

Λογικός Προγραμματισμός

Μανόλης Μαρακάκης, Καθηγητής
mmarak@cs.hmu.gr

Τμήμα Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών
Σχολή Μηχανικών
Ελληνικό Μεσογειακό Πανεπιστήμιο

Περίγραμμα Ύλης Θεωρίας

- ❑ 1. Εισαγωγή
- ❑ 2. Αναπαράσταση γνώσης και συλλογιστική στον Προτασιακό Λογισμό (ΠΛ) .
 - Τύποι και η αλήθειά τους.
 - Λογικές ισοδυναμίες και μετασχηματισμοί τύπων σε κανονικές μορφές στο προτασιακό λογισμό.
 - Σημασιολογική συνέπεια.
 - Εξαγωγή συμπερασμάτων και η επίλυση (Resolution) στο προτασιακό λογισμό.

- ❑ 3. Αναπαράσταση γνώσης και συλλογιστική στον Κατηγορηματικό Λογισμός (ΚΛ) ή Λογική Πρώτης Τάξης (ΛΠΤ).
 - Συντακτικά συστατικά του κατηγορηματικού λογισμού.
 - Ερμηνείες προτάσεων, σημασιολογική συνέπεια.
 - Λογικές ισοδυναμίες, μετασχηματισμοί τύπων σε κανονικές μορφές.
 - Εξαγωγή συμπερασμάτων στο κατηγορηματικό λογισμό.

Περίγραμμα Ύλης Θεωρίας

- ❑ 4. Αντικατάσταση. Ενοποίηση (Unification). Η επίλυση στον κατηγορηματικό λογισμό. SLD-Επίλυση και SLD δέντρα.
- ❑ 5. Οριστικά (Definite) λογικά προγράμματα.
- ❑ 6. Άρνηση στο λογικό προγραμματισμό. Κανονικά λογικά προγράμματα (Normal logic programs).
- ❑ 7. Γενικά λογικά προγράμματα.
- ❑ 8. Εφαρμογές του Λογικού Προγραμματισμού.

Αξιολόγηση Θεωρητικού Μέρους του Μαθήματος

❑ **Πρόοδος** (προαιρετική):

- Ένα διαγώνισμα στη μισή ύλη του μαθήματος, βαθμολογείται με 50% του τελικού βαθμού.

❑ **Τελικό διαγώνισμα**

- εφόσον ο φοιτητής έχει περάσει το διαγώνισμα προόδου τότε θα εξεταστεί στην υπόλοιπη μισή ύλη του μαθήματος, βαθμολογείται με 50% του τελικού βαθμού
- διαφορετικά (δηλαδή ο φοιτητής είτε δεν πέρασε τη πρόοδο ή δεν έδωσε το διαγώνισμα προόδου), θα εξεταστεί σε όλη την ύλη του μαθήματος, βαθμολογείται 100% του τελικού βαθμού.

Βιβλιογραφία Μαθήματος

1. Μ. Μαρακάκης, Prolog: Προγραμματισμός σε Λογική για Τεχνητή Νοημοσύνη, Εκδόσεις Νέων Τεχνολογιών, 2019, ISBN: **978-960-578-055-5**. Κωδικός Βιβλίου στον Εύδοξο: 86200975.
2. U. Nilsson and J. Maluszynski, Logic, Programming and Prolog, Second edition, John Wiley & Sons, 1995, ISBN: 0 471 95996 0.
3. I. Bratko, Prolog Programming for Artificial Intelligence, Pearson Education Canada, 4th edition, 2011, ISBN 13: 9780321417466.
4. Η. Σακελλαρίου, Ν. Βασιλειάδης, Π. Κεφαλάς, Δ. Σταμάτης, Τεχνικές Λογικού Προγραμματισμού – Η Γλώσσα Prolog, Ελληνικά Ακαδημαϊκά Συγγράμματα και Βοηθήματα, www.kallipos.gr, ISBN: 978-960-603-246-2, Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών (ΣΕΑΒ), 2015.
5. Γ. Μητακίδης, Από τη Λογική στο Λογικό Προγραμματισμό και την Prolog, εκδόσεις Καρδαμίτσα, 1992, ISBN: 960-7262-59-X.

