ePublishing
ΕΘΝΙΚΟ ΚΕΝΤΡΟ ΤΕΚΜΗΡΙΩΣΗΣ
NATIONAL DOCUMENTATION CENTRE

Συνέδρια της Ελληνικής Επιστημονικής Ένωσης Τεχνολογιών Πληροφορίας & Επικοινωνιών στην Εκπαίδευση

Τόμ. 1 (2008)

4ο Συνέδριο Διδακτική Πληροφορικής

Σχεδιασμός ενός Περιβάλλοντος Δυναμικής Οπτικοποίησης Αλγορίθμων: Το σύστημα DAVE*

Ε. Βραχνός, Α. Τζιμογιάννης

Βιβλιογραφική αναφορά:

Βραχνός Ε., & Τζιμογιάννης Α. (2023). Σχεδιασμός ενός Περιβάλλοντος Δυναμικής Οπτικοποίησης Αλγορίθμων: Το σύστημα DAVE* . *Συνέδρια της Ελληνικής Επιστημονικής Ένωσης Τεχνολογιών Πληροφορίας & Επικοινωνιών στην Εκπαίδευση*, *1*, 171–180. ανακτήθηκε από https://eproceedings.epublishing.ekt.gr/index.php/cetpe/article/view/5862

https://epublishing.ekt.gr | **e-Εκδότης:** EKT | **Πρόσβαση:** 26/03/2026 20:30:19

Σχεδιασμός ενός Περιβάλλοντος Δυναμικής Οπτικοποίησης Αλγορίθμων: Το σύστημα DAVE*

Ε. Βραχνός, Α. Τζιμογιάννης

Τμήμα Κοινωνικής και Εκπαιδευτικής Πολιτικής, Πανεπιστήμιο Πελοποννήσου
evrachnos@gmail.com, ajimoyia@uop.gr

Περίληψη

Η κατανόηση και η χρήση των αλγοριθμικών δομών από αρχάριους στον προγραμματισμό αποτελεί μια ιδιαίτερα δύσκολη και πολύπλοκη διαδικασία. Τα τελευταία χρόνια έχουν αναπτυχθεί εκπαιδευτικά περιβάλλοντα οπτικοποίησης αλγορίθμων, τα οποία έχουν ως στόχο την ανάδειξη των βασικών αλγοριθμικών χαρακτηριστικών και την υποστήριξη των μαθητών/φοιτητών ώστε να οικοδομήσουν επαρκείς αναπαραστάσεις για τις αλγοριθμικές και τις προγραμματιστικές δομές. Στην εργασία αυτή γίνεται μια επισκόπηση των σημαντικότερων συστημάτων οπτικοποίησης που αναφέρονται στη διεθνή βιβλιογραφία. Στη συνέχεια παρουσιάζεται ο σχεδιασμός του συστήματος DAVE, το οποίο αποτελεί ένα περιβάλλον δυναμικής οπτικοποίησης αλγορίθμων που στοχεύει στην υποστήριξη της μάθησης στα εισαγωγικά μαθήματα αλγοριθμικής και προγραμματισμού στο Ενιαίο Λύκειο.
Λέξεις κλειδιά: *Δυναμική οπτικοποίηση αλγορίθμων, αλγοριθμική, εκπαιδευτικό περιβάλλον*

Abstract

Understanding and using algorithmic structures is a particularly difficult and complex process for novice programmers. During the last years various algorithm visualization systems have been developed which aim at revealing the dynamic nature of algorithms while supporting students to construct effective representations. This article presents an overview of the most important and influential algorithm visualization systems reported in the literature. Following, it presents the design of an open, fully interactive and dynamic algorithm visualization system, named DAVE, which aims to support high school students in introductory courses about algorithms and programming.
Keywords: *Dynamic algorithm visualization, algorithms, educational environment*

1. Εισαγωγή

Η διδασκαλία του προγραμματισμού και της αλγοριθμικής επίλυσης προβλημάτων σε αρχάριους, μαθητές της δευτεροβάθμιας εκπαίδευσης ή φοιτητές, συνιστά ένα δύσκολο αλλά εξαιρετικά ενδιαφέρον έργο, τόσο από διδακτική όσο και από γνωστική άποψη. Κατά την επίλυση αλγοριθμικών προβλημάτων χρησιμοποιούνται

---

* Dynamic Algorithm Visualization Environment

έννοιες και δομές, οι οποίες είναι δύσκολο να οικοδομηθούν με τα παραδοσιακά διδακτικά μέσα (αλγόριθμος, μεταβλητή, αρχικοποίηση, δομή επιλογής, δομή επανάληψης κ.λπ.). Τα συνήθη προγραμματιστικά περιβάλλοντα και οι γλώσσες προγραμματισμού που χρησιμοποιούνται για εκπαιδευτικούς σκοπούς έχουν σχεδιαστεί για την ανάπτυξη εφαρμογών και όχι για τη διδασκαλία του προγραμματισμού. Είναι, συνεπώς, προσαρμοσμένα στο πλαίσιο γνώσεων και δεξιοτήτων των έμπειρων προγραμματιστών, γεγονός που ενισχύει τις δυσκολίες και τα εμπόδια που συναντούν οι μαθητές και οι αρχάριοι στον προγραμματισμό (Κόμης & Τζιμογιάννης, 2006).

