

# Προϋπάρχουσες γνώσεις των μαθητών Γυμνασίου και Λυκείου στους αλγόριθμους ταξινόμησης. Μια συγκριτική ανάλυση

Ευριπίδης Βραχνός<sup>1</sup>, Αθανάσιος Τζιμογιάννης<sup>1</sup>,  
[evrachnos@gmail.com](mailto:evrachnos@gmail.com), [ajimoyia@uop.gr](mailto:ajimoyia@uop.gr)

<sup>1</sup> Τμήμα Κοινωνικής και Εκπαιδευτικής Πολιτικής, Πανεπιστήμιο Πελοποννήσου

## Περίληψη

Η ταξινόμηση αποτελεί ένα από τα θεμελιώδη προβλήματα της επιστήμης της πληροφορικής. Οι αλγόριθμοι ταξινόμησης χρησιμοποιούνται ως εισαγωγικά παραδείγματα σε μαθήματα ανάλυσης και σχεδίασης αλγορίθμων. Σύμφωνα με την θεωρία του εποικοδομισμού η νέα γνώση χτίζεται πάνω σε προϋπάρχουσα γνώση ή στις εμπειρίες που έχουν οι μαθητές, οπότε η διερεύνηση των προϋπάρχουσών γνώσεων και αναπαραστάσεων των μαθητών είναι πολύ χρήσιμη για τον σχεδιασμό διδακτικών στρατηγικών. Στην εργασία αυτή παρουσιάζουμε τα αποτελέσματα μιας έρευνας που έγινε σε μαθητές γυμνασίου στους οποίους δόθηκε ένα πρόβλημα ταξινόμησης και τους ζητήθηκε να καταγράψουν στο χαρτί τον αλγόριθμο με τον οποίο θα ταξινομήσουν ένα σύνολο αριθμών σε αύξουσα σειρά. Τα αποτελέσματα παρουσιάζονται συγκριτικά με τα αποτελέσματα αντίστοιχης έρευνας που έγινε σε μαθητές Λυκείου και γίνεται μια προσπάθεια ερμηνείας των ευρημάτων των δυο ερευνών. Οι δυο αυτές έρευνες είναι οι μοναδικές μέχρι στιγμής διεθνώς σχετικά με τη διερεύνηση των προϋπάρχουσών γνώσεων μαθητών σχετικά με τους αλγόριθμους ταξινόμησης.

**Λέξεις κλειδιά:** Αλγόριθμοι ταξινόμησης, Προϋπάρχουσες γνώσεις

## Εισαγωγή

Η διδασκαλία του προγραμματισμού υπολογιστών και της αλγοριθμικής σχεδίασης συνιστούν ένα έργο με ιδιαίτερες δυσκολίες. Πολλοί μαθητές, ακόμη και φοιτητές, δεν μπορούν να σχεδιάσουν ολοκληρωμένα και λογικά ορθά προγράμματα ακόμη και μετά από πολλά μαθήματα στον προγραμματισμό. (Κόμης & Τζιμογιάννης, 2006; Τζιμογιάννης, 2005; De Raadt, 2007; Robins et al., 2003). Ένας από τους λόγους που οι αρχάριοι προγραμματιστές συναντούν τόσες δυσκολίες, είναι οι αφηρημένες αλγοριθμικές έννοιες που πρέπει να διαχειριστούν και οι οποίες έχουν μικρή σχέση με τα αντικείμενα της καθημερινότητας. Έννοιες όπως βρόχος, μετρητής, δείκτης πίνακα, αναδρομή, είναι δύσκολο να χτιστούν πάνω στις εμπειρίες των μαθητών ή πάνω σε φυσικά μοντέλα που είναι οικεία σε αυτούς.

Σύμφωνα με την θεωρία του εποικοδομισμού (Ben-Ari, 2001), η μάθηση συνίσταται στην τροποποίηση ή επέκταση των γνώσεων που έχουν ήδη οι μαθητές. Δηλαδή πολλές νέες έννοιες οικοδομούνται σε προϋπάρχουσες έννοιες ή εμπειρίες, αφού οι μαθητές πριν ακόμα διδαχθούν κάποιο αντικείμενο διαθέτουν ήδη κάποιες γνώσεις ή εμπειρίες σχετικά με αυτό. Ο σχεδιασμός των διδακτικών στρατηγικών που θα πρέπει να ακολουθηθούν έτσι ώστε να διευκολυνθούν οι μαθητές στην πρόσκτηση νέων γνώσεων θα πρέπει να λαμβάνει υπόψη του τις προϋπάρχουσες γνώσεις των μαθητών στο γνωστικό αντικείμενο της διδασκαλίας. Για να γίνει όμως αυτό θα πρέπει πρώτα να εντοπιστούν οι γνώσεις αυτές.

