράσματα. Θέλουμε αυτό να γίνεται με τυπικό τρόπο έτσι ώστε τα συμπεράσματα
 - να είναι *έγκυρα* δηλ. να αιτιολογούνται με μαθηματική αυστηρότητα ή
 - να μπορούν να εκτελεστούν σε μία μηχανή (proof assistant)

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ II

ΠΑΡΑΔΕΙΓΜΑ

Αν το τραίνο καθυστερήσει και δεν υπάρχουν στο σταθμό ταξί, τότε ο Κώστας αργεί στο ραντεβού του.

Ο Κώστας δεν αργεί στο ραντεβού του.

Το τραίνο έφτασε με καθυστέρηση.

ΑΡΑ, υπήρχαν ταξί στο σταθμό.

ΑΝ p **ΚΑΙ** **ΟΧΙ** q , **TOTE** r .

ΟΧΙ r . p .

TOTE q .

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ III

- Η γλώσσα που θα χρησιμοποιήσουμε, δηλ. ο προτασιακός λογισμός αναδεικνύει τη λογική δομή των ισχυρισμών και βασίζεται σε *δηλωτικές προτάσεις* ή απλά *προτάσεις*, δηλ. εκφράσεις για τις οποίες μπορούμε να ισχυριστούμε ή ότι είναι αληθείς ή ότι είναι ψευδείς.

ΠΑΡΑΔΕΙΓΜΑ ΔΗΛΩΤΙΚΗΣ ΠΡΟΤΑΣΗΣ

Κάθε άρτιος φυσικός αριθμός δίνεται ως άθροισμα δύο πρώτων αριθμών. (εικασία Goldbach)

ΠΑΡΑΔΕΙΓΜΑ ΜΗ ΔΗΛΩΤΙΚΗΣ ΠΡΟΤΑΣΗΣ

Μπορεί η προσπάθεια να είναι επιτυχής.

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ IV

- Στην πληροφορική, ο προτασιακός λογισμός επεξεργάζεται δηλωτικές προτάσεις ή απλά *δηλώσεις για τη συμπεριφορά υπολογιστικών συστημάτων ή προγραμμάτων*. Δεν ενδιαφερόμαστε όμως απλά για τον ορισμό τέτοιων δηλώσεων αλλά και για να ελέγξουμε αν ένα πρόγραμμα ή ένα σύστημα πληροί τον *ορισμό* που μελετάμε.
- Έτσι, μας ενδιαφέρει τελικά ένας *λογισμός* που θα μας επιτρέψει να θεμελιώνουμε ισχυρισμούς και να καταλήγουμε σε συμπεράσματα με βάση δοθείσες *υποθέσεις* όπως π.χ. τις αρχικές τιμές μεταβλητών.
- Ένας τέτοιος λογισμός είναι αξιόπιστος με την έννοια ότι διασφαλίζει την αλήθεια: *αν όλες οι υποθέσεις είναι αληθείς τότε είναι αληθές και το συμπέρασμα*.

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ V

- Μία άλλη πιο δύσκολη χρήση του προτασιακού λογισμού είναι αν δοθείσης μιας ιδιότητας που είναι αληθής για ένα πρόγραμμα μπορούμε να βρούμε στο λογισμό μας έναν ισχυρισμό που να αποδίδει τη συγκεκριμένη ιδιότητα ως συμπέρασμα (βλ. δυσκολία απόδειξης της εικασίας Goldbach).
- Ο λογισμός που παρουσιάζουμε είναι *συμβολικός* με την έννοια ότι αναπαριστούμε ένα μεγάλο υποσύνολο όλων των δηλωτικών προτάσεων που μπορούμε να εκφράσουμε σε φυσική γλώσσα με σύμβολα.

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ VI

- Θεωρούμε κάποιες δηλωτικές προτάσεις ως *πρωταρχικές* ή *μη διασπάσιμες (ατομικές)*, π.χ.

«Ο αριθμός 5 είναι άρτιος.»

Αυτές τις συμβολίζουμε με p, q, r ή μερικές φορές με $p_1, p_2, \dots$ και τελικά κατασκευάζουμε πιο πολύπλοκες προτάσεις με σύνθεση αυτών των πρωταρχικών προτάσεων.

