

Σχέση έναντι Δράσης: Μια αναγκαία διάκριση για μια ολοκληρωμένη προσέγγιση της έννοιας της μεταβλητής με χρήση υπολογιστή

Δειμέζης Μωυσής^{1,2}, Γυφτοδήμος Γεώργιος²

moissis@phs.uoa.gr, gyftodim@phs.uoa.gr

¹ Εκπαιδευτικός Πληροφορικής ΠΕ19

² Τμήμα Μεθοδολογίας, Ιστορίας και Θεωρίας της Επιστήμης (Μ.Ι.Θ.Ε.), Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

Περίληψη

Η εξοικείωση των μαθητών Γυμνασίου και Λυκείου με την έννοια της μεταβλητής παρουσιάζει δυσκολίες που εν μέρει οφείλονται στο ότι η έννοια αυτή θεμελιώνεται στις μαθηματικές - λογικές θεωρίες διαφορετικά απ' ότι στις αλγοριθμικές. Το γεγονός ότι τα θέματα μελέτης συχνά προσεγγίζονται με συσχετισμούς λογικών - μαθηματικών σχέσεων και εκτελέσεων αλγορίθμων δημιουργεί την ανάγκη ενός ενοποιημένου πλαισίου μελέτης που θα επιτρέπει στον μαθητή να εφαρμόζει ενοποιημένα τόσο θεωρητικές ιδιότητες όσο και υπολογιστικές μεθόδους επεξεργασίας. Για το σκοπό αυτό διακρίνονται οι εκφράσεις σε "σχέσης" και "δράσης", που υπακούουν αντίστοιχα σε λογικές - μαθηματικές ή αλγοριθμικές αρχές που διέπουν και τις μεταβλητές των εκφράσεων. Ως "ενοποιημένο πλαίσιο" θεωρείται στο παρόν ο προγραμματισμός διότι, παρά τους θεωρητικούς και πρακτικούς περιορισμούς, δίνει τη δυνατότητα αποτύπωσης και επεξεργασίας γνωσιακών εκφράσεων σχεδόν κάθε μορφής. Δίνονται ενδεικτικά περιπτώσεις υπολογιστικής παράστασης τέτοιων ζητημάτων μέσω της Logo, ως παραδείγματα σχετικής διδασκαλίας.

Λέξεις κλειδιά: Εκφραση σχέσης, Εκφραση δράσης, Μεταβλητή, Διαδικασία, Logo

Εισαγωγή

Η εργασία αυτή αποσκοπεί στην εκμετάλλευση του προγραμματισμού από μαθητές για την αντιμετώπιση θεμάτων που συμπεριλαμβάνουν από το ένα μέρος σχέσεις που είναι μαθηματικές ή/και λογικές εκφράσεις και από το άλλο μέρος, εκτελέσεις απλών ή συνθέτων βημάτων που έχουν λειτουργικό νόημα δράσης.

Για το σκοπό αυτό, διακρίνονται οι εκφράσεις και οι μεταβλητές τους σε σχέσης ή δράσης ώστε, σε πρώτο βήμα, να αξιοποιηθούν οι υπολογιστικές αναπαραστάσεις τους ως υπολογιστικά εργαλεία της αντίστοιχης κατεύθυνσης και σε επόμενα, να αξιοποιηθούν ως αντίστοιχα δομικά στοιχεία προς διεύρυνση του αρχικού θέματος μελέτης. Η διάκριση αυτών των δύο κατευθύνσεων διευκολύνει τον μαθητή σε χειρισμό των εκφράσεων ως γνωσιακών στοιχείων που όταν πρόκειται για σχέσεις αυτός γίνεται με μετασχηματισμούς τους μέσω εφαρμογής αληθών σχέσεων της αντίστοιχης θεωρίας, ενώ όταν πρόκειται για δράσεις γίνεται μέσω εκτέλεσης λειτουργικών βημάτων, αλγοριθμικά εκφρασμένων, που οδηγούν σε παραγωγή αποτελεσμάτων. Ως "μαθητές" στα επόμενα νοούνται μαθητές και μαθήτριες Λυκείου.

Εκφράσεις "σχέσης" ή "δράσης" και ο ρόλος των μεταβλητών τους.

Η συνύπαρξη σχέσεων με δράσεις είναι παρούσα σχεδόν παντού σε ένα θέμα μελέτης, ακόμα και καθαρά μαθηματικού περιεχομένου: Τα βήματα μιας απόδειξης, ακόμα και αν είναι μεταξύ τους λογικά ισοδύναμα και κατά συνέπεια από τυπική άποψη είναι αντιμεταθετικά, για να γίνει κατανοητή η απόδειξη πρέπει να παρουσιάζονται με ρητά ακολουθιακή διαδοχή διότι στην ουσία αποτελούν βήματα εκτέλεσης αλγόριθμου. Η εκτέλεση αριθμητικών πράξεων σε μια παράσταση μπορεί να γίνει αναγκαία σε κάποια φάση και αυτό είναι επίσης μια αλγοριθμική δράση. Κατά κανόνα, η κατανόηση ενός μαθηματικού θέματος συχνά γίνεται ευκολότερη μέσω μιας λειτουργικής αναπαράστασής του, δηλαδή εκτέλεσης αλγόριθμου που θεωρείται ότι "εκφράζει" αυτό το θέμα.

Οι σχέσεις είναι δυνατό να προσδιορίζονται σε αξιωματικό χώρο, όπως οι μαθηματικές, οι λογικές και οι συνθέσεις αυτών, ή ως σχέσεις εφαρμογών των προηγούμενων σε ένα γνωσιακό χώρο αρχών (πχ. Φυσική) ή εφαρμογών τους κάτω από σημασιολογικές παραδοχές (πχ. ένας συλλογισμός της καθημερινότητας) ή κάτω από πραγματολογικές θέσεις (πχ. η δομή μιας μηχανής). Στο παρόν, ως δράσεις νοούνται οι εκτελέσεις αλγορίθμων που έχουν νόημα διαμόρφωσης ή λειτουργικής μεταβολής ενός "κόσμου" (πχ. ένα φυσικό φαινόμενο, μια χημική αντίδραση) ή υπολογισμού τελικής τιμής από σημειωμένες μαθηματικές ή λογικές πράξεις. Η διαφοροποίηση μεταξύ σχέσεων και δράσεων στο πλαίσιο ενός θέματος που μελετάται, είναι ουσιαστική, διότι προκαθορίζει τι είναι δυνατό να εφαρμοστεί σε κάθε έκφραση και από αυτό απορρέει πώς μπορεί αυτή να αξιοποιηθεί.

Σε κάθε περίπτωση, μια δράση, ως εκτέλεση μιας αλγοριθμικής έκφρασης μπορεί να θεωρηθεί ως σχέση και συγκεκριμένα, ως συσχέτιση των δεδομένων της με το αποτέλεσμα της εκτέλεσης. Ακόμα και η ίδια έκφραση μπορεί να παίζει ρόλο άλλοτε σχέσης και άλλοτε δράσης, ανάλογα με τον επιδιωκόμενο σκοπό και το συλλογισμό που απορρέει από αυτόν. Για παράδειγμα: Η έκφραση $5+3$ ως σχέση παράγει την αληθή σχέση $5+3=8$ αλλά και την $5+3=9-1$ και άπειρες άλλες, ενώ ως δράση νοείται ότι παράγει ένα και μοναδικό αποτέλεσμα, 8. Το σύμβολο της ισότητας στην πρώτη περίπτωση είναι καθαρά μαθηματικό, άρα αντιμεταθετικό, ενώ στη δεύτερη "αναμειγνύεται" με ένα νόημα δράσης που νοείται "κατευθυνόμενο προς το αποτέλεσμα" και ως εκ τούτου διαφοροποιείται το πώς θα χρησιμοποιηθεί.

