

Η ΣΗΜΑΣΙΑ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ ΚΑΙ ΤΑ ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΗΣ ΑΛΓΟΡΙΘΜΙΚΗΣ ΕΠΙΛΥΣΗΣ ΣΤΟ ΜΑΘΗΜΑ ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΣΕ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

Κοίλιας Χρήστος
Επίκουρος Καθηγητής
ΤΕΙ Αθήνας, Τμήμα
Πληροφορικής
ckoilias@teiath.gr

Δουκάκης Σπύρος
Εργαστήριο Γνωσιακής
Επιστήμης, τΜΙΘΕ
Πανεπιστημίου Αθηνών
sdoukakis@mail.ntua.gr

Ψαλτίδου Αλεξάνδρα
Καθηγήτρια
Πληροφορικής ΠΕ19,
Δευτεροβάθμια
εκπαίδευση
alex_psa@yahoo.com

ΠΕΡΙΛΗΨΗ

Στην εργασία αυτή παρουσιάζεται η σημασία της αλγοριθμικής επίλυσης προβλημάτων. Μεταφέρονται οι εμπειρίες των εκπαιδευτικών από τη διδασκαλία του μαθήματος Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον της Τεχνολογικής Κατεύθυνσης των Ενιαίων και Εσπερινών Λυκείων και διατυπώνονται προβληματισμοί για το μάθημα.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Αλγόριθμος, Ψευδογλώσσα, Προγραμματισμός, Επίλυση προβλημάτων, Διδακτική Πληροφορικής.

ABSTRACT

This article focuses on the importance of solving problems using algorithms. It presents personal experiences from High School teachers, teaching the Greek Lyceum course "Application Development in Programming Environment".

ΕΙΣΑΓΩΓΗ

Η αλγοριθμική προσέγγιση ενός προβλήματος δίνει έμφαση στον τρόπο επίλυσης του προβλήματος, ανεξάρτητα από το χώρο από τον οποίο προέρχεται το πρόβλημα. Η ικανότητα περιγραφής της επίλυσης ενός προβλήματος με μια μορφή αναπαράστασης αλγορίθμου εφοδιάζει το μαθητή με έναν γενικό οδηγό για την αντιμετώπιση προβλημάτων. Η αλγοριθμική σχεδίαση επίλυσης προβλημάτων αναπτύσσει γνωστικές δεξιότητες υψηλού επιπέδου και αποτελεί ένα ισχυρό μέσο για τη διδασκαλία βασικών εννοιών που βρίσκουν εφαρμογή σε όλους τους τομείς δραστηριοτήτων [Papert, 1980].

Η ΣΗΜΑΣΙΑ ΚΑΙ ΟΙ ΔΥΝΑΤΟΤΗΤΕΣ ΤΗΣ ΑΛΓΟΡΙΘΜΙΚΗΣ ΣΧΕΔΙΑΣΗΣ

Οι μαθητές καθημερινά, τόσο στο σχολείο όσο και σε άλλες τους δραστηριότητες καλούνται να επιλύσουν προβλήματα. Αυτό δεν είναι πάντα εύκολο. Πολλές φορές το πρόβλημα δεν είναι κατανοητό, αφού μπορεί να μην έχει διατυπωθεί σωστά από το δημιουργό του ή μπορεί να ερμηνεύεται λάθος από τη μεριά του επίδοξου λύτη. Άλλες φορές, η επίλυση του προβλήματος δεν είναι προφανής και υπάρχει και η περίπτωση

οι μαθητές να έχουν καταλήξει στον τρόπο επίλυσης, αλλά να μην μπορούν να επικοινωνήσουν (δηλαδή να διατυπώσουν ή να αναπαράγουν) τον τρόπο επίλυσης που έχουν σκεφθεί.

Σε όλα τα σχολικά μαθήματα, από την έκθεση έως και τα μαθηματικά, οι μαθητές καλούνται να αναπτύξουν θέματα με σκοπό να επιλύσουν ένα πρόβλημα (συγγραφή έκθεσης, σχολιασμός κειμένου, επίλυση μιας εξίσωσης, τακτική άμυνας σε έναν αγώνα ποδοσφαίρου). Συνεπώς, οι μαθητές λύνουν προβλήματα χωρίς να έχουν ποτέ διδαχτεί τον αλγοριθμικό τρόπο σκέψης.

Η αλγοριθμική προσέγγιση ενός προβλήματος δίνει έμφαση στον τρόπο επίλυσης του προβλήματος, ανεξάρτητα από το χώρο από τον οποίο προέρχεται το πρόβλημα.

