

# Απόψεις Μαθητών για τα Προγραμματιστικά Περιβάλλοντα που Χρησιμοποιούνται για Εισαγωγή στον Προγραμματισμό

Ευάγγελος Κανίδης<sup>1</sup>, Μαργαρίτα Καραλιοπούλου<sup>2</sup>, Γεωργία Μενούνου<sup>3</sup>

<sup>1</sup>Δρ. Σχολ. Σύμβουλος Πληροφορικής, Β' Αθήνας & Ανατολικής Αττικής  
vkanidis@sch.gr

<sup>2</sup>Δρ. Εκπαιδευτικός Πληροφορικής, 2<sup>ο</sup> Γυμνάσιο Παιανίας  
mkaraliop@sch.gr

<sup>3</sup>ΜEd Εκπαιδευτικός Πληροφορικής, ΚΕ ΠΛΗ.ΝΕ.Τ Ανατ. Αττικής  
gmenounou@sch.gr

## Περίληψη

Η παρούσα εργασία ερευνά την αξιοποίηση προγραμματιστικών περιβαλλόντων που χρησιμοποιούνται για την εκμάθηση του προγραμματισμού στην Δευτεροβάθμια εκπαίδευση. Τα περιβάλλοντα αυτά διαχωρίζονται σε προγραμματιστικά περιβάλλοντα με πλακίδια και σε προγραμματιστικά περιβάλλοντα που βασίζονται στη συγγραφή εντολών με τη μορφή κειμένου. Στην εργασία εξετάζονται, τόσο σε θεωρητικό επίπεδο όσο και με ερωτηματολόγιο, οι πρόσφατες εξελίξεις σχετικά με την ευκολία χρήσης κάθε κατηγορίας περιβάλλοντος, αν η χρήση του περιβάλλοντος με πλακίδια διευκολύνει τη μετάβαση σε ένα περιβάλλον προγραμματισμού βασισμένου σε πληκτρολόγηση εντολών καθώς και ποιο από τα δύο περιβάλλοντα θεωρούνται από τους μαθητές πιο κοντά στον «πραγματικό» προγραμματισμό.

**Λέξεις κλειδιά:** Προγραμματιστικά περιβάλλοντα, προγραμματισμός, πλακίδια

## 1. Εισαγωγή

Ο στόχος της εισαγωγής του προγραμματισμού στην Πρωτοβάθμια και Δευτεροβάθμια Εκπαίδευση είναι η ανάπτυξη της υπολογιστικής σκέψης του μαθητή, ώστε να λύνει προβλήματα χρησιμοποιώντας μια αλγοριθμική διαδικασία και να εκφράζει τη λύση αυτή σε μια γλώσσα προγραμματισμού.

Στην παρούσα εργασία θα διακρίνουμε 2 κατηγορίες προγραμματιστικών περιβαλλόντων ανάλογα με τον τρόπο με τον οποίο δομείται το πρόγραμμα:

- Προγραμματιστικά περιβάλλοντα με πλακίδια (Block based programming/ Graphical User Interface/ Drag and drop editing) στα οποία γίνεται χρήση «πλακιδίων» κώδικα, τα οποία συναρμολογούνται για να δημιουργηθεί ένα πρόγραμμα
- Προγραμματιστικά περιβάλλοντα κειμένου (Text-based editing/ Command User Interface) όπου ο χρήστης πρέπει να πληκτρολογήσει τις εντολές του προγράμματος.