Ιδιαίτερες κατηγορίες σχέσεων είναι αυτές που εκφράζουν ισότητα ή γενικότερα ισοδυναμία και, αντίστοιχα, ανισότητα ή γενικότερα διάταξη, διότι είναι αυτές που εκτός του ότι επιδέχονται επεξεργασία βάσει κανόνων του χώρου του θέματος, λαβαίνουν λογική τιμή και συνεπώς είναι επιπλέον λογικά επεξεργάσιμες.

Το κεντρικό ζήτημα στην παρούσα μελέτη είναι ότι μια έκφραση σχέσης είτε δράσης, μπορεί να περιλαμβάνει μεταβλητές, όπου ο ρόλος τους καθορίζεται αντίστοιχα διαφορετικά: Μια σχέση είναι δυνατό να συντεθεί με άλλες σχέσεις μέσω των μεταβλητών τους σε αφαιρετικό επίπεδο (δηλ. χωρίς να απαιτείται να έχουν λάβει τιμή οι μεταβλητές). Στο επίπεδο αυτό οι μεταβλητές "λειτουργούν" ως αντιπρόσωποι όλων των δυνατών περιπτώσεων συγκεκριμενοποίησής τους, αλλά ταυτόχρονα και ως αυτόνομα στοιχεία της σχέσης τα οποία υπόκεινται σε μετασχηματισμούς βάσει γενικών ιδιοτήτων (πχ. αλγεβρικές εκφράσεις). Από την άλλη πλευρά, μπορεί μια σχέση να αποτελεί ένα δεδομένο σε μια δράση, όπως καθορίζοντας μια συνθήκη, ή η ίδια να αποτελεί και εκτελέσιμο αλγόριθμο. Αντίστοιχα, με την παραδοχή ότι κάθε δράση μπορεί να περιγραφεί κατά διακριτά βήματα (έστω και επιλέγοντας "στιγμιότυπα της εκτέλεσης" που επαρκούν να την προσδιορίσουν) μια δράση είναι αποτέλεσμα εκτέλεσης αλγόριθμου ο οποίος (δυνητικά) θα έχει μεταβλητές

εισόδου, οι οποίες νοείται εν γένει ότι θα έχουν λάβει τιμή κατάλληλου τύπου, προκειμένου να εκτελεστεί. Μέσω των μεταβλητών γενικεύεται το νόημα του αλγόριθμου με τρόπο που να αντιπροσωπεύει το σύνολο των ειδικότερων αλγόριθμων - περιπτώσεων εφαρμογής του, (Δεϊμέζης & Γυφτοδήμος, 2016) που ακριβώς είναι δράσεις.

Μια σχέση μπορεί να αποτελέσει "πηγή" μιας ομάδας αλγόριθμων που ο καθένας προσδιορίζεται από τη σχέση ανάλογα με την εκάστοτε επίλυσή της και την αντίστοιχη διάκριση των μεταβλητών σε "εισόδου" και "εξόδου".

Μια λογική ή μαθηματική έκφραση μπορεί να χρησιμοποιηθεί σε πεδία εφαρμογής (πχ. Φυσικής) ως σχέση που επιδέχεται συσχέτισή της με άλλες σχέσεις του πεδίου, ή ως δράση εφαρμογής στο πεδίο αυτό, κάτω από την προϋπόθεση υπακοής των "βημάτων χρήσης" στους νόμους του πεδίου. Η εφαρμογή έκφρασης ενός γνωστικού πεδίου σε άλλο, αφ' ενός μεταφέρει το ρόλο της από το πρώτο πεδίο στο δεύτερο μαζί με τις αντίστοιχες του δυνατότητες επεξεργασίας (ενδεχομένως κάτω από περιορισμούς που επιβάλλει το δεύτερο) και αφ' ετέρου αποκτά επιπρόσθετα νέες δυνατότητες χρήσης, μέσω της ταυτοποίησης μεταβλητών μεταξύ διαφόρων εκφράσεων, εννοιολογικά ή τυπικά. Για παράδειγμα:

Η μαθηματική σχέση $x=y*z$ με πεδίο εφαρμογής τη Φυσική, θεωρώντας τη μεταβλητή x ως δύναμη, την y ως μάζα και την z ως επιτάχυνση, παριστά το νόμο του Νεύτωνα $F=m*a$ που, αφ' ενός επιδέχεται εφαρμογή όλων των ιδιοτήτων των μαθηματικών πράξεων και ταυτοτήτων και αφ' ετέρου, ως σχέση Φυσικής μπορεί να συσχετιστεί με άλλες σχέσεις - φυσικούς νόμους, όπως της έλξης των μαζών, μέσω μεταβλητών τους με ίδιο όνομα που εκφράζει την ίδια φυσική έννοια. Ως επιλυμένη σχέση, η $F=m*a$ μπορεί να θεωρηθεί αλγόριθμος που η εκτέλεσή του θα δώσει την τιμή της προκαλούμενης δύναμης F με την προϋπόθεση ότι θα έχουν δοθεί σταθερές αριθμητικές τιμές στις μεταβλητές m και a . Πιο αφαιρετικά, η σχέση αυτή προσδιορίζει και αλγόριθμο που δίνει έξοδο τη δύναμη F συναρτήσει της μάζας m , η οποία κατά την εκτέλεση πρέπει να έχει σταθερά τιμή (ως μεταβλητή δράσης) με παράμετρο την επιτάχυνση a που κατά την εκτέλεση μπορεί να παραμένει ελεύθερη (ως μεταβλητή σχέσης). Οι επιλυμένες μορφές της ίδιας σχέσης ως προς άλλη μεταβλητή, αποτελούν άλλους αλγόριθμους. Η ίδια η σχέση, ανεξάρτητα του ότι στην συγκεκριμένη περίπτωση είναι επιλύσιμη, αποτελεί αλγόριθμο τριών μεταβλητών εισόδου, f , m , a , με έξοδο τιμή Boole (αληθές/ψευδές) που μπορεί να χρησιμοποιηθεί είτε αυτόνομα είτε ως κριτήριο σε βήμα επιλογής στο πλαίσιο ενός ευρύτερου αλγόριθμου.

Προγραμματιστική αποτύπωση σχέσεων και δράσεων.

Η έννοια της μεταβλητής, ή ακριβέστερα οι διάφορες έννοιες που εκφράζει ο όρος αυτός, καθορίζεται ανάλογα με το αντίστοιχο θεωρητικό ή πρακτικό μοντέλο όπου εντάσσεται η σχέση που την περιλαμβάνει. Πιο συγκεκριμένα, η "μαθηματική μεταβλητή" υπακούει στα αξιώματα και νόμους της λεγόμενης Μεταμαθηματικής (Kleene, 1962), η "λογική μεταβλητή" υπακούει στα αξιώματα και τους συμπερασματικούς νόμους της Κατηγορηματικής Λογικής (Mendelson, 2009), η "αλγοριθμική μεταβλητή" στους κανόνες του λ-λογισμού (Church, 1941) ή άλλης ισοδύναμης αλγοριθμικής θεωρίας όπως η Μηχανή Turing, ενώ η "μεταβλητή κατά το κοινό νόημα" εκφράζει ένα δυναμικά μεταβαλλόμενο ποσό στο πλαίσιο κάποιου context και οι "νόμοι" χρήσης της εξαρτώνται από αυτό.