Ο αλγόριθμος είναι ένα σύνολο από λειτουργίες, οι οποίες περιγράφουν βήμα προς βήμα τις ενέργειες που πρέπει να εκτελεστούν για να ολοκληρωθεί μία εργασία ή να επιλυθεί ένα πρόβλημα σε πεπερασμένο αριθμό ενεργειών [Beekman, 1997].

Αφού, λοιπόν, ο μαθητής έχει κατανοήσει το πρόβλημα και έχει σκεφτεί αλγοριθμικά την επίλυσή του, πρέπει να καταγράψει τη λύση. Έτσι, στην έκθεση αφού κατανοήσει το θέμα δημιουργεί ένα σχεδιάγραμμα, το οποίο ακολουθεί για τη συγγραφή της. Στα μαθηματικά, για την επίλυση της δευτεροβάθμιας εξίσωσης, ο μαθητής συγκεντρώνει τα δεδομένα και με εφαρμογή των μαθηματικών κανόνων προσδιορίζει τις λύσεις της εξίσωσης. Στη σχολική ομάδα καλαθοσφαίρισης, ο γυμναστής-προπονητής δημιουργεί ένα σχεδιάγραμμα θέσεων των μαθητών για τον τρόπο επίθεσης.

Η αλγοριθμική επίλυση ενός προβλήματος, λοιπόν, μπορεί να καταγραφεί με τους ακόλουθους τρόπους:

1. **Ελεύθερο κείμενο**, το οποίο αποτελεί τον πιο ανεπεξέργαστο και αδόμητο τρόπο παρουσίασης αλγορίθμου.
2. **Φυσική γλώσσα με βήματα**.
3. **Διαγραμματικές τεχνικές** που συνιστούν ένα γραφικό τρόπο παρουσίασης του αλγορίθμου.
4. **Κωδικοποίηση** δηλαδή με ένα πρόγραμμα γραμμένο είτε με μία ψευδογλώσσα, είτε σε κάποιο προγραμματιστικό περιβάλλον που όταν εκτελεσθεί θα δώσει τα ίδια αποτελέσματα με τον αλγόριθμο [Βακάλη et al., 1999].

Η περιγραφή της επίλυσης ενός προβλήματος με έναν από τους παραπάνω τρόπους αναπαράστασης εφοδιάζει το μαθητή με έναν γενικό οδηγό για την αντιμετώπιση των προβλημάτων που του θέτονται προς επίλυση.

Η εκπαιδευτική διαδικασία, λοιπόν, απαιτεί από τους εκπαιδευόμενους να μπορούν να επιλύουν καθημερινά ή επιστημονικά προβλήματα, γεγονός που δηλώνει ότι πρέπει να σκέπτονται αλγοριθμικά και στη συνέχεια να καταγράφουν τη λύση του προβλήματος χρησιμοποιώντας την αλγοριθμική προσέγγιση.

ΑΝΑΠΑΡΑΣΤΑΣΗ ΑΛΓΟΡΙΘΜΟΥ Ή ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ;

Όταν το πρόβλημα πρόκειται να επιλυθεί με τη χρήση του υπολογιστή, τότε η αλγοριθμική σχεδίαση είναι το πρώτο στάδιο για την υλοποίηση του προγράμματος σε κατάλληλη γλώσσα προγραμματισμού. Ο αλγόριθμος, λοιπόν, παίζει ένα κεντρικό

ρόλο και στην επιστήμη των υπολογιστών, γεγονός που υποδηλώνεται και από τη ρήση του Wirth: Πρόγραμμα = Αλγόριθμος + Δομές Δεδομένων.

Έτσι, λοιπόν, για να κωδικοποιηθεί η λύση ενός προβλήματος σε γλώσσα προγραμματισμού, πρέπει να έχει σχεδιαστεί και καταγραφεί η αλγοριθμική επίλυση του προβλήματος. Η καταγραφή της αλγοριθμικής επίλυσης χαρακτηρίζεται από "ελευθερίες". Τέτοιες "ελευθερίες" βρίσκουμε τόσο στο λεξιλόγιο, όσο και στο συντακτικό. Ας δούμε ένα παράδειγμα:

Όταν αλγοριθμικά χρειάζεται να αναπτυχθεί μία δομή επανάληψης, τότε είναι αποδεκτό οποιοδήποτε από τα παρακάτω:

Όσο συνθήκη **επανάλαβε**

ή

Όσο συνθήκη **κάνε**

ή

Επανάλαβε όσο συνθήκη

Οι παραπάνω δομές υποδηλώνουν την ίδια ενέργεια. Είναι μάλιστα εύκολη η κωδικοποίηση της παραπάνω δομής επανάληψης σε μία γλώσσα προγραμματισμού.

