

ΕΙΣΑΓΩΓΗ

Ενώ δεκάδες (ίσως και εκατοντάδες) γλώσσες προγραμματισμού έχουν αναπτυχθεί και υλοποιηθεί από διάφορες ερευνητικές ομάδες, εταιρείες ή και διεθνείς επιτροπές, πολλές από αυτές, δεν χρησιμοποιήθηκαν ποτέ έξω από τα πλαίσια των ομάδων που τις σχεδίασαν. Άλλες πάλι ενώ χρησιμοποιήθηκαν ιδιαίτερα για ένα μικρό διάστημα, αντικαταστάθηκαν από πιο σύγχρονες. Παρ' όλα αυτά υπάρχει σήμερα ένας αρκετά μεγάλος αριθμός από γλώσσες προγραμματισμού εν' ενεργεία. Ποιά η αναγκαιότητα λοιπόν για μία ακόμα γλώσσα προγραμματισμού; Γιατί η PROLOG;

Η Λογική ως Γλώσσα Προγραμματισμού

Η **Συμβολική Λογική** (*Symbolic Logic*) αποτελεί το προϊόν των προσπαθειών για ένα μαθηματικοποιημένο φορμαλισμό των ιδιοτήτων της λογικής της ανθρώπινης σκέψης. Μέχρι πρόσφατα μελετιόταν ανεξάρτητα από την προοπτική να χρησιμοποιηθεί σαν γλώσσα για την επικοινωνία ανθρώπου - Υπολογιστή. Η προοπτική αυτή άρχισε να γίνεται ορατή μετά από τις εξελίξεις στην Υπολογιστική Λογική, με τη διατύπωση της **"Αρχής της Ανάλυσης"** (*Resolution principle*) από το Robinson το 1965 [ROBINSON, 1965]. Λίγα χρόνια αργότερα ο R. Kowalski [KOWALSKI, 1974] απέδειξε ότι οι **"Φράσεις του Horn"** (*Horn Clauses*), που αποτελούν ένα περιορισμένο υποσύνολο του **Κατηγορηματικού Λογισμού 1ης τάξης** (*1st Order Predicate Calculus*), μπορούν να αποτελέσουν τη βάση για τη δημιουργία γλώσσας **Λογικού Προγραμματισμού**.

Διατακτικές και Δηλωτικές Γλώσσες Προγραμματισμού

Η βασική ιδέα που κρύβεται πίσω από το λογικό προγραμματισμό είναι ο προγραμματισμός διά περιγραφής (programming by description). Η ιδέα αυτή

βασίζεται στην παραδοχή ότι κάθε αλγόριθμος αποτελείται από δύο μέρη τα οποία χρειάζεται να περιγραφούν - δηλωθούν:

- ♦ μία λογική προδιαγραφή (logical specification) και
- ♦ μία περιγραφή του πώς πρέπει να εκτελεστεί αυτή η προδιαγραφή (ο έλεγχος (control))

Η βασική αυτή ιδέα περιγράφεται στην εξίσωση του Kowalski [KOWALSKI, 1979]:

$$\text{Αλγόριθμος} = \text{Λογική} + \text{Έλεγχος}$$

Ο στόχος εδώ είναι να γίνει δυνατόν το λογικό προγραμματιστικό σύστημα να αναλάβει από μόνο του το μεγαλύτερο μέρος του ελέγχου (κάτι που στις άλλες γλώσσες γίνεται με τη βοήθεια των προγραμματιστικών δομών ελέγχου if-then-else, while κ.λπ.), έτσι ώστε ο προγραμματιστής να ενδιαφέρεται μόνον για την περιγραφή της λογικής του αλγορίθμου. Από αυτήν την άποψη ο λογικός προγραμματισμός καθίσταται μία τάξη μεγέθους πιο περιγραφικός σε σχέση με τις καθιερωμένες γλώσσες προγραμματισμού.¹

Οι καθιερωμένες γλώσσες προγραμματισμού, όπως είναι η FORTRAN, η PASCAL και η C/C++, ονομάζονται **διατακτικές**. Ένα πρόγραμμα γραμμένο σε μια διατακτική γλώσσα είναι μια σειρά από "διαταγές" τις οποίες ο υπολογιστής πρέπει να καταλάβει και να εκτελέσει για να επιλύσει κάποιο πρόβλημα. Ο προγραμματιστής αναγκάζεται δηλαδή να περιγράψει το **πώς** θα επιλυθεί το πρόβλημα. Αντίθετα η PROLOG ανήκει στις **δηλωτικές** γλώσσες προγραμματισμού. Ένα πρόγραμμα γραμμένο σε PROLOG αποτελεί μια δήλωση του προβλήματος που πρόκειται να επιλυθεί και όχι των διαδοχικών βημάτων-διαταγών που πρέπει να ακολουθηθούν για την επίλυση του προβλήματος. Ο προγραμματιστής αντί να ασχολείται με το πώς θα επιλυθεί το πρόβλημα, δηλώνει με τη βοήθεια ειδικών φράσεων **ποιό** είναι το πρόβλημα, αφήνοντας στη γλώσσα την ευθύνη του πώς θα το επιλύσει.

