Προηγούμενη Αρχική Σελίδα Θεματικά Πεδία Επόμενη

Αξιοποιώντας τις Εκπαιδευτικές Δυνατότητες

του Περιβάλλοντος Ptolemy

Θ.Χ. Κασκάλης[1] *Β. Δαγδιλέλης*[2] *Γ. Ευαγγελίδης*[1] *Κ.Γ. Μαργαρίτης*[1]

[1]Τμήμα Εφαρμοσμένης Πληροφορικής

[2]Τμήμα Εκπαιδευτικής και Κοινωνικής Πολιτικής

Πανεπιστήμιο Μακεδονίας

Εγνατίας 156, 54006 Θεσσαλονίκη

E-mail: {kaskalis, dagdil, gevan, kmarg }@uom.gr

Περίληψη

Οι μαθησιακές δυσκολίες που παρουσιάζονται στη διδασκαλία του προγραμματισμού υπολογιστικών συστημάτων μας οδηγούν στην αναζήτηση διαφορετικών προσεγγίσεων του θέματος. Το περιβάλλον Ptolemy αποδεικνύεται ένα χρήσιμο εργαλείο ανάπτυξης αλγορίθμων μέσω οπτικών μεθόδων. Μέσα από μία σειρά παραδειγμάτων παρουσιάζεται η βήμα προς βήμα ανάπτυξη ορισμένων πρότυπων ασκήσεων, με σκοπό τη διαπίστωση των πλεονεκτημάτων προς αυτήν την κατεύθυνση. Χρήσιμα χαρακτηριστικά όπως το υψηλό επίπεδο αφαίρεσης, η φιλικότητα προς το χρήστη, η άμεση ιεραρχική δόμηση και η εύκολη επαναληπτική χρησιμοποίηση επιμέρους δομικών στοιχείων, αποκαλύπτονται προοδευτικά. Ταυτοχρόνως, διαπιστώνονται οι δυνατότητες υποστήριξης ενός απομακρυσμένου μοντέλου εκπαίδευσης και εντοπίζεται η άμεση αναπαράσταση των αρχών της παράλληλης εκτέλεσης των διαδικασιών. Τα στοιχεία αυτά συνθέτουν μία αποτελεσματική πρόταση, που σκοπό έχει να διατηρεί τη διαχρονικότητα των γνώσεων, αποσυνδέοντας την εκμάθηση του προγραμματισμού από την αποστήθιση δυσνόητων εντολών με δυσνόητο χαρακτήρα. Το περιβάλλον Ptolemy προτείνεται σαν μία νέα πλατφόρμα ανάπτυξης εκπαιδευτικών μεθόδων μέσω της οπτικής δόμησης ποικίλων αλγοριθμικών εννοιών τα οποία μπορούν να χρησιμοποιηθούν από απόσταση.

Abstract

The area of computer systems' programming, regarded as a teaching topic, presents a certain amount of educational difficulties. Attempting to resolve these difficulties, we propose the Ptolemy environment as a useful tool for the visual implementation of algorithmic methods. Through an incremental creation of a series of examples, we attempt to identify the advantages of such an approach. Useful features, such as the high level of abstraction, user friendliness, direct hierarchical construction and effortless reusability of created components, are exhibited through the various implemented exercises. At the same time, we identify the potential for a distance learning model and we recognize the immediate representation of the parallel execution principles, included in such processes. These features comprise an effective proposal, directing towards long lasting knowledge, since the memorizing of strict programming commands appears quite inefficient. The Ptolemy environment suggests a new platform for the development of innovative educational methods, through the visual construction of algorithmic notions.

1. Εισαγωγή

Η διδασκαλία του προγραμματισμού υπολογιστικών συστημάτων αποτελεί ένα πολύ ενδιαφέρον και σημαντικό κεφάλαιο στο χώρο της εκπαίδευσης. Οι σχετικές έρευνες δείχνουν την ύπαρξη ενός μεγάλου πλήθους προβλημάτων κατανόησης και εμβάθυνσης του θέματος (για παράδειγμα [2], [3], [4], [5], [6], [7]) καθιστώντας την αποτελεσματική προσέγγισή του μία πολύπλοκη διαδικασία. Στο άρθρο αυτό προτείνουμε μία εισαγωγή στον προγραμματισμό που στηρίζεται σε ένα "οπτικοποιημένο" ενδιάμεσο. Συγκεκριμένα, αξιοποιούμε το περιβάλλον Ptolemy [1] για τη μεταφορά βασικών προγραμματιστικών μεθόδων σε μία φιλική προς τον χρήστη άποψη των υπό μελέτη αλγορίθμων.

Η χρησιμοποίηση γραφικών δομικών στοιχείων επιτρέπει στο σπουδαστή να εκφράσει με σαφήνεια τα βήματα που απαιτούνται για την επίλυση του προτεινόμενου προβλήματος. Ταυτοχρόνως, ο σπουδαστής δε δεσμεύεται από περίπλοκες εντολές και κανόνες σύνταξης, που συνήθως απαιτείται να αποστηθίσει στις κλασικές γλώσσες προγραμματισμού. Μέσω ενός φιλικού περιβάλλοντος "χτίζει" το κύκλωμα, που υλοποιεί την ιδέα του, και μπορεί εύκολα να διαπιστώσει την ορθή λειτουργία του. Ο ιεραρχικός τρόπος σχεδίασης, επιτρέπει στο χρήστη να κατασκευάσει δομικά στοιχεία, τα οποία μπορεί να χρησιμοποιήσει επανειλημμένα σε ένα ανώτερο επίπεδο. Η εύκολη επαλήθευση της σωστής λειτουργίας των επιμέρους κυκλωμάτων τον απαλλάσσει από την δύσκολη διαδικασία της εκ των υστέρων εκσφαλμάτωσης (debugging) και βρίσκεται σε ευθεία αναλογία με την έννοια της "συνάρτησης" ή "υπορουτίνας".

Όπως θα δούμε και στη συνέχεια, η προσέγγιση αυτή οδηγεί και σε περαιτέρω χρήσιμα αποτελέσματα, καθώς ο μαθητευόμενος εξοικειώνεται με την έννοια του *παραλληλισμού* των διαδικασιών. Μπορεί, δηλαδή, να διαπιστώσει τα τμήματα του υπό ανάπτυξη αλγορίθμου, που μπορούν να εκτελεσθούν ταυτοχρόνως, καθώς οι συγκεκριμένες επιμέρους ενέργειες δεν απαιτούν χρονική αλληλουχία. Επίσης, η δημιουργία τέτοιων δομικών στοιχείων επιτρέπει τον εξ αποστάσεως έλεγχο της ορθότητάς τους, δημιουργώντας κατάλληλες προϋποθέσεις για ένα απομακρυσμένο μοντέλο εκπαίδευσης, καθώς το συγκεκριμένο περιβάλλον Ptolemy μπορεί να χρησιμοποιηθεί από απόσταση. Τέλος, αξίζει να σημειωθεί η δυνατότητα πειραματισμού ενός υπό μελέτη προβλήματος για διαφορετικές μεταβλητές εισόδου. Το πλήθος των γραφικών αναπαραστάσεων, που προσφέρει το Ptolemy, επιτρέπει την οπτικοποίηση των ιδιαίτερων χαρακτηριστικών ενός αλγορίθμου.