Όμως αυτά τα θεωρητικά μοντέλα είναι δυσνόητα για ένα μαθητή, που είναι αναγκασμένος να προσεγγίσει τις έννοιες από τους τρόπους χρήσης τους. Έτσι περνά από σφάλματα που, στην αρχή τουλάχιστον, αδυνατεί να τα αναγνωρίσει ή να τα ερμηνεύσει μόνος του. Συνήθως διαμορφώνει σταδιακά μια "κάπως αφαιρετική" γνώση που είναι λίγο -

πολύ "αγκιστρωμένη" σε κάποιες ειδικές συνθήκες ή σημασιολογικές δεσμεύσεις ("situated abstractions"). Οι δεσμεύσεις αυτές είναι δυνατό να αξιοποιηθούν διδακτικά ώστε να αποτελέσουν "σκαλοπάτι" που θα βοηθήσει τον μαθητή να κτίσει αφαιρετικές έννοιες όπως οι μαθηματικές, μέσω προγραμματισμού (Noss & Hoyles, 1996).

Στο πρόβλημα αυτό έρχεται να προσφέρει ενεργό υποστήριξη ο προγραμματισμός διότι, παρά τις δυσχέρειες που δημιουργούνται, τόσο από την ανάγκη πρόσθετων γνώσεων όσο και επαναδόμησης των συλλογισμών σε ένα πλαίσιο διαφορετικών αρχών και πρωτογενών στοιχείων, αποτελεί ένα "χειροπιαστό" τρόπο αποτύπωσης νοημάτων, συλλογισμών και στη συνέχεια, τρόπο ελέγχου της ορθότητάς τους μέσω της εκτέλεσης.

Ένα ειδικότερο πρόβλημα που γεννάται, είναι οι δυσκολίες που συναντούν οι μαθητές στην χρήση των αλγοριθμικών μεταβλητών, διαπιστώνοντας ότι διαφέρουν από τις μαθηματικές: Είναι δυνατό να μεταβάλλεται η τιμή τους δυναμικά, απέναντι στη "στατική" συγκεκριμενοποίηση μεταβλητής στα μαθηματικά (Τζιμογιάννης Α. & Κόμης Β., 2000).

Για να αξιοποιηθεί ο προγραμματισμός από τον μαθητή στο ζήτημα που μελετάμε, πρέπει να αντιμετωπιστεί κατά περίπτωση το "πέραςμα" των εκφράσεων και μεταβλητών σχέσης σε αντίστοιχες αλγοριθμικές. Στην περίπτωση μαθηματικών εκφράσεων, αυτό είναι σχετικά απλό, βέβαια κάτω από τις δυσκολίες που θέτει το προηγούμενο και οπωσδήποτε κάτω από τη ρητή διευκρίνηση ότι κάθε μαθηματική πράξη είναι υλοποιημένη υπολογιστικά ως αλγόριθμος με είσοδο τα μέλη της πράξης, άρα μπορεί να ορίζεται με μέλη μεταβλητές αλλά εκτελείται μόνον με συγκεκριμένες τιμές τους. Σε ό,τι αφορά κατηγορηματικές λογικές εκφράσεις, η υπολογιστική αναπαράστασή τους απαιτεί ανάπτυξη αλγόριθμου που "μεταφράζει" το λογικό νόημα σε διαδικασία που περιλαμβάνει δράσεις. Για παράδειγμα, η μεταβλητή ενός ποσοδείκτη ($\forall X, \exists X$) αποδίδεται έμμεσα, μέσω δράσης "σάρωσης" της λίστας όλων των περιπτώσεων και ελέγχου κάθε όρου της και μάλιστα δεν απαιτείται καν να υπάρχει όνομα αντίστοιχης μεταβλητής στο πρόγραμμα.

Η επιλογή προγραμματιστικής γλώσσας για το σκοπό αυτό, πρέπει να πληροί τα εξής: i) να είναι διαθέσιμη και πρακτικά εκμεταλλεύσιμη στο σχολικό πλαίσιο, ii) να είναι εύληπτη από τους μαθητές σε βαθμό που να μην αποτελεί αυτοσκοπό γνώσης αλλά απλώς βοηθητικό εργαλείο σκέψης και iii) να επιτρέπει την αποτύπωση και επεξεργασία εκφράσεων με μεταβλητές κάθε είδους και, στο πλαίσιο του εφικτού, κάθε μορφής εκφράσεων. Με κατάλληλους περιορισμούς, που θα αφορούν το είδος των θεμάτων αλλά και τις προγραμματιστικές δυσκολίες που συνεπάγεται η αποτύπωσή τους, είναι δυνατό, όχι μόνο να γίνονται αναπαραστάσεις τέτοιων γνωσιακών θεμάτων με τη Logo αλλά και αυτή είναι η βέλτιστη επιλογή για το σημερινό εκπαιδευτικό πλαίσιο και για τον επιδιωκόμενο σκοπό (Δειμέζης & Γυφτοδήμος, 2016).

Συγκεκριμένα, παρατηρούμε για τη Logo τα εξής:

α) Πέραν των τυπικών περιορισμών που θέτει εν γένει η έννοια του αλγόριθμου, δηλ. το "βηματικό", το "πεπερασμένο" και το "επαρκώς αποτελεσματικό για το πλαίσιο χρήσης", περιορίζονται οι δυνατότητες και από το ποιά είναι τα διαθέσιμα πρωτογενή ή αρχικά στοιχεία της γλώσσας. Στο ζήτημα αυτό η Logo υστερεί κρίσιμα, τόσο σε αποτελεσματικότητα όσο και σε κάλυψη κάθε τεχνικής απαίτησης, αλλά υπερκαλύπτει το παρόντα σκοπό, της κατανόησης της έννοιας της μεταβλητής από το μαθητή.

β) Γενική προϋπόθεση στον διαδικαστικό προγραμματισμό είναι ότι αν η έκφραση είναι σε μορφή ισότητας ή ισοδυναμίας, πρέπει να είναι επιλυμένη ως προς μια μεταβλητή, αυτή που θα αποτελέσει τον φορέα του τελικού αποτελέσματος της εκτέλεσης, που μπορεί τελικά να παραλειφθεί στο πρόγραμμα. Η δυνατότητα αυτή ενισχύεται από το ότι η Logo δέχεται

ως τιμή μεταβλητής "οτιδήποτε είναι αναγνωρίσιμο και επεξεργάσιμο", απλό ή σύνθετο, τυποποιώντας εκ των υστέρων την μεταβλητή βάσει του τύπου της τιμής της.

γ) Συχνά κατά τον προγραμματισμό υπεισέρχονται και διάφορες πρόσθετες εκφράσεις τεχνικής "υφής" (ως υποαλγόριθμοι) που δεν αφορούν άμεσα το γνωσιακό θέμα αλλά δρουν καταλυτικά, επιτρέποντας ή έστω διευκολύνοντας την προγραμματιστική αποτύπωση, όπως μια επανάληψη, ή η πρόσθετη διαδικασία που θα συνενώσει μια ομάδα αυτόνομα ορισμένων διαδικασιών σε μια δομημένη λειτουργία. Τέτοιες εκφράσεις ενδέχεται να εισάγουν πρόσθετες αλγοριθμικές μεταβλητές. Για να μη γίνεται σύγχυση, είναι βοηθητικό, τα ονόματα τέτοιων μεταβλητών και διαδικασιών να διαφοροποιούνται φραστικά από τα ονόματα των μεταβλητών και σχέσεων ή δράσεων του θέματος.