Τα προγραμματιστικά περιβάλλοντα με πλακίδια υποστηρίζουν οπτικές γλώσσες προγραμματισμού στις οποίες δεν είναι αναγκαία η τυπική συγγραφή κώδικα. Οι μαθητευόμενοι μπορούν να δημιουργήσουν λειτουργικά προγράμματα με χρήση μόνο του ποντικιού συναρμολογώντας πλακίδια που αντιστοιχούν σε εντολές. Τα πλακίδια παρέχουν οπτικά σήματα στο χρήστη για τον τρόπο και τη θέση που μπορούν να χρησιμοποιηθούν, ανάλογα με το σχήμα τους, το χρώμα τους (το οποίο σχετίζεται με την κατηγορία παρόμοιων πλακιδίων) και τη χρήση ετικέτας επάνω τους με την εντολή την οποία αντιπροσωπεύουν. Τα προγραμματιστικά περιβάλλοντα με πλακίδια είναι ο πιο διαδεδομένος τρόπος για εισαγωγή στον προγραμματισμό και στην επιστήμη των υπολογιστών, καθώς θεωρείται ότι βοηθούν το μαθητή να αναπτύσσει με μη τυπικό τρόπο τις προγραμματιστικές ιδέες (Chetty & Barlow-Jones, 2012). Περιβάλλοντα όπως το Scratch, το Alice, το App inventor και οι δραστηριότητες της «ώρας του κώδικα» του Code.org είναι αρκετά δημοφιλή. Στο τελευταίο περιβάλλον, ο χρήστης καλείται να λύσει μια σειρά από puzzle, ενώνοντας πλακίδια τα οποία είναι τα βασικά στοιχεία της «γλώσσας».

Μια από τις πρώτες γλώσσες που δημιουργήθηκε για την εκμάθηση προγραμματισμού από παιδιά είναι η Logo (Feurzeig et al., 1970· Papert, 1980). Η γλώσσα Logo, αν και οι εντολές της πληκτρολογούνται, εισήγαγε έναν αριθμό από χαρακτηριστικά τα οποία εμφανίζονται στα περιβάλλοντα με πλακίδια, με αξιοσημείωτα την ύπαρξη των εντολών κίνησης, της παρουσίας ενός avatar για να υλοποιήσει αυτές τις εντολές, συντακτικό της γλώσσας κατάλληλο για αρχάριους προγραμματιστές, η έμφαση στην αυτοκαθοδηγούμενη και ενεργή διαδικασία μάθησης καθώς και στη σημασία να δημιουργούν δημόσια, διαμοιράσιμες δημιουργίες, όπως παιχνίδια, ιστορίες και εφαρμογές.

Μερικοί ερευνητές θεωρούν ότι η χρήση πλακιδίων τοποθετεί τα θεμέλια για την τελική μετάβαση σε προγραμματισμό με πληκτρολόγηση εντολών. Αυτό όμως παραμένει ένα ανοικτό ερευνητικό ερώτημα.

Στην εργασία των Καραλιοπούλου και Κανίδη (2016) γίνεται μια επισκόπηση της πρόσφατης βιβλιογραφίας και δίνεται μια περιγραφή των χαρακτηριστικών των περιβαλλόντων με πλακίδια, τα πλεονεκτήματα και μειονεκτήματα αυτών των προγραμματιστικών περιβαλλόντων και συμπεράσματα από την εφαρμογή τους στην τάξη. Επίσης προτείνεται μια νέα προσέγγιση στη χρήση αυτών των περιβαλλόντων.

Η παρούσα εργασία αποτελεί την ερευνητική συνέχεια της προηγούμενης εργασίας. Σκοπό έχει τη διερεύνηση των απόψεων των μαθητών σχετικά με την ταχύτητα και ευκολία συγγραφής προγραμμάτων στα δύο περιβάλλοντα, το πόσο βοηθήθηκαν και αν ήταν εύκολη η μετάβαση από το περιβάλλον με πλακίδια στο περιβάλλον με εντολές καθώς και πιο από τα δύο περιβάλλοντα θεωρούν ότι είναι πιο κοντά σε μια γλώσσα προγραμματισμού.