- Μπορούμε να σχηματίσουμε σύνθετες προτάσεις με χρήση των:
 - $\neg$: άρνηση. Η άρνηση της πρότασης p συμβολίζεται με $\neg p$
 - $\vee$: διάζευξη. Η $p \vee r$ σημαίνει ότι τουλάχιστο μία από τις p, r είναι αληθής.
 - $\wedge$: σύζευξη. Η $p \wedge r$ σημαίνει ότι οι p και r είναι αληθείς.
 - $\rightarrow$: συνεπαγωγή. Η $p \rightarrow q$ σημαίνει ότι η q είναι λογική συνέπεια της p

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ VII

ΠΑΡΑΔΕΙΓΜΑ ΣΥΝΘΕΤΗΣ ΠΡΟΤΑΣΗΣ

$$p \wedge q \rightarrow \neg r \vee q$$

«Αν p και q , τότε όχι p ή q »

ΠΡΟΤΕΡΑΙΟΤΗΤΑ

$\neg$	$\uparrow$ υψηλότερη προτεραιότητα
$\wedge \vee$	$ $
$\rightarrow$	$\downarrow$ χαμηλότερη προτεραιότητα

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ VIII

ΛΟΓΙΚΟ ΣΥΜΠΕΡΑΣΜΑ

- Για την εξαγωγή *λογικών συμπερασμάτων* χρησιμοποιούμε ένα σύνολο *αποδεικτικών κανόνων* που μας επιτρέπει την μετάβαση από έναν τύπο σε έναν άλλο τύπο. Με τη διαδοχική εφαρμογή τέτοιων κανόνων καταλήγουμε τελικά σε ένα συμπέρασμα με βάση ένα σύνολο *προϋποθέσεων*.

Έστω λοιπόν οι τύποι $\phi_1, \phi_2, \dots, \phi_n$, που τους αποκαλούμε προϋποθέσεις και ένας άλλος τύπος ψ που αποτελεί το συμπέρασμα. Θέλουμε με την εφαρμογή κανόνων στις προϋποθέσεις να δημιουργήσουμε τύπους που απ' αυτούς να καταλήξουμε στο συμπέρασμα. Αυτό το γράφουμε με μια *ακολουθία*:

$$\phi_1, \phi_2, \dots, \phi_n \vdash \psi$$

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ IX

ΛΟΓΙΚΟ ΣΥΜΠΕΡΑΣΜΑ (συνέχεια)

- Μία ακολουθία είναι έγκυρη αν μπορούμε να κάνουμε γι' αυτή μία απόδειξη.

ΠΑΡΑΔΕΙΓΜΑ

Αν το τραίνο καθυστερήσει (p) και δεν υπάρχουν στο σταθμό ταξί ($\neg q$), τότε ο Κώστας αργεί στο ραντεβού του (r).

Ο Κώστας δεν αργεί στο ραντεβού του ($\neg r$).

Το τραίνο έφτασε με καθυστέρηση (p).

ΑΡΑ, υπήρχαν ταξί στο σταθμό (q).

$$p \wedge \neg q \rightarrow r, \neg r, p \vdash q$$

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ X

ΚΑΝΟΝΕΣ ΣΥΖΕΥΞΗΣ

$$\frac{\phi \quad \psi}{\phi \wedge \psi} \wedge i$$

εισαγωγή and

$$\frac{\phi \wedge \psi}{\phi} \wedge e_1$$

$$\frac{\phi \wedge \psi}{\psi} \wedge e_2$$

απαλοιφή and

ΠΑΡΑΔΕΙΓΜΑ

προϋποθέσεις

Αποδείξτε ότι $p \wedge q, r \vdash q \wedge r$

Απόδειξη:

$$\frac{\frac{p \wedge q}{q} \wedge e_2 \quad r}{q \wedge r} \wedge i$$

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XI

ΚΑΝΟΝΕΣ ΔΙΠΛΗΣ ΑΡΝΗΣΗΣ

$$\frac{\neg\neg\phi}{\phi} \quad \neg\neg e$$

απαλοιφή $\neg \neg$

$$\frac{\phi}{\neg\neg\phi} \quad \neg\neg i$$

εισαγωγή $\neg \neg$

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XII

ΚΑΝΟΝΕΣ ΣΥΝΕΠΑΓΩΓΗΣ

$$\frac{\phi \quad \phi \rightarrow \psi}{\psi} \rightarrow e$$

απαλοιφή $\rightarrow$ (modus ponens)

$$\frac{\phi \rightarrow \psi \quad \neg \psi}{\neg \phi} \text{ MT}$$

modus tollens

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XIII

ΚΑΝΟΝΑΣ ΕΙΣΑΓΩΓΗΣ ΣΥΝΕΠΑΓΩΓΗΣ

Ας υποθέσουμε ότι $p \rightarrow q$. Αν προσωρινά υποθέσουμε ότι ισχύει $\neg q$ τότε

1	$p \rightarrow q$	προϋπόθεση
2	$\neg q$	(προσωρινή) υπόθεση
3	$\neg p$	MT 1, 2
4	$\neg q \rightarrow \neg p$	$\rightarrow$ i 2-3

Το συμπέρασμα της γραμμής 4 δεν εξαρτάται από την υπόθεση $\neg q$ και γι' αυτό είναι έξω από το πλαίσιο. Είναι σα να λέμε «Αν είσαι Έλληνας, τότε είσαι Ευρωπαίος» όπου η αλήθεια της πρότασης δεν εξαρτάται από το αν είσαι Έλληνας ή όχι.