Τα τελευταία χρόνια, αναπτύσσεται μεγάλο ερευνητικό και εκπαιδευτικό ενδιαφέρον για τη χρήση ειδικών περιβαλλόντων προγραμματισμού, όπως περιβάλλοντα Logo (γλώσσες Logo, MicroWolds Pro), εκπαιδευτική ρομποτική (π.χ. LEGO/LOGO, JKarelRobot), μικρογλώσσες, προγραμματιστικοί μικρόκοσμοι (π.χ. Karel, Karel++, KarelJ, JEROO), περιβάλλοντα προσομοίωσης και οπτικοποίησης αλγορίθμων, ολοκληρωμένα εκπαιδευτικά προγραμματιστικά περιβάλλοντα κ.λπ. Τα περιβάλλοντα αυτά παρέχουν νέες δυνατότητες για την οικοδόμηση γνώσεων και την ανάπτυξη δεξιοτήτων στον προγραμματισμό.

Παράλληλα διαπιστώνεται αυξημένο ενδιαφέρον για το σχεδιασμό μαθησιακών δραστηριοτήτων που βασίζονται στις αρχές του εποικοδομισμού (Ramadhan, 2000; Kolikant & Pollack, 2004; Γρηγοριάδου κ.α., 2004; Κόμης, 2005; Τσέλιος κ.α., 2006). Κύριοι άξονες των εποικοδομιστικών διδακτικών προσεγγίσεων είναι η εκτίμηση των προϋπαρχουσών γνώσεων και αντιλήψεων των μαθητών και η οργάνωση διδακτικών-μαθησιακών δραστηριοτήτων που να ευνοούν τη *διερευνητική*, *ανακαλυπτική* και *συνεργατική* μάθηση. Οι προσεγγίσεις αυτές δίνουν έμφαση στον παιδαγωγικό σχεδιασμό της διδασκαλίας του προγραμματισμού και στη μετατόπιση από το συντακτικό στην καλλιέργεια δεξιοτήτων επίλυσης προβλημάτων (αναλυτική-συνθετική σκέψη, αφαιρετική ικανότητα, μοντελοποίηση λύσεων). Στο πλαίσιο αυτό, έχει προταθεί ο σχεδιασμός κατάλληλων μαθησιακών δραστηριοτήτων με χρήση εκπαιδευτικών περιβαλλόντων προσομοίωσης-οπτικοποίησης αλγορίθμων και προγραμμάτων (Stasko, 1997; Hundhausen et.al., 2002; Sajaniemi, 2005; Τζιμογιάννης κ.α., 2006).

Η εργασία αυτή αποτελεί μια πρόταση εφαρμογής λογισμικού δυναμικής οπτικοποίησης αλγορίθμων στη διδασκαλία των βασικών αρχών της αλγοριθμικής, στο πλαίσιο των μαθημάτων προγραμματισμού του Ενιαίου Λυκείου. Γίνεται μια επισκόπηση των σημαντικότερων συστημάτων οπτικοποίησης αλγορίθμων και προγραμμάτων από τη διεθνή βιβλιογραφία. Στη συνέχεια παρουσιάζεται η σχεδίαση του λογισμικού DAVE, το οποίο υποστηρίζει διαδικασίες δυναμικής οπτικοποίησης αλγορίθμων και έχει ως στόχο την υποστήριξη των μαθητών ώστε να οικοδομήσουν αποτελεσματικές αναπαραστάσεις για βασικές προγραμματιστικές έννοιες, δομές και αλγορίθμους.

2. Συστήματα Οπτικοποίησης Αλγορίθμων: Μια Επισκόπηση

Οι εφαρμογές οπτικοποίησης αλγορίθμων έχουν μακρά ιστορία στην εκπαίδευση. Το βίντεο *Sorting Out Sorting* (Baecker, 1981), το οποίο παρουσίαζε εικόνες δεδομένων που ταξινομούνταν από διαφορετικούς αλγορίθμους θεωρείται η πρώτη σημαντική αναφορά στην οπτικοποίηση αλγορίθμων. Από τότε έχουν αναπτυχθεί πολλά συστήματα προσομοίωσης και οπτικοποίησης αλγορίθμων που βασίζονταν στις διαθέσιμες κάθε φορά τεχνολογίες. Με τον όρο ***οπτικοποίηση (visualization)*** αλγορίθμου περιγράφεται μια διαδραστική οπτικοποιημένη παρουσίαση της λογικής που κρύβεται πίσω από τον αλγόριθμο, βασισμένη σε μια σειρά εικόνων και αναπαραστάσεων που εστιάζουν στα βασικά χαρακτηριστικά της συμπεριφοράς του (Hundhausen & Brown, 2007). Τα συνήθη λογισμικά προσομοίωσης-οπτικοποίησης δεν αντικαθιστούν το προγραμματιστικό περιβάλλον, αφού δεν εκτελούν κάποιον αλγόριθμο, αλλά παρουσιάζουν, οπτικοποιούν και προσομοιώνουν την εκτέλεσή του για προκαθορισμένα δεδομένα εισόδου.