## **2. Ερευνητικά δεδομένα**

Σε πρόσφατο συνέδριο, το Μάρτιο του 2016 (SIGSE'16), υπήρξε panel με θέμα “Future Directions of Block based Programming” (Brown et al., 2016). Συζητήθηκαν μεταξύ άλλων τα εξής ζητήματα: για ποιο λόγο τα περιβάλλοντα με πλακίδια είναι δημοφιλή, ποια είναι τα προβλήματα που αντιμετωπίζουμε κατά τη χρήση τέτοιων περιβαλλόντων στην τάξη, όπως για παράδειγμα το γεγονός ότι ο προγραμματισμός με πλακίδια δεν θεωρείται «πραγματικός» προγραμματισμός. Με τον όρο «πραγματικός» προγραμματισμός αναφερόμαστε στη συγγραφή προγραμμάτων σε μια εμπορική γλώσσα προγραμματισμού όπως η C, C++, Java, Python κ.λπ. Επίσης ένα σημαντικό ζήτημα που απασχόλησε το panel είναι το «κενό» ανάμεσα στα περιβάλλοντα με πλακίδια και στα περιβάλλοντα με σύνταξη εντολών δηλαδή πως μεταφέρονται έννοιες και πρακτικές που οι μαθητές διδάχτηκαν στα περιβάλλοντα με πλακίδια σε μια συμβατική γλώσσα προγραμματισμού με εντολές. Οι έρευνες έχουν αναφέρει πετυχημένες αλλά και ανεπιτυχείς εννοιολογικές μεταφορές. Το ζήτημα της μετάβασης από το ένα περιβάλλον στο άλλο και της σχέσης των περιβαλλόντων με πλακίδια με τον πραγματικό προγραμματισμό, είχε ήδη απασχολήσει τους ερευνητές.

Στη βάση αυτού του προβληματισμού, οι Powers., Ecott & Hirshfield (2004), χρησιμοποιώντας το προγραμματιστικό περιβάλλον Alice, διαπίστωσαν ότι οι σπουδαστές είχαν προβλήματα στη μετάβαση από το περιβάλλον με πλακίδια σε μια γλώσσα προγραμματισμού όπως η C++ ή η Java. Οι μαθητές δεν αναγνώρισαν τη σημασία της ακριβούς σύνταξης εντολών όταν ήρθε η στιγμή να τις πληκτρολογήσουν. Επίσης τα περιβάλλοντα με πλακίδια θύμιζαν παιχνίδι στα παιδιά και δεν αντιμετωπίστηκαν ως περιβάλλοντα «πραγματικού» προγραμματισμού. Αργότερα οι Parsons και Haden

(2007) ερευνήσαν τη διαδικασία της διδασκαλίας των δομών επιλογής και επανάληψης στο προγραμματιστικό περιβάλλον Alice. Διαπίστωσαν ότι αρκετοί μαθητές δυσκολεύτηκαν να βρουν συσχέτιση μεταξύ της δουλειάς τους στο Alice και τον «πραγματικό» προγραμματισμό. Οι Weintrop και Wilensky (2016) επίσης εξέτασαν τα ζητήματα που πιθανόν θα προκύψουν στη χρήση προγραμματιστικών περιβαλλόντων με πλακίδια, ειδικά στην περίπτωση της Δευτεροβάθμιας εκπαίδευσης. Μερικοί μαθητές θεωρούν ότι τα περιβάλλοντα με πλακίδια είναι μακριά από τον «πραγματικό» προγραμματισμό, και ότι είναι κυρίως για εκπαιδευτικούς σκοπούς. Την άποψη αυτή σύμφωνα με τον DiSalvo (2014), συμμερίζονται και μεγαλύτεροι σε ηλικία σπουδαστές.

Σε έρευνα που πραγματοποίησε ο Lewis (2010) συνέκρινε την αυτοπεποίθηση και την ευχέρεια μαθητών στον προγραμματισμό διδάσκοντας αντίστοιχες δραστηριότητες στους μισούς σε Scratch και στους άλλους σε Logo. Οι μαθητές που εργάστηκαν σε Logo είχαν περισσότερες θετικές απαντήσεις στην πρόταση «Είμαι καλός στο να γράφω προγράμματα στον υπολογιστή». Αργότερα, οι Lewis et al. (2014) θέλησαν να μελετήσουν τις αντιλήψεις των μαθητών ως προς τι θεωρούν ότι είναι σημαντικό για την επιλογή μιας γλώσσας προγραμματισμού. Ένα περιβάλλον με πλακίδια όπως το Scratch μπορεί να είναι πιο εύκολο για τους μαθητές στο να δημιουργήσουν ένα πρόγραμμα αλλά, αν οι μαθητές δε το θεωρούν πραγματικό προγραμματισμό, τότε πιθανόν να μην τους εξάπτει το ενδιαφέρον για το αντικείμενο και να μην αποκτούν εμπιστοσύνη στις προγραμματιστικές τους ικανότητες. Η έρευνα που διεξήγαγαν έδειξε ότι οι μαθητές ανέμεναν ο προγραμματισμός να είναι πιο «πολύπλοκος» και μακριά από παιχνίδια.