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XIV

ΚΑΝΟΝΑΣ ΕΙΣΑΓΩΓΗΣ ΣΥΝΕΠΑΓΩΓΗΣ

$$\frac{\boxed{\begin{array}{c} \phi \\ \cdot \\ \cdot \\ \psi \end{array}}}{\phi \rightarrow \psi} \rightarrow i$$

Για να αποδειχθεί το $\phi \rightarrow \psi$ γίνεται μία προσωρινή υπόθεση ϕ και αποδεικνύεται το ψ με χρήση του ϕ και οποιασδήποτε άλλης προϋπόθεσης ή συμπεράσματος που έχει αποδεχθεί.

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XV

ΘΕΩΡΗΜΑΤΑ

Λογικοί τύποι της μορφής ϕ τέτοιοι ώστε να ισχύει $\vdash \phi$ (δηλαδή η ϕ ισχύει χωρίς προϋποθέσεις) ονομάζονται *θεωρήματα*.

ΠΑΡΑΔΕΙΓΜΑ

1	p	(προσωρινή) υπόθεση
2	$p \rightarrow p$	$\rightarrow$ i 1-1

Αποδείχθηκε ότι $\vdash p \rightarrow p$

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XVI

ΚΑΝΟΝΕΣ ΔΙΑΖΕΥΞΗΣ

$$\frac{\phi}{\phi \vee \psi} \quad \vee i_1$$

εισαγωγή or

$$\frac{\psi}{\phi \vee \psi} \quad \vee i_2$$

εισαγωγή or

$$\frac{\phi \vee \psi \quad \begin{array}{|c|c|} \hline \phi & \psi \\ \hline \cdot & \cdot \\ \cdot & \cdot \\ \hline x & x \\ \hline \end{array}}{x} \quad \vee e$$

απαλοιφή or

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XVII

ΚΑΝΟΝΑΣ ΑΝΤΙΓΡΑΦΗΣ

ΠΑΡΑΔΕΙΓΜΑ

1	p	(προσωρινή) υπόθεση
2	q	(προσωρινή) υπόθεση
3	p	<i>αντιγραφή 1</i>
4	$q \rightarrow p$	$\rightarrow$ i 2-3
5	$p \rightarrow (q \rightarrow p)$	$\rightarrow$ i 1-4

Αποδείχθηκε ότι $\vdash p \rightarrow (q \rightarrow p)$

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ ΧVΙΙΙ

ΑΤΟΠΟ

Άτοπο είναι όλες οι εκφράσεις της μορφής $\phi \wedge \neg \phi$ ή $\phi \wedge \neg \phi$, όπου ϕ είναι κάποιος τύπος.

Στον προτασιακό λογισμό *από το άτοπο μπορούμε να συμπεράνουμε κάθε πιθανό τύπο* και *όλοι οι τύποι που εκφράζουν άτοπο είναι ισοδύναμοι μεταξύ τους*.

$$\neg(r \vee s \rightarrow q) \wedge (r \vee s \rightarrow q) \vdash (p \rightarrow q) \wedge \neg(p \rightarrow q)$$

Το άτοπο το συμβολίζουμε με το $\perp$ ('bottom'), ενώ τύπους της μορφής $\phi \vee \neg \phi$ τους συμβολίζουμε με $\top$ ('top').