Το γεγονός ότι υπάρχει ελευθερία στην καταγραφή του αλγορίθμου, διευκολύνει σχετικά μαθήματα (σχολικά ή πανεπιστημιακά) που διδάσκονται στην τάξη και εξετάζονται στο "χαρτί". Ο αλγόριθμος ούτε μεταγλωττίζεται, ούτε εκτελείται από υπολογιστή, αλλά είναι μία παρουσίαση της λογικής της επίλυσης του προβλήματος. Πρέπει, λοιπόν, οι διδάσκοντες, οι εξεταστές και οι διδασκόμενοι, να εκμεταλλευτούν αυτό το πλεονέκτημα, ώστε να ενισχύουν τη διδασκαλία της λογικής της επίλυσης και να μην αναλώνονται στη διδασκαλία της συντακτικής αυστηρότητας.

Με δεδομένο την υλικοτεχνική υποδομή των σχολικών και πανεπιστημιακών μονάδων, είναι αδύνατο να πραγματοποιηθεί εξέταση μαθημάτων αλγοριθμικής σχεδίασης σε υπολογιστή. Έτσι το πρόσφορο μέσο είναι η γραπτή εξέταση. Στο "χαρτί" δεν έχει ιδιαίτερη αξία η αξιολόγηση του μαθητή σε συντακτικούς κανόνες. Το αποτέλεσμα του αλγορίθμου δεν επηρεάζεται από τη χρήση μονού ή διπλού εισαγωγικού, τη χρήση του συγκριτικού τελεστή \neq ή του $\langle \rangle$, τη δήλωση των μεταβλητών κτλ.

Η διαδικασία αλγοριθμικής επίλυσης, διευκολύνει τον επίδοξο επίλυτη, αφού του επιτρέπει να καταγράψει τη λύση του προβλήματος σε μία φυσική γλώσσα. Ένας αλγόριθμος γραμμένος στα ελληνικά θα έχει το ίδιο αποτέλεσμα με τον ίδιο αλγόριθμο γραμμένο σε οποιαδήποτε φυσική γλώσσα, αφού και οι δύο αλγόριθμοι παρουσιάζουν την ίδια σειρά ενεργειών.

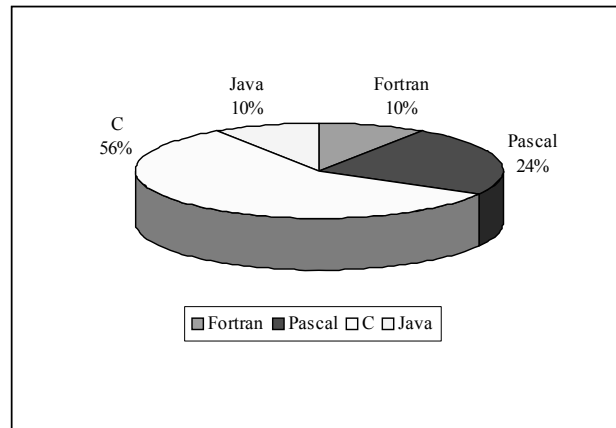
Φαίνεται, λοιπόν, ότι μια αλγοριθμική επίλυση είναι ανεξάρτητη από τη φυσική γλώσσα που έχει χρησιμοποιήσει ο λύτης, γεγονός που κάνει την κατανόησή της ευκολότερη σε σχέση με την κωδικοποίηση σε γλώσσα προγραμματισμού. Πολλοί διαφωνούν με τη χρήση αλγορίθμου για την επίλυση ενός προβλήματος, αφού δεν είναι δυνατή η υπολογιστική εκτέλεσή του. Ένας αλγόριθμος δεν εκτελείται, ενώ με το πρόγραμμα ο υπολογιστής "λαμβάνει" τον κώδικα του προγράμματος και τα δεδομένα, τα επεξεργάζεται και επιστρέφει τα αποτελέσματα. Δεν είναι όμως σκοπός η εξέταση της δυνατότητας του υπολογιστή να εκτελέσει εντολές. Σκοπός είναι ο μαθητής να έχει τη δυνατότητα να καταγράψει τα λογικά βήματα του αλγορίθμου και να ελέγχει την αποδοτικότητα του. Είναι πολύ σημαντικό ο διδασκόμενος να μπορεί να "εκτελεί" στο

"χαρτί" τον αλγόριθμο για να γνωρίζει τι αποτελέσματα θα δώσει για διαφορετικά δεδομένα εισόδου. Τα λογικά λάθη δεν μπορούν να διαπιστωθούν από την εκτέλεση του προγράμματος με τη βοήθεια του υπολογιστή, αλλά μόνο με συγκεκριμένα δεδομένα ελέγχου και "τρέξιμο" με το "χέρι", τα οποία και βοηθούν στη διαπίστωσή τους.