Στη βιβλιογραφία υπάρχουν λίγες έρευνες σχετικά με τις προϋπάρχουσες γνώσεις σε προγραμματιστικές/αλγοριθμικές έννοιες (Kolikant, 2005; Lewandowski, 2007; McCartney, 2009) και αναφέρονται μόνο σε φοιτητές μαθημάτων προγραμματισμού υπολογιστών. Όσον

αφορά τη διερεύνηση των προϋπαρχουσών γνώσεων στους αλγόριθμους ταξινόμησης έχουν διενεργηθεί μόνο δυο έρευνες από την ομάδα του “Common sense computing” (Chen et al., 2007; Simon et al. 2006) σε φοιτητές. Για μαθητές δεν υπάρχει αντιστοιχη έρευνα διεθνώς από όσο γνωρίζουμε εκτός από μια έρευνα που έγινε πρόσφατα σε μαθητές Λυκείου (Βραχνός & Τζιμογιάννης, 2016). Τα αποτελέσματα της έρευνας της ομάδας του “Common Sense Computing” είναι αρκετά διαφορετικά από τα αποτελέσματα της παραπάνω έρευνας η οποία έγινε σε μαθητές και αυτό οφείλεται κυρίως στο υπολογιστικό μοντέλο που “επιβλήθηκε” έμμεσα στους μαθητές.

Στην εργασία αυτή παρουσιάζουμε τα αποτελέσματα μιας αντιστοιχης έρευνας που έγινε σε 180 μαθητές των δυο πρώτων τάξεων ενός Πειραματικού Γυμνασίου. Αποτελεί την πρώτη και μοναδική μέχρι στιγμής έρευνα διεθνώς από όσο γνωρίζουμε, για τη διερεύνηση των προϋπαρχουσών γνώσεων μαθητών Γυμνασίου στους αλγόριθμους ταξινόμησης.

Στους μαθητές του δείγματος δόθηκε ένα πρόβλημα με τέτοιο τρόπο ώστε να υποχρεωθούν να περιγράψουν τον αλγόριθμο που θα ακολουθούσαν με όσο το δυνατόν πιο αναλυτικό τρόπο. Η συντριπτική πλειοψηφία των μαθητών χρησιμοποίησε τους αλγόριθμους εισαγωγής και επιλογής όπως ακριβώς συνέβη στην αντιστοιχη έρευνα που διενεργήθηκε σε μαθητές Λυκείου. Ωστόσο αρκετοί μαθητές σε αντίθεση με τους μαθητές Λυκείου δεν κατάφεραν να διατυπώσουν έναν αλγόριθμο για την επίλυση του προβλήματος.

Επίσης ενώ οι μαθητές Λυκείου είχαν χρησιμοποιήσει κατά συντριπτική πλειοψηφία τον αλγόριθμο ταξινόμησης με εισαγωγή, οι επιλογές των μαθητών Γυμνασίου μοιράστηκαν μεταξύ του αλγόριθμου με εισαγωγή και αυτού με επιλογή. Μια πιθανή εξήγηση για αυτό είναι ότι οι μαθητές του Γυμνασίου έχτισαν διαφορετική αναπαράσταση για τη ροή των δεδομένων από αυτή των μαθητών Λυκείου.

Τα αποτελέσματα της έρευνας θεωρούμε ότι είναι χρήσιμα για τη διάρθρωση του προγράμματος σπουδών εισαγωγικών μαθημάτων αλγορίθμων και προγραμματισμού στην ενότητα της ταξινόμησης, αφού μπορούν να χρησιμοποιηθούν για την επιλογή και την διδακτική προσέγγιση των κατάλληλων αλγορίθμων ταξινόμησης που μπορούν να διδαχθούν οι μαθητές.

## Θεωρητικό Πλαίσιο

Στη βιβλιογραφία υπάρχουν δυο μόνο έρευνες (Chen et al., 2007; Simon et al., 2006) σχετικά με τις προϋπαρχουσες γνώσεις των μαθητών σε αλγόριθμους ταξινόμησης, από την ομάδα του “Common Sense Computing”. Οι ερευνητές αυτής της ομάδας μελετούν τις αντιλήψεις των μαθητών σε διάφορες έννοιες της πληροφορικής, όπως πολυπλοκότητα, λογική και άλλα. Στις δυο αυτές έρευνες διερευνήθηκαν οι προϋπαρχουσες γνώσεις πρωτοετών φοιτητών στους αλγόριθμους ταξινόμησης. Οι ερευνητές ζήτησαν από τους φοιτητές να περιγράψουν τον τρόπο με τον οποίο θα ταξινομούσαν μια σειρά από 10 αριθμούς σε αύξουσα σειρά. Τα αποτελέσματα έδειξαν ότι οι περισσότεροι φοιτητές αντιμετώπισαν τους αριθμούς ως συμβολοσειρές από ψηφία και έκαναν συγκρίσεις ψηφίο-ψηφίο, ομαδοποιώντας τους αριθμούς σε δεκάδες, εκατοντάδες κλπ., χτίζοντας ένα ευρετήριο με βάση το πλήθος των ψηφίων. Η δεύτερη πιο δημοφιλής μέθοδος ήταν ο αλγόριθμος επιλογής (selection sort). Άλλοι αλγόριθμοι που χρησιμοποιήθηκαν ήταν ο αλγόριθμος εισαγωγής (insertion sort) και σε πολύ λίγες περιπτώσεις ο αλγόριθμος ευθείας ανταλλαγής (bubble sort). Επίσης το 30% των φοιτητών δεν μπόρεσαν να περιγράψουν σωστά έναν αλγόριθμο ταξινόμησης ενώ ενδιαφέρον παρουσιάζει το γεγονός ότι οι φοιτητές που είχαν διδαχθεί προγραμματισμό (με τη γλώσσα Java) τα πήγαν χειρότερα από αυτούς που δεν είχαν καμία εμπειρία ή εκπαίδευση στον προγραμματισμό.