Υπάρχουν πολλοί λόγοι που συνηγορούν στη χρήση των προσομοιώσεων-οπτικοποιήσεων στα εισαγωγικά μαθήματα του προγραμματισμού. Οι προσομοιώσεις αλγορίθμων αποτελούν δυναμικά μαθησιακά περιβάλλοντα τα οποία (Τζιμογιάννης κ.α., 2006):

- επιτρέπουν την οπτικοποίηση της λειτουργίας ενός αλγορίθμου, υποστηρίζοντας τη διάλεξη του εκπαιδευτικού και την εργαστηριακή εξάσκηση των μαθητών
- επιτρέπουν στους μαθητές να απομονώσουν τις μεταβλητές και το ρόλο τους στο πρόγραμμα, με στόχο την κατανόηση πιο σύνθετων υπολογιστικών δομών και διαδικασιών
- βοηθούν τους μαθητές να εμβαθύνουν στη λογική του προγραμματισμού και να κατανοήσουν δύσκολες υπολογιστικές έννοιες και διαδικασίες
- ενεργοποιούν το ενδιαφέρον των μαθητών, παρέχοντας δυνατότητες να εκφράσουν τις δικές τους αναπαραστάσεις για αλγορίθμους και διαδικασίες
- διευκολύνουν την ενεργητική μάθηση μέσα από διαδικασίες παράθεσης υποθέσεων, μεταβολής των τιμών εισόδου και άμεσου ελέγχου των αποτελεσμάτων στην οθόνη
- βοηθούν τους μαθητές να μάθουν μέσα από διερευνητικές δραστηριότητες.

Φαίνεται ότι η τυπική χρήση λογισμικών προσομοίωσης-οπτικοποίησης αλγορίθμων δεν είναι περισσότερο αποτελεσματική σε σχέση με την παραδοσιακή διδασκαλία. Πρόσφατες έρευνες έχουν δείξει ότι τα λογισμικά προσομοίωσης αλγορίθμων θα πρέπει να προωθούν τη διερευνητική μάθηση και να ευνοούν την ενεργητική συμμετοχή και όχι την παθητική συμμόρφωση των μαθητών (Hundhausen et.al., 2002; Hundhausen & Brown, 2006). Ο μαθητής δεν αρκεί απλά να παρακολουθεί την οπτικοποίηση ως παθητικός θεατής αλλά θα πρέπει να πειραματιστεί με την εκτέλεση του αλγορίθμου, έτσι ώστε να διερευνήσει τις διάφορες λογικές πτυχές του.

Με βάση τον τύπο τους, διακρίνουμε τα εκπαιδευτικά περιβάλλοντα οπτικοποίησης σε δύο μεγάλες κατηγορίες: ***συστήματα οπτικοποίησης αλγορίθμων*** και ***συστήματα οπτικοποίησης προγραμμάτων***.

2.1 Συστήματα Οπτικοποίησης Αλγορίθμων

Τα συστήματα οπτικοποίησης αλγορίθμων, όπως εμφανίζονται στη βιβλιογραφία, είναι περιβάλλοντα τα οποία παράγουν μια γραφική αναπαράσταση του αλγορίθμου με σκοπό να αναδειχθούν τα ιδιαίτερα χαρακτηριστικά του, τα οποία είναι δύσκολο να οικοδομήσουν ή να προσεγγίσουν οι μαθητευόμενοι. Διακρίνονται σε:

Συστήματα με γλώσσα συγγραφής σεναρίων: Είναι η πιο δημοφιλής κατηγορία συστημάτων οπτικοποίησης αλγορίθμων. Ο χρήστης έχει τη δυνατότητα να δημιουργήσει τη δική του οπτικοποίηση του αλγορίθμου, αφού εισάγει στον κώδικά του εντολές οπτικοποίησης μέσω μιας γλώσσας σεναρίων (script language). Αυτό δεν είναι κάτι που μπορεί να γίνει εύκολα από έναν αρχάριο προγραμματιστή ή έναν μαθητή που παρακολουθεί τα εισαγωγικά μαθήματα αλγοριθμικής, καθώς θα πρέπει να μάθει μια ακόμη γλώσσα προγραμματισμού, τη γλώσσα σεναρίων του συστήματος. Τα συστήματα οπτικοποίησης-προσομοίωσης αλγορίθμων μπορούν να συμβάλλουν στην υποστήριξη των διαλέξεων του διδάσκοντα, ειδικά σε φοιτητές που έχουν παρακολουθήσει αρκετά μαθήματα προγραμματισμού. Τα πρώτα συστήματα αυτής της κατηγορίας ανήκουν στην οικογένεια των Tango, Polka, Samba και JSamba που αναπτύχθηκαν από την ομάδα του Stasko (1997). Ένα άλλο σύστημα που παρουσιάζει ομοιότητες με το JSamba είναι το JAWAA (Pierson & Rodger, 1998) που οπτικοποιεί δομές δεδομένων όπως είναι η στοίβα, η ουρά κ.α. Νεότερα συστήματα που ακολουθούν την ίδια τεχνική είναι το JHave (Naps, 2005), το Animal (Rößling 2000) και το Alvis (Hundhausen & Brown, 2007), το οποίο είναι ένα από τα καλύτερα συστήματα της κατηγορίας αυτής και χαρακτηρίζεται από τον υψηλό βαθμό αλληλεπίδρασης με τον χρήστη.