Ο αλγόριθμος παρέχει τη λογική δομή του προγράμματος. Για το λόγο αυτό, η μεταφορά του αλγόριθμου σε πρόγραμμα είναι μία απλή διαδικασία. Οι ενέργειες του αλγορίθμου μεταφράζονται άμεσα σε εντολές προγράμματος σε οποιαδήποτε γλώσσα προγραμματισμού. Ο αλγόριθμος περιγράφει τη λύση και η κωδικοποίηση της λύσης σε μία γλώσσα προγραμματισμού γίνεται με σκοπό να εκφράσουμε τις ενέργειες και να επικοινωνούμε με τη μηχανή.

Κάποιοι υποστηρίζουν, ότι η εκμάθηση της αλγοριθμικής σχεδίασης θα μπορούσε να γίνει με τη χρήση μιας γλώσσας προγραμματισμού. Το πρόβλημα που προκύπτει είναι η επιλογή της κατάλληλης γλώσσας προγραμματισμού. Η λύση θα ήταν η χρήση μίας από τις γνωστές γλώσσες προγραμματισμού (Basic, Pascal, C, Java), είτε η κατασκευή μιας ελληνικής γλώσσας προγραμματισμού, που θα ήταν κατάλληλη για εκπαιδευτική χρήση. Είναι αποδεκτό από όλους, ότι δεν υπάρχει μία αντικειμενικά καλύτερη εκπαιδευτική γλώσσα. Αυτό φαίνεται και από την υπάρχουσα κατάσταση που επικρατεί στα ανώτατα εκπαιδευτικά ιδρύματα.

Με μία μελέτη για το τι διδάσκεται στο πρώτο εξάμηνο στα ελληνικά Πανεπιστήμια και τα ΑΤΕΙ διαπιστώθηκαν τα ακόλουθα:



Γράφημα 1: Ποσοστιαία παρουσίαση επιλογής γλώσσας προγραμματισμού για το πρώτο εξάμηνο για Τμήματα Πανεπιστημίων - ΑΤΕΙ Πληροφορικής, Υπολογιστών, Μηχανικών Πληροφορικής και Επικοινωνιών.

Παρατήρηση: Η μελέτη έγινε με βάση τα διαθέσιμα στοιχεία του Διαδικτύου και αφορά 21 τμήματα Πληροφορικής Πανεπιστημίων - ΑΤΕΙ.

Έτσι, η διδασκαλία ενός μαθήματος αλγοριθμικής σχεδίασης στη δευτεροβάθμια εκπαίδευση πρέπει να προεκπαιδεύει τους μαθητές στον προγραμματισμό και να τους βοηθάει να μεταβούν εύκολα σε ένα προγραμματιστικό περιβάλλον στο Πανεπιστήμιο.

Θα ήταν εξειδίκευση η εκμάθηση μιας γλώσσας προγραμματισμού στη δευτεροβάθμια η οποία μπορεί και να μην διδάσκεται στο Πανεπιστήμιο.

Από την άλλη, η επιλογή της κατασκευής μιας ελληνικής γλώσσας προγραμματισμού με συντακτικούς κανόνες και ελληνικό λεξιλόγιο και η χρήση της για την εκπαίδευση, δεν θα αποφέρει ουσιαστικό κέρδος στους εκπαιδευόμενους. Αυτή η γλώσσα θα είναι μία γλώσσα για τη συγκεκριμένη ομάδα εκπαιδευόμενων και η χρήση της θα έχει τοπική εμβέλεια.

ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ ΣΕ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ ΠΕΡΙΒΑΛΛΟΝ

Στα πλαίσια του προγράμματος σπουδών για την πληροφορική στο σχολείο διδάσκεται για τέταρτη συνεχή χρονιά, στην Τεχνολογική Κατεύθυνση της Γ΄ τάξης Ενιαίων και στη Δ΄ τάξη Εσπερινών Λυκείων, το μάθημα Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον. Το μάθημα υποστηρίζεται από διδακτικό πακέτο τριών βιβλίων που συμπεριλαμβάνουν βιβλίο μαθητή, τετράδιο μαθητή και βιβλίο καθηγητή. Το βιβλίο του καθηγητή εξηγεί με σαφήνεια στους διδάσκοντες τους στόχους του μαθήματος, όπου τονίζονται τα ακόλουθα:

"Το μάθημα Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον δεν έχει σαν στόχο τη διδασχία και την εκμάθηση κάποιου συγκεκριμένου προγραμματιστικού περιβάλλοντος, ούτε την καλλιέργεια προγραμματιστικών δεξιοτήτων από τη μεριά των μαθητών. Δεν αποσκοπεί στη λεπτομερειακή εξέταση της δομής, του ρεπερτορίου και των συντακτικών κανόνων κάποιας γλώσσας προγραμματισμού. Δεν προτίθεται να επιχειρήσει να δημιουργήσει προγραμματιστές. Το μάθημα δεν αφορά την εκμάθηση εξεζητημένων τεχνικών προγραμματισμού, αλλά ως εργαλείο δόμησης της σκέψης πρέπει να εστιάζει στις προσεγγίσεις και στις τεχνικές επίλυσης προβλημάτων."

Ερμηνεύοντας το στόχο και μελετώντας το υλικό, καταλήξαμε στο συμπέρασμα ότι στο μάθημα αυτό οι μαθητές πρέπει πρώτα να εκπαιδεύονται στην ανάλυση προβλημάτων και στην αλγοριθμική επίλυσή τους και ύστερα στη μετάβαση από τον αλγόριθμο σε πρόγραμμα με τη χρήση μιας γλώσσας προγραμματισμού.

Η διδασκαλία του μαθήματος στην τάξη και η συμμετοχή μας στη διόρθωση των γραπτών των πανελληνίων μας οδήγησαν στις ακόλουθες παρατηρήσεις:

1. Το διδακτικό πακέτο αρχικά παρουσιάζει την επίλυση των προβλημάτων χρησιμοποιώντας ψευδογλώσσα (κεφάλαια 2 έως 5) και στη συνέχεια η ψευδογλώσσα κωδικοποιείται με τη χρήση της ΓΛΩΣΣΑΣ (κεφάλαια 7 έως 14). Έτσι, η παρουσίαση των αλγοριθμικών δομών και των δομών δεδομένων με ψευδογλώσσα και ΓΛΩΣΣΑ, είναι αρωγός στο έργο του εκπαιδευτικού, αφού μπορεί να διδάξει στην τάξη αλγόριθμο και να παρουσιάσει στο εργαστήριο το πρόγραμμα καλύπτοντας με τον τρόπο αυτό τους στόχους του μαθήματος.

Η πραγματικότητα όμως είναι διαφορετική. Το πρόγραμμα σπουδών περιορίζει το μάθημα σε δύο διδακτικές ώρες, με αποτέλεσμα οι εκπαιδευτικοί να μην έχουν τον απαραίτητο χρόνο να παρουσιάσουν την ύλη του μαθήματος με πληρότητα. Ακόμα και στην περίπτωση που υπήρχε μια επιπλέον διδακτική ώρα, όπως είχε προταθεί από το Παιδαγωγικό Ινστιτούτο, η παρουσίαση και χρήση της ΓΛΩΣΣΑΣ στο εργαστήριο δεν θα ήταν εφικτή, αφού δεν παρέχεται κατάλληλο λογισμικό ως εργαλείο προγραμματισμού. Θα μπορούσε βέβαια να χρησιμοποιηθεί το προγραμματιστικό

περιβάλλον της Basic ή της Pascal για την εξοικείωση των μαθητών στον προγραμματισμό. Η επιλογή αυτή επιβαρύνει τους καθηγητές με τη διδασκαλία μιας τεχνητής γλώσσας και τους μαθητές με την εκμάθηση της γλώσσας αυτής. Επιπλέον, μια τέτοια επιλογή θα ήταν πέρα από τους στόχους του μαθήματος, αφού σύμφωνα με τη φετινή εξεταστέα ύλη, οι μαθητές μπορούν να αναπτύξουν τα θέματα των εξετάσεων είτε σε οποιαδήποτε μορφή παράστασης αλγορίθμου, είτε σε ΓΛΩΣΣΑ, αποκλείοντας έτσι τη δυνατότητα, που οι μαθητές είχαν τις προηγούμενες χρονιές, να αναπτύξουν τα θέματα σε Basic ή Pascal.