Όλα τα παραπάνω χαρακτηριστικά, θα παρουσιαστούν στη συνέχεια, μέσω αναλυτικών παραδειγμάτων. Ξεκινώντας από μία σύντομη παρουσίαση του περιβάλλοντος Ptolemy, θα προχωρήσουμε σε μία βήμα προς βήμα κατασκευή ενός πιθανού προγραμματιστικού σεναρίου. Συγκεκριμένα, θα κατασκευάσουμε μία απλή δευτεροβάθμια εξίσωση, με σκοπό να μελετήσουμε την επίδραση της μεταβολής του σταθερού όρου στη γραφική της παράσταση. Ακολούθως, θα αναπτύξουμε γραφικά τον αλγόριθμο επίλυσης της δευτεροβάθμιας εξίσωσης, αξιοποιώντας τα δομικά στοιχεία, που κατασκευάσαμε προηγουμένως και αναπτύσσοντας ακόμη περισσότερα. Στα πλαίσια της εκ των υστέρων μελέτης, θα δούμε και πάλι πώς επηρεάζει ο σταθερός όρος τις ρίζες της εξίσωσης και θα διαπιστώσουμε τα τμήματα του αλγορίθμου, που εκτελούνται παράλληλα.

Τέλος, θα εξηγήσουμε την απλότητα της αυτόματης και εξ αποστάσεως επαλήθευσης ενός κυκλώματος, που έχει κατασκευαστεί από έναν σπουδαστή. Βεβαίως, η αξία της διαδικασίας αυτής δεν έγκειται μόνον στην ευκολία της αυτόματης επαλήθευσης, αλλά κυρίως αφορά τη δυνατότητα του σπουδαστή να ελέγχει μόνος του την ορθότητα του κυκλώματός του, πριν υποβάλει την τελική μορφή της προτάσεώς του.

2. Παρουσίαση του Ptolemy

Το περιβάλλον Ptolemy [1] αναπτύχθηκε στο Πανεπιστήμιο της Καλιφόρνια στο Berkeley. Αποτελεί μία ευέλικτη πλατφόρμα πάνω στην οποία μπορούν να δημιουργηθούν, με τρόπο απλό και άμεσο, διαφόρων μορφών πρωτότυπα. Υπάρχουν πολλά παραδείγματα ανεπτυγμένων πρωτοτύπων, συμπεριλαμβάνοντας γραφικούς προγραμματισμούς επεξεργασία σημάτων, μοντελοποίηση δικτύων επικοινωνίας μέσω διακριτών γεγονότων, σχεδιασμό κυκλωμάτων σε επίπεδο καταχωρητών, περιβάλλοντα σύνθεσης καθώς και σχεδιαστικά προγράμματα που βοηθούν την ανάπτυξη προτύπων υλικού/λογισμικού.

Αν και η πλειοψηφία των χρηστών περιορίζεται σε ένα μόνο περιβάλλον, το Ptolemy συνδυάζει ποικίλα περιβάλλοντα σύνθεσης και εξομοίωσης παραδειγμάτων. Αποτελεί λογισμικό τρίτης γενιάς, υποστηρίζοντας ετερογενείς περιγραφές συστημάτων, σχεδιασμό και εξομοίωση. Επίσης, εκτός από τη συνήθη χρήση του ιεραρχικού τρόπου σχεδιασμού για λόγους απλότητας, αξιοποιεί τις ιεραρχικές του δομές για να επιτρέψει τη συνεργασία διαφορετικών μοντέλων υπολογισμού. Το αποτέλεσμα είναι ένα ενοποιημένο περιβάλλον προγραμματισμού, που επεκτείνει τη φιλοσοφία ανάπτυξης και εξομοίωσης κυκλωμάτων έως τη γρήγορη παραγωγή προτύπων σύνθετων συστημάτων.

Κάθε μοντέλο προγραμματισμού στο Ptolemy ονομάζεται πεδίο (domain). Μία από τις πιο επεξεργασμένες και ώριμες, από πλευρά λειτουργικότητας, περιπτώσεις πεδίων αποτελεί το Synchronous Dataflow (SDF) πεδίο. Αυτό χρησιμοποιείται ιδιαίτερα αποτελεσματικά για την ανάπτυξη αλγορίθμων επεξεργασίας σημάτων. Άλλα πεδία είναι επίσης τα Dynamic Dataflow (DDF), Boolean Dataflow (BDF), Discrete Event modelling (DE), Message Queues (MQ) και Communicating Processes (CP). Εδώ θα εστιάσουμε την προσοχή μας στο πεδίο SDF, το οποίο αποδεικνύεται αρκετό για τα παραδείγματα, που θα παρουσιάσουμε

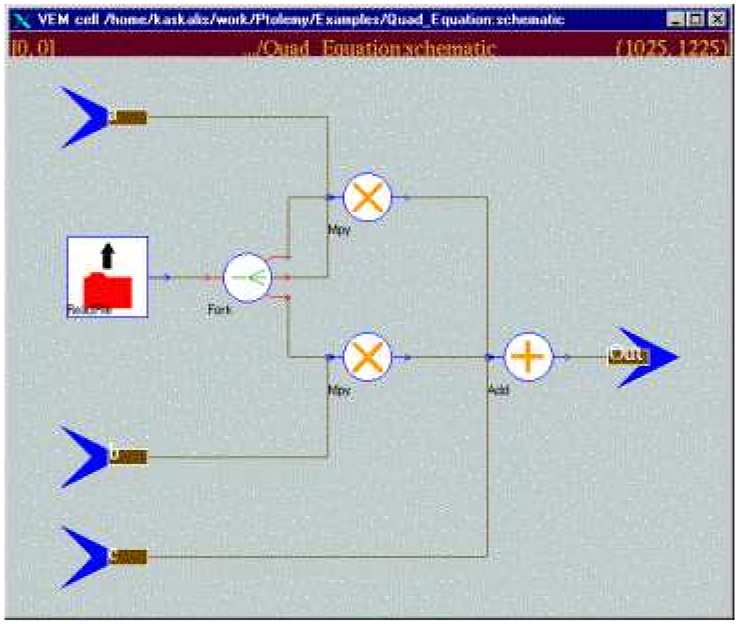
Στο Ptolemy η συνολική περιγραφή ενός συστήματος προκύπτει από τη σύνθεση μικρότερων τμημάτων που ονομάζονται **blocks**. Αυτά τα blocks καλούνται σε επίπεδο εκτέλεσης από τον δρομολογητή (**scheduler**) και ανταλλάσσουν δεδομένα μεταξύ τους, ενώ ο δρομολογητής καθορίζει τις λειτουργικές απαιτήσεις των διασυνδεδεμένων τμημάτων. Από την πλευρά του χρήστη, υπάρχουν δύο τύποι blocks: το **star** και το **galaxy**. Το star βρίσκεται στο κατώτερο επίπεδο της ιεραρχίας, υπό την έννοια ότι υλοποιείται μέσω κώδικα που γράφεται από τον χρήστη. Πρέπει να επισημάνουμε ότι ένα μεγάλο πλήθος από προγραμματισμένα stars προσφέρεται ήδη μέσω της βιβλιοθήκης του Ptolemy. Το galaxy αποτελεί εκείνο το block που περιέχει stars και, ενδεχομένως, άλλα galaxies. Το galaxy είναι, επομένως, η δομή που περιγράφει και εξασφαλίζει την ιεραρχική οργάνωση της εξομοίωσης. Τέλος, το **universe**, αποτελεί ένα ολοκληρωμένο πρόγραμμα ή εφαρμογή.