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XIX

ΚΑΝΟΝΕΣ ΓΙΑ ΤΟ ΑΤΟΠΟ

$$\frac{}{\phi} \perp e$$

απαλοιφή άτοπου

$$\frac{\phi \quad \neg\phi}{\perp} \neg e$$

απαλοιφή $\neg$

ΠΑΡΑΔΕΙΓΜΑ: απόδειξη ότι $\neg p \vee q \vdash p \rightarrow q$

1 $\neg p \vee q$

2 $\neg p$ υπόθεση

3 p υπόθεση

4 $\perp$ $\neg e$ 3, 2

5 q $\perp e$ 4

6 $p \rightarrow q$ $\rightarrow i$ 3-5

7 $p \rightarrow q$

q υπόθεση

p υπόθεση

q αντιγραφή του 2

$p \rightarrow q$ $\rightarrow i$ 3-4

$\vee e$ 1, 2-6

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XX

ΚΑΝΟΝΕΣ ΓΙΑ ΤΟ ΑΤΟΠΟ

ϕ
.
.
$\perp$

$\neg i$
 $\neg \phi$

εισαγωγή άρνησης

ΠΑΡΑΔΕΙΓΜΑ: απόδειξη ότι $p \rightarrow q, p \rightarrow \neg q \vdash \neg p$

1	$p \rightarrow q$	προϋπόθεση
2	$p \rightarrow \neg q$	προϋπόθεση
3	p	υπόθεση
4	q	$\rightarrow e$ 1, 3
5	$\neg q$	$\rightarrow e$ 2, 3
6	$\perp$	$\neg e$ 4, 5
7	$\neg p$	$\neg i$ 3-6

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXI

ΠΑΡΑΔΕΙΓΜΑ: απόδειξη ότι $p \rightarrow \neg p \vdash \neg p$

1	$p \rightarrow \neg q$	προϋπόθεση
2	p	υπόθεση
3	$\neg p$	$\rightarrow$ e 1, 2
4	$\perp$	$\neg$ e 2, 3
5	$\neg p$	$\neg$ i 2-4

Αυτό το συμπέρασμα φαίνεται ότι δεν είναι λογικό. Αυτό συμβαίνει γιατί η σχέση $\vdash$ καθορίζει το συμπεράσματα που μπορούμε να εξάγουμε λαμβάνοντας ως δεδομένες τις προϋποθέσεις αριστερά του $\vdash$. *Δεν καθορίζει αν οι προϋποθέσεις αυτές είναι λογικές.*

Στη συνέχεια θα δούμε πως το $\vdash$ θα ταυτιστεί με σημασιολογικούς ελέγχους που θα τυποποιηθούν με πίνακες αληθείας: αν όλες οι προϋποθέσεις είναι αληθείς τότε και το συμπέρασμα είναι επίσης αληθές.

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXII

ΠΑΡΑΓΩΓΟΣ ΚΑΝΟΝΑΣ

$$\frac{\phi \rightarrow \psi \quad \neg \psi}{\neg \phi} \text{ MT} \quad \textit{modus tollens}$$

ΑΠΟΔΕΙΞΗ:

1	$\phi \rightarrow \psi$	προϋπόθεση
2	$\neg \psi$	προϋπόθεση
3	ϕ	υπόθεση
4	ψ	$\rightarrow$ e 1, 3
4	$\perp$	$\neg$ e 4, 2
5	$\neg \phi$	$\neg$ i 3-5

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXIII

ΠΑΡΑΓΩΓΟΣ ΚΑΝΟΝΑΣ

$$\frac{\phi}{\neg\neg\phi} \neg\neg i$$

εισαγωγή $\neg\neg$

ΑΠΟΔΕΙΞΗ:

1 ϕ προϋπόθεση

2 $\neg\phi$ υπόθεση

3 $\perp$ $\neg e$ 1, 2

5 $\neg\neg\phi$ $\neg i$ 2-3

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXIV

ΠΑΡΑΓΩΓΟΣ ΚΑΝΟΝΑΣ

$$\frac{\boxed{\begin{array}{c} \neg \phi \\ \cdot \\ \cdot \\ \perp \end{array}}}{\phi} \text{ RAA}$$

*εις άτοπο απαγωγή
(reductio ad absurdum)*

ΑΠΟΔΕΙΞΗ:

1	$\neg \phi \rightarrow \perp$	δεδομένο
2	$\neg \phi$	υπόθεση
3	$\perp$	$\neg$ e 1, 2
4	$\neg \neg \phi$	$\neg$ i 2-3
5	ϕ	$\neg \neg$ e 4

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXV

ΠΑΡΑΓΩΓΟΣ ΚΑΝΟΝΑΣ

$$\frac{}{\phi \vee \neg \phi} \text{LEM}$$

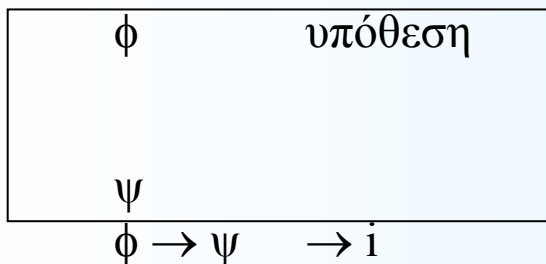
νόμος του αποκλειόμενου μέσου
(*Law of Excluded Middle*
tertium non datur)