Με δεδομένο ότι οι ώρες διδασκαλίας δεν είναι αρκετές και ότι οι μαθητές μπορούν να επιλέξουν οποιαδήποτε μορφή αναπαράστασης αλγορίθμου για την ανάπτυξη των θεμάτων, γίνεται προσπάθεια κάλυψης της ύλης δίνοντας μεγαλύτερη βαρύτητα σε μια από τις δύο μορφές κωδικοποίησης (ψευδογλώσσα ή ΓΛΩΣΣΑ). Με την παρούσα κατάσταση, η ΓΛΩΣΣΑ θα πρέπει να διδαχθεί στον "πίνακα" και να εξετασθεί στο "χαρτί", περιορισμός που δεν θυμίζει καθόλου προγραμματισμό. Η επιλογή να δοθεί βαρύτητα στην ψευδογλώσσα καλύπτει σε μεγάλο βαθμό τους στόχους του μαθήματος και χρήζει των πλεονεκτημάτων αλγοριθμικού σχεδιασμού που προαναφέραμε σε σχέση με τον προγραμματισμό στη ΓΛΩΣΣΑ.

2. Με τη σπειροειδή προσέγγιση που προτείνεται από τους συγγραφείς του διδακτικού πακέτου, οι έννοιες να μην προσεγγίζονται επαναληπτικά, αλλά οι διαφορετικές παρουσιάσεις του ίδιου θέματος, οδηγούν σε σύγχυση μαθητές και συναδέλφους. Έτσι, παρουσιάζονται προβλήματα κατά τη διδασκαλία, τη διόρθωση και την αξιολόγηση των γραπτών των πανελλήνιων. Ας δούμε κάποια παραδείγματα. Πότε χρησιμοποιούνται τα **Δεδομένα** και τα **Αποτελέσματα**; Πότε χρησιμοποιείται η εντολή **Εκτύπωσε**, πότε η εντολή **Γράψε** και πότε η εντολή **Εμφάνισε**; Είναι αποδεκτή η εντολή **Τύπωσε**; Μέσα από το διδακτικό πακέτο τα θέματα αυτά προσεγγίζονται με ποικίλους τρόπους, δυσκολεύοντας καθηγητές και μαθητές να προσδιορίσουν τον σωστό.

3. Θεωρούμε ότι πρέπει να υπάρξει ένας σαφής διαχωρισμός μεταξύ ψευδογλώσσας και ΓΛΩΣΣΑΣ για την ορθότητα της εξέτασης του μαθήματος. Στις εξετάσεις, ο μαθητής πρέπει να έχει τη δυνατότητα να επιλέξει οποιαδήποτε από τους τέσσερις τρόπους αναπαράστασης αλγορίθμων, δηλαδή Ελεύθερο κείμενο, Φυσική γλώσσα με βήματα, Διαγραμματικές τεχνικές ή Κωδικοποίηση (ψευδογλώσσα ή πρόγραμμα στη ΓΛΩΣΣΑ) χωρίς να υποχρεώνεται να χρησιμοποιήσει μία από αυτές. Για να επιτρέπεται στο μαθητή να αναπτύξει τον αλγόριθμο σε όποια μορφή επιθυμεί, η ιδανική εκφώνηση στις εξετάσεις πρέπει να έχει την παρακάτω μορφή: "Να αναπτύξετε αλγόριθμο ο οποίος...". Είναι χρέος μας όμως, να τονίζουμε στους μαθητές ότι ο καλύτερος τρόπος αναπαράστασης είναι η ψευδογλώσσα. Αυτή η δυνατότητα επιλογής αφορά μόνο την ανάπτυξη αλγορίθμων και όχι τη θεωρητική εξέταση του προγραμματισμού ή του "τρεξίματος" ενός προγράμματος. Για παράδειγμα οι μαθητές πρέπει να είναι σε θέση να απαντήσουν ερωτήσεις σχετικά με τον τύπο των μεταβλητών ή με τα αποτελέσματα της εκτέλεσης ενός προγράμματος.

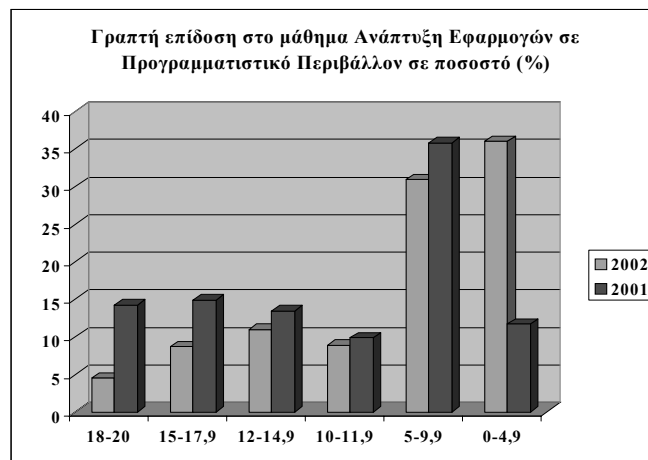
Τονίζουμε αυτό το σημείο, λόγω της άτοπης επιλογής του 4ου θέματος των εξετάσεων των Ενιαίων Λυκείων τον Ιούνιο του 2002. Το θέμα έλεγε:

"Μια εταιρεία αποθηκεύει είκοσι (20) προϊόντα σε δέκα (10) αποθήκες. Να γράψετε πρόγραμμα στη γλώσσα προγραμματισμού "ΓΛΩΣΣΑ", το οποίο:

α. περιέχει τμήμα δήλωσης των μεταβλητών του προγράμματος

Μονάδες 3"

Με το ερώτημα αυτό, τυπικά καταρρίφθηκε η επιλογή των μαθητών να γράψουν σε οποιαδήποτε μορφή αναπαράστασης θέλουν βάσει του καθορισμού της εξεταστέας ύλης. Η διόρθωση ήρθε καθυστερημένα, και δεν αφορούσε τους μαθητές αλλά τους διορθωτές και έλεγε ότι: "...Αν οι μαθητές έχουν χρησιμοποιήσει στο τέταρτο θέμα την αλγοριθμική προσέγγιση της ψευδογλώσσας ή κάποια από τις γλώσσες προγραμματισμού όπως Pascal, Basic, Turbo Pascal ή Quick Basic, τότε εφόσον η λύση είναι αλγοριθμικά ορθή, η απάντηση θεωρείται πλήρης". Το ερώτημα που ποτέ δεν απαντήθηκε είναι ο τρόπος βαθμολόγησης της απάντησης διατυπωμένης σε ψευδογλώσσα ή Basic, όπου δεν είναι απαραίτητο το τμήμα δηλώσεων των μεταβλητών. Η καθυστέρηση δε αυτή, διαφοροποίησε τη βαθμολόγηση των φυσικώς αδυνάτων μαθητών, αφού η πλειοψηφία αυτών είχε περατώσει την εξέτασή της. Συγκρίνοντας μάλιστα, τη γραπτή επίδοση των μαθητών στο μάθημα της Ανάπτυξης Εφαρμογών σε Προγραμματιστικό Περιβάλλον για τα έτη 2002 και 2001, αναδεικνύεται το πρόβλημα στην επιλογή του θέματος.



Γράφημα 2: Διαγραμματική παρουσίαση γραπτής επίδοσης στο μάθημα της Ανάπτυξης Εφαρμογών σε Προγραμματιστικό Περιβάλλον για τα έτη 2002 και 2001.

Τέλος, η διάρθρωση της φετινής ύλης περιλαμβάνει τα υποπρογράμματα, τα οποία παρουσιάζονται μόνο σε προγραμματιστικό περιβάλλον. Αν, λοιπόν, υπάρξει σχετικό θέμα ανάπτυξης υποπρογράμματος, οι μαθητές καλούνται να χρησιμοποιήσουν τη ΓΛΩΣΣΑ. Με αυτό τον τρόπο καταρρίπτεται η δυνατότητα επιλογής των μαθητών να διατυπώνουν τις λύσεις των ασκήσεων των εξετάσεων σε οποιαδήποτε μορφή παράστασης αλγορίθμου.

Οι εξεταστές θα μπορούσαν να εξετάσουν τις γνώσεις των μαθητών στον προγραμματισμό με άλλους τρόπους. Για παράδειγμα, στο θέμα των εξετάσεων των Ενιαίων Λυκείων τον Ιούνιο του 2002 θα έπρεπε να επιτρέπεται στους μαθητές να χρησιμοποιήσουν οποιαδήποτε μορφή αναπαράστασης για την ανάπτυξη του

αλγορίθμου και με ένα υποερώτημα να ζητείται από τους μαθητές να προσδιορίσουν τον τύπο μεταβλητών που χρησιμοποίησαν. Ελπίζουμε, κατά την επιλογή των θεμάτων, να αποφευχθούν ερωτήματα που να υποχρεώνουν τους μαθητές να αναπτύξουν την απάντησή τους με συγκεκριμένο τρόπο αναπαράστασης αλγορίθμου.

ΤΟ ΜΕΛΛΟΝ ΤΟΥ ΜΑΘΗΜΑΤΟΣ

Αν και από τα παραπάνω αναδείχθηκαν ορισμένα προβλήματα που διέπουν το μάθημα, η χρησιμότητά του και η ανταπόκριση των μαθητών είναι οι καλύτεροι σύμμαχοί μας για να αγωνιστούμε για τη βελτίωσή του.