Το γραφικό περιβάλλον που επιτρέπει τη σχεδίαση των διαφόρων εφαρμογών στο Ptolemy είναι το Ptolemy Interactive Graphical Interface (pigi). Με το pigi ο χρήστης αναπτύσσει εφαρμογές, διασυνδέοντας εικόνες με γραφικό τρόπο, με αποτέλεσμα το όλο περιβάλλον να είναι φιλικό και ταυτοχρόνως λειτουργικό. Ένα σύνολο από universes μπορεί να κατασκευαστεί και καθένα από αυτά να αποτελεί και μια διαφορετική εφαρμογή. Όπως έχει ήδη γίνει κατανοητό, κάθε universe περιέχει μια ιεραρχική δομή από galaxies και stars, με τα τελευταία να αποτελούν τους ακρογωνιαίους λίθους κάθε σχεδίασης.

Στη συνέχεια ο χρήστης μπορεί να "τρέξει" ένα universe, έτσι ώστε να διαπιστώσει την ορθή λειτουργία του. Γίνεται προφανές, επομένως, ότι το Ptolemy μπορεί πράγματι να χρησιμοποιηθεί για την ανάπτυξη προτύπων, η εσωτερική δομή των οποίων περιγράφεται με τρόπο ικανοποιητικά λεπτομερή.

3. Κατασκευή και Μελέτη Δευτεροβάθμιας Εξίσωσης

Ένα απλό παράδειγμα αναπαράστασης των δυνατοτήτων του Ptolemy είναι αυτό της κατασκευής και μελέτης μιας δευτεροβάθμιας εξίσωσης: $f(x) = ax^2+bx+c$. Όπως αναφέρθηκε και προηγουμένως, θα εστιάσουμε την προσοχή μας στο πεδίο SDF και θα χρησιμοποιήσουμε τις ήδη υπάρχουσες οικογένειες stars, που έρχονται εξ ορισμού με το πρόγραμμα. Στο Σχήμα 1. βλέπουμε την κατασκευή ενός βασικού galaxy, που υλοποιεί την επιθυμητή συνάρτηση.

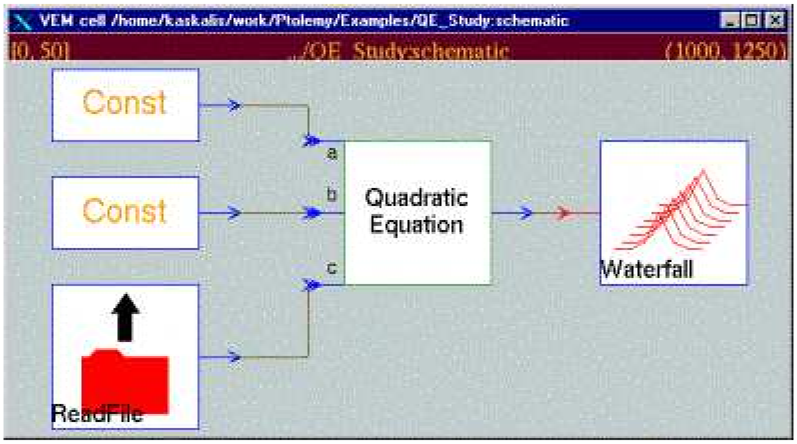


Σχήμα 1 Υλοποίηση Δευτεροβάθμιας Εξίσωσης

Παρατηρούμε την ύπαρξη δύο πολλαπλασιαστών και ενός αθροιστή. Τα stars αυτά ανήκουν στην οικογένεια Arithmetic του πεδίου SDF και δέχονται οποιοδήποτε αριθμό εισόδων. Ο τρόπος τοποθέτησής τους συμβολίζει τη διαδοχή της εκτέλεσης των πράξεων: οι πολλαπλασιασμοί μπορούν να εκτελεστούν ταυτοχρόνως και έπεται η άθροιση. Το fork star απλά διακλαδίζει τα δεδομένα, που πηγάζουν από το star ReadFile (οικογένεια Signal Sources). Το τελευταίο αυτό στοιχείο διαβάζει δεδομένα από ένα αρχείο. Στο αρχείο αυτό έχουμε φροντίσει να αποθηκεύσουμε ένα σύνολο από τιμές της μεταβλητής x, βάσει των οποίων θέλουμε να μελετήσουμε τη δοσμένη συνάρτηση. Στο συγκεκριμένο παράδειγμα αποθηκεύσαμε τις 200 ακέραιες τιμές του διαστήματος [-99,100].

Τα στοιχεία που απομένουν και συμβολίζουν τους σταθερούς όρους a,b,c καθώς και το αποτέλεσμα της συνάρτησης (Out στο Σχήμα 1.) αποτελούν βασικά stars του Ptolemy (πεδίο System) και εξυπηρετούν την επικοινωνία των galaxies σε ένα ανώτερο ιεραρχικό επίπεδο. Καταλαβαίνουμε, επομένως, ότι το κύκλωμα που κατασκευάσαμε μπορεί να αποτελέσει ένα δομικό στοιχείο με τρεις εισόδους και μία έξοδο, το οποίο θα αξιοποιηθεί σε ένα ανώτερο galaxy.

Το Ptolemy μας δίνει τη δυνατότητα να αναπαραστήσουμε το δομικό αυτό στοιχείο με ένα νέο σύμβολο αναπαράστασης, τη μορφή του οποίου, μάλιστα, μπορούμε να αλλάξουμε κατά βούληση. Η ιδέα αυτή αναπαριστάται στο Σχήμα 2. όπου έχουμε περάσει σε ένα ανώτερο ιεραρχικό επίπεδο και έχουμε τροφοδοτήσει το νέο δομικό στοιχείο, που μόλις κατασκευάσαμε (galaxy Quadratic Equation), με τις απαραίτητες τιμές *a*,*b*,*c*. Όσον αφορά τις δύο πρώτες σταθερές, χρησιμοποιούμε το Const star (οικογένεια Signal Sources). Το δομικό αυτό στοιχείο παράγει μία σταθερή τιμή, η οποία καθορίζεται από το χρήστη με τρόπο γραφικό. Στο συγκεκριμένο παράδειγμα εφαρμόσαμε τις τιμές *a*=-1 και *b*=-3.



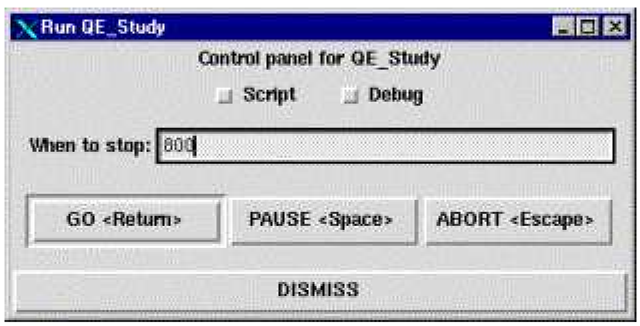
Σχήμα 2 Μελέτη Δευτεροβάθμιας Εξίσωσης

Θα μπορούσαμε να εφαρμόσουμε την ίδια προσέγγιση και για τη σταθερά *c*, αν θέλαμε μία απλή γραφική αναπαράσταση της συνάρτησης. Μπορούμε, όμως, να μελετήσουμε περαιτέρω τη συγκεκριμένη εξίσωση, χρησιμοποιώντας μεταβαλλόμενες τιμές του *c*, ώστε να διαπιστώσουμε την επίδραση της σταθεράς, με τρόπο γραφικό. Έτσι λοιπόν, εισάγουμε μέσω ενός αρχείου τις επιθυμητές τιμές. Τα περιεχόμενα του αρχείου θα εξηγηθούν στη συνέχεια.