ΑΠΟΔΕΙΞΗ:

1	$\neg(\phi \vee \neg \phi)$	υπόθεση
2	ϕ	υπόθεση
3	$\phi \vee \neg \phi$	$\vee i_1 2$
4	$\perp$	$\neg e 3, 1$
5	$\neg \phi$	$\neg i 2-4$
6	$\phi \vee \neg \phi$	$\vee i_2 5$
7	$\perp$	$\neg e 6, 1$
8	$\neg\neg(\phi \vee \neg \phi)$	$\neg i 1-7$
9	$\phi \vee \neg \phi$	$\neg\neg e 8$

**ΠΩΣ ΧΡΗΣΙΜΟΠΟΙΟΥΜΕ ΤΟΥΣ ΚΑΝΟΝΕΣ ΛΟΓΙΚΩΝ
ΣΥΝΠΕΡΑΣΜΑΤΩΝ ΓΙΑ ΤΗΝ ΚΑΤΑΣΚΕΥΗ ΜΙΑΣ ΑΠΟΔΕΙΞΗΣ;**

·
·
προϋποθέσεις



ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXVII

Πολύ χρήσιμος κανόνας εκτός από τον $\rightarrow i$ είναι επίσης και ο $\neg i$ γιατί μας αποφέρει μία έξτρα προϋπόθεση και συχνά απλοποιεί την πορεία προς το επιθυμητό συμπέρασμα.

Σε κάθε βήμα μπορεί να υπάρχουν περισσότεροι από έναν κανόνες που είναι πιθανό να βρίσκουν εφαρμογή. Πέρα από τους $\rightarrow i$ και $\neg i$ που συνιστάται η χρήση τους όποτε αυτό είναι εφικτό καλό είναι σε κάθε βήμα να καταγράφουμε όλους τους κανόνες που είναι υποψήφιοι για εφαρμογή και να επιλέγουμε τελικά τον κανόνα που είναι πιο πιθανό να βελτιώνει την κατάσταση.

ΟΡΙΣΜΟΣ Δύο τύποι ϕ και ψ λέμε ότι είναι *αποδείξιμα ισοδύναμοι* αν και μόνο αν

$$\phi \vdash \psi$$

$$\psi \vdash \phi$$

Αυτό το γράφουμε με $\phi \dashv\vdash \psi$, που σημαίνει ότι $\vdash(\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)$

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXVIII

ΑΠΟΔΕΙΞΙΜΑ ΙΣΟΔΥΝΑΜΟΙ ΤΥΠΟΙ

$$\neg(p \wedge q) \dashv\vdash \neg q \vee \neg p$$

$$\neg(p \vee q) \dashv\vdash \neg p \wedge \neg q$$

$$p \rightarrow q \dashv\vdash \neg q \rightarrow \neg p$$

$$p \rightarrow q \dashv\vdash \neg p \vee q$$

$$p \wedge q \rightarrow r \dashv\vdash p \rightarrow (q \rightarrow r)$$

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXIX

ΣΗΜΑΣΙΑ ΠΡΟΤΑΣΙΑΚΟΥ ΛΟΓΙΣΜΟΥ

Εδώ, αντί για λογικά συμπεράσματα στη βάση της σχέσης

$$\phi_1, \phi_2, \dots, \phi_n \vdash \psi$$

μας ενδιαφέρουν λογικά συμπεράσματα στη βάση της σχέσης

$$\phi_1, \phi_2, \dots, \phi_n \models \psi$$

όπου για κάθε τύπο πρέπει να μπορούμε να ελέγξουμε την αλήθειά του βάση των πινάκων αληθείας για τους λογικούς τελεστές που χρησιμοποιήσαμε.