δ) Στον προγραμματισμό, οι μεταβλητές διακρίνονται επίσης σε είδη, που παίζουν διαφορετικούς ρόλους στο πρόγραμμα, ανάλογα με την επιδίωξη και την αντιστοιχία της τεχνική. Αυτό προσφέρει στον μαθητή τη δυνατότητα να κάνει μια λεπτομερέστερη αντιστοιχίση των αρχικών εννοιών προς τις προγραμματιστικές, όπως και να αντιληφθεί την ουσία του προγραμματισμού, σε κάποιο βαθμό ανεξάρτητα από τη γλώσσα (Sajaniemi, 2005). Αν και στη Logo τα είδη μεταβλητών δεν διαφοροποιούνται τυπικά πάντοτε ούτε διακρίνονται ρητά τύποι και υπο-τύποι, οι ρόλοι αυτοί εξακολουθούν να υφίστανται και η διάκρισή τους γίνεται από τον μαθητή σε νοηματικό επίπεδο, όπου αυτό έχει νόημα για το μελετώμενο θέμα. Αυτό έχει το θετικό ότι επιτρέπει στο μαθητή να διακρίνει τα στοιχεία δράσης και ποιά είδος αλγοριθμικής δράσης, ακόμα και εκ των υστέρων όταν θα έχει επιτύχει το σκοπό μετά από τοπικές δοκιμές και διορθώσεις, διαπλάθοντας έτσι πιο ευέλικτα τη διαμόρφωση του προγράμματος, διακρίνοντας και αναμειγνύοντας με καθαρά διακριτούς ρόλους λειτουργικά στοιχεία με στοιχεία σχέσης.

ε) Μαθηματικές εκφράσεις όπως και σχέσεις Boole, είναι εν γένει εκφράσιμες προγραμματιστικά, με άμεση αντιστοιχίση των μεταβλητών τους σε υπολογιστικές. Όμως δεν είναι εν γένει ισοδύναμες οι αποτυπωμένες εκφράσεις με τα πρότυπά τους, όπως όταν έχουμε λογικές εκφράσεις (με πράξεις Boole "και", "είτε", "όχι" και παράγωγές τους) διότι οι αντιστοιχίες τους προγραμματιστικές εκφράσεις παράγουν μεν τα ίδια αποτελέσματα, αλλά "μονόδρομα". Για παράδειγμα, αν είναι αληθής η λογική σχέση (A και B) τότε συμπεραίνουμε πως είναι αληθή και τα δύο μέλη A, B, όπως και το αντίστροφο, αλλά αυτό υπολογιστικά ισχύει μόνον προς τη μία πλευρά: Από το αληθές των μελών A, B προκύπτει αυτόματα το αληθές της σύζευξης (A AND B), εξ ορισμού της πράξης AND ως αλγόριθμος, ενώ από το δεύτερο δεν προκύπτει το πρώτο παρά μόνον αν ορίσουμε άλλον αλγόριθμο σύζευξης που να καλύπτει και τις δύο κατευθύνσεις. Η μη-ενσωμάτωση τέτοιων αλγόριθμων εξ αρχής στις απλές γλώσσες όπως η Logo οφείλεται σε λόγους αποτελεσματικότητας. Αυτό είναι ένα παράδειγμα για εμβάθυνση στη διάκριση σχέσεων από δράσεις, εισάγοντας και την παράμετρο της "αποτελεσματικότητας".

Ακόμα συνθετότερο ζήτημα, που αφορά μαθητές Λυκείου αλλά κεντρικής σημασίας για τη διαμόρφωση τη σκέψης, είναι η προσπάθεια αντιστοιχίσης της λογικής σχέσης "αν A τότε B" προς την υπολογιστική έκφραση "IF A THEN B", προσπάθεια που συχνά οδηγεί σε σφάλματα, ακόμα και πιο προχωρημένους μελετητές: Λογικά, αν "A ψευδής" και "B αληθής" τότε η σχέση "αν A τότε B" είναι εξ ορισμού αληθής. Υπολογιστικά, όταν εκτελεστεί το βήμα "IF A THEN B" και αν συμβεί τη στιγμή εκείνη το A να είναι αληθές (δηλ. ή να έχει τιμή "αληθές" ή να είναι αλγόριθμος που να εκτελείται και να επιστρέφει τιμή "αληθές") τότε προκαλείται εκτέλεση του B (δηλ. δράση) ενώ αν A ψευδές δεν εκτελείται το B. Αυτό είναι ένα ακόμα σημείο που κάνει φανερή στους μαθητές την ουσιαστική διαφορά που διακρίνει σχέσεις από δράσεις. Αντίστοιχο είναι και το ζήτημα του συμπερασμού: "έχω

A, άρα B", το οποίο είναι η βάση κάθε νοήμονος συλλογισμού. Η υπολογιστική αντιστοίχιση του σε επαρκώς περιορισμένα για το επίπεδο των μαθητών θέματα, είναι μια αρχή για να εξηγηθεί το νόημα "Τεχνητή Νοημοσύνη".

Αν και όλοι οι αλγόριθμοι μπορούν έστω και ακραία να θεωρηθούν ως σχέσεις, διακρίνονται δύο βασικά είδη:

α) Αυτοί που το αποτέλεσμα της εκτέλεσής τους είναι μια έκφραση υπολογιστικά αναγνωρίσιμη, που είναι οι λεγόμενοι "συναρτησιακοί" ή "συναρτήσεις", όπως λέγονται σε κάποιες γλώσσες προγραμματισμού, όρος που διαφέρει του μαθηματικού. Τέτοιοι αλγόριθμοι αποτελούν κατά κανόνα "σχέσεις που παράγουν σχέσεις".

β) Αυτοί που η εκτέλεσή τους έχει νόημα δράσης και είναι οι λεγόμενοι "διαδικαστικοί" ή απλά "διαδικασίες". Τέτοιοι αλγόριθμοι αποτελούν κατά κανόνα "προσδιορισμό δράσης", η εκτέλεσή τους τη "δράση" και το αποτέλεσμα "στοιχείο παρεχόμενο προς τον χρήστη για την τελική κατάσταση του συστήματος που αυτός εννοεί".

Η "ειδοποιός διαφορά" μεταξύ των δύο είναι ότι: Σε συναρτησιακό αλγόριθμο υπάρχει πάντα έξοδος τελικού αποτελέσματος σε μορφή "επιστρεφόμενης τιμής" (όπως μέσω της εντολής Op στη Logo) όπου αυτή η τιμή μπορεί να είναι σταθερά ή επίσης αλγόριθμος, και εν δυνάμει αποτελεί στοιχείο εισόδου αλγόριθμου (άλλου, ή του ίδιου σε αναδρομή). Ειδικά για τη Logo, αν και το νόημα αυτού του είδους διαδικασιών είναι ότι η έξοδός τους θα δίνει τιμή εισόδου προς άλλη διαδικασία, κατά βάση έχουν το γενικότερο νόημα σχέσης, αποτυπωμένης αλγοριθμικά. Αντίστοιχα, σε διαδικαστικό αλγόριθμο το τελικό αποτέλεσμα δεν μπορεί να αποτελέσει στοιχείο εισόδου σε άλλον αλγόριθμο. Επειδή στη Logo οι αλγόριθμοι αποκαλούνται γενικά "διαδικασίες", θα χρησιμοποιούμε στα επόμενα τους όρους "συναρτησιακές διαδικασίες" και "διαδικασίες δράσης" αντίστοιχα.

Η χρήση υπολογιστή για γνωσιακά ζητήματα που περιλαμβάνουν μεταβλητές, αξιοποιείται όχι μόνο από τη μελέτη των αλγοριθμικά αποτυπωμένων σχέσεων ή δράσεων και των σταθερών αποτελεσμάτων, αλλά και από την ευρύτερη πληροφορία που μπορεί να δώσει η εκτέλεση του προγράμματος που είναι δυνατό να δίνει στον μελετητή ένα "κόσμο" σε εξέλιξη, με διαφαινόμενες σχέσεις και δράσεις, ή ακόμα και με περαιτέρω δυνατότητες υπολογισμών βάσει στοιχείων που θα εισάγει ο μελετητής. Η συν-αξιοποίηση και των δύο όψεων του προγράμματος (δηλ. πηγαίου κώδικα - αποτελέσματος) ανοίγει πολύ περισσότερους δρόμους προς τη γνώση απ' ό,τι η μία μόνον, όπου οι δρόμοι αυτοί γίνονται πληρέστερα αντιληπτοί κάτω από τη διάκριση σχέσεων - δράσεων. Πχ., μια γεωμετρική κατασκευή μέσω της Logo-χελώνας μπορεί να αποτελέσει τέτοιο "κόσμο", παρέχοντας πληροφορία μέσω της λειτουργικής εξέλιξης του σχήματος.