Τα βασικά ερευνητικά ερωτήματα αυτής της έρευνας είναι τα παρακάτω:

- Με ποια μέθοδο / αλγόριθμο ταξινομούν οι μαθητές μια σειρά από αντικείμενα;
- Πώς περιγράφουν οι μαθητές τα βήματα ενός αλγορίθμου;
- Πώς περιγράφουν/κωδικοποιούν μια επαναληπτική διαδικασία;
- Διαφέρουν οι αναπαραστάσεις των μαθητών Γυμνασίου για τη διαδικασία της ταξινόμησης από αυτές των μαθητών Λυκείου;

Επίσης ένα άλλο θέμα που εξετάζει αυτή η έρευνα είναι κατά πόσο ο αλγόριθμος ταξινόμησης της ευθείας ανταλλαγής που διδάσκεται στα περισσότερα εισαγωγικά μαθήματα προγραμματισμού προκύπτει από τις εμπειρίες των μαθητών. Η υπόθεσή μας είναι ότι ένας από τους λόγους που οι μαθητές δυσκολεύονται στην κατανόηση του αλγορίθμου ευθείας ανταλλαγής (Βραχνός & Τζιμογιάννης, 2014) είναι ότι δεν προκύπτει άμεσα από την εμπειρία, σε αντίθεση με άλλους αλγορίθμους όπως ο αλγόριθμος με εισαγωγή ή με επιλογή.

### Μεθοδολογία της Έρευνας

Τα αποτελέσματα της έρευνας που θα παρουσιάσουμε στη συνέχεια είναι αρκετά διαφορετικά από τα αποτελέσματα των παραπάνω ερευνών κυρίως επειδή μέσω της εκφώνησης του προβλήματος που τέθηκε στους μαθητές ορίστηκε με έμμεσο τρόπο το υπολογιστικό μοντέλο που μπορούσαν να χρησιμοποιήσουν σε αντίθεση με τις παραπάνω έρευνες, που παρείχαν μεγαλύτερη ελευθερία έκφρασης.

Αποτέλεσμα αυτής της ελευθερίας που δόθηκε στους μαθητές ήταν, πολλές από τις απαντήσεις να μην “δείχνουν” κάποιον γνωστό αλγόριθμο ταξινόμησης, ή κάποιον αλγόριθμο ταξινόμησης που να είναι αποδεκτός από το υπολογιστικό μοντέλο που χρησιμοποιούμε. Για παράδειγμα ένας μαθητής μπορεί να απαντήσει ότι με μια ματιά βρίσκει αμέσως το μικρότερο στοιχείο, χωρίς να περιγράψει αναλυτικά τα βήματα που ακολούθησε για να το πετύχει.

Το σύνολο των επιτρεπών εντολών που μπορούν να χρησιμοποιηθούν ορίζει τη νοητή μηχανή που προγραμματίζουμε για την επίλυση του προβλήματος. Η πρώτη αναφορά στην έννοια της νοητής μηχανής (notional machine) έγινε από τον Du Boulay (1986). Σύμφωνα με τον Du Boulay η νοητή μηχανή περικλείει το σύνολο των γενικών χαρακτηριστικών που έχει η μηχανή στην οποία προγραμματίζουμε τη λύση ενός προβλήματος.

Στη δική μας έρευνα η νοητή μηχανή την οποία προγραμματίζουν οι μαθητές προκύπτει έμμεσα από την εκφώνηση του προβλήματος που τους δόθηκε:

Σας δίνουν 20 κλειστούς φακέλους. Σε κάθε έναν από αυτούς υπάρχει ένα χαρτί με έναν αριθμό γραμμένο πάνω σε αυτό. Σας ζητείται να βάλετε τους φακέλους σε αύξουσα σειρά (δηλαδή από το μικρότερο στο μεγαλύτερο) με βάση τα νούμερα αυτά.

Μπορείτε να ανοίξετε κάθε φάκελο όσες φορές θέλετε αλλά μπορείτε να έχετε ανοικτούς την ίδια στιγμή το πολύ δύο φακέλους και να βλέπετε τι έχουν μέσα. Μετά τους κλείνετε πάλι και τους τοποθετείτε όπου εσείς κρίνετε.