Συστήματα με βιβλιοθήκες γραφικών: Ανάλογα μειονεκτήματα έχουν επίσης και τα συστήματα που χρησιμοποιούν βιβλιοθήκες γραφικών για την οπτικοποίηση του αλγορίθμου. Ο μαθητής/φοιτητής είναι υποχρεωμένος να εισάγει τις κατάλληλες εντολές στον κώδικά του, ώστε να παρατηρήσει την οπτικοποίηση του αλγορίθμου που σχεδίασε. Χαρακτηριστικά συστήματα του τύπου αυτού είναι το Zeus (Brown, 1991) και το Algorithm Explorer (Carson et al., 2007).

2.2 Συστήματα Οπτικοποίησης Προγραμμάτων

Τα συστήματα οπτικοποίησης προγραμμάτων χρησιμοποιούν αναπαραστάσεις και διαδικασίες που είναι χρήσιμες στους προγραμματιστές κατά την εκσφαλμάτωση (debugging) του προγράμματος. Για παράδειγμα, γίνεται παρακολούθηση των τιμών των μεταβλητών σε κάθε βήμα του αλγορίθμου με στόχο την ανίχνευση και τη διόρθωση των λαθών κατά την εκτέλεση του προγράμματος.

Χαρακτηριστικός εκπρόσωπος της κατηγορίας αυτής, παρότι ενσωματώνει αρκετά στοιχεία οπτικοποίησης δομών δεδομένων, είναι το σύστημα Leonardo (Demetrescu & Finocchi, 1999) που αναπτύχθηκε στο Πανεπιστήμιο της Ρώμης. Η βασική έκδοσή του δημιουργήθηκε για υπολογιστές Macintosh και, μέχρι σήμερα, δεν υποστηρίζεται από άλλες πλατφόρμες. Η οπτικοποίηση του προγράμματος δεν γίνεται αυτόματα αλλά ο μαθητής είναι υποχρεωμένος να χρησιμοποιήσει τη γλώσσα συγγραφής σεναρίων Alpha.

Το PlanAni αποτελεί ένα περιβάλλον προσομοίωσης της εκτέλεσης προγραμμάτων, η οποία βασίζεται στην προσέγγιση της διδασκαλίας των ρόλων μεταβλητών (Sajaniemi & Kuittinen, 2003). Το λογισμικό αυτό χρησιμοποιεί διαφορετικές αναπαραστάσεις για τις διάφορες μεταβλητές, οι οποίες σχηματίζονται κατά τη φάση της δήλωσής τους και ανάλογα με το ρόλο τους στο πρόγραμμα. Ο μαθητής έχει τη δυνατότητα να παρακολουθεί την προσομοίωση της εκτέλεσης του προγράμματος ορίζοντας τα δεδομένα εισόδου. Το PlanAni έχει και ελληνική έκδοση, η οποία έχει εφαρμοστεί πιλοτικά για την κατανόηση της έννοιας της προγραμματιστικής μεταβλητής και της εντολής εκχώρησης (Τζιμογιάννης κ.α., 2006).

Τέλος, η πιο δημοφιλής εφαρμογή της κατηγορίας αυτής είναι το Jeliot (Ben-Bassat Levy et al., 2003). Το σημαντικότερο χαρακτηριστικό της είναι η αυτόματη οπτικοποίηση των προγραμμάτων σε Java, η οποία όμως επικεντρώνεται στις αντικειμενοστρεφείς δομές του προγράμματος (όπως είναι τα αντικείμενα και οι μέθοδοι) και όχι στη συμπεριφορά-λογική του αλγορίθμου.

3. Δυναμική Οπτικοποίηση Αλγορίθμων

Οι ερευνητικές μελέτες σχετικά με την αποτελεσματικότητα των συστημάτων οπτικοποίησης-προσομοίωσης αλγορίθμων έδειξαν ότι τα μαθησιακά οφέλη είναι περιορισμένα όταν ο μαθητής παρακολουθεί απλά μια οπτικοποίηση χωρίς να εμπλέκεται σε γνωστικές διαδικασίες χειρισμού και αλληλεπίδρασης με τον αλγόριθμο (Byrne et al., 1999; Hundhausen & Brown, 2006; Stasko, 1993; 1997). Ο Hundhausen (2002), μετά από την ανάλυση των αποτελεσμάτων 24 πειραματικών διδασκαλιών, κατέληξε στο συμπέρασμα ότι *''όσο πιο ενεργά συμμετέχουν οι μαθητές σε δραστηριότητες με το εκπαιδευτικό λογισμικό, τόσο καλύτερα αποδίδουν αργότερα, όταν καλούνται να υλοποιήσουν τον αλγόριθμο που διδάχθηκαν''*.