Προς μια πιο ολοκληρωμένη γνωσιακή εκμετάλλευση της υπολογιστικής αναπαράστασης.

Ο οποιοσδήποτε περιορισμός της μελέτης του μαθητή μόνον στα αποτελέσματα της εκτέλεσης, περιορίζει τη μάθηση στην προκαθορισμένη ειδική γνώση που προσφέρει ο "κόσμος" της εκτέλεσης (πχ. σε προγράμματα προσομοίωσης). Η ύπαρξη παραμέτρων στο πρόγραμμα ή/και στο αποτέλεσμα διευρύνει το χώρο γνώσης, αλλά εξακολουθεί να λείπει η δυνατότητα διερεύνησης των νόμων μέσω αμφισβητήσεων της μορφής "αν δεν ... αλλά ..." (ως τεχνική "δοκιμής και πλάνης"). Για παράδειγμα: Αν δεν ίσχυε ο νόμος του Νεύτωνα στη βαρύτητα αλλά ο $F=m \cdot g^2$, τι θα συνέβαινε; Από τέτοιες παρατηρήσεις ο μαθητής θα έβλεπε πώς θα συμπεριφερόταν ο πραγματικός κόσμος και σταδιακά, θα κατανοούσε την αναγκαιότητα του νόμου ως αληθή σχέση - στοιχείο της πραγματικότητας.

Κατά συνέπεια, η μελέτη ενός θέματος από το μαθητή με χρήση υπολογιστή θα είναι πληρέστερη στο ενοποιημένο τρίπτυχο "γνωσιακό θέμα" - "πηγαιός κώδικας ως υπολογιστική αναπαράσταση του θέματος" - "αποτέλεσμα της εκτέλεσης", όπου το τρίτο μπορεί να παίζει συνθετότερους ρόλους, όπως λειτουργικής αναπαράστασης του θέματος (πχ. προσομοίωση). Η διάκριση του είδους των εκφράσεων και των μεταβλητών τους σε σχέση ή δράσης μπορεί να αξιοποιηθεί συσχετίζοντας και τις τρεις πτυχές ως διαδοχικά "επίπεδα" της μελέτης, ενδεχομένως σε επαναλαμβανόμενη ανάδραση όποτε προκύψει κάποια νέα ιδέα για τον μαθητή. Πιο συγκεκριμένα, μελέτη:

α) Στο επίπεδο των συλλογισμών κατάστροφης και επίλυσης, πριν τον προγραμματισμό. Εκεί θα διακρίνει ο μαθητής τις μεταβλητές των σχέσεων (ανεξάρτητες ή εξαρτημένες) από αυτές των λειτουργιών-δράσεων. Δυνατό να υπάρχουν στο θέμα και μεταβλητές που εννοούνται ή δηλώνονται περιφραστικά. Τέτοιες μεταβλητές, για τη μετάβαση σε πρόγραμμα, πρέπει να εντοπιστούν και να καταγραφούν ρητά, να αναγνωριστεί ο ρόλος τους και η σχέση ή δράση που τις χρησιμοποιεί, ώστε αυτή να αποτυπωθεί ως διαδικασία κατάλληλης μορφής. Για παράδειγμα, από την έλλειψη χρήσης μεταβλητών είναι που προκύπτει η δυσκολία επίλυσης προβλημάτων της πρακτικής αριθμητικής, σε σύγκριση με την επίλυσή τους μέσω αλγεβρικών σχέσεων-εξισώσεων.

β) Στο επίπεδο του πηγαιού κώδικα. Εδώ, παρ' όλο που όλες οι μεταβλητές αναπαρίστανται ως αλγοριθμικές, πρέπει -και μπορούν- να φέρουν το αρχικό νόημά τους, γενικά διακρινόμενες ως μεταβλητές σχέσης ή δράσης και ειδικότερα αν παριστούν κάποιο μέγεθος, να το μεταφέρουν επεξηγηματικά κατά την πορεία γραφής των βημάτων, με τρόπο εύληπτο για τον "επόμενο αναγνώστη". Αυτό βοηθιέται τόσο με επιλογή από κατάλληλα ονόματα των μεταβλητών (πχ. `μάζα_gr`) όσο και με δόμηση του προγράμματος ανάλογη προς την αντίστοιχη νοηματική δόμηση της επεξεργασίας του θέματος. Σχέσεις "κατηγοριοποίησης", αν και δεν υποστηρίζονται άμεσα από τη Logo, έμμεσα μπορούν να αποδοθούν μέσω λιστών.

Στο επίπεδο αυτό διακρίνονται και "αυτόνομα" οι μεταβλητές ως προγραμματιστικά στοιχεία, σε διάφορα είδη. Αυτό αποτελεί ένα ιδιαίτερο πεδίο μελέτης μεταβλητών, που αφορά τη δόμηση της προγραμματιστικής έκφρασης (Τζιμογιάννης et al, 2005).

γ) Στο επίπεδο των αποτελεσμάτων (έξοδος στην οθόνη, ρομποτική κίνηση, ήχος...). Σ' αυτό, ο μαθητής μπορεί να εντοπίσει γενικευμένες έννοιες που δεν παρουσιάζονται επεξηγηματικά ως μεταβλητές αλλά εμφανίζονται ως μεταβαλλόμενα νοήματα, σχηματικά ή λειτουργικά. Για παράδειγμα, η μεταβολή της θέσης ενός σχήματος στην οθόνη μπορεί να γίνεται αντιληπτή ως "κίνηση οντότητας". Η θεώρηση του μεταβαλλόμενου νοήματος ως μεταβλητής στον παρατηρούμενο "κόσμο" (πείραμα, φαινόμενο...) θα οδηγήσει το μαθητή σε ανακάλυψη των σχέσεων ή δράσεων που διέπουν την μεταβολή. Μεταβαίνοντας στο επίπεδο του πηγαιού κώδικα μπορεί να κάνει δοκιμαστικές αλλαγές τύπου "δοκιμή και πλάνη" και να συνάγει τη διαδικασία σχέσης ή δράσης που διέπει τη μεταβολή. Επιτρέποντας στο πρώτο επίπεδο, μπορεί να δει ευρύτερα και βαθύτερα το θέμα.

Μια άμεση μορφή υπολογιστικής αναπαράστασης μαθηματικο-λογικών νοημάτων, που συνδυάζει "εσωτερικά" και τα τρία επίπεδα, είναι αυτή της "σωματικής έκφρασης" του θέματος από το μαθητή, ο οποίος αναλαμβάνει να οργανώσει τις κινήσεις του βάσει ορισμένων αρχικών στοιχειωδών κινήσεων. Είναι εφαρμόσιμη άμεσα με κινήσεις "χελώνας Logo" του σώματος από μαθητές Δημοτικού, που με χρήση αποκλειστικά των στοιχείων κίνησης {fd, bk, rt, lt} καλούνται να εκφράσουν και να επιλύσουν συνήθη πρακτικά θέματα και να σχηματίσουν έτσι την κατάλληλη διαδικασία (Κυνίγος et al, 1993).