Εκπαιδευτική βοήθεια

- ❑ Ώρες σύγχρονης (ζωντανής) διαδικτυακής επικοινωνίας μέσω του eclass.
 - Κάθε **Τρίτη 14:00-16:00**.
- ❑ Για θέματα που αφορούν το μάθημα όπως απορίες σε ασκήσεις, εργασίες, προγραμματιστικά προβλήματα με Prolog, κτλ.
 - Από τον διδάσκοντα στις ώρες σύγχρονης (ζωντανής) διαδικτυακής επικοινωνίας.
 - Θα ανακοινωθεί μέσω του eClass επιπλέον βοήθεια για προγραμματιστικά προβλήματα.

Λογικός Προγραμματισμός

Μάθημα 1

□ 1. Εισαγωγή

- 1.1 Γιατί Λογικός Προγραμματισμός;
- 1.2 Τι είναι Λογικός Προγραμματισμός;

1. Εισαγωγή.

1.1. Γιατί Λογικός Προγραμματισμός;

- ❑ **Προγραμματιστική Μέθοδος/Προσέγγιση (programming paradigm)** είναι ο τρόπος που προγραμματίζουμε. Δηλαδή, ο τρόπος που βλέπουμε ένα πρόβλημα (τον κόσμο) όταν το γράφουμε σε πρόγραμμα.
 - Μια **μέθοδος/προσέγγιση προγραμματισμού** τείνει **να επιβάλει κάποια εικόνα/άποψη για το πρόβλημα** (τον κόσμο) στους χρήστες της.
 - **Κάθε μέθοδος/προσέγγιση προγραμματισμού** έχει μια σημασιολογία η οποία μας οδηγεί να σκεφτόμαστε και να βλέπουμε το πρόβλημα με βάση αυτή τη σημασιολογία.

1. Εισαγωγή.

1.1. Γιατί Λογικός Προγραμματισμός;

❑ Οι κυριότερες προγραμματιστικές μέθοδοι/προσεγγίσεις (programming paradigms) είναι οι εξής:

➤ **Προστακτική (Imperative) μέθοδος/προσέγγιση.** Με αυτή την προσέγγιση η ροή ελέγχου είναι μια ακολουθία από εντολές. Το πρόγραμμα λέει στον υπολογιστή **πώς** να κάνει κάτι και ως συνέπεια της ακολουθίας αυτών των πράξεων **τί** αποτέλεσμα θα πάρει. Οι γλώσσες που ακολουθούν αυτή την προσέγγιση χαρακτηρίζονται ως **προστακτικές γλώσσες**.

➤ **Δηλωτική (Declarative) μέθοδος/προσέγγιση.** Το πρόγραμμα λέει στον υπολογιστή **τί** θέλει να κάνει το πρόγραμμα (δηλαδή **τί** πρόβλημα να λύσει), και ο υπολογιστής καθορίζει **πώς** θα το κάνει (δηλαδή καθορίζει **πώς** θα λύσει το πρόβλημα). Οι γλώσσες που ακολουθούν αυτή την προσέγγιση χαρακτηρίζονται ως **δηλωτικές γλώσσες**.

1. Εισαγωγή.

1.1. Γιατί Λογικός Προγραμματισμός;

❑ Κύρια χαρακτηριστικά των **προστακτικών (imperative)** γλωσσών προγραμματισμού.

- Η σημασιολογία τους βασίζεται σε καταστάσεις.
- Οι **υπολογισμοί (computations)** θεωρούνται ως μια διαδικασία μετάβασης από μια κατάσταση σε μια άλλη. Η σειρά με την οποία γίνονται οι υπολογισμοί είναι σημαντική. Δηλαδή η σειρά εκτέλεσης των πράξεων και των εντολών είναι σημαντική.
- Κατηγορίες προστακτικών γλωσσών προγραμματισμού.
 - ❖ Διαδικαστικές γλώσσες.
 - ❖ Αντικειμενοστραφείς γλώσσες.
- Παραδείγματα: C, Pascal, Java, κτλ .
- Κάθε βήμα **υπολογισμών (computation)** σε αυτές τις γλώσσες περιγράφει τη διαδικασία αλλαγής μιας **κατάστασης**.