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ XXX

ϕ	ψ	$\phi \wedge \psi$
T	T	T
T	F	F
F	T	F
F	F	F

ϕ	ψ	$\phi \vee \psi$
T	T	T
T	F	T
F	T	T
F	F	F

ϕ	ψ	$\phi \rightarrow \psi$
T	T	T
T	F	F
F	T	T
F	F	T

ϕ	$\neg \phi$
T	F
F	T

$\top$
T

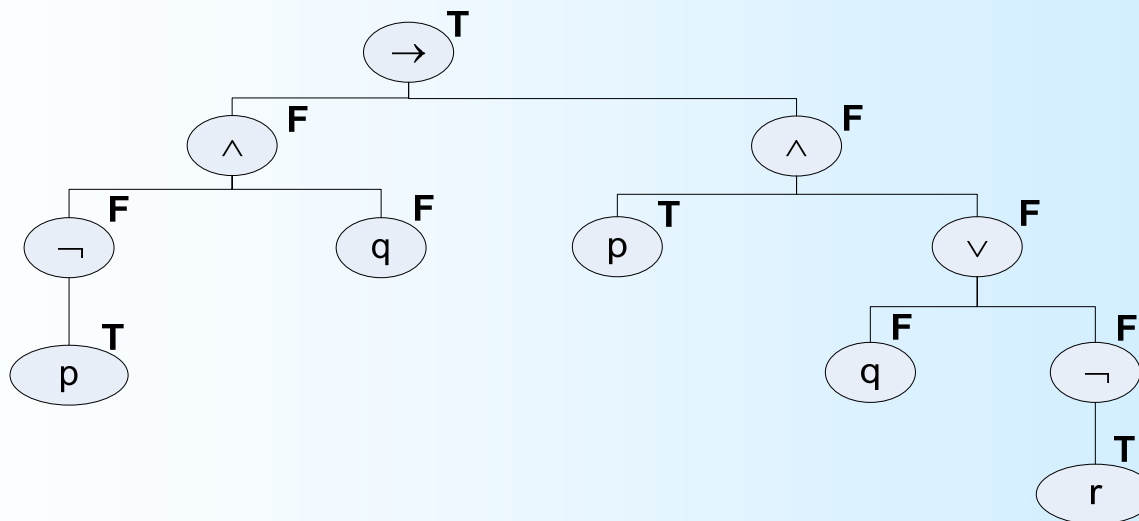
$\perp$
F

ελέγχεται αν διατηρείται η αλήθεια

ΠΡΟΤΑΣΙΑΚΟΣ ΛΟΓΙΣΜΟΣ ΧΧΧΙ

Συντακτικό δέντρο τύπου: $\neg p \wedge q \rightarrow p \wedge (q \vee \neg r)$

Η αποτίμηση της τιμής του γίνεται με βάση τους στοιχειώδεις τύπους p , q και r και με postorder διάσχιση του δένδρου με χρήση των πινάκων αληθείας.



ΚΑΤΗΓΟΡΗΜΑΤΙΚΟΣ ΛΟΓΙΣΜΟΣ I

- Για τον προτασιακό λογισμό παρουσιάσαμε
 - την αποδεικτική θεωρία (natural deduction/λογικό συμπέρασμα)
 - τη σύνταξη (ορίζεται με γραμματική χωρίς συμφραζόμενα και εκφράζεται με συντακτικά δέντρα)
 - τη σημασία (πίνακες αληθείας)
- Αυτά είναι τα βασικά στοιχεία που συγκροτούν μία τυπική γλώσσα.
- Ο προτασιακός λογισμός είναι μία **τυπική γλώσσα κατάλληλη για δηλωτικές προτάσεις**, προτάσεις δηλαδή για τις οποίες μπορεί να δοθεί μία τιμή αληθείας.

ΚΑΤΗΓΟΡΗΜΑΤΙΚΟΣ ΛΟΓΙΣΜΟΣ II

- Ενώ ο προτασιακός λογισμός λειτουργεί ικανοποιητικά για προτάσεις με τμήματα *not*, *and*, *or* δεν είναι επαρκής για την περιγραφή άλλων προσδιορισμών που απαντώνται σε φυσικές ή τεχνητές γλώσσες όπως για παράδειγμα υπάρχει (*there exists*), όλοι (*all*), μόνο (*only*) κλπ.
- Για να καλυφθεί αυτό το κενό δημιουργήθηκε ο **κατηγορηματικός λογισμός** (*predicate logic*), που είναι επίσης γνωστός ως **λογική πρώτης τάξης** (*first order logic*).

ΚΑΤΗΓΟΡΗΜΑΤΙΚΟΣ ΛΟΓΙΣΜΟΣ ΙΙΙ

ΠΑΡΑΔΕΙΓΜΑ

«Κάθε φοιτητής είναι νεότερος από κάποιο καθηγητή»

Στον προτασιακό λογισμό η παραπάνω πρόταση δεν μπορεί παρά να είναι ένας πρωταρχικός (ατομικός) τύπος με συγκεκριμένη τιμή αληθείας, αλλά εμείς ενδιαφερόμαστε να αναδείξουμε μία πιο λεπτομερή εικόνα της λογικής δομής της πρότασης.