Να περιγράψετε όσο πιο αναλυτικά μπορείτε τον αλγόριθμο με τον οποίο θα βάλετε τους φακέλους στη σειρά

Ορίζουμε μια νοητή μηχανή στην οποία οι φάκελοι παίζουν το ρόλο των μεταβλητών και οι αριθμοί το περιεχόμενό τους. Το άνοιγμα ενός φακέλου μοντελοποιεί την προσπέλαση στη μνήμη της αντίστοιχης μεταβλητής. Επειδή κάθε φορά μπορεί να συγκριθούν μόνο δύο στοιχεία, θέσαμε τον περιορισμό των δύο ανοικτών φακέλων για κάθε βήμα του αλγορίθμου, ώστε να υποχρεώσουμε τους μαθητές να συγκρίνουν μόνο δύο αντικείμενα κάθε φορά.

Οι βασικοί κανόνες-περιορισμοί που θέσαμε έμμεσα στους μαθητές είναι οι εξής:

- Την ίδια στιγμή μπορούν να έχουν ανοικτούς το πολύ δύο φακέλους

- Αφού δουν το περιεχόμενο ενός φακέλου τον ξανακλείνουν και τον τοποθετούν πάλι στη θέση του, εκτός αν θέλουν να τον μετακινήσουν σε άλλο σημείο.

Επίσης, οι μαθητές είχαν τη δυνατότητα να τοποθετούν τους φακέλους όπου ήθελαν στο τραπέζι. Αυτή η απόφαση αποδείχθηκε κρίσιμη διότι όπως θα δούμε στην ανάλυση των αποτελεσμάτων, πολλοί μαθητές επινόησαν αναδρομικούς αλγόριθμους όπως η Merge Sort οι οποίοι χρειάζονται περισσότερες από μια δομές δεδομένων. Είναι συνηθισμένο όταν βάζουμε στη σειρά μια ομάδα από αντικείμενα να τοποθετούμε τα ταξινομημένα ξεχωριστά από τα υπόλοιπα και όχι δίπλα τους. Αυτό το είδαμε και στην παρούσα έρευνα αλλά και στην έρευνα που έγινε σε μαθητές Λυκείου (Βραχνός & Τζιμογιάννης 2016). Η τοποθέτηση των ταξινομημένων αντικειμένων σε ξεχωριστή περιοχή υποδηλώνει τη χρήση μιας επιπλέον βοηθητικής δομής δεδομένων, αν προσπαθήσουμε να μεταφράσουμε το πρόγραμμα της νοητής μηχανής σε μια συμβατική γλώσσα προγραμματισμού.

Η έρευνα έλαβε χώρα τον Φεβρουάριο του 2017 σε ένα δείγμα 80 μαθητών της Α΄ τάξης και 100 μαθητών της Β΄ τάξης του Ζαννείου Πειραματικού Γυμνασίου, συνολικά 180 μαθητών. Οι μαθητές είχαν διδαχθεί κάποιες βασικές έννοιες προγραμματισμού στο περιβάλλον Scratch, αλλά δεν είχαν ακόμα φτάσει στη δομή επανάληψης ούτε σε λίστες. Επίσης δεν είχαν έρθει σε επαφή με κανέναν αλγόριθμο ταξινόμησης.

Για τη συμπλήρωση του ερωτηματολογίου από τους μαθητές διατέθηκε μια διδακτική ώρα. Το ερωτηματολόγιο ήταν επώνυμο, ώστε να έχουμε στη συνέχεια τη δυνατότητα να διερευνήσουμε ακόμη περισσότερο, μέσω συνεντεύξεων, τις ιδέες των μαθητών που θα παρουσίαζαν ερευνητικό ενδιαφέρον. Περιλάμβανε το πρόβλημα με τους φακέλους που αναλύθηκε παραπάνω. Ζητήθηκε από τους μαθητές να επινοήσουν και να διατυπώσουν έναν αλγόριθμο ταξινόμησης σε όποια μορφή ήθελαν και με όσο πιο αναλυτικό τρόπο γινόταν.

## Ανάλυση των αποτελεσμάτων

Στον Πίνακα 1 παρουσιάζονται τα συγκεντρωτικά αποτελέσματα της έρευνας. Οι περισσότεροι μαθητές (53.4%) χρησιμοποίησαν τους αλγόριθμους επιλογής και εισαγωγής. Τα αποτελέσματα αυτά συμφωνούν σε κάποιο βαθμό με τα αποτελέσματα της αντίστοιχης έρευνας που έγινε σε μαθητές Λυκείου (Βραχνός & Τζιμογιάννης 2016) αλλά και με τις έρευνες των (Chen et. al., 2007; Simon et. al., 2006).

Πίνακας 1: Απαντήσεις των μαθητών Γυμνασίου ανά τάξη

α/α	Απάντηση (N=180)	Κατανομή Απαντήσεων ανά τάξη			
		Α΄ Τάξη	Β΄ Τάξη	Σύνολο	Ποσοστό (%)
1	Δεν απάντησαν	1	7	8	4.4
2	Ημιτελής περιγραφή αλγορίθμου	30	35	65	36.1
3	Ταξινόμηση Εισαγωγής	22	26	48	26.7
4	Ταξινόμηση Επιλογής	25	23	48	26.7
5	Ταξινόμηση Συγχώνευσης	2	8	10	5.6
6	Ταξινόμηση Φυσαλίδας	0	1	1	0.5

Ωστόσο υπάρχουν και σημαντικές διαφορές που θα αναλύσουμε στη συνέχεια και οφείλονται στη μεθοδολογία που ακολουθήσαμε.