Το ζήτημα της διαφορετικής θεωρητικής θεμελίωσης της έννοιας της μεταβλητής στις διάφορες θεωρίες, αν και δεν αφορά άμεσα το μαθητή, οδηγεί σε καταστάσεις που πρέπει να εξηγηθούν από τον εκπαιδευτικό, που θα διακρίνει είδη εκφράσεων και μεταβλητών του πρότυπου θέματος και θα οδηγήσει τους μαθητές να τις φέρουν σε αντιστοιχία, άμεση ή έμμεση, προς τις εκφράσεις του προγράμματος. Όμως το θεωρητικό ζήτημα που ανακόπτει, έχει και άλλη μια μορφή, που πρέπει να αντιμετωπίσει ο εκπαιδευτικός: Η υπολογιστική αναπαράσταση μαθηματικών και λογικών θεμάτων σημαίνει "υποταγή και περιορισμό" των αρχικών εκφράσεων στους αλγοριθμικούς νόμους και δυνατότητες. Αυτό δημιουργεί παρανοήσεις που αν δεν αρθούν, θα οδηγήσουν τον μαθητή σε περιορισμένη γνώση, "δεμένη" με τις υπολογιστικές δυνατότητες. Ο εκπαιδευτικός με κατάλληλες συμπληρώσεις θα επισημάνει ότι οι υπολογιστικές αναπαραστάσεις είναι απλώς ένα μέσον προσέγγισης κάποιων τόπων και όχι σφαιρικά της γνώσης.

Οργάνωση σχολικού μαθήματος προγραμματισμού με τη Logo.

Στο τμήμα αυτό περιγράφεται ένας τρόπος προγραμματιστικής προσέγγισης βασικών εννοιών σχέσης και δράσης, ώστε να γίνει κατανοητή από τους μαθητές η αναγκαιότητα διάκρισής τους, ώστε να γίνουν σαφέστεροι οι ρόλοι που παίζουν οι μεταβλητές τους και οι δυνατότητες έκφρασης που ανοίγουν. Για το σκοπό αυτό, αναπτύσσονται στα επόμενα ορισμένες διαδικασίες ενδεικτικά, με έμφαση προς την κατεύθυνση της "σχέσης" διότι η κατεύθυνση αυτή είναι που κυρίως επιτρέπει ανάπτυξη αφαιρετικής σκέψης.

Το θέμα είναι επιλεγμένο από το μαθηματικό χώρο, που επιτρέπει εμβάθυνση σε έννοιες μέσα από μια συνοπτική παρουσίαση. Αντίστοιχα θέματα προς μελέτη της έννοιας της μεταβλητής που θα φθάνουν μέχρι αξιοποίηση των γνώσεων που αποκτά ο μαθητής, μπορούν να τεθούν στο πλαίσιο κάθε γνωστικού χώρου (Λογικής, Φυσικής, Γλώσσας, Ζωολογίας, Φυτολογίας κ.α.).

Το πρόγραμμα αναπτύσσεται με σκοπό τη σταδιακή εξοικείωση του μαθητή με την έννοια της μεταβλητής στις δύο εννοιολογικές μορφές της, "σχέσης" και "δράσης". Με βάση αυτή τη διάκριση, ο μαθητής θα αντιμετωπίσει προγραμματιστικά την έννοια της μαθηματικής σύνθεσης συναρτήσεων, αλλά και της σύνθεσης δράσης με σχέση ως αλγόριθμο που επεξεργάζεται συνάρτηση, φθάνοντας στο επίπεδο της υλοποίησης αφαιρετικών διαδικασιών. Για το σκοπό αυτό αξιοποιούνται οι δυνατότητες της Logo να δέχεται κλήση μεταβλητών συναρτήσεων και να δέχεται συναρτησιακές διαδικασίες. Ο μαθητής θα κινηθεί στο νόημα-τρόπο υπολογισμού "απ' έξω προς τα μέσα" ("call by name" ή "call by reference") σε σύγκριση με τον υπολογισμό "από μέσα προς τα έξω" ("call by value"), εντοπίζοντας τα κοινά που έχει το μαθηματικό νόημά τους με το προγραμματιστικό.

Ο κώδικας είναι υλοποιημένος σε compiler FMSLogo (Robertson, 2003), χωρίς να χρησιμοποιεί ειδικά στοιχεία αυτού του compiler. Οι διαδικασίες νοούνται υλοποιούμενες διαδοχικά, προσθετικά στο ίδιο πρόγραμμα.

Τα σχόλια νοούνται ως υποδείξεις που θα απευθύνονται προς τους μαθητές.

Προκαταρκτικές διαδικασίες.

α) Ορισμός του "τριωνόμου" $f(x)=ax^2+bx+c$ ως συναρτησιακή διαδικασία (ΣΔ) με τέσσερις μεταβλητές εισόδου, όπου οι a , b , c νοούνται πραγματικές παράμετροι και x η ανεξάρτητη πραγματική μεταβλητή. Επιστρέφει την τιμή του τριωνόμου για δεδομένες τιμές των μεταβλητών:


```

To trionymo :a :b :c :x
Op (:a * :x * :x + :b * :x + :c)
End

```

Η διαφοροποίηση των παραμέτρων a, b, c από την ανεξάρτητη μεταβλητή x εναπόκειται στον τρόπο χρήσης του αποτελέσματος.

β) Ορισμός της συνάρτησης $f(x)=x^3$ ως ΣΔ με μία μεταβλητή εισόδου. Επιτρέφει τον κύβο της τιμής εισόδου, με πεδίο ορισμού το R :

```

To cube :x
Op (:x * :x * :x)
End

```

γ) Εξοικείωση με "μεταβλητές διαδικασίες" (μέσω της εντολής apply): Ορίζουμε ΣΔ με όνομα f , δύο μεταβλητών εισόδου, g και x , όπου η g θα δέχεται ως τιμή όνομα μιας ΣΔ και η x την τιμή που θα δοθεί ως εισοδος στην g

```

To f :g :x
Op apply :g :x ;εφάρμοσε τη διαδικασία g στην τιμή της x
End

```

Κλήση: print f "cube [2]. *Απάντηση:* 8 (ίδιο με: print cube 2)

Σημ.: Η εντολή apply απαιτεί να τεθεί σε αγκύλες η τιμή της μεταβλητής εισόδου x για να καλυφθεί τεχνικά και η περίπτωση να έχει η g περισσότερες της μιας μεταβλητές εισόδου, που οι αντίστοιχες τιμές τους θα δίνονταν επίσης στη λίστα αυτή, όπως:

```

Κλήση: print f "sum [1 2 3 4] . Απάντηση: 10
(αποτέλεσμα ίδιο με της κλήσης: print (sum 1 2 3 4) )
Κλήση: print f "count [[2 3 4]] . Απάντηση: 3
(αποτέλεσμα ίδιο με της κλήσης: print count [2 3 4] )

```

Σύνθεση ήδη ορισμένων ΣΔ.

Στην περίπτωση σύνθεσης συγκεκριμένων συναρτήσεων δεν απαιτείται χρήση του αφηρημένου ορισμού της σύνθεσης.

α) Ορισμός διαδικασίας που αποδίδει τη μαθηματική σύνθεση $h(x)=f(g(x))$ όπου $f(x)=ax^2+bx+c$ και $g(x)=x^3$ (δηλ. η συνάρτηση $h(x)=(ax^2+bx+c)^3$) όταν είναι ήδη ορισμένες οι συναρτήσεις ως διαδικασίες cube και trionymo. Πεδίο ορισμού το R .

```

To cube_trionymo :a :b :c :x
Op cube (trionymo :a :b :c :x)
End
Κλήση: 1 + cube_trionymo 1 2 3 4
Απάντηση: You don't say what to do with 19684

```

Ερμηνεία: Το νόημα της op είναι να δώσει το αποτέλεσμα "συναρτησιακά", δηλαδή να αποτελέσει τιμή εισόδου προς αλγόριθμο "γραμμένο αριστερά της". Αν αυτός δεν υπάρχει, τυπώνει το μήνυμα ως καθοδήγηση διόρθωσης για τον χρήστη. Όμως δεν σημαίνει "σφάλμα", αντίθετα εδώ αποτελεί ένδειξη ότι η συναρτησιακή σύνθεση "λειτουργεί σωστά".