Οι Weintrop και Wilensky (2015) διεξήγαγαν έρευνα σε μαθητές Δευτεροβάθμιας εκπαίδευσης. Η πρώτη τους ερευνητική ερώτηση ήταν αν ο προγραμματισμός με πλακίδια είναι πιο εύκολος από τον προγραμματισμό σε περιβάλλον με σύνταξη εντολών. Η σύγκριση έγινε ανάμεσα στο Snap! – προγραμματιστικό περιβάλλον με πλακίδια- και τη Java. Τα αποτελέσματα έδειξαν ότι οι μαθητές θεώρησαν την προσέγγιση με το Snap! πιο εύκολη από τη χρήση της Java. Τα αποτελέσματα αυτά επιβεβαιώνονται από τη μελέτη των Price και Barnes (2015), οι οποίοι συνέκριναν τα αποτελέσματα εργασίας στο προγραμματιστικό περιβάλλον Tiled Grace το οποίο επιτρέπει στο χρήστη να επιλέγει είτε τη διεπαφή με πλακίδια είτε αυτή με πληκτρολόγηση εντολών. Υποστηρίζουν ότι η εργασία σε περιβάλλον με πλακίδια είναι ευκολότερη, γιατί ένας αρχάριος προγραμματιστής συγκεντρώνει την προσοχή του να βρει τον αλγόριθμο για τη λύση του προβλήματος. Επίσης κατέληξαν ότι η χρήση των πλακιδίων μπορεί να βελτιώσει σημαντικά τους αρχάριους προγραμματιστές σε μερι-

κές προγραμματιστικές δραστηριότητες, ειδικά μέσω του αυξημένου χρόνου στην εργασία καθώς και στη γρηγορότερη και πιο συχνή επίτευξη των προγραμματιστικών στόχων.

Το θέμα της μετάβασης από την εκμάθηση προγραμματισμού με Scratch στην αρχή της Δευτεροβάθμιας εκπαίδευσης στη εκμάθηση μιας επαγγελματικής γλώσσας όπως η C# ή η Java στο τέλος της Δευτεροβάθμιας εκπαίδευσης εξέτασαν οι Armoni et al. (2015). Η έρευνα έδειξε ότι η πρότερη εμπειρία των μαθητών στο Scratch βοήθησε να μάθουν πιο προχωρημένα θέματα στην επόμενη βαθμίδα, αν και στο τέλος της εκπαιδευτικής διαδικασίας δεν υπήρχε σημαντική διαφορά στην επίτευξη των στόχων σε σχέση με μαθητές που δεν είχαν διδαχθεί Scratch. Παρόμοια, οι Wolz et al. (2009) παρατήρησαν ότι φοιτητές που είχαν αρχικά διδαχθεί Scratch για έστω μόνο μια εβδομάδα, έκαναν πιο εύκολα τη μετάβαση σε Java ή C.

Σε μια άλλη μελέτη, οι Dorling και White (2015) προτείνανε διάφορες στρατηγικές ώστε να πετύχει η μετάβαση από περιβάλλον με πλακίδια σε περιβάλλον με σύνταξη εντολών. Συγκεκριμένα εργάστηκαν με το Scratch από τη μια και τη Logo και την Python από την άλλη. Κατέληξαν ότι ο συνδυασμός των δυο διαφορετικών τύπων περιβαλλόντων, η παράλληλη χρήση τους, συμβάλει στο να αποκτήσουν οι μαθητές εμπιστοσύνη στις ικανότητες τους, ανεξαρτησία και ενδιαφέρον στην εκμάθηση μιας γλώσσας κειμένου, τόσο στην Πρωτοβάθμια όσο και στη Δευτεροβάθμια εκπαίδευση.