Τέλος, η έξοδος του galaxy Quadratic Equation οδηγείται σε ένα ειδικό star, με όνομα Waterfall. Το star αυτό ανήκει στην οικογένεια Signal Sinks και προσφέρει τη γραφική αναπαράσταση των δεδομένων εισόδου του. Το ιδιαίτερο χαρακτηριστικό του συγκεκριμένου star είναι η αναδίπλωση του γραφήματος ανά συγκεκριμένο πλήθος δεδομένων εισόδου, καθώς και η εμφάνιση του άξονα *x*. Αυτό που επιδιώκουμε, ουσιαστικά, είναι να "τρέξουμε" το κύκλωμα που φαίνεται στο Σχήμα Error!

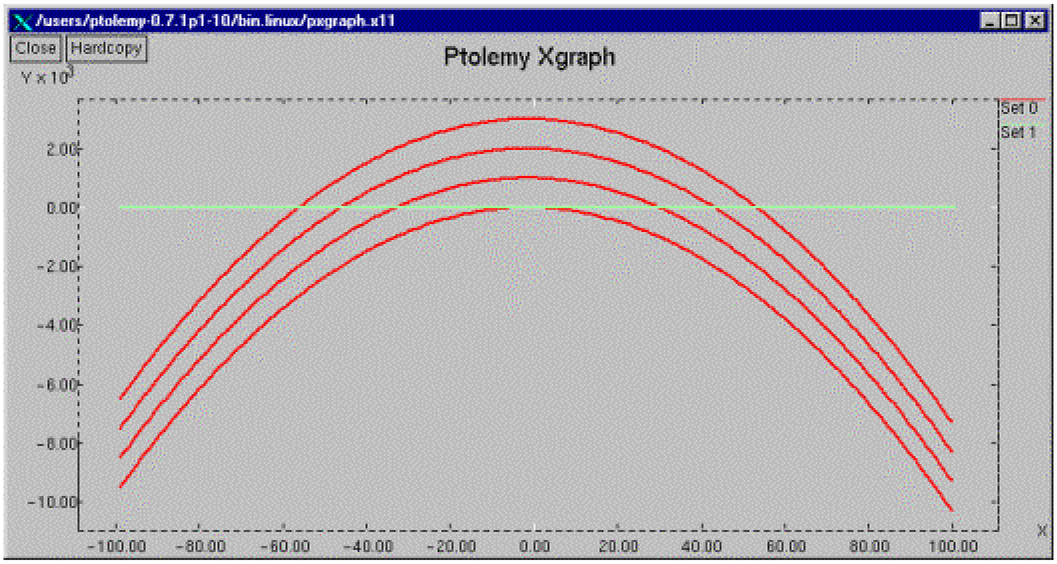
Unknown switch argument., διατηρώντας σταθερά τα *a*,*b*,*c* για τις 200 αποθηκευμένες τιμές του *x*. Στη συνέχεια, να αλλάξουμε την τιμή του *c* και να επαναλάβουμε το ίδιο για τις 200 τιμές του x. Η νέα γραφική παράσταση, που θα προκύψει, δεν πρέπει να έπεται της πρώτης, αλλά πρέπει να επικαλύπτονται, αφού αναφέρονται στο ίδιο εύρος τιμών του *x*.

Καταλαβαίνουμε, επομένως, ότι στο ReadFile star, που τροφοδοτεί το *c* (Σχήμα Error! Unknown switch argument.), πρέπει να έχει 200 ίδιες διαδοχικές τιμές και στη συνέχεια άλλες 200, που θα αντιπροσωπεύουν το νέο *c*. Το αρχείο που ορίζει τις τιμές του *x*, δεν χρειάζεται να το επαναλαμβάνουμε, αφού στη φάση του τρεξίματος τα δεδομένα ξαναδιαβάζονται από την αρχή, σε περίπτωση που αυτά είναι λιγότερα από τα πραγματικά βήματα εκτέλεσης.



Σχήμα 3 Παράθυρο Εκτέλεσης Κυκλώματος

Μετά από αυτά, είμαστε έτοιμοι να ζητήσουμε την προσομοίωση του universe, που έχουμε κατασκευάσει. Μέσω του ειδικού παραθύρου, που φαίνεται στο Σχήμα Error! Unknown switch argument. ζητούμε την εκτέλεση 800 βημάτων, επιθυμώντας να δούμε την επίδραση της σταθεράς *c* για 4 διαφορετικές τιμές της. Το συνολικό κύκλωμα θα παράγει ένα σύνολο δεδομένων εξόδου, το οποίο, μέσω του Waterfall star, θα εμφανίσει τη γραφική παράσταση, που απεικονίζεται στο Σχήμα 4..

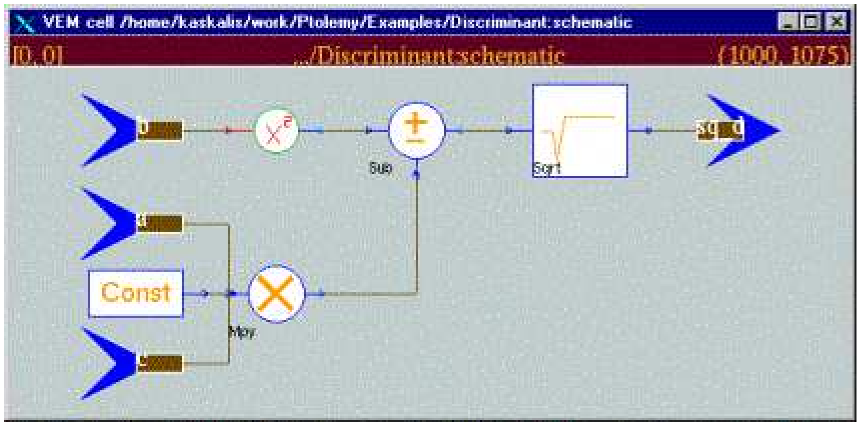


Σχήμα 4 Μελέτη της Επίδρασης του Σταθερού Όρου στη Δευτεροβάθμια Εξίσωση

4. Επίλυση Δευτεροβάθμιας Εξίσωσης

Η αναλυτική παρουσίαση της προηγούμενης ενότητας μας οδήγησε στην "γραφική" κατασκευή μιας συνάρτησης. Αυξάνοντας τα επίπεδα ιεραρχίας και πολυπλοκότητας, θα προχωρήσουμε στην παρουσίαση της επίλυσης μιας δευτεροβάθμιας εξίσωσης. Το παράδειγμα αυτό αποτελεί κλασική περίπτωση άσκησης, κατά τη φάση εκμάθησης μιας γλώσσας προγραμματισμού. Αξίζει να παρατηρήσουμε τη μαθησιακή δυσκολία που συχνά παρουσιάζεται, όταν εισέρχονται οι έννοιες των *μεταβλητών* σε μια παραδοσιακή γλώσσα προγραμματισμού ([8], [9], [10]). Πριν ο μαθητευόμενος γνωρίσει άλλες απλούστερες έννοιες, καλείται να κατανοήσει το δύσκολο ζήτημα των τύπων μεταβλητών, ώστε να γράψει ακόμη και το πρώτο του πρόγραμμα.