Στη συνέχεια δίνουμε κάποιες χαρακτηριστικές απαντήσεις μαθητών για κάθε περίπτωση. Να σημειωθεί ότι στις απαντήσεις των μαθητών έχουμε παρέμβει μόνο ως προς τη διόρθωση των ορθογραφικών λαθών που υπήρχαν.

### **Ταξινόμηση Εισαγωγής (Insertion sort)**

Ο αλγόριθμος ταξινόμησης με εισαγωγή είναι ο ένας από τους δυο αλγόριθμους που χρησιμοποίησαν οι περισσότεροι μαθητές. Μια χαρακτηριστική συνοπτική περιγραφή μαθητή είναι η παρακάτω:

*Χωρίζω τους φακέλους σε δυο ομάδες, αυτούς που είναι ταξινομημένοι και αυτούς που δεν είναι. Κάθε φορά παίρνω έναν από τους μη ταξινομημένους και τον βάζω στην σωστή θέση στους ταξινομημένους.*

Στην παραπάνω απάντηση φαίνεται ξεκάθαρα η τάση που έχουν οι μαθητές να ομαδοποιούν τα αντικείμενα κατά την ταξινόμηση σε δύο σύνολα, τα ταξινομημένα και τα μη ταξινομημένα. Διαδοχικά αφαιρούν ένα στοιχείο από το αρχικό σύνολο και το εισάγουν στο ήδη ταξινομημένο σύνολο αφού πρώτα βρουν τη θέση του.

Μια άλλη χαρακτηριστική απάντηση είναι η παρακάτω:

*Ανοίγω τον πρώτο και τον δεύτερο φάκελο. Τους συγκρίνω, βάζω τον μικρότερο αριστερά και τον μεγαλύτερο δεξιά. Ανοίγω τον επόμενο φάκελο και τον συγκρίνω με τον μικρότερο που έβαλα στην άκρη. Αν είναι μικρότερος από αυτόν τον βάζω αριστερά του ενώ αν είναι μεγαλύτερος δεξιά του. Ανοίγω τον επόμενο φάκελο, αν είναι μικρότερος από τον μικρότερο τον βάζω πρώτο ενώ αν είναι μεγαλύτερος από τον μεγαλύτερο τελευταίο, αλλιώς τον βάζω στην μέση. Συνεχίζω με τους υπόλοιπους φακέλους.*

Η δομή αυτής της απάντησης είναι και η πιο συνηθισμένη που είδαμε. Δηλαδή οι μαθητές περιγράφουν τα 2-3 πρώτα βήματα και τελειώνουν με μια φράση όπως “κ.ο.κ”, “ομοίως” κλπ. Δεν δίνουν δηλαδή τον αλγόριθμο στην μορφή που γνωρίζουμε χρησιμοποιώντας τις γνωστές δομές ακολουθίας, επιλογής και επανάληψης, αλλά με ελεύθερο κείμενο στο οποίο ωστόσο είναι διακριτή η βασική ιδέα του αλγορίθμου.

Επίσης είναι φανερό ότι η μαθήτρια που έδωσε την παραπάνω απάντηση έχει χωρίσει τους φακέλους σε δύο ομάδες, τους ταξινομημένους και τους μη ταξινομημένους. Κάθε φορά παίρνει έναν φάκελο από το αρχικό σύνολο και το τοποθετεί στη σωστή θέση μεταξύ των ταξινομημένων φακέλων, μέχρι να τους τοποθετήσει όλους. Δεν το αναφέρει ρητά γιατί το θεωρεί αυτονόητο. Στη συνέντευξη όμως που ακολούθησε ανέφερε ότι ακολούθησε αυτή την προσέγγιση.

### **Ταξινόμηση Επιλογής (Selection sort)**

Ο άλλος εξίσου δημοφιλής αλγόριθμος που επινοήθηκε από τους μαθητές για την επίλυση του προβλήματος της ταξινόμησης ήταν ο αλγόριθμος με επιλογή (selection sort). Ενώ στους μαθητές Λυκείου ο αλγόριθμος αυτός ήταν δεύτερος στις επιλογές τους με μεγάλη διαφορά από τον πρώτο που ήταν ο αλγόριθμος με εισαγωγή, στους μαθητές Γυμνασίου χρησιμοποιήθηκε όσες και ο αλγόριθμος με εισαγωγή. Παρακάτω δίνουμε μερικές χαρακτηριστικές απαντήσεις μαθητών:

*Αρχικά θα πάρουμε 2 φακέλους και να τους συγκρίνουμε, τον μικρότερο τον κρατάμε, ενώ τον άλλον τον κλείνουμε. Αφού λοιπόν κάνουμε αυτή τη διαδικασία σε όλους τους φακέλους, βάζουμε τον μικρότερο στην αρχή. Με την ίδια διαδικασία βρίσκουμε τον αμέσως μικρότερο και τον βάζουμε δεύτερο. Αυτό θα γίνει και στους υπόλοιπους φακέλους με βάση την προηγούμενη διαδικασία.*

Από την παραπάνω περιγραφή φαίνεται η απλότητα του αλγορίθμου με επιλογή αφού συνίσταται σε μια επαναλαμβανόμενη εύρεση μεγίστου κάτι που επίσης καθιστά την υλοποίησή του αρκετά απλή.