Προτείνουμε, την αύξηση των ωρών διδασκαλίας και την ύπαρξη κατάλληλου λογισμικού το οποίο να έχει πρόσθετη αξία. Αυτό σημαίνει ότι χρειαζόμαστε διερευνητικό λογισμικό, με εργαλεία για επεξεργασία και οπτικοποίηση πληροφοριών και πολλαπλούς τρόπους έκφρασης. Χρειαζόμαστε λογισμικό που θα συνοδεύεται με προσεχτικά σχεδιασμένα σενάρια χρήσης και οδηγίες παιδαγωγικής αξιοποίησης. Χρειαζόμαστε λογισμικό που θα προσφέρει τη δυνατότητα δοκιμής υποθέσεων και σεναρίων με στόχο την επαλήθευση της ορθότητάς και της αποτελεσματικότητας του αλγορίθμου και την καταγραφή της αλληλουχίας των δοκιμών [Κυνηγός et al]. Με τον τρόπο αυτό, το μάθημα θα γίνει πραγματικά εργαστηριακό, αφού ο διδάσκων, θα ενισχύσει τη διδασκαλία των αλγορίθμων με εργαστηριακά παραδείγματα.

Η χρήση του διδακτικού πακέτου τα τέσσερα αυτά χρόνια, ανέδειξε ελλείψεις και παραβλέψεις. Πολλοί συνάδελφοι σε προσωπικούς δικτυακούς τόπους ή μέσω του σχολικού δικτύου, καθώς και η συγγραφική ομάδα έχουν δημοσιεύσει παροράματα. Ελπίζουμε, το συντομότερο δυνατό, το Παιδαγωγικό Ινστιτούτο να προχωρήσει στην ανανέωσή του διδακτικού πακέτου.

Τέλος, θεωρούμε ότι η συνεχής βελτίωση και εξέλιξη του μαθήματος μπορεί και πρέπει να οδηγήσει την ΕΠΥ καθώς και σχολές - τμήματα Πληροφορικής Πανεπιστημίων - ΑΤΕΙ να απαιτήσουν να γίνει το μάθημα της Ανάπτυξης Εφαρμογών σε Προγραμματιστικό Περιβάλλον, μάθημα αυξημένης βαρύτητας για την εισαγωγή σε σχολές και τμήματα Πληροφορικής.

ΕΠΙΛΟΓΟΣ

Όπως έγινε φανερό από τις προηγούμενες ενότητες, η αλγοριθμική σχεδίαση έχει πρωτεύοντα ρόλο στη γνωστική ανάπτυξη των μαθητών. Το μάθημα Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον πρέπει να δίνει έμφαση στην αλγοριθμική σχεδίαση επίλυσης προβλημάτων. Για να επιτευχθεί αυτός ο στόχος, η διδασκαλία θα πρέπει να επικεντρωθεί στη δόμηση της σκέψης και στις τεχνικές επίλυσης προβλημάτων και όχι στους συντακτικούς κανόνες κάποιας γλώσσας προγραμματισμού.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Beekman, G. (1997), Computer Confluence, Addison - Wesley.
2. Papert S. (1980), Νοητικές Θύελλες: Παιδιά, ηλεκτρονικοί υπολογιστές και δυναμικές ιδέες, Εκδόσεις Οδυσσέας.
3. Βακάλη Α., Γιαννόπουλος Η., Ιωαννίδης Ν., Κοΐλιας Χ., Μάλαμας Κ., Μανωλόπουλος Ι., Πολίτης Π. (1999), Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον, Υπουργείο Εθνικής Παιδείας και Θρησκευμάτων.
4. Δουκάκης Σ. (2002), Παρατηρήσεις για τις Εξετάσεις στο Μάθημα Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον, <http://users.ntua.gr/sdoukakis>.
5. Δουκάκης Σ., Ψαλτίδου Α. (2002), Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον, Εκδόσεις Νέων Τεχνολογιών.
6. Κυνηγός Χ., Κουτλής Μ., (), Λογισμικό υπό ... συνθήκες
7. Τζιμογιάννης Α., Κόμης Β. (2000), Η έννοια της μεταβλητής στο προγραμματισμό: Δυσκολίες και παρανοήσεις μαθητών του Ενιαίου Λυκείου, 2^ο Πανελλήνιο Συνέδριο: Οι Τεχνολογίες της Πληροφορίας και της Επικοινωνίας στην Εκπαίδευση, Εκδόσεις Νέων Τεχνολογιών.