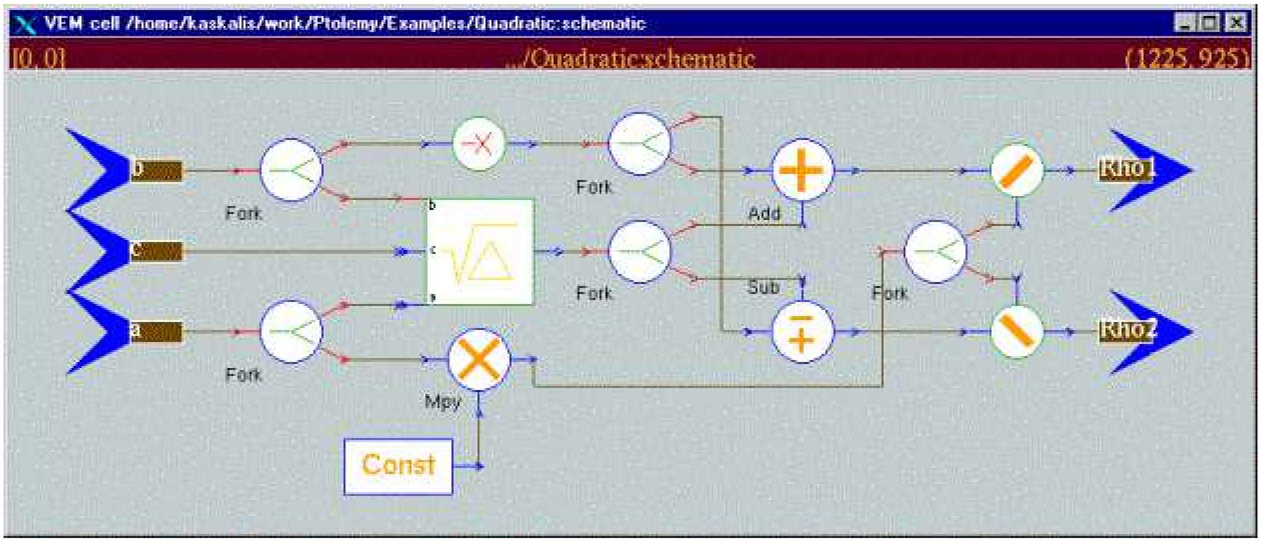
Το παράδειγμα της επίλυσης μιας δευτεροβάθμιας εξίσωσης, αν και απλό κατασκευαστικά, προϋποθέτει την κατανόηση συγκεκριμένων τύπων μεταβλητών. Στο Ptolemy η ανάγκη αυτή δεν αποτελεί επιτακτικό παράγοντα, με αποτέλεσμα η διάκριση και η μετατροπή μεταβλητών από έναν τύπο σε άλλο να γίνεται με τρόπο διαφανή, όσο αυτό δεν παραβιάζει προφανείς κανόνες. Το πρώτο βήμα του προβλήματος αναφέρεται στον υπολογισμό της *διακρίνουσας*, που θα μας οδηγήσει στην κατασκευή των επιμέρους ριζών. Συγκεκριμένα, επιλέξαμε να κατασκευάσουμε την τετραγωνική ρίζα της διακρίνουσας σαν ένα αυτόνομο δομικό στοιχείο, ώστε να απλοποιηθεί η φάση του τελικού σχεδιασμού και να αναδειχθεί ο δομημένος τρόπος σκέψης, που πρέπει να ακολουθείται. Ο υπολογισμός της συνάρτησης: $\sqrt{b^2 - 4ac}$ παρουσιάζεται στο Σχήμα 5..



Σχήμα 5 Υλοποίηση Διακρίνουσας

Το γινόμενο 4*ac* προκύπτει από ένα απλό star πολλαπλασιασμού, που εκτός των δύο σταθερών *a* και *c*, δέχεται είσοδο από ένα Const star, που ορίζεται να έχει την τιμή 4. Η σταθερά *b* οδηγείται μέσα από ένα star, που έχουμε κατασκευάσει εμείς, και αναλαμβάνει τον τετραγωνισμό των δεδομένων εισόδου του. Η δομή αυτού του star είναι πολύ απλή και για το λόγο αυτό παραλείπεται. Αξίζει όμως να παρατηρήσουμε τη δυνατότητα του Ptolemy να επιτρέπει την "οπτική" διαμόρφωση ενός δομικού στοιχείου, όπως αυτό θα εμφανίζεται σε οποιοδήποτε ανώτερο ιεραρχικό επίπεδο. Η διαμόρφωση αυτή μας δίνει τη δυνατότητα να μορφοποιούμε έτσι τα δικά μας stars, ώστε να προσδίδουν οπτική πληροφορία για τις συναρτήσεις που υλοποιούν. Η παρόμοια κατακόρυφη τοποθέτηση και πάλι υπονοεί τη δυνατότητα παραλληλισμού των δύο ανεξάρτητων πράξεων, πριν οδηγηθούν τα αποτελέσματα στα stars Sub (αφαίρεση) και Sqrt (τετραγωνική ρίζα) thw οικογένειας Arithmetic. Η σειριακή τοποθέτηση των τελευταίων ολοκληρώνει την ορθή άποψη που πρέπει να έχει ο μαθητευόμενος σχετικά με τις δυνατότητες ταυτόχρονων υπολογισμών και χρονικών περιορισμών.

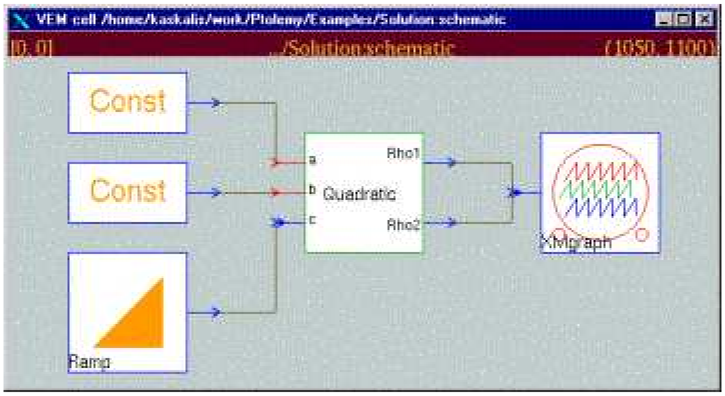
Το επόμενο βήμα είναι ο πραγματικός υπολογισμός των ριζών $\rho_{1,2} = \frac{-b \pm \sqrt{\Delta}}{2a}$, ο οποίος παρουσιάζεται στο Σχήμα 6.. Κεντρικό ρόλο στο συγκεκριμένο κύκλωμα κατέχει το δομικό στοιχείο, που κατασκευάσαμε προηγουμένως και το οποίο λαμβάνει σαν είσοδο τις σταθερές *a*,*b*,*c*. Από αυτές, οι *a* και *b* συμμετέχουν και σε άλλες πράξεις της συνάρτησης, γι' αυτό και διακλαδίζονται μέσω του Fork star.