Η παραπάνω απάντηση περιγράφει αρκετά καλά τον αλγόριθμο επιλογής. Ωστόσο παρατηρούμε ότι και εδώ οι μαθητές αποθηκεύουν τους ταξινομημένους αριθμούς σε ξεχωριστή δομή. Αν για παράδειγμα υλοποιήσουμε τον παραπάνω αλγόριθμο με πίνακες θα χρειαστούμε δύο πίνακες. Έναν για τα αρχικά δεδομένα και έναν για τους ταξινομημένους αριθμούς. Οι μαθητές έδωσαν και μερικές παραλλαγές της ταξινόμησης επιλογής. Σε μια από

αυτές υπολογίζουν σε κάθε βήμα και το μέγιστο και το ελάχιστο στοιχείο του πίνακα και τοποθετούν το ένα στην αρχή και το άλλο στο τέλος του νέου πίνακα.

### Ταξινόμηση Συγχώνευσης (Merge sort)

Οι περιγραφές των μαθητών που πρότειναν αυτόν τον αλγόριθμο δεν είναι ιδιαίτερα ακριβείς, κυρίως λόγω της δυσκολίας περιγραφής του αναδρομικού βήματος και της συγχώνευσης των ταξινομημένων λιστών. Ακολουθούν δυο ενδεικτικές απαντήσεις μαθητών:

*Για να μπορέσω να βάλω και τους 20 φακέλους σε αύξουσα σειρά, θα ανοίξω τους πρώτους δυο και τους χωρίζω σε μικρό και μεγάλο και συνεχίζω να κάνω το ίδιο και με τους υπόλοιπους 18. Αφού, τους έχω ξεχωρίσει σε μικρούς και μεγάλους, ανοίγω δυο φακέλους από τους μικρότερους αριθμούς και τους χωρίζω σε μεγάλους και μικρούς. Το ίδιο κάνω με τους μεγαλύτερους. Μετά αρχίζω και τοποθετώ τους μικρότερους, πιο μετά τους μεγαλύτερους κ.ο.κ.*

*Εγώ θα πάρω 5 φακέλους θα τους ανοίξω έναν-έναν (και θα τους κλείνω φυσικά) και θα τους βάλω στη σειρά. Μετά θα πάρω άλλους 5 και θα τους βάλω στη σειρά. Θα φτιάξω 4 σειρές από 5 φακέλους. Μετά 2 σειρές των 10 φακέλων, μετά θα προσθέσω άλλους 5 και μετά άλλους 5.*

Οι παραπάνω περιγραφές παραπέμπουν σε αλγόριθμο διαιρεί και βασιλεύει, αφού το πρόβλημα διαιρείται διαδοχικά σε απλούστερα προβλήματα. Οι μαθητές δυσκολεύονται πολύ στην περιγραφή της ένωσης των δύο ταξινομημένων λιστών. Ωστόσο φαίνεται ότι οι μαθητές έχουν κατανοήσει ότι η διαίρεση του προβλήματος οδηγεί σε άλλα απλούστερα.

Κάποιοι μαθητές διαχώρισαν τους αριθμούς με κριτήριο το πλήθος των ψηφίων (μονάδες, δεκάδες, εκατοντάδες). Αυτή την προσέγγιση ακολούθησαν οι περισσότεροι φοιτητές στις δυο διεθνείς έρευνες (Chen et al., 2007; Simon et al. 2006) της ομάδας “common sense computing”.

*Πρώτα θα χωρίσω τους αριθμούς σε μονοψήφιους, διψήφιους, τριψήφιους κλπ. Μετά ανά ομάδα τους βάζω σε αύξουσα σειρά και μετά ενώνω τις ομάδες, βάζοντας μπροστά τους μονοψήφιους, μετά τους διψήφιους, κλπ.*

Τέλος θα πρέπει να σημειώσουμε ότι από τους 180 μαθητές μόνο ένας έδωσε αλγόριθμο ο οποίος μοιάζει με τον αλγόριθμο ευθείας ανταλλαγής (φυσάλιδας). Αυτό επιβεβαιώνει τα ευρήματα άλλων εργασιών (Βραχνός & Τζιμογιάννης, 2016, 2014), σύμφωνα με τα οποία η ταξινόμηση φυσάλιδας δεν συνάδει με τις αρχές του εποικοδομισμού, αφού είναι πολύ δύσκολο όχι μόνο να χτιστεί σε προϋπάρχουσα γνώση των μαθητών αλλά και να συνδεθεί με εμπειρίες τους από την καθημερινή ζωή. Αυτός ίσως να είναι ο βασικότερος λόγος που οι μαθητές δυσκολεύονται τόσο πολύ στην κατανόηση της λειτουργίας του.