Στα περισσότερα από τα συστήματα που παρουσιάστηκαν ο μαθητής έχει περιορισμένες δυνατότητες παρέμβασης στην οπτικοποίηση του αλγορίθμου, ορίζοντας απλά τα δεδομένα εισόδου, σταματώντας την εκτέλεση ή μεταβαίνοντας σε όποιο σημείο της εκτέλεσης επιθυμεί. Κανένα από τα εργαλεία αυτά *δεν έχει τη δυνατότητα να οπτικοποιεί αυτόματα έναν αλγόριθμο που γράφεται από τον ίδιο το μαθητή σε μια δεδομένη γλώσσα προγραμματισμού*.

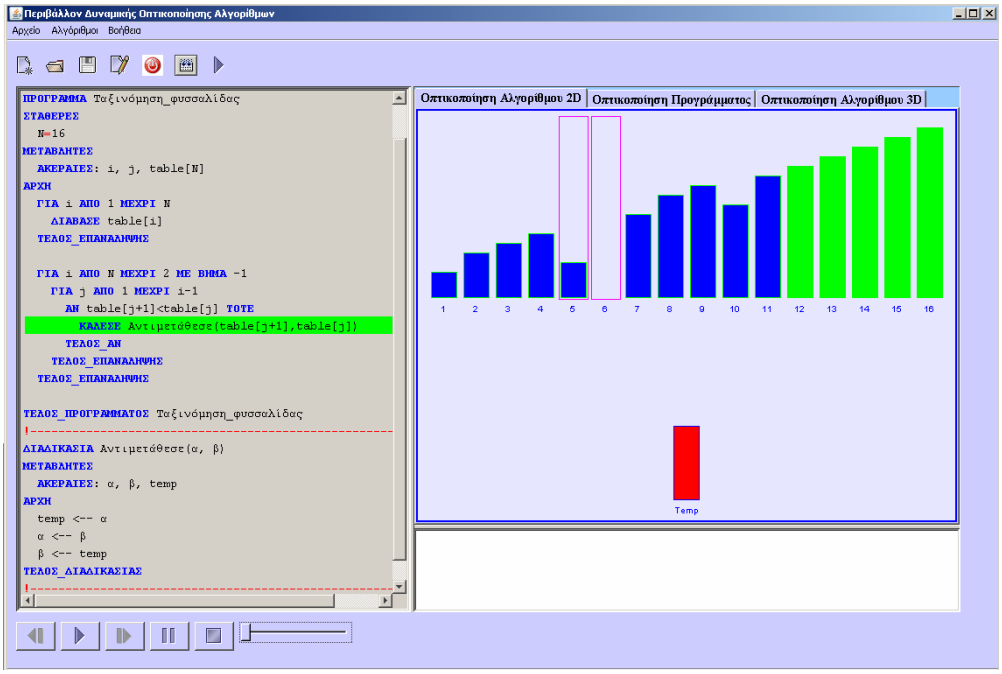
Απαιτείται, συνεπώς, ένα σύστημα με υψηλό βαθμό αλληλεπίδρασης, το οποίο να ενσωματώνει δυνατότητες αυτοματοποιημένης παραγωγής της οπτικοποίησης του κώδικα του μαθητή. Η δυνατότητα αυτή ενισχύει σημαντικά τη μαθησιακή διαδικασία, καθώς προσθέτει μια άλλη διάσταση στην αλληλεπίδραση του μαθητή με το λογισμικό, όπως για παράδειγμα συμβαίνει με τη γλώσσα προγραμματισμού Logo. Ο μαθητής βλέπει τη γραφική αναπαράσταση των βασικών χαρακτηριστικών της συμπεριφοράς του αλγορίθμου που έχει υλοποιήσει, ενώ παράλληλα έχει τη δυνατότητα να ανιχνεύσει άμεσα και να διορθώσει τα λογικά λάθη του. Επιπλέον, έχει τη δυνατότητα να λειτουργήσει διερευνητικά και να λάβει ανάδραση από το ίδιο το σύστημα ελέγχοντας την οπτικοποίηση του υπό μελέτη αλγορίθμου, ανακαλύπτοντας σημαντικά χαρακτηριστικά της δομής και της συμπεριφοράς του, μελετώντας τον τρόπο που ανταποκρίνεται ο αλγόριθμος στις αλλαγές που δοκιμάζει κ.ο.κ.. Αυτό ενισχύεται ακόμη περισσότερο όταν βλέπει στην ίδια οθόνη τον κώδικα του αλγορίθμου και την οπτικοποίησή του, έτσι ώστε να είναι εύκολη η αντιστοίχηση της εκτελούμενης εντολής με το αποτέλεσμα της οπτικοποίησης.