Σχήμα 6 Υπολογισμός Ριζών Δευτεροβάθμιας Εξίσωσης

Η σταθερά *b* τροφοδοτεί ένα star, που αναλαμβάνει την αντιστροφή του προσήμου και έχει κατασκευαστεί από εμάς, στο ίδιο πνεύμα με αυτό του star τετραγωνισμού, που παρουσιάσαμε προηγουμένως (Σχήμα 5.). Επίσης κατασκευασμένο είναι και το star της διαίρεσης, που φαίνεται στη δεξιά πλευρά του κυκλώματος. Τα δομικά αυτά στοιχεία δεν προσφέρονται εξ ορισμού από τις οικογένειες του Ptolemy, λόγω της απλής κατασκευής τους μέσω άλλων stars. Επιλέξαμε όμως τη δημιουργία τους για λόγους μεγαλύτερης οπτικής σαφήνειας. Το αποτέλεσμα του star αλλαγής προσήμου προστίθεται και αφαιρείται με το αποτέλεσμα της τετραγωνικής ρίζας της διακρίνουσας, μετά τη μεσολάβηση των απαραίτητων fork stars. Η λογική διαδοχή των πράξεων οδηγεί στις δύο τελικές διαιρέσεις με τον παράγοντα 2*a*, που έχει προκύψει από τον πολλαπλασιασμό της σταθεράς με το περιεχόμενο του Const star.

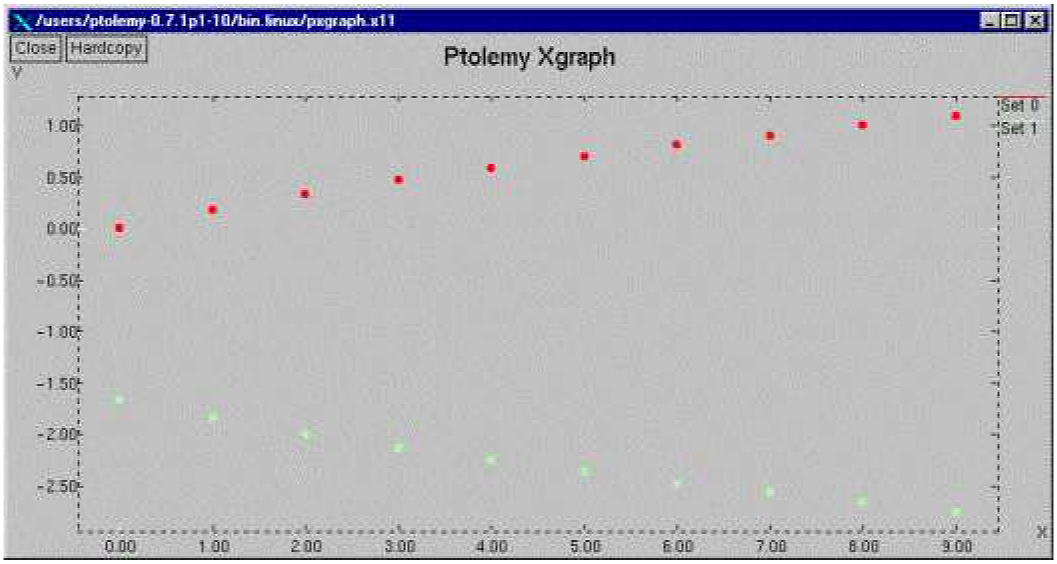
Το επόμενο και τελευταίο ιεραρχικό επίπεδο είναι αυτό του universe, που θα μας επιτρέψει τον έλεγχο της ορθότητας του συνολικού σχεδίου μας και τον ενδεχόμενο πειραματισμό για διαφορετικά δεδομένα εισόδου. Για λόγους ομοιομορφίας, επιλέξαμε και πάλι τη μελέτη της επίδρασης του σταθερού όρου *c* στις δύο ρίζες της δευτεροβάθμιας εξίσωσης. Το επιθυμητό κύκλωμα παρουσιάζεται στο Σχήμα 7..



Σχήμα 7 Μελέτη Επίλυσης Δευτεροβάθμιας Εξίσωσης

Διατηρώντας σταθερές τις τιμές των *a* και *b* (-3 και -5 αντίστοιχα), χρησιμοποιούμε το Ramp star (οικογένεια Signal Sources) για να δημιουργήσουμε μία σειρά από διαδοχικές τιμές για το *c*. Το συγκεκριμένο star είναι παραμετροποιήσιμο και ορίσαμε σαν αρχική τιμή το 0 και τιμή βήματος το 1. Αυτό σημαίνει ότι θα δημιουργηθεί μία σειρά από αύξοντες φυσικούς αριθμούς, που θα αποτελέσουν το σύνολο δεδομένων εισόδου για την περαιτέρω μελέτη της συνάρτησης, που μας ενδιαφέρει. Το Xmgraph star (οικογένεια Signal Sinks), που εμφανίζεται στο Σχήμα 7., επιτρέπει την ταυτόχρονη γραφική παράσταση περισσότερων της μιας μεταβλητών, όπως είναι η περίπτωσή μας για τις δύο ρίζες της δευτεροβάθμιας εξίσωσης.

Στη συνέχεια, ζητούμε την "εκτέλεση" του κυκλώματός μας, μέσω ενός παραθύρου παρόμοιου με αυτό που φαίνεται στο Σχήμα 3.. Εξομοιώνοντας 10 χρονικά βήματα, προκύπτει η γραφική παράσταση που παρουσιάζεται στο Σχήμα 8.. Αξίζει να σημειώσουμε τους διαφορετικούς τρόπους αναπαράστασης, που μπορούν να προκύψουν, παραμετροποιώντας τα διάφορα stars γραφικών αναπαραστάσεων του Ptolemy.