### Συζήτηση

Στον Πίνακα 2 παρουσιάζονται τα αποτελέσματα της έρευνας συγκριτικά με αυτά της αντίστοιχης έρευνας που είχε γίνει σε 106 μαθητές Λυκείου.

Οι μαθητές Λυκείου χρησιμοποίησαν τον αλγόριθμο εισαγωγής πολύ περισσότερο από αυτόν της επιλογής. Για κάποιο λόγο οι μαθητές Λυκείου έκαναν την υπόθεση ότι τα δεδομένα έρχονται σταδιακά και δεν τα έχουν όλα στη διάθεσή τους από την αρχή. Αυτή η σκέψη οδηγεί στον αλγόριθμο της εισαγωγής. Ωστόσο η διαφοροποίηση αυτή είναι αρκετά σημαντική και χρήζει περισσότερης διερεύνησης.

Πίνακας 2: Απαντήσεις των μαθητών ανά τύπο σχολείου

α/α	Απάντηση	Απαντήσεις ανά τύπο σχολείου (ποσοτά %)	
		Γυμνάσιο N=180	Λύκειο N=106
1	Δεν απάντησαν	4.4	2.4

2	Ημιτελής περιγραφή αλγορίθμου	36.1	10
3	Ταξινόμηση Εισαγωγής	26.7	50
4	Ταξινόμηση Επιλογής	26.7	15
5	Ταξινόμηση Συγχώνευσης	5.6	14.1
6	Ταξινόμηση Φυσαλίδας	0.5	3.8
7	Γρήγορη Ταξινόμηση	0	4.7

Μια άλλη σημαντική διαφορά είναι ότι το 4.4% των μαθητών του Γυμνασίου δεν απάντησε καθόλου ενώ το 36% έδωσε απαντήσεις οι οποίες είχαν σοβαρές ελλείψεις ή λάθη και δεν μπορούσαν να κατηγοριοποιηθούν σε κάποιον γνωστό αλγόριθμο ταξινόμησης. Το ποσοστό αυτό (40%) είναι πολύ μεγάλο δεδομένου ότι το αντίστοιχο ποσοστό στο Λύκειο ήταν μόλις 12.4%. Είναι φανερό ότι οι μαθητές Γυμνασίου δυσκολεύτηκαν πολύ περισσότερο από τους μαθητές Λυκείου στη διατύπωση του αλγορίθμου κάτι που ήταν αναμενόμενο. Επίσης οι μαθητές Λυκείου είχαν παρακολουθήσει περισσότερα μαθήματα προγραμματισμού στη σχολική τους ζωή από ότι οι μαθητές γυμνασίου και έχουν σίγουρα μεγαλύτερη ευχέρεια στον γραπτό λόγο.

### Συμπεράσματα

Σύμφωνα με την θεωρία του εποικοδομητισμού η νέα γνώση χτίζεται πάνω σε προϋπάρχουσα γνώση και στις εμπειρίες που έχουν οι μαθητές από την καθημερινότητά τους. Για αυτό η μελέτη της προϋπάρχουσας γνώσης των μαθητών για ένα αντικείμενο όπως οι αλγόριθμοι ταξινόμησης έχει πολύ μεγάλη σημασία για το σχεδιασμό των διδακτικών στρατηγικών από τους εκπαιδευτικούς.

Στην έρευνα που διενεργήσαμε τέθηκε στους μαθητές ένα πρόβλημα το οποίο ήταν σχεδιασμένο με τέτοιο τρόπο ώστε οι επιτρεπτές κινήσεις που μπορούσαν να χρησιμοποιηθούν να είναι όσο το δυνατόν πιο κοντά στο μοντέλο υπολογισμού των περισσότερων γλωσσών προγραμματισμού χωρίς όμως να περιορίζεται η φαντασία τους.

Περισσότεροι από τους μισούς μαθητές χρησιμοποίησαν δυο αλγόριθμους ταξινόμησης, τον αλγόριθμο επιλογής και τον αλγόριθμο εισαγωγής, οι οποίοι από ότι φαίνεται προκόπουν πιο εύκολα από την καθημερινή εμπειρία. Ο αλγόριθμος εισαγωγής ήταν ο αλγόριθμος που είχε χρησιμοποιηθεί περισσότερο από μαθητές Λυκείου σε αντίστοιχη έρευνα. Μια σημαντική διαφορά που υπήρξε σε σχέση με την αντίστοιχη έρευνα σε μαθητές Λυκείου ήταν ότι πολλοί μαθητές Γυμνασίου έδωσαν ημιτελή ή μη ορθά τμήματα αλγορίθμων. Ένα πολύ μικρό ποσοστό μαθητών επινόησε τον αλγόριθμο συγχώνευσης επιλέγοντας διάφορα κριτήρια για τον διαχωρισμό των αριθμών, όπως συνέβη και στην περίπτωση των μαθητών του Λυκείου. Η σημαντικότερη διαφορά μεταξύ των δυο ερευνών ήταν ότι στην περίπτωση του Λυκείου ο αλγόριθμος της εισαγωγής ήταν ο επικρατέστερος με μεγάλη διαφορά ενώ στο Γυμνάσιο είχε τον ίδιο ποσοστό με τον αλγόριθμο επιλογής.