¹ Για να γίνει κατανοητό τι σημαίνει μία τάξη μεγέθους, αρκεί να σκεφτούμε την περιγραφική ικανότητα που εισήγαγε η γλώσσα προγραμματισμού FORTRAN σε σχέση με τη συμβολική γλώσσα μηχανής (ASSEMBLY)

Σύντομη Ιστορία της PROLOG

Η γλώσσα προγραμματισμού **PROLOG** είναι μια ειδική περίπτωση υλοποίησης ενός συστήματος λογικού προγραμματισμού. Η ιστορία της PROLOG ξεκινάει στα 1971-1972 με τη δημιουργία του πρώτου διερμηνέα (interpreter) στο πανεπιστήμιο της Μασσαλίας [COLMERAUER & al, 1973], [BATTANI & MELONI, 1973]. Το όνομα PROLOG προτάθηκε από τη Jacqueline Roussel, γυναίκα του Phillipe Roussel που δούλευε στην ομάδα του πανεπιστημίου της Μασσαλίας) σαν συνγροπτόμενη λέξη της φράσης "**PRO**grammation en **LOG**ique" (Προγραμματίζοντας στη Λογική).

Μια προηγούμενη προσέγγιση ενός συστήματος Λογικού Προγραμματισμού μπορεί να θεωρηθεί αυτή που οδήγησε στην υλοποίηση των Planner - MicroPlanner [HEWITT, 1969], [SUSSMAN, 1970] στο MIT της Αμερικής. Τα συστήματα αυτά όμως δεν κατέληξαν σε γλώσσα προγραμματισμού.

Ο πρώτος μεταγλωττιστής (*compiler*) υλοποιήθηκε το 1977 στο πανεπιστήμιο του Εδιμβούργου [PEREIRA & al, 1978]. Στην επόμενη δεκαετία υπήρξε ένα ολοένα αυξανόμενο ενδιαφέρον για το λογικό προγραμματισμό από διάφορα ερευνητικά κέντρα της Ευρώπης, της Ιαπωνίας και της Αμερικής και μέχρι σήμερα έχουν γίνει πολλές πραγματοποιήσεις της γλώσσας PROLOG. Σημαντικό σταθμό στην εξέλιξη της PROLOG αποτέλεσε το πρόγραμμα 5ης γενιάς υπολογιστών της Ιαπωνίας (Fifth Generation Computer Systems) [MOTOOKA, 1982], [SHAPIRO, 1983]. Σημαντική στην ανάπτυξη του λογικού προγραμματισμού θεωρείται η συνεισφορά διαφόρων ευρωπαϊκών ερευνητικών ομάδων, όπως είναι αυτές των Πορτογάλων (πανεπιστήμιο Λισσαβώνας) και των Ούγγρων (πανεπιστήμιο Βουδαπέστης) και στη συνέχεια και Αμερικανικών όπως αυτή του ερευνητικού κέντρου SRI.

Αρχικά η χρήση της PROLOG περιορίστηκε σε καθαρά ερευνητικά πλαίσια και κυρίως για εφαρμογές της Τεχνητής Νοημοσύνης. Στη συνέχεια όμως κατάφερε να ξεπεράσει τα στενά πλαίσια της ερευνητικής κοινότητας και

σήμερα χρησιμοποιείται σε ένα μεγάλο φάσμα από κλάδους εφαρμογών όπως είναι:

- ♦ οι βάσεις δεδομένων,
- ♦ η ανάπτυξη μεταφραστών γλωσσών προγραμματισμού,
- ♦ τα έμπειρα συστήματα,
- ♦ η ρομποτική,
- ♦ η αναγνώριση έντυπης φυσικής γλώσσας και
- ♦ τα έξυπνα συστήματα διδασκαλίας.

Βιβλιογραφικές Αναφορές

[BATTANI & MELONI, 1973]

Battani G., Meloni H., Interpreteur du Language de Programmation PROLOG, Groupe d' Intelligence Artificielle, Univ. d' Aix-Marseille, Luminy, France, 1973

[BOBROW, 1985]

Bobrow D. G., If PROLOG is the Answer, What is the Question? or What it Takes to Support AI Programming Paradigms, IEEE Trans. on Soft. Engineering, Vol.11, No.11, Nov. 1985

[BRATKO, 1990]

Bratko I., PROLOG Programming for Artificial Intelligence, 2nd Ed., Addison Wesley, 1990

[COELHO & al, 1982]

Coelho H., Cotta J. C., Pereira L. M., How to Solve it with PROLOG, (Ed.) Laboratorio National de Engenharia Civil, Lisboa, Portugal, 1982

[COHEN, 1988]