Στην παρούσα έρευνα οι μαθητές εργάστηκαν παράλληλα επαναλαμβάνοντας κάθε άσκηση πρώτα σε ένα περιβάλλον με πλακίδια και κατόπιν σε ένα περιβάλλον κειμένου. Η συγκεκριμένη μέθοδος είναι μια περίπτωση μεταφοράς της γνώσης από ένα πλαίσιο σε ένα άλλο, γνωστή ως mediated transfer (Perkins & Salomon, 1988). Για παράδειγμα ένα πρόγραμμα σε Scratch μπορεί να αντικατοπτριστεί ή να συγκριθεί εντολή προς εντολή με το ίδιο παράδειγμα σε Java (Chetty & Barlow-Jones, 2012).

Οι Dann et al. (2012) αναπτύξανε μια έκδοση του Alice που διευκολύνει στη μεταφορά της εμπειρίας από το Alice στη Java με mediated transfer. Σε έρευνα που κάνανε οι μαθητές στους οποίους έγινε διδασκαλία με mediated transfer πέτυχανε καλύτερη επίδοση από ότι σε αυτούς που δεν είχε χρησιμοποιηθεί κατά τη διδασκαλία η μέθοδος αυτή. Σε νεώτερη έρευνα οι Wagner et al. (2013) επαναλαμβάνοντας ασκήσεις πρώτα με App Inventor, ένα περιβάλλον με πλακίδια, και στη συνέχεια με Java, περιβάλλον με σύνταξη εντολών, κατέληξαν ότι οι μαθητές μπορούσαν να αντιστοιχίσουν τις διαδικασίες με πλακίδια σε διαδικασίες με κείμενο.

### **3. Η έρευνα**

#### **3.1 Ερευνητικά ερωτήματα**

Με βάση το θεωρητικό προβληματισμό που προηγήθηκε προέκυψαν τα ακόλουθα ερευνητικά ερωτήματα:

1. Ποιες είναι οι απόψεις των μαθητών σχετικά με την ταχύτητα και την ευκολία στη συγγραφή προγραμμάτων στο περιβάλλον με πλακίδια και στο προγραμματιστικό περιβάλλον κειμένου;
2. Πόσο δυσκολεύτηκαν οι μαθητές στη μετατροπή του κώδικα που δημιούργησαν με τη βοήθεια του περιβάλλοντος με πλακίδια στο προγραμματιστικό περιβάλλον κειμένου;
3. Σε ποιο βαθμό βοήθησε η χρήση του περιβάλλοντος με πλακίδια για τη δημιουργία ενός προγράμματος, στον προγραμματισμό με τη χρήση του περιβάλλοντος κειμένου;
4. Ποιες οι απόψεις των μαθητών σχετικά με το πιο περιβάλλον είναι πιο κοντά σε μια γλώσσα προγραμματισμού;

#### **3.2 Η μέθοδος**

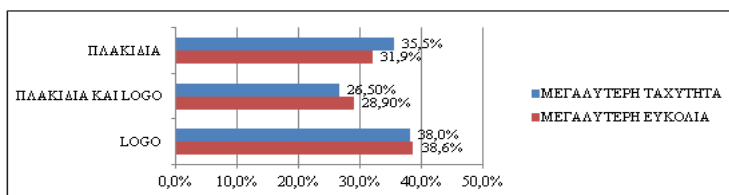
Στην εργασία των Καραλιοπούλου και Κανίδη (2016) προτάθηκε και υλοποιήθηκε η παράλληλη εργασία σε ένα περιβάλλον με πλακίδια και σε ένα αντίστοιχο περιβάλλον κειμένου. Η προτεινόμενη διαδικασία ήταν η εξής: Ο μαθητής καλείται να δώσει λύση σε ένα πρόβλημα σε προγραμματιστικό περιβάλλον με πλακίδια (χρησιμοποιήθηκε το code.org-προγραμματισμός με την Έλσα και την Άννα) και στη συνέχεια καλείται να μεταφέρει τη δομημένη λύση σε μια γλώσσα κειμένου η οποία έχει κοινά χαρακτηριστικά με τη γλώσσα που κρύβεται πίσω από τα πλακίδια (χρησιμοποιήθηκε το περιβάλλον Logo Microworlds Pro). Το περιβάλλον code.org χρησιμοποιήθηκε διότι παρέχει δραστηριότητες κατάλληλες για την κατανόηση της δομής της επανάληψης για αρχάριους προγραμματιστές.