Για το σκοπό αυτό μπορούμε να χρησιμοποιούμε *κατηγορήματα* της μορφής:

S (Μαρία) που εκφράζει το ότι η Μαρία είναι φοιτήτρια

I (Παναγιώτης) που εκφράζει το ότι ο Παναγιώτης είναι καθηγητής

Y (Μαρία, Παναγιώτης) που εκφράζει ότι η Μαρία είναι νεότερη από τον Παναγιώτη

Πρέπει να είμαστε προσεκτικοί στον ορισμό της σημασίας των κατηγορημάτων γιατί θα μπορούσε π.χ. η σημασία του Y να ερμηνεύεται με διαφορετικό τρόπο από τον επιθυμητό.

ΚΑΤΗΓΟΡΗΜΑΤΙΚΟΣ ΛΟΓΙΣΜΟΣ IV

Θέλουμε επίσης να μπορούμε να εκφράζουμε κατηγορήματα με προσδιοριστές της μορφής *για κάθε ή υπάρχει*. Αυτό θα μπορούσε να γίνει αν γράφαμε

$$S(\cdot)$$

όπου η $\cdot$ θα μπορούσε να αντικαθιστάται από το όνομα του κάθε φοιτητή.

Αυτό όμως δεν θα βόλευε γιατί θα είχε ως συνέπεια όταν θα θέλαμε να κωδικοποιήσουμε μία πρόταση για την εκτέλεση ενός προγράμματος να πρέπει να γράψουμε την ίδια πρόταση για κάθε κατάσταση του προγράμματος.

Για το λόγο αυτό χρησιμοποιούμε *μεταβλητές*.

ΚΑΤΗΓΟΡΗΜΑΤΙΚΟΣ ΛΟΓΙΣΜΟΣ V

Οι μεταβλητές γράφονται ως

$x, y, z, \dots$ ή x_1, x_2, x_3

και ουσιαστικά είναι σύμβολα αντικαθιστάμενα από συγκεκριμένες τιμές.

Με τη χρήση μεταβλητών μπορούμε να ορίσουμε και τυπικά την σημασία των κατηγορημάτων:

$S(x)$: ο x είναι φοιτητής

$I(x)$: ο x είναι καθηγητής

$Y(x, y)$: ο x είναι νεότερος από τον y

Η χρήση μεταβλητών δεν αρκεί για τον ορισμό κατηγορημάτων, αλλά χρειαζόμαστε και **ποσοδείκτες** όπως $\forall$ (για κάθε), $\exists$ (υπάρχει), που πάντα συνοδεύονται από όνομα μεταβλητής.

ΚΑΤΗΓΟΡΗΜΑΤΙΚΟΣ ΛΟΓΙΣΜΟΣ VI

ΠΑΡΑΔΕΙΓΜΑ

«Κάθε φοιτητής είναι νεότερος από κάποιο καθηγητή»

«**Κάθε** φοιτητής x είναι νεότερος από **κάποιο** καθηγητή y »

$$\forall x (S(x) \rightarrow (\exists y (I(y) \wedge Y(x, y))))$$

ΠΑΡΑΔΕΙΓΜΑ II

«Δεν μπορούν να πετάξουν **όλα** τα πουλιά»

$B(x)$: το x είναι πουλί

$F(x)$: το x πετάει

$$\neg (\forall x (B(x) \rightarrow F(x))) \text{ ή}$$

$$\exists x (B(x) \wedge \neg F(x))$$

ΚΑΤΗΓΟΡΗΜΑΤΙΚΟΣ ΛΟΓΙΣΜΟΣ VII

Οι δύο προτάσεις του τελευταίου παραδείγματος είναι *σημασιολογικά ισοδύναμες*.

- Εκείνο που χρειάζεται είναι μία αποδεικτική θεωρία που να επιτρέπει την εξαγωγή συμπερασμάτων συμβολικά ($\vdash$) ή σημασιολογικά ($\models$).
- Επίσης, η αποτίμηση τύπων κατηγορηματικού λογισμού είναι αρκετά διαφορετική από τον υπολογισμό με βάση τους πίνακες αληθείας που είδαμε για τον προτασιακό λογισμό.

ΤΕΧΝΙΚΕΣ ΤΥΠΙΚΗΣ ΕΠΑΛΗΘΕΥΣΗΣ Ι

- *Τυπική επαλήθευση βάση απόδειξης*

Η περιγραφή του συστήματος δίνεται ως ένα σύνολο τύπων Γ σε κάποια γλώσσα λογικής και η ιδιότητα που πρέπει να επαληθευτεί δίνεται από κάποιο τύπο έστω ϕ .