β) Διαδικασία που ορίζει τη μαθηματική σύνθεση $h(x)=f(g(x))$ όπου $f(x)=x^3$ και $g(x)=ax^2+bx+c$ (δηλ. τη συνάρτηση $h(x)=a(x^3)^2+bx^3+c$). Πεδίο ορισμού της σύνθεσης, R^+_0

```

To trionymo_sqrt :a :b :c :x
Op trionymo :a :b :c sqrt :x
End
Κλήση: print trionymo_sqrt 1 2 3 4 . Απάντηση: 11

```

Σύνθεση μεταβλητών διαδικασιών.

Η καλούμενη διαδικασία πρέπει να είναι μορφής ΣΔ ενώ η καλούσα μπορεί να είναι οποιαδήποτε, αρκεί να δέχεται ως εισοδο την έξοδο της καλούμενης.

α) Ορίζουμε τη ΣΔ `compose` που ορίζει τη μαθηματική σύνθεση δύο μεταβλητών συναρτήσεων. Οι μεταβλητές `f1` και `f2` θα παίρνουν, κατά την κλήση της `compose`, ως τιμή το όνομα της καλούσας και της καλούμενης συνάρτησης αντίστοιχα, ενώ η μεταβλητή `x` θα παίρνει ως τιμή τη **λίστα** των τιμών εισόδου της `f2` (από καμία μέχρι `n`):

```
To compose :f1 :f2 :x
Op apply :f1 (list apply :f2 :x)
End
α' κλήση: print (compose "cube "sqrt [9] ). Απάντηση: 27
β' κλήση: print (compose "sqrt "trionymo [3 4 5 6])
```

Παρατήρηση: Το β' είναι το ίδιο με `print (sqrt trionymo 3 4 5 6)` αλλά ο υπολογισμός μέσω της `compose` πλεονεκτεί νοηματικά κατά το ότι είναι σε αφηρημένο επίπεδο, με μεταβλητές διαδικασίες εισόδου.

β) Διαδοχική σύνθεση τριών συναρτήσεων:

```
To compose3 :f1 :f2 :f3 :x
Op apply :f1 (list (compose :f2 :f3 :x))
End
Κλήση: print compose3 "cube "sqrt "trionymo [1 2 3 4].
```

Παρόμοια, ορίζεται η σύνθεση τεσσάρων, πέντε κ.ο.κ. ΣΔ.

Σύνθεση δεδομένης διαδικασίας δράσης (ΔΔ) με μεταβλητή ΣΔ.

α) Ορίζουμε διαδικασία δράσης `plot` που εμφανίζει τη μορφή της γραφικής παράστασης της μαθηματικής συνάρτησης $f(x)$ που θα δοθεί ως τιμή εισόδου. Η διαδικασία αυτή αποτελεί προκαταρκτικό βήμα για τη σύνταξη της επόμενης.

Η `plot` σχεδιάζει ένα σημείο της συνάρτησης στη θέση (x,y) με $y=f(x)$, συνδυάζοντας την `setxy` (μετάβαση στο σημείο (x,y)) με την `setpixel` (αποτύπωση μιας τελείας) και αναδρομικά σχεδιάζει κάθε φορά "άλλο ένα σημείο", αυξάνοντας το x κατά ένα βήμα.

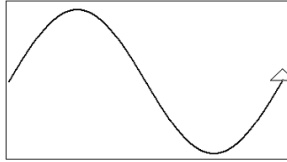
Η `plot` έχει πέντε μεταβλητές εισόδου: i) `f`, που θα λαβαίνεται ως τιμή τη συνάρτηση, ii) `firstx`, που παίζει **διπλό ρόλο**, ως μεταβλητή της f και ως αρχική τιμή του x , iii) `lastx`, που παριστά το τελευταίο σημείο στον άξονα X , iv) `stepx`, που είναι το βήμα μεταβολής για τα διαδοχικά x και v) `magny`, που είναι ο συντελεστής "διαστολής" του άξονα Y για να γίνει πιο ευκρινής η μορφή της καμπύλης. Οι άξονες εδώ είναι νοητοί και είναι οι άξονες της Logo, οι οποίοι ορίζονται με μηδέν στο κέντρο της οθόνης. Εσωτερικά στη διαδικασία, χρησιμοποιείται βοηθητικά και η μεταβλητή `yp` για λόγους απλοστευσης της έκφρασης.

```
To plot :f :firstx :lastx :stepx :magny
local "xp make "xp :firstx
local "yp make "yp ((apply :f (list :firstx))* :magny)
pu setxy :xp :yp pd setpixel [0 0 0]
if (:firstx > :lastx) [stop]
plot :f (:firstx + :stepx) :firstx :lastx :stepx :magny ;αναδρομή
End
```

Παρατήρηση: Η `stepx` για τον ορισμό της συνάρτησης f παίζει ρόλο "μεταβλητής σε σχέση", ως αντίστοιχη της ανεξάρτητης μεταβλητής της $f(x)$ ενώ η ίδια, για την κατασκευή των γραφικών, παίζει ρόλο "μεταβλητής σε δράση". Εδώ, οι ρόλοι αυτοί "συμβαδίζουν".

Εφαρμογή: Γραφική παράσταση της $\sin(x)$, για x από 0 έως 360, με βήμα 0.1 για το x και μεγέθυνση του άξονα Y επί 80

Κλήση: `cs plot "sin 0 360 0.1 80`



Χρήσιμη επέκταση αυτής της διαδικασίας είναι να σχεδιάζει και τους άξονες βαθμολογημένους κατάλληλα ώστε το σχήμα να συμπεριλάβει την ή τις απαιτούμενες περιόδους.

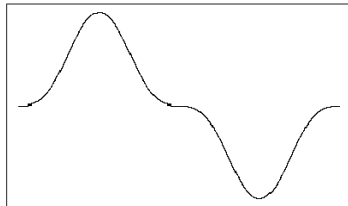
β) Γραφική παράσταση της σύνθεσης $f(g(x))$ δύο μεταβλητών συναρτήσεων, f και g . Ορίζουμε διαδικασία δράσης `plot2` ανάλογη προς την `plot`.

Για το σκοπό αυτό, αρκεί να αλλάξουμε μια "λειπομέρεια" στην `plot`: Να προσδιορίσουμε, χρησιμοποιώντας και τη διαδικασία `compose` που προαναφέρθηκε, το σημείο y της σύνθεσης που αντιστοιχεί σε δεδομένο x :

```
To plot2 :f :g :firstx :lastx :stepx :magny
  local "yp make "yp (compose :f :g (list :firstx))* :magny
  pu setxy :firstx :yp
  pd setpixel [0 0 0]
  if (:firstx > :lastx) [stop]
  plot2 :f :g (:firstx + :stepx) :lastx :stepx :magny ;αναδρομή
End
```

Εφαρμογή: Γραφική παράσταση της σύνθετης συνάρτησης $(\text{cube}(\sin(x)))$, για x από 0 έως 360, με βήμα 0.2 για το x και μεγέθυνση του άξονα Y επί 100.

Κλήση: `cs plot2 "cube "sin 0 360 0.2 100`



Παρατηρούμε στη γραφική παράσταση την περιοδικότητα της σύνθετης συνάρτησης, αποτέλεσμα του ότι η καλούμενη συνάρτηση είναι περιοδική. Αυτό αποτελεί ένα ιδιαίτερο θέμα μελέτης για το μαθητή.

Η `plot2`, ως σύνθεση της δράσης "γραφική παράσταση" με τη συνάρτηση που είναι το αποτέλεσμα της σύνθεσης δύο μεταβλητών, αποτελεί μια "νοηματική πράξη" που επιτρέπει στον μαθητή την ανάπτυξη συλλογισμών που ξεκινούν από το γενικό νόημα "γραφική παράσταση σύνθετης συνάρτησης" με άμεση δυνατότητα εξειδίκευσής του σε οποιοδήποτε συνθέσεις. Το πρόγραμμα αυτό ως σύνολο διαδικασιών δεν δίνεται απ' ευθείας ως "προκατασκευασμένο εργαλείο" στον μαθητή αλλά προκαλείται αυτός να εμβαθύνει στο πώς κτίζεται, με τη βοήθεια του διδάσκοντα.