4. Το σύστημα DAVE

Η σχεδίαση του λογισμικού δυναμικής οπτικοποίησης αλγορίθμων DAVE έγινε με βάση τις παραπάνω απαιτήσεις. Βασικός στόχος είναι όχι απλά η εμπλοκή και ο πειραματισμός του μαθητή με προκαθορισμένους αλγορίθμους αλλά, κυρίως, η αυτόματη οπτικοποίηση αλγορίθμων που έχει σχεδιάσει ο ίδιος. Το λογισμικό DAVE παρέχει τη δυνατότητα στο μαθητή να συσχετίσει κάθε εντολή-δομή του κώδικα, που έχει αναπτύξει ο ίδιος, με κατάλληλες γραφικές αναπαραστάσεις δεδομένων έτσι ώστε να αναδεικνύονται τα ειδικά χαρακτηριστικά κάθε αλγορίθμου. Με βάση την προσέγγιση αυτή, ο μαθητής βρίσκεται σε άμεση και συνεχή επαφή με τον κώδικα του αλγορίθμου, καθώς κάθε αλλαγή στον κώδικα έχει άμεσο αντίκτυπο στην οπτικοποίηση. Για να δημιουργήσει μια νέα οπτικοποίηση αρκεί να γράψει τον κώδικα του υπό μελέτη αλγορίθμου στη γλώσσα προγραμματισμού και δεν χρειάζεται να μάθει μια νέα γλώσσα σεναρίων ή να χειριστεί βιβλιοθήκες γραφικών.

Σύμφωνα με τους Stasko et al. (1993), η υλοποίηση της οπτικοποίησης του αλγορίθμου από τον ίδιο το μαθητή είναι ο καλύτερος τρόπος να κατανοήσει τα βασικά χαρακτηριστικά του υπό μελέτη αλγορίθμου. Το πρωτότυπο χαρακτηριστικό του DAVE είναι ότι οι διάφορες αναπαραστάσεις δημιουργούνται αυτόματα από τον κώδικα του μαθητή, χωρίς να χρειάζεται να προσθέσει ο ίδιος κάποια εντολή οπτικοποίησης, όπως συμβαίνει στα συνήθη λογισμικά προσομοίωσης αλγορίθμων. Ο μαθητής βλέπει την οπτικοποίηση της προσομοίωσης του δικού του αλγορίθμου, πράγμα που σημαίνει ότι οποιοδήποτε λάθος στον κώδικα θα φανεί άμεσα στην οπτικοποίηση. Η ανατροφοδότηση από το σύστημα προσφέρει στο μαθητή μια συνολική εικόνα των ενεργειών του με στόχο να βοηθηθεί στην ανάπτυξη και βελτίωση του υπό μελέτη αλγορίθμου.

Η οπτικοποίηση ενός αλγορίθμου μέσω του DAVE δεν περιορίζεται στην απλή εμφάνιση των τιμών των μεταβλητών κατά την εκτέλεση, αλλά περιλαμβάνει την δυναμική οπτικοποίηση των δομών που προσομοιώνουν κατάλληλα τη λειτουργία του αλγορίθμου. Η επιλογή των μέσων αναπαράστασης έγινε έτσι ώστε ο μαθητής να μπορεί εύκολα να παρατηρήσει τις διαφορές μεταξύ των αναμενόμενων και των δικών του αποτελεσμάτων. Οι χρησιμοποιούμενες αναπαραστάσεις είναι διαφορετικές για κάθε αλγόριθμο. Για παράδειγμα, στον αλγόριθμο της ταξινόμησης μονοδιάστατου πίνακα με τη μέθοδο της φυσαλίδας χρησιμοποιούμε για την αναπαράσταση των στοιχείων του πίνακα ράβδους μεταβλητού μεγέθους, ώστε να είναι εμφανής η σχέση διάταξης (Σχήμα 1).



Σχήμα 1: *Οθόνη του λογισμικού DAVE με τον αλγόριθμο ταξινόμησης φυσαλίδας*

Το σύστημα DAVE περιλαμβάνει (Σχήμα 1):

- *Συντάκτη κειμένου*, όπου ο μαθητής έχει τη δυνατότητα να γράψει το πρόγραμμά του στη ΓΛΩΣΣΑ του σχολικού εγχειριδίου
- *Παράθυρο οπτικοποίησης*, στο οποίο εμφανίζεται το αποτέλεσμα της οπτικοποίησης της εκτέλεσης του αλγορίθμου
- *Πλήκτρα ελέγχου* της οπτικοποίησης και πειραματισμού (έναρξη, τερματισμός, παύση, βήμα εμπρός, βήμα πίσω)

- *Μεταγλωττιστή*, ο οποίος δέχεται ως είσοδο τον κώδικα του μαθητή στη γλώσσα προγραμματισμού ΓΛΩΣΣΑ και παράγει αυτόματα τον κώδικα της οπτικοποίησης που εκτελείται στη συνέχεια.

Το παράδειγμα εφαρμογής του λογισμικού DAVE αφορά σε βασικούς αλγορίθμους ταξινόμησης και στην οπτικοποίησή τους στις δύο διαστάσεις. Ο μαθητής μπορεί να αναπτύξει τον δικό του αλγόριθμο ταξινόμησης και να παρακολουθήσει την οπτικοποίησή του βήμα προς βήμα. Ο μεταγλωττιστής εντοπίζει τα σημαντικά γεγονότα που φαίνονται στην οπτικοποίηση, για παράδειγμα τη σύγκριση ή την ανταλλαγή δύο στοιχείων του πίνακα, και εισάγει τις κατάλληλες εντολές οπτικοποίησης. Η επέκταση του συστήματος σε άλλα είδη αλγορίθμων μπορεί να γίνει εύκολα με την προσθήκη και άλλων συμβάντων στον κώδικα του μεταγλωττιστή.