Η μέθοδος επαλήθευσης αποσκοπεί στο να βρεθεί μία απόδειξη ότι $\Gamma \vdash \phi$.

- *Τυπική επαλήθευση βάση μοντέλου*

Στις προσεγγίσεις που βασίζονται σε μοντέλα το σύστημα περιγράφεται από ένα πεπερασμένο μοντέλο M κατάλληλο για κάποια γλώσσα λογικής. Η ιδιότητα επίσης εκφράζεται από κάποιο τύπο ϕ και η επαλήθευση είναι ένας υπολογισμός που εξακριβώνει αν ένα μοντέλο M ικανοποιεί τον τύπο ϕ ($M \models \phi$). Αυτός ο έλεγχος γίνεται συνήθως αυτόματα.

ΤΕΧΝΙΚΕΣ ΤΥΠΙΚΗΣ ΕΠΑΛΗΘΕΥΣΗΣ II

- Μία αποδεικτική θεωρία όταν λέμε ότι είναι *καλά ορισμένη* (sound) και *πλήρης* (complete) και αυτό αποδεικνύεται, εννοούμε ότι $\Gamma \vdash \phi$ αν και μόνο αν $\Gamma \models \phi$, όπου το τελευταίο σημαίνει ότι για όλα τα μοντέλα M αν $M \models \Gamma$, τότε $M \models \phi$.
- Άρα δυνητικά η τυπική επαλήθευση βάση μοντέλου είναι απλούστερη από την τυπική επαλήθευση βάση απόδειξης γιατί βασίζεται σε ένα μόνο μοντέλο και όχι σε ένα πιθανά άπειρο σύνολο μοντέλων.

ΕΠΑΛΗΘΕΥΣΗ ΠΡΟΓΡΑΜΜΑΤΩΝ I

- Η τυπική επαλήθευση βάση μοντέλου είναι κατάλληλη για συστήματα επικοινωνούντων διεργασιών (π.χ. κατανεμημένα συστήματα) όπου το βασικό πρόβλημα είναι ο έλεγχος αλλά γενικά δεν υπάρχουν σύνθετα δεδομένα. Βασικά στηρίζομαστε στην υπόθεση ότι τα μοντέλα που προκύπτουν από μία αφαιρετική θεώρηση των συστημάτων που περιγράφουν είναι *πεπερασμένων καταστάσεων*.
- Αυτή όμως η υπόθεση δεν ισχύει για (σειριακά) εκτελέσιμα προγράμματα που μπορεί να διαχειρίζονται μη τετριμμένα δεδομένα και μπορεί π.χ. να έχουν μεταβλητές τύπου integer, λίστας, δέντρου οπότε ουσιαστικά μιλάμε για μηχανές με *άπειρο χώρο καταστάσεων*.

ΕΠΑΛΗΘΕΥΣΗ ΠΡΟΓΡΑΜΜΑΤΩΝ II

- Οι τεχνικές επαλήθευσης που χρησιμοποιούνται για προγράμματα δεν κάνουν εξαντλητικό έλεγχο κάθε κατάστασης του συστήματος αφού αυτό θα ήταν ανέφικτο μια και μιλάμε για άπειρα πολλές καταστάσεις.
- Αντί γι' αυτό συνήθως αποβλέπουμε στην κατασκευή μιας απόδειξης για την ιδιότητα που ενδιαφέρει χρησιμοποιώντας μία θεωρία αποδεικτικής που παρακάμπτει την απαίτηση να ελεγχθούν άπειρα πολλά μοντέλα ενός κάποιου συνόλου τύπων κατηγορηματικού λογισμού.

ΕΠΑΛΗΘΕΥΣΗ ΠΡΟΓΡΑΜΜΑΤΩΝ III

- Παρά το γεγονός ότι πολλά από τα βήματα που απαιτούνται για να επαληθευτεί ότι ένα πρόγραμμα ικανοποιεί κάποια προδιαγραφή είναι μηχανιστικά (και άρα αυτοματοποιούνται) δεν υπάρχει εγγύηση ότι αυτό μπορεί να γίνει για όλες τις περιπτώσεις αλγοριθμικά. Αν και υπάρχουν προσεγγίσεις που βοηθούν τον προγραμματιστή να φέρει σε πέρας μία τέτοια διαδικασία, είναι βέβαιο ότι μιλάμε για μία ημιαυτόματη διαδικασία.
- Παράδειγμα τεχνικής επαλήθευσης προγραμμάτων: Hoare triples
- Παράδειγμα προγραμματιστικών βοηθημάτων: Java Modeling Language ή Object Constraint Language (C++)