1. Εισαγωγή.

1.1. Γιατί Λογικός Προγραμματισμός;

❑ Κύρια χαρακτηριστικά των **δηλωτικών (declarative)** γλωσσών προγραμματισμού.

- Η σημασιολογία τους εστιάζεται στη λογική (δηλαδή στο **τί** κάνει το πρόγραμμα) και όχι στον έλεγχο (δηλαδή στο **πώς** θα το κάνει).
- Κατηγορίες δηλωτικών γλωσσών:
 - ❖ **Λογικός (Logic) προγραμματισμός.** Οι **υπολογισμοί (computations)** είναι μια διαδικασία συλλογισμών, π.χ. Prolog.
 - ❖ **Συναρτησιακός (Functional) προγραμματισμός.** Οι **υπολογισμοί (computations)** είναι υπολογισμοί συναρτήσεων, π.χ. Lisp, Scheme, κτλ. Ο συναρτησιακός προγραμματισμός αντιμετωπίζει/βλέπει την λύση του προβλήματος σαν ένα σύνολο από συναρτήσεις που πρέπει να εκτελεστούν για να λυθεί.
 - ❖ **Περιοριστικός (Constraint) προγραμματισμός.** Οι **υπολογισμοί (computations)** θεωρούνται ως ένα πρόβλημα ικανοποίησης περιορισμών, π.χ. Prolog III, CHIP, κτλ. Οι περιορισμοί καθορίζουν τις ιδιότητες της λύσης που πρέπει να βρεθεί.

1. Εισαγωγή.

1.1. Γιατί Λογικός Προγραμματισμός;

❑ Επίπεδο προγραμματισμού που υποστηρίζει η γλώσσα.

➤ Χαμηλό επίπεδο.

❖ Προγραμματισμός σε χαμηλό επίπεδο βλέπει τον κόσμο (δηλαδή το πρόβλημα) **κοντά στον τρόπο με το οποίο ο υπολογιστής κάνει τους υπολογισμούς.**

➤ Υψηλό επίπεδο.

❖ Προγραμματισμός σε υψηλό επίπεδο βλέπει τον κόσμο (δηλαδή το πρόβλημα) **κοντά στις προδιαγραφές του προβλήματος.** Δηλαδή με αυτό το είδος προγραμματισμού περιγράφεις το πρόβλημα που χρειάζεται να λυθεί και όχι τον τρόπο λύσης του.

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ Υπάρχουν πολλές απόψεις με επικαλύψεις, από τις οποίες μπορούμε να δούμε το Λογικό Προγραμματισμό (ΛΠ).

- 1. Οι **υπολογισμοί (computations)** είναι μια διαδικασία **παραγωγής (deduction)**.
- 2. Έχουμε απόδειξη θεωρημάτων.
- 3. Έχουμε δηλωτικό προγραμματισμό σε αντίθεση με τον προστακτικό ή διαδικαστικό προγραμματισμό.
- 4. Υπάρχει άλλη έννοια για το τι είναι αλγόριθμος.

Αλγόριθμος = Λογική + Έλεγχος.

- 5. Είναι μια πολύ υψηλού επιπέδου γλώσσα προγραμματισμού.
- 6. Έχουμε διαδικαστική ερμηνεία μιας δηλωτικής προδιαγραφής.

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ 1. Οι υπολογισμοί σαν διαδικασία παραγωγής (deduction).

- Ο λογικός προγραμματισμός προσφέρει ένα διαφορετικό παράδειγμα για τους υπολογισμούς.

Ο υπολογισμός (computation) είναι μια λογική παραγωγή (deduction).

- Χρησιμοποιεί τη γλώσσα της λογικής για να εκφράσει δεδομένα και προγράμματα.