Ο παραγόμενος από το σύστημα κώδικας είναι σε γλώσσα Java, πράγμα που επιτρέπει την εκτέλεση της εφαρμογής σε οποιοδήποτε λειτουργικό περιβάλλον, καθώς και σε μορφή applet στον Παγκόσμιο Ιστό. Η οπτικοποίηση γίνεται στις δύο διαστάσεις ενώ υπάρχει δυνατότητα επέκτασης στις τρεις διαστάσεις με χρήση κατάλληλων βιβλιοθηκών γραφικών όπως είναι οι Java3D και JOGL. Η αρχιτεκτονική του συστήματος επιτρέπει την επέκτασή του και σε άλλες γλώσσες προγραμματισμού, για παράδειγμα Pascal, C, C++, Java κ.λπ. Το λογισμικό θα διατίθεται δωρεάν από την ιστοσελίδα των συγγραφέων με άδεια χρήσης Creative Commons, έτσι ώστε να μπορεί να χρησιμοποιηθεί ή/και να επεκταθεί από τρίτους.

Κατά τη διερευνητική και ανακαλυπτική προσέγγιση τους, μέσω του DAVE, οι μαθητές γίνονται ενεργά υποκείμενα της μάθησης και υποστηρίζονται με στόχο την κατανόηση της δυναμικής συμπεριφοράς του αλγορίθμου, την ανάδειξη των δικών τους αναπαραστάσεων για τον αλγόριθμο ή μέρη αυτού, την ανίχνευση των σφαλμάτων τους κατά την ανάπτυξη του προγράμματος και τη διαπίστωση των παρανοήσεών τους που σχετίζονται με αυτά. Σε τελική ανάλυση, οι μαθητές μαθαίνουν μέσα από διαδικασίες δοκιμής και άμεσης παρατήρησης του αποτελέσματος στην οθόνη του υπολογιστή. Η κατανόηση του αλγορίθμου βασίζεται στη μαθησιακή χρησιμότητα του λάθους, κατά τη διαδικασία της ανάπτυξης του αλγορίθμου από τους ίδιους τους μαθητές, και στην αξιοποίηση των διαφορών ανάμεσα στα αναμενόμενα και στα παρατηρούμενα αποτελέσματα, με στόχο την οικοδόμηση αποτελεσματικών αναπαραστάσεων για τις αλγοριθμικές δομές και τα προγραμματιστικά αντικείμενα.

5. Επίλογος

Συμπερασματικά, το λογισμικό οπτικοποίησης αλγορίθμων DAVE είναι ένα ανοικτό περιβάλλον, το οποίο παρέχει στο μαθητή δυνατότητες αλληλεπίδρασης σε υψηλό βαθμό, με στόχο την ανάπτυξη αλγορίθμων και την οικοδόμηση επαρκών αναπαραστάσεων για τις αλγοριθμικές δομές. Ο μαθητής αναπτύσσει ο ίδιος τον

αλγόριθμο που πρόκειται να οπτικοποιηθεί και δεν περιορίζεται απλά στον καθορισμό των δεδομένων εισόδου του αλγορίθμου. Έτσι έχει τη δυνατότητα να πειραματιστεί με τον δικό του αλγόριθμο και να ανιχνεύσει τα λάθη του μέσω οικείων αναπαραστάσεων που προκύπτουν από την οπτικοποίηση. Επιπλέον, έχει τη δυνατότητα να επικεντρωθεί στα σημαντικά χαρακτηριστικά κάθε αλγορίθμου και όχι στις τεχνικές λεπτομέρειες της χρησιμοποιούμενης γλώσσας προγραμματισμού. Η προτεινόμενη προσέγγιση από ένα ανοικτό εκπαιδευτικό λογισμικό οπτικοποίησης-προσομοίωσης αλγορίθμων, όπως είναι το DAVE, δίνει έμφαση στον παιδαγωγικό σχεδιασμό της μάθησης του προγραμματισμού και στη μετατόπιση από το συντακτικό-τεχνικό μέρος στην καλλιέργεια δεξιοτήτων επίλυσης προβλημάτων.

Στους μελλοντικούς στόχους μας είναι η επέκταση του συστήματος και η ενσωμάτωση πρόσθετων δυνατοτήτων αυτόματης οπτικοποίησης νέων αλγορίθμων, όπως είναι οι αλγόριθμοι για γράφους. Παράλληλα, σχεδιάζεται η πειραματική χρήση του λογισμικού από μαθητές της Γ' Λυκείου στα πλαίσια του μαθήματος Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον.

Βιβλιογραφία

Baecker, R. (1981). Sorting out Sorting. Narrated colour videotape, 30 minutes, presented at ACM SIGGRAPH'81.

Ben-Bassat Levy, R., Ben-Ari, M., & Uronen, A. P. (2003). The Jeliot 2000 program animation system, *Computers & Education*, 40(1), 15–21.