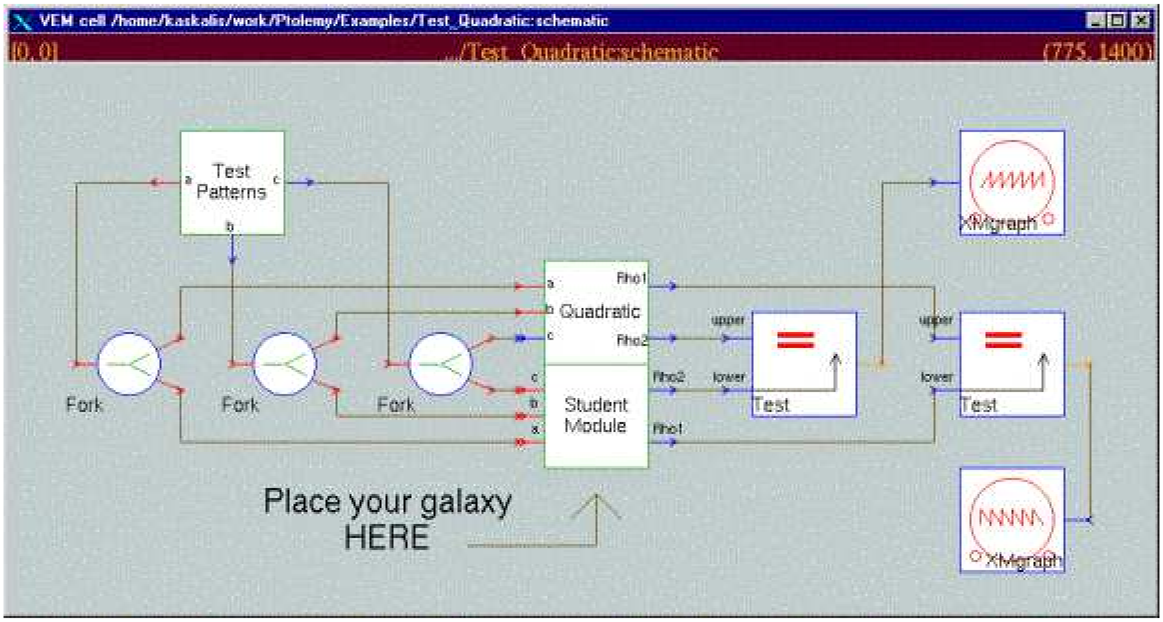


Σχήμα 8 Επίδραση του Σταθερού Όρου στις Ρίζες Δευτεροβάθμιας Εξίσωσης

5. Δυνατότητες Εξ Αποστάσεως Υποβολής και Ελέγχου

Μέχρι στιγμής έχουν παρουσιαστεί εκείνα τα χαρακτηριστικά του περιβάλλοντος Ptolemy, που το κάνουν ιδιαίτερα ελκυστικό στην εκμάθηση προγραμματισμού με οπτικό τρόπο, σε συνάρτηση με το υψηλό επίπεδο αφαίρεσης, που προσφέρει. Όπως θα δούμε όμως και στη συνέχεια, μπορούμε να αξιοποιήσουμε τα ιδιαίτερα χαρακτηριστικά του και στην εφαρμογή ενός μοντέλου εξ αποστάσεως εκπαίδευσης.

Συγκεκριμένα, ο εκπαιδευόμενος καλείται να κατασκευάσει ένα κύκλωμα, που θα υλοποιεί ένα δοθέν πρόγραμμα ή συνάρτηση. Ακολουθώντας τις απαραίτητες οδηγίες, μπορεί να εξομοιώνει τη λειτουργία του κυκλώματός του βήμα προς βήμα, χτίζοντας ιεραρχικά την ενδεχόμενη υλοποίησή του. Ταυτοχρόνως, ο εκπαιδευτής έχει φροντίσει να κατασκευάσει ένα αυτοματοποιημένο περιβάλλον ελέγχου, στο οποίο μπορεί να "τοποθετείται" το τελικό (ή οποιοδήποτε επιμέρους) κύκλωμα, με σκοπό την τελική (ή ενδιάμεση) αξιολόγησή του. Μία πρότυπη κατασκευή ενός τέτοιου περιβάλλοντος ελέγχου παρουσιάζεται στο Σχήμα 9..

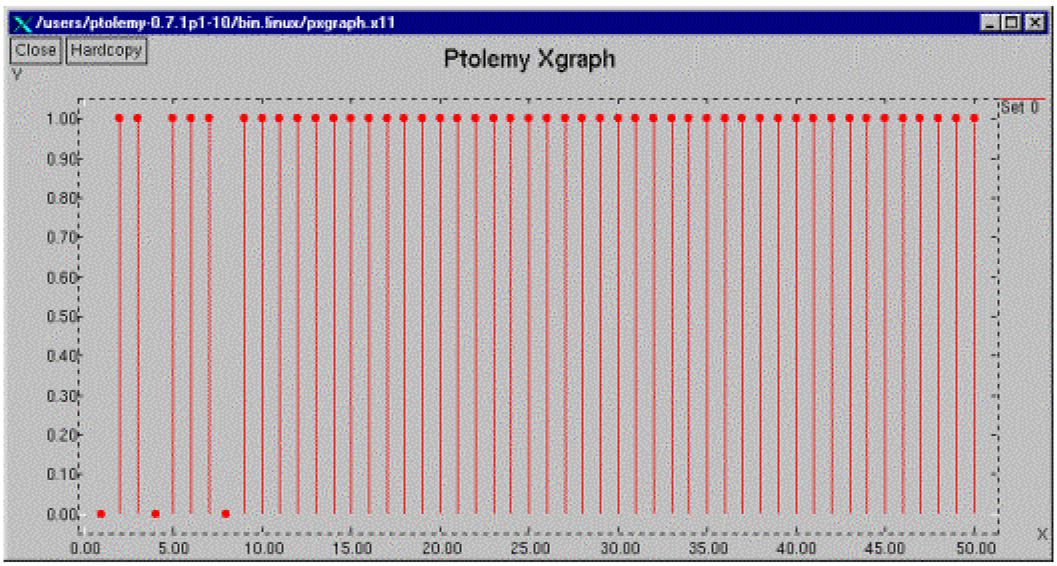


Σχήμα 9 Περιβάλλον Εξ Αποστάσεως Ελέγχου Παραδείγματος Δευτεροβάθμιας Εξίσωσης

Ο εκπαιδευτής έχει φροντίσει να δημιουργήσει ένα σύνολο δεδομένων εισόδου (galaxy Test Patterns), που τροφοδοτεί διακλαδιζόμενο (Fork stars) δύο κυκλώματα, που θεωρούμε ότι υλοποιούν την υπόψη συνάρτηση. Το επάνω κύκλωμα αποτελεί την ορθή λύση και το κάτω προστίθεται κατά περίπτωση από τον εκπαιδευόμενο, μέσω γραφικών τρόπων. Η ορθή λύση, βεβαίως, είναι έτσι παραμετροποιημένη, ώστε να μην αποκαλύπτεται η δομή της, ενώ το σύνολο των δεδομένων εισόδου μπορεί επίσης να παραμένει κρυφό.

Όπως φαίνεται και στο Σχήμα 9., τα αποτελέσματα των δύο galaxies εισάγονται σε μία δομή ελέγχου (Test star της οικογένειας Logic). Ο έλεγχος αυτός στη συνέχεια παρουσιάζεται γραφικά μέσω των Xmgraph stars, που ολοκληρώνουν το σχήμα ελέγχου. Ο εκπαιδευόμενος μπορεί είτε απ' ευθείας στο τέλος, είτε κατά τη φάση ανάπτυξης, να τοποθετεί το κύκλωμα που έχει κατασκευάσει, με σκοπό τη σύγκριση με ένα "πρότυπο", κατασκευασμένο από τον εκπαιδευτή. Το αποτέλεσμα της πράξης αυτής, του επιτρέπει την άμεση διαπίστωση της ορθότητας της προτάσεώς του, απλοποιώντας ταυτόχρονα τη διαδικασία της τελικής βαθμολόγησης.

Η εκτέλεση του κυκλώματος, που φαίνεται στο Σχήμα 9., για 50 διαφορετικά δεδομένα εισόδου και για ένα απλό galaxy εκπαιδευομένου, που κατασκευάστηκε για αυτόν το σκοπό (galaxy Student Module), παρουσιάζεται στο Σχήμα 10.. Από τις δύο αναμενόμενες γραφικές παραστάσεις επιλέξαμε να παρουσιάσουμε τη μία, εισάγοντας ορισμένα σφάλματα στο υποτιθέμενο κύκλωμα του εκπαιδευόμενου. Η συγκεκριμένη γραφική παράσταση δείχνει την ορθότητα ή μη (τιμή 1 ή 0) για κάθε ένα από τα 50 βήματα εισόδου. Είναι φανερό ότι για τη συντριπτική πλειοψηφία των τιμών τα αποτελέσματα των δύο κυκλωμάτων συμπίπτουν. Όμως για τις τιμές 1, 4 και 8 δεν υπάρχει απόλυτη συμφωνία, υπονοώντας την ύπαρξη λαθών στο υπόψη κύκλωμα.