Συμπεράσματα και περαιτέρω έρευνα.

Τα παραδείγματα διαδικασιών που αναφέρθηκαν, είναι περιορισμένα στον μαθηματικό χώρο. Αντίστοιχου επιπέδου παραδείγματα είναι δυνατό να αναπτυχθούν στον λογικό χώρο καθώς και σε χώρους εφαρμογών σε επιστήμες ή σε σημασιολογικά θέματα. Η μεθόδευση που προτείνεται, είναι ενδεικτική προς την κατεύθυνση μιας λεπτομερέστερης και πληρέστερης οργάνωσης, με πειραματικό έλεγχο "στην τάξη", που για το στάδιο αυτό αποτελεί μελλοντική έρευνα.

Για μια μεθόδευση διδασκαλίας όπως η προτεινόμενη, μπορούμε -και πρέπει- να λαβαίνουμε υπ' όψη ότι οι μαθητές σήμερα είναι εξοικειωμένοι σε μεγάλο βαθμό με την έννοια της υπολογιστικής διαδικασίας "δράσης", έχοντας αποκτήσει δεξιότητες από πολλές τεχνολογικές ενασχολήσεις στη ζωή τους: Internet, smartphones, παιχνίδια, οικιακές συσκευές κ.α. Το "βήμα γνώσης" που χρειάζεται να κάνουν, είναι προς την κατεύθυνση οργάνωσης και εκμετάλλευσης εκφράσεων σχέσης και αυτό θεωρούμε ότι θα εξυπηρετηθεί από τον προγραμματισμό αντίστοιχων διαδικασιών, σε βαθμό που θα φανεί από την εφαρμογή της προτεινόμενης διδασκαλίας στην τάξη.

Με βάση τα προηγούμενα, η μελλοντική μας έρευνα προσανατολίζεται στην "εφαρμογή και αξιολόγηση στην τάξη": Σχεδιάζονται δύο ομάδες ωριαίων μαθημάτων που θα αποτελέσουν μέρος του μαθήματος Πληροφορικής, σε δύο επίπεδα γνώσεων: Γυμνασίου, όπου ο μαθητής έχει τις πρώτες επαφές με την έννοια της μαθηματικής μεταβλητής και είναι σημαντικό να εναρμονιστεί από τα πρώτα στάδια αυτή η γνώση με το λειτουργικό νόημα της μεταβλητής και Λυκείου σε πιο ολοκληρωμένο από πλευράς ύλης περιβάλλον.

Απώτερος σκοπός είναι η ανάπτυξη ικανοτήτων σκέψης απ' ευθείας σε επίπεδο σχέσεων, που θα επιτρέπει "από πάνω προς τα κάτω" δομημένους συλλογισμούς, όχι μόνο ως αναλυτική μέθοδος επίλυσης (την οποία εννοείται πως θα ακολουθήσει η αντίστοιχη συνθετική) αλλά και ως ικανότητα οργάνωσης σκέψης σε αφαιρετικό επίπεδο, δυναμικά εφαρμόσιμης "προς τα κάτω" όπου και όταν χρειαστεί. Επομένως το ζήτημα που πρέπει να ερευνηθεί είναι, αν με την προτεινόμενη διδακτική μεθόδευση ο μαθητής θα φθάσει ευκολότερα να κάνει συλλογισμούς σε πιο αφαιρετικά επίπεδα, που είναι θεμελιακό βήμα για την ανάπτυξη μαθηματικής σκέψης.

Μια κριτική για την επιλογή μας της γλώσσας προγραμματισμού, είναι ότι η γλώσσα Logo είναι από κατασκευής κατάλληλη για να διευκολύνει τη διαδικαστική σκέψη "δράσης" και μέσα από αυτή επιχειρήσαμε μια αξιοποίηση προς συλλογισμούς "πάνω σε σχέσεις". Πιο άμεση και πληρέστερη προσέγγιση θα ήταν με τη γλώσσα Scheme, που είναι από κατασκευής συναρτησιακή, χωρίς να διαφοροποιεί "κοινές μεταβλητές" από "μεταβλητές συναρτήσεις", διευκολύνοντας πολύ την αναπαράσταση συλλογισμών σχέσης (πχ. δεν χρειάζεται εντολή τύπου "apply"). Αλλά, αν και προσιτή η γλώσσα αυτή για το επίπεδο των μαθητών Γυμνασίου και Λυκείου (Friedman et al, 1988; Harvey, 1999) θα απαιτούσε προσαρμογή του εκπαιδευτικού προγράμματος καθώς και μετεκπαίδευση διδασκόντων.

Αναφορές

- Church, A. (1941). *The Theory of Lambda Conversion*. Princeton University Press.
Friedman, D. P. & Felleisen, M. (1998). *The Little SCHEMEr*. MIT Press.
Harvey, B. (1997). *Computer Science Logo Style*, MIT Press (3 volumes).
Harvey, B. (1999). *Simply Scheme: Introducing Computer Science*. MIT Press.
Kleene, S.C. (1962). *Introducion to Metamathematics*. North-Holland.

- Kynigos, C., Gyftodimos, G. & Georgiadis, P. (1993). Empowering a Society of Future Users of Information Technology. In: *European Journal of Information Systems* 4 , p. 139-148.
- Mendelson, E. (2009). *Introduction to Mathematical Logic*. CRC Press.
- Noss, R., & Hoyle, C. (1996). *Windows on mathematical meanings*. Dordrecht, The Netherlands: Kluwer.
- Robertson, J. (2003). *FMS Logo Manual*. <http://fmslogo.sourceforge.net/manual/>
- Sajaniemi, J. (2005). Roles of Variables and Learning to Program. Proceedings of the 3rd Panhellenic Conference "*Didactics of Informatics*", A. Jimoyiannis (ed.). University of Peloponnese, Korinthos, Greece.
- Δεϊμέζης, Μ. & Γυφτοδήμος, Γ. (2016). Προς μια ολοκληρωμένη διδασκαλία των εννοιών της μεταβλητής στα διάφορα γνωστικά πεδία, μέσω προγραμματισμού. Πρακτικά συνεδρίου "*Η Πληροφορική στην Εκπαίδευση*" (CIE 2016), Πανεπιστήμιο Πειραιά, σ. 520-530.
- Δεϊμέζης, Μ. (2016). *Προγραμματισμός Υπολογιστών. Θεωρία & Λυμένες Ασκήσεις Προγραμματισμού σε MSW LOGO με σχόλια*, Αθήνα: Δεϊμέζης Μ.
- Τζιμογιάννης, Α. & Κόμης, Β. (2000). Η έννοια της μεταβλητής στον προγραμματισμό: Δυσκολίες και παρανοήσεις μαθητών του Ενιαίου Λυκείου. Πρακτικά 2ου Πανελληνίου Συνεδρίου "*Οι Τεχνολογίες της Πληροφορίας και της Επικοινωνίας στην Εκπαίδευση*", σ. 103-114, Πάτρα.
- Τζιμογιάννης, Α., Πολίτης, Π. & Κόμης, Β. (2005). Μελέτη των αναπαραστάσεων τελειόφοιτων μαθητών Ενιαίου Λυκείου για την έννοια της μεταβλητής. Στο: Α. Τζιμογιάννης (επιμ.), Πρακτικά 3ου Πανελληνίου Συνεδρίου "*Διδακτική της Πληροφορικής*", σ. 61-70, Κόρινθος.