Cohen J., A view of the origins and development of PROLOG, Communications of the ACM, vol. 31, no 1, Jan. 1988

[COLMERAUER & al, 1973]

Colmerauer A., Kanoui H., Pasero R. Roussel P., "Une Systeme de Communication Homme-machine en Francais", Research Report, Groupe d' Intelligence Artificielle, Univ. d' Aix-Marseille, Luminy, France, 1973

[CLOCKSIN & MELLISH, 1984]

Clocksin W. F. and Mellish C. S., Programming in PROLOG, 2nd Ed., Springer-Verlag, 1984, (1st Ed. 1981)

[DERANSART & al, 1996]

Deransart B., Ed.Dbali A., Cervoni L., PROLOG: The Standard Reference Manual, Springer erlang, 1996

[HEWITT, 1969]

Hewitt C., PLANNER: A Language for proving theorems in Robots, In Proceedings of IJCAI, Washington D.C., May 1969

[KOWALSKI, 1974]

Kowalski R., Predicate Logic as a Programming Language, In Proceedings of IFIP 1974 (Stockholm, Sweden), North-Holland Publishing Co., 1974

[KOWALSKI, 1979a]

Kowalski R., Algorithm = Logic + Control, Commun. of the ACM, Vol.22, No. 7, July 1979.

[KOWALSKI, 1979b]

Kowalski R., Logic for Problem Solving, North-Holland, 1979

[KOWALSKI, 1988]

Kowalski R., The early years of Logic Programming, Communications of the ACM, vol. 31, no 1, Jan. 1988

[LLOYD, 1984]

Lloyd J.W., Foundations of Logic Programming, Springer-Verlang, 1984

[MOTO-OKA & al, 1981]

Moto-oka T. et al., Challenge for Knowledge Information Processing Systems (preliminary report on Fifth Generation Computer Systems), In Proceedings of International Conference on Fifth Generation Computer Systems, JIPDEC, 1981

[MOTO-OKA, 1982]

Moto-oka T. (Ed.), Fifth Generation Computer Systems, North-Holland 1982

[O'KEEFE, 1990]

O'Keefe R., The Craft of PROLOG, MIT Press, Cambridge Mass., 1990

[PEREIRA & al, 1978]

Pereira L.M., Pereira F., Warren D.H.D, User's Guide to DECsystem-10 PROLOG, University of Edinburg, Department of Artificial Intelligence, 1978.

[ROBINSON, 1965]

Robinson J., A machine-oriented logic based on the resolution principle, Journal of the ACM, vol. 12 no 1, Jan. 1965

[ROUSSEL, 1975]

Roussel P., PROLOG: Manuel de Référence et d' Utilisation, Groupe d' Intelligence Artificielle, Univ. d' Aix-Marseille, Luminy, France, 1975

[SHAPIRO, 1983]

Shapiro E.Y., The Fifth Generation Project - A Trip Report, Commun. of the ACM, Vol.26, No. 9, Sep. 1983.

[STERLING & SHAPIRO, 1986]

Sterling L. and Shapiro E., The Art of PROLOG - Advanced Programming Techniques, MIT Press, 1986

[SUSSMAN, 1970]

Sussman G.J., Winograd T. and Charniak E., "MICROPLANNER Reference MAnnual", AI Memo 203, Cambridge MA, MIT Artificial Intelligence Laboratory, 1970

[VANCANEGHEM, 1986]

Van Caneghem M., L' Anatomie de PROLOG, InterEditions, Paris, 1986

✓ Βιβλία αναφοράς για τη γλώσσα προγραμματισμού PROLOG είναι αυτά των [CLOCKSIN & MELLISH, 1984], [BRATKO, 1990], [STERLING & SHAPIRO, 1986], και [O'KEEFE, 1990]

✓ Η πρώτη ενδιαφέρουσα συλλογή προγραμματιστικών προβλημάτων σε PROLOG μπορεί να βρεθεί στο βιβλίο των Coelho, Cotta και Pereira [COELHO & al, 1982]

✓ Πολύ ενδιαφέροντα στοιχεία για την ιστορική εξέλιξη της γλώσσας προγραμματισμού PROLOG αλλά και του Λογικού Προγραμματισμού γενικότερα μπορούν να βρεθούν στις αναφορές των Cohen [COHEN, 1988] και Kowalski [KOWALSKI, 1988] καθώς επίσης και στο βιβλίο του Van Caneghem [VANCANEGHEM, 1986]

✓ Παραπέμπουμε τον αναγνώστη που ενδιαφέρεται για τη σχέση της λογικής με το λογικό προγραμματισμό στις αναφορές του Kowalski [KOWALSKI, 1979] και Lloyd [LLOYD, 1984]

✓ Για τα μειονεκτήματα-πλεονεκτήματα της PROLOG μια ενδιαφέρουσα μελέτη μπορεί να βρεί ο αναγνώστης στην αναφορά του [BOBROW, 1985]