Τα αποτελέσματα αυτά επαληθεύουν την υπόθεση ότι ο αλγόριθμος της ευθείας ανταλλαγής δεν προκόπει φυσικά από τις εμπειρίες των μαθητών οπότε δεν μπορεί να χτιστεί πάνω σε προϋπάρχουσα γνώση. Για αυτό κατά τη διδασκαλία του παρουσιάζονται πολλά προβλήματα (Geller & Dios, 1998; Astrachan, 2003; Simon et al., 2006; Βραχνός & Τζιμογιάννης, 2014). Επίσης επιβεβαιώνεται η τάση των μαθητών να καταχωρούν τα ταξινομημένα στοιχεία σε μια νέα βοηθητική δομή ξεχωρίζοντάς τα από τα μη ταξινομημένα, κάτι που ήταν πολύ εύκολο στο υπολογιστικό μοντέλο της νοητής μηχανής που ορίσαμε.

### Αναφορές

Astrachan, O. (2003). Bubble sort: an archaeological algorithmic analysis. *Proceedings of the 34th SIGCSE technical symposium on Computer science education (SIGCSE '03)*. (pp. 1-5). NY: ACM.

- Ben-Ari, M. (2001). Constructivism in computer science education. *Journal of Computers in Mathematics and Science Teaching*, 20(1):45-73.
- de Raadt, M. (2007). A review of Australian investigations into problem solving and the novice programmer. *Computer Science Education*, 17(3), 201-213.
- du Boulay, B. (1986). Some Difficulties of Learning to Program. In E. Soloway & J. C. Spohrer (eds.), *Studying the Novice Programmer* (pp. 283-299). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Geller, J., & Dios, R. (1998). A low-tech, hands-on approach to teaching sorting algorithms to working students, *Computers & Education*, 31(1), 89-103.
- Kolikant, B.-D., Y. (2005). Students' alternative standards for correctness. In *Proceedings of the 1st International Computing Education Research Workshop*, ACM, New York, 37-43.
- McCartney, R., Bouvier, J., D., Tzu-Yi Chen, Gary Lewandowski, Kate Sanders, Beth Simon, and Tammy VanDeGrift. (2009). Commonsense computing (episode 5): algorithm efficiency and balloon testing. In *Proceedings of the fifth international workshop on Computing education research workshop (ICER '09)*. ACM, New York, NY, USA, 51-62.
- Lewandowski, G., Dennis J. Bouvier, Robert McCartney, Kate Sanders, and Beth Simon. (2007). Commonsense computing (episode 3): concurrency and concert tickets. In *Proceedings of the third international workshop on Computing education research workshop (ICER '07)*. ACM, New York, NY, USA, 51-62.
- Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education*, 13(2), 137-172.
- Simon, B., Chen, T.-Y., Lewandowski, G., McCartney, R., & Sanders, K. (2006). Commonsense computing: What students know before we teach (Episode 1): Sorting. *Proceedings of the 2nd International Workshop on Computing Education Research* (pp. 29-40). NY: ACM.
- Chen, T.-Y., Lewandowski, G., McCartney, R., Sanders, K., and Simon, B. (2007). Commonsense computing: using student sorting abilities to improve instruction. *SIGCSE Bulletin*. 39(1), 276-280.
- Βραχνός, Ε., & Τζιμογιάννης, Α. (2014). Αναπαραστάσεις μαθητών και φοιτητών για τον αλγόριθμο ταξινόμησης ευθείας ανταλλαγής: Μια ανάλυση βασισμένη στην ταξινόμηση SOLO. *Πρακτικά 7ου Πανελληνίου Συνεδρίου "Διδακτική της Πληροφορικής"*. 3-5 Οκτωβρίου, Ρέθυμνο.
- Βραχνός, Ε., & Τζιμογιάννης, Α. (2016). Προϋπάρχουσες γνώσεις των μαθητών στους αλγόριθμους ταξινόμησης. Ποιον αλγόριθμο επινοούν; *Πρακτικά 8ου Πανελληνίου Συνεδρίου "Διδακτική της Πληροφορικής"*. 23-25 Σεπτεμβρίου, σελ. 61-68, Ιωάννινα.
- 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. (2005). Προς ένα παιδαγωγικό πλαίσιο διδασκαλίας του προγραμματισμού στη δευτεροβάθμια εκπαίδευση. Στο Α. Τζιμογιάννης (επιμ.), *Πρακτικά 3ου Πανελληνίου Συνεδρίου "Διδακτική της Πληροφορικής"* (σ. 99-111). Κόρινθος.