Brown, H., M. (1991). Zeus: A system for algorithm animation and multi-view editing. In *IEEE Workshop on Visual Languages*, Japan, 4-9.

Byrne, D., Catrambone, R., & Stasko, J. (1999). Evaluating animations as student aids in learning computer algorithms. *Computers & Education*, 33(4), 253–278.

Carson, E., Parberry, I., & Jensen, B. (2007). Algorithm explorer: visualizing algorithms in a 3D multimedia environment. In *38th SIGCSE Technical Symposium on Computer Science Education*, 155–159.

Demetrescu, C., & Finocchi, I. (1999). A technique for generating graphical abstractions of program data structures. In *3rd International Conference on Visual Information Systems*, LNCS 1614, pp. 785-792.

Hundhausen, D., & Brown, J. (2006). Designing, visualizing, and discussing algorithms within a CS 1 studio experience: An empirical study. *Computers & Education*, 50(1), 301-326.

Hundhausen, D., & Brown, J. (2007). What you see is what you code: A 'live' algorithm development and visualization environment for novice learners. *Journal of Visual Languages and Computing*, 18(1) 22-47.

Hundhausen, D., Douglas, S., & Stasko, J. (2002). A metastudy of algorithm visualization effectiveness. *Journal of Visual Languages & Computing*, 3(3), 259-290.

Kolikant, Y. B.-D., & Pollack, S. (2004). Establishing computer science professional norms among high-school students. *Computer Science Education*, 14(1), 21-35

Naps, T. L. (2005). JHAVE: supporting algorithm visualization. *Computer Graphics and Applications*, 25(5), 49-55.

Pierson, W., & Rodger, S. (1998). Web-based animation of data Structures Using JAWAA, *In 29th SIGCSE Technical Symposium on Computer Science Education*. pp. 267-271.

Ramadhan, H. A. (2000). Programming by discovery, *Journal of Computer Assisted Learning*, 16, 83-93.

Rößling, G. (2000). The Animal algorithm animation tool. *In 5th Annual SIGCSE/SGCUE Conference on Innovation and Technology in Computer Science Education, ITiCSE'00*. ACM, Helsinki, Finland, pp.37-40.

Sajaniemi, J. (2005). Roles of variables and learning to program. Στο Α. Τζιμογιάννης (επιμ.), *Πρακτικά 3ου Πανελλήνιου Συνεδρίου ''Διδακτική της Πληροφορικής''*, Κόρινθος.

Sajaniemi, J., & Kuittinen, M. (2003). PlanAni: A program animator based on the roles of variables. *ACM Symposium on Software Visualization SoftVis*, Demo/Poster Program, San Diego, CA.

Stasko, J. T. (1997). Using student-built algorithm animations as learning aids, In *28th SIGCSE Technical Symposium on Computer Science Education*, pages 25-29.

Stasko, J. T., Badre, A., & Lewis, C., (1993). Do algorithm animations assist learning? An empirical study and analysis. *INTERCHI Conference*, pp. 61–66.

Γρηγοριάδου, Μ., Γόγουλου, Α., Γούλη, Ε., & Σαμαράκου, Μ. (2004). Σχεδιάζοντας "Διερευνητικές+Συνεργατικές" δραστηριότητες σε εισαγωγικά μαθήματα προγραμματισμού, Στο Π. Πολίτης (επιμ.). Πρακτικά *2ης Διημερίδας με Διεθνή Συμμετοχή "Διδακτική της Πληροφορικής"*, 86-96, Βόλος.

Κόμης, Β. (2005). *Εισαγωγή στη Διδακτική της Πληροφορικής*, Αθήνα: Κλειδάριθμος

Κόμης, Β., & Τζιμογιάννης, Α. (2006). Ο προγραμματισμός ως μαθησιακή δραστηριότητα: από τις εμπειρικές προσεγγίσεις στη γνώση παιδαγωγικού περιεχομένου, *Θέματα στην Εκπαίδευση,* 7(3)*,* 229-255.

Τζιμογιάννης, Α., Τσιωτάκης, Π., & Sajaniemi, J. (2006). Μελετώντας το ρόλο των προσομοιώσεων αλγορίθμων στη διδασκαλία του προγραμματισμού στο Ενιαίο Λύκειο, Στο Ε. Σταυρίδου & Χ. Σολομωνίδου (επιμ.), *Πρακτικά Πανελλήνιου Συνεδρίου ''Ψηφιακό Εκπαιδευτικό Υλικό: Ζητήματα δημιουργίας, διδακτικής αξιοποίησης και αξιολόγησης'',* 99-108, Βόλος.

Τσέλιος, Ν., Γεωργόπουλος, Α., Πολίτης, Π., Πύρζα, Φ., Φανίκος, Α., Κουμπούρη, Δ., & Κόμης, Β. (2006). Μελέτη της χρήσης πολλαπλών αναπαραστάσεων προγράμματος σε εκπαιδευτικό περιβάλλον προγραμματισμού, *Θέματα στην Εκπαίδευση*, 7(3), 251-275.