Σχήμα 10 Αποτέλεσμα Εξ Αποστάσεως Ελέγχου για τη Διαπίστωση της Ορθότητας

Ασφαλώς, τα περιθώρια τροποποίησης και βελτίωσης της πρότασης του παραδείγματος είναι μεγάλα. Το κατά περίπτωση αυτοματοποιημένο περιβάλλον ελέγχου μπορεί να είναι περισσότερο ή λιγότερο καθοδηγητικό, ανάλογα με τον τρόπο ελέγχου και διόρθωσης, που θέλει να εισάγει ο εκπαιδευτής. Το παράδειγμα αυτό, όμως, αναπαριστά τις μεγάλες δυνατότητες που μπορεί να προσφέρει το περιβάλλον Ptolemy και στον τομέα του εξ αποστάσεως, αυτοματοποιημένου ελέγχου.

6. Συμπεράσματα και Μελλοντική Έρευνα

Μέσα από ένα σύνολο παραδειγμάτων παρουσιάσαμε μία νέα πρόταση προσέγγισης της διδασκαλίας του προγραμματισμού, μέσω του περιβάλλοντος Ptolemy. Τα συμπεράσματα, στα οποία μπορούμε να καταλήξουμε, όσον αφορά τη χρησιμότητα της πρότασης αυτής είναι:

· Η αναπαράσταση της αρχικής ιδέας ενός αλγορίθμου διατηρεί έναν υψηλό βαθμό ευκρίνειας, λόγω της οπτικής κατασκευής των επιμέρους διαδικασιών.

· Αίρεται η ανάγκη της αποστήθισης αυστηρών προγραμματιστικών εντολών, οι οποίες εκτός της δυσκολίας τους, μπορεί σύντομα να αποδειχθούν μη χρήσιμες, λόγω της διαφοροποίησης και εξέλιξης των γλωσσών προγραμματισμού.

· Διατηρείται η ιεραρχική δόμηση των επιμέρους συναρτήσεων και υπορουτινών, μέσω της κατασκευής αυτόνομων δομικών στοιχείων, που μπορούν να χρησιμοποιηθούν χωρίς δυσκολία και σε άλλα αλγοριθμικά σχέδια.

· Η δύσκολη έννοια των τύπων μεταβλητών, μπορεί να παραμείνει διαφανής, χωρίς να καταλήγει απαγορευτική η κατασκευή "προχωρημένων" αλγορίθμων.

· Η διαδικασία της εκ των υστέρων εκσφαλμάτωσης είναι δυνατό να γίνει τόσο στα επιμέρους, όσο και στο συνολικό κύκλωμα.

· Η έννοια του παραλληλισμού των πράξεων εισάγεται αβίαστα, εξυπηρετώντας ταυτόχρονα και άλλους μαθησιακούς σκοπούς του ίδιου γνωστικού αντικειμένου.

· Τέλος, ο εξ αποστάσεως έλεγχος της ορθότητας ενός κυκλώματος, επιτρέπει την ανάπτυξη ενός απομακρυσμένου μοντέλου εκπαίδευσης, εξυπηρετώντας ταυτόχρονα εκπαιδευτές και εκπαιδευόμενους.

Όλες οι παραπάνω διαπιστώσεις, επιβεβαιώνουν τη χρησιμότητα του περιβάλλοντος Ptolemy σε σκοπούς εκπαιδευτικής φύσεως. Ο τομέας του προγραμματισμού, στον οποίο εστιάσαμε κυρίως την προσοχή μας, αποτελεί βεβαίως μία σημαντική θεματική ενότητα. Είναι όμως εύκολο να διαπιστώσουμε ότι το Ptolemy, μπορεί να αξιοποιηθεί και σε αρκετά ακόμη γνωστικά πεδία.

Γενικά εκπαιδευτικά θέματα μαθηματικών, φυσικής, μηχανικής (engineering) και επεξεργασίας σήματος μπορούν να αντιμετωπιστούν με παρόμοιο τρόπο. Η μελλοντική μας έρευνα περιλαμβάνει την ανάπτυξη μεθόδων και παραδειγμάτων, που θα αποκαλύψουν την περαιτέρω χρησιμότητα του συγκεκριμένου εργαλείου στα γνωστικά αυτά πεδία. Ο απώτερος στόχος είναι μία ενοποιημένη πρόταση ανάπτυξης εκπαιδευτικών μεθόδων και η διαπίστωση των πλεονεκτημάτων της προσέγγισης αυτής.

Αναφορές

[1] *The Almagest - Ptolemy 0.7 User's Manual*, College of Engineering, Dept. of Electrical Engineering and Computer Sciences, Berkeley, 1997.

[2] P. Bayman, R.E. Mayer, "A Diagnosis of Beginning Programmers' Misconceptions of BASIC Programming Statements", *Communications of the ACM*, vol. 26, no. 9, pp. 677-679, 1983.

[3] B. Du Boulay, "Some Difficulties Of Learning To Program", *Studying The Novice Programmer*, E. Soloway, J. Sprohrer (Eds.), Lawrence Erlbaum Associates, pp. 283-300, 1989.

[4] B. Du Boulay, T. O'Shea, J. Monk, "The Black Box Inside the Glass Box: Presenting Computing Concepts to Novices", *Studying The Novice Programmer*, E. Soloway, J. Sprohrer (Eds.), Lawrence Erlbaum Associates, pp. 431-446, 1989.

[5] Β. Δαγδιλέλης, *Διδακτική της πληροφορικής. Η διδασκαλία του προγραμματισμού: αντιλήψεις των σπουδαστών για την κατασκευή κι επικύρωση προγραμμάτων και διδακτικές καταστάσεις για τη διαμόρφωσή τους*, Διδακτορική Διατριβή, Βιβλιοθήκη Τμ. Εφ. Πληροφορικής, Πανεπιστήμιο Μακεδονίας, 1996.

[6] P. Mendelsohn, T.R.G. Green, P. Brna, "Programming Languages in Education: The Search for an Easy Start", *Psychology of Programming*, J. Hoc, T. Green, R. Samurcay, D. Gilmore (Eds.), Academic Press, 175-200, 1990.

[7] J. Rogalski, "Alphabétisation Informatique", *Bulletin de l'APMEP*, no. 352, pp. 61-74, 1985.

[8] R. Samurçay, "Signification et fonctionnement du concept de variable informatique chew les éléves débutants", *Educational Studies in Mathematics*, no 16, 1985.

[9] D.C. Smith, A. Cypher, "KidSim: Child Constructible Simulations", *Proceedings of IMAGINA '95*, Monte-Carlo, pp. 87-99, 1995.

[10] E. Soloway, K. Ehrlich, "Empirical Studies of Programming Knowledge", *IEEE TSE*, vol. SE-10, no. 5, pp. 595-609, 1984.