$\forall X \ \forall Y \text{ πατέρας}(X, Y) \text{ εάν γονέας}(X, Y) \text{ και } \text{άρρεν}(X).$

- Οι γλώσσες του ΛΠ χρησιμοποιούν την Λογική Πρώτης Τάξης (ΛΠΤ) ή γνωστό ως Κατηγορηματικό Λογισμό Πρώτης Τάξης ή απλά Κατηγορηματικό Λογισμό (ΚΛ).

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ 1. Οι υπολογισμοί σαν διαδικασία παραγωγής (συνέχεια) .

- Η έννοια της πρώτης τάξης αναφέρεται στο ότι μπορούμε να έχουμε ποσοδείκτες πάνω σε **αντικείμενα** (σταθερές, μεταβλητές) αλλά όχι πάνω σε **σχέσεις (κατηγορήματα)** ή **συναρτήσεις**.

Μπορούμε να πούμε «Όλοι οι άνθρωποι είναι θνητοί»

$\forall X/\text{Άνθρωπος} \quad \text{θνητός}(X)$

αλλά δεν μπορούμε να πούμε «Όλες οι ιδιότητες του Γιάννη,
π.χ. ψηλός, εργατικός, επιμελής, κτλ»,

$\forall P \quad P(\text{γιαννης})$

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ 2. Απόδειξη θεωρημάτων.

- Ο λογικός προγραμματισμός χρησιμοποιεί την έννοια της **αυτόματης απόδειξης θεωρημάτων** σαν ένα μεταφραστή.
- Η διαδικασία απόδειξης θεωρημάτων (theorem proving or the proof) παράγει μια επιθυμητή λύση από ένα **αρχικό σύνολο αξιωμάτων**.
- Η λύση πρέπει να είναι «**δημιουργική**» (constructive) ώστε περισσότερες από μια αληθείς ή ψευδείς απαντήσεις να εξαχθούν.

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ 3. Δηλωτικός προγραμματισμός.

- Οι γλώσσες του ΛΠ είναι δηλωτικές γλώσσες προγραμματισμού.
- Σε μια δηλωτική γλώσσα καθορίζεται **τί** χρειάζεται να υπολογιστεί αλλά όχι **πώς** θα υπολογιστεί.
- Ο προγραμματιστής πρέπει να καθορίσει.
 - ❖ Το σύνολο των αντικειμένων που εμπλέκονται στους υπολογισμούς.
 - ❖ Τις σχέσεις που υπάρχουν μεταξύ αυτών των αντικειμένων.
 - ❖ Οι περιορισμοί που πρέπει να ισχύουν στο πρόβλημα για να λυθεί.
- Ο μεταφραστής ή ο μεταγλωττιστής της γλώσσας του λογικού προγραμματισμού αποφασίζει **πώς** θα ικανοποιηθούν οι περιορισμοί.

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ 4. Άλλη έννοια για το τι είναι αλγόριθμος.

- Ο προστακτικός ή διαδικαστικός προγραμματισμός εκφράζεται από την αποφθεγματική εξίσωση του Wirth

Προγράμματα = Αλγόριθμοι + Δομές Δεδομένων

- Ο δηλωτικός προγραμματισμός σε γλώσσες Τεχνητής Νοημοσύνης όπως είναι η Prolog εκφράζεται από την αποφθεγματική εξίσωση του Kowalski

Αλγόριθμος = Λογική + Έλεγχος.

ή ισοδύναμα

Πρόγραμμα = Λογική + Έλεγχος.

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

□ 4. Άλλη έννοια για το τι είναι αλγόριθμος (συνεχεια).

➤ **Αλγόριθμος = Λογική + Έλεγχος.**

ή ισοδύναμα

Πρόγραμμα = Λογική + Έλεγχος.

➤ Μπορούμε να δούμε το τμήμα της **Λογικής** ως:

❖ «**μια προδιαγραφή των ουσιαστικών περιορισμών του προβλήματος**».

➤ Μπορούμε να δούμε το τμήμα του **ελέγχου** ως:

❖ «**ένα σύμβουλο (ένα μηχανισμό) σε μια υπολογιστική μηχανή (μεταφραστή ή μεταγλωττιστή) που την καθοδηγεί πώς πρέπει να προχωρήσει, η υπολογιστική μηχανή, για να ικανοποιηθούν οι περιορισμοί του προβλήματος**».

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ 5. Μια πολύ υψηλού επιπέδου γλώσσα προγραμματισμού.

- Μια καλή γλώσσα προγραμματισμού δεν πρέπει να επιβαρύνει το προγραμματιστή με μη ουσιαστικές λεπτομέρειες.
- Η εξέλιξη των γλωσσών προγραμματισμού ήταν προς την κατεύθυνση της απεμπλοκής του προγραμματιστή από λεπτομέρειες που δεν σχετιζόταν με το πρόβλημα αλλά αφορούσαν τη διαδικασία λύσης του προβλήματος και τον υπολογιστή (συμβολικές /assembly γλώσσες, Fortran, ALGOL, Pascal, C, Java) .
- Οι γλώσσες του ΛΠ είναι μια ομάδα γλωσσών οι οποίες προσπαθούν να απεμπλέξουν τον προγραμματιστή από τον να εισάγει στο πρόγραμμα και τον μηχανισμό ελέγχου των υπολογισμών.

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ 6. Διαδικαστική ερμηνεία μια δηλωτικής προδιαγραφής.

➤ Ας θεωρήσουμε την λογική έκφραση

❖ $\forall X \ \forall Y \text{ πατέρας}(X, Y) \text{ εάν γονέας}(X, Y) \text{ και } \text{άρρεν}(X).$

➤ Μπορεί να γραφτεί σε Prolog ως εξής:

❖ $\text{πατέρας}(X, Y) \text{ :- } \text{γονέας}(X, Y), \text{ } \text{άρρεν}(X).$

➤ Αυτή η έκφραση έχει δύο ερμηνείες:

❖ Τη **δηλωτική ερμηνεία**:

- «Ο X είναι πατέρας του Y εάν ο X είναι γονέας του Y και ο X είναι άρρεν».
- Δηλαδή ως μια πρόταση αληθών προϋποθέσεων, «ο X είναι γονέας του Y και ο X είναι άρρεν», η οποία πρέπει να είναι αληθής για να ικανοποιείται η σχέση του «πατέρας».

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός;

❑ 6. Διαδικαστική ερμηνεία μια δηλωτικής προδιαγραφής (**συνέχεια**) .

➤ Ας θεωρήσουμε την λογική έκφραση

❖ $\forall X \ \forall Y \text{ πατέρας}(X, Y) \text{ εάν γονέας}(X, Y) \text{ και } \text{άρρεν}(X).$

➤ Μπορεί να γραφτεί σε Prolog ως εξής:

❖ $\text{πατέρας}(X, Y) \text{ :- } \text{γονέας}(X, Y), \text{άρρεν}(X).$

➤ Αυτή η έκφραση έχει δύο ερμηνείες:

❖ Τη **διαδικαστική ερμηνεία**:

- «Για να αποδειχθεί ότι ο X είναι πατέρας του Y πρέπει να αποδειχθεί ότι ο X είναι γονέας του Y και ότι ο X είναι άρρεν».
- Δηλαδή μια περιγραφή τι πρέπει να αποδειχθεί για να είναι αληθής η σχέση πατέρας

1. Εισαγωγή.

1.2. Τι είναι ο Λογικός Προγραμματισμός.

- ❑ **Ορισμός: Λογικός Προγραμματισμός** είναι μια προγραμματιστική μέθοδος/προσέγγιση η οποία βασίζεται στη τυπική λογική.
 - Ένα πρόγραμμα σε μια γλώσσα του λογικού προγραμματισμού είναι ένα σύνολο από προτάσεις σε λογική οι οποίες εκφράζουν γεγονότα και κανόνες για το πεδίο ενός προβλήματος.

Τέλος Μαθήματος 1

Ευχαριστώ!