Στην παρούσα εργασία υλοποιήθηκε η πρόταση αυτή σε τέσσερα σχολεία (3 σχολεία της Β' Αθήνας και 1 της Αν. Αττικής) με τη συνεργασία πέντε καθηγητών Πληροφορικής. Οι δραστηριότητες που δόθηκαν στους μαθητές αποσκοπούσαν στην κατανόηση της δομής της επανάληψης και υλοποιήθηκαν σε μια διδακτική ώρα. Στη συνέχεια πραγματοποιήθηκε έρευνα με στόχο να απαντηθούν συγκεκριμένα ερευνητικά ερωτήματα που προέκυψαν από τον θεωρητικό προβληματισμό. Για τη διεξαγωγή

της έρευνας επιλέχθηκε ποσοτική προσέγγιση με χρήση ερωτηματολογίου. Το τελικό δείγμα της έρευνας περιλάμβανε 166 μαθητές τρίτης Γυμνασίου. Για τη στατιστική ανάλυση χρησιμοποιήθηκε το στατιστικό πρόγραμμα SPSS. Το δείγμα των μαθητών που χρησιμοποιήθηκε στην έρευνα δεν ήταν αντιπροσωπευτικό του συνόλου του μαθητικού πληθυσμού της Ελλάδος. Συνεπώς τα αποτελέσματα της έρευνας δεν μπορούν να γενικευτούν για το σύνολο του πληθυσμού.

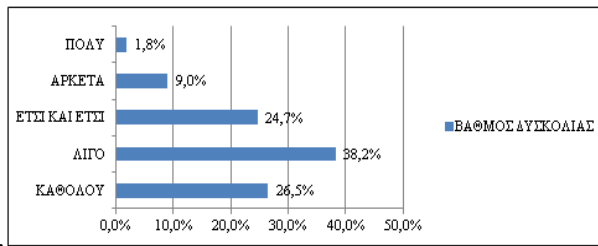
#### 4. Αποτελέσματα

Σε ερωτήσεις σύγκρισης της ταχύτητας δημιουργίας και της ευκολίας συγγραφής ενός προγράμματος στα δύο περιβάλλοντα το 38% των μαθητών (63 μαθητές) απάντησε πως η συγγραφή προγραμμάτων είναι πιο γρήγορη σε περιβάλλοντα με πλακίδια από ότι στο περιβάλλον της γλώσσας Logo. Πολύ κοντά στο ποσοστό αυτό το 35,5 % (59 μαθητές) απάντησε πως πιο γρήγορα δημιουργούν προγράμματα με τη βοήθεια της Logo, ενώ σε ένα σημαντικό ποσοστό 26,5% (49 μαθητές) απάντησε ότι ο χρόνος είναι ίδιος στα δύο περιβάλλοντα (Σχήμα 1). Παρόμοια εικόνα υπήρξε και στην ερώτηση για την ευκολία χρήσης, το 38,6% των μαθητών (64 μαθητές) απάντησε πως η συγγραφή προγραμμάτων είναι πιο εύκολη σε περιβάλλοντα με πλακίδια από ότι στο περιβάλλον της γλώσσας Logo. Πολύ κοντά στο ποσοστό αυτό το 31,9 % (53 μαθητές) απάντησε πως πιο εύκολα δημιουργούν προγράμματα με τη βοήθεια της Logo, ενώ σε ένα σημαντικό ποσοστό 28,9% (48 μαθητές) απάντησε ότι με την ίδια ευκολία δημιουργούν προγράμματα και στα δύο περιβάλλοντα (Σχήμα 1).



Σχήμα 1. Σύγκριση ταχύτητας και ευκολίας στα δύο περιβάλλοντα

Όταν οι μαθητές ρωτήθηκαν αν δυσκολεύτηκαν στη μετατροπή του κώδικα που προέκυψε από το περιβάλλον με πλακίδια στο περιβάλλον της γλώσσας Logo, το 64,5% (107 μαθητές) απάντησε πως δε δυσκολεύτηκαν ή δυσκολεύτηκαν λίγο και τελικά σε σύνολο 166 μαθητών μόνο 18 (10,8%) δυσκολεύτηκαν πολύ ή αρκετά (Σχήμα 2).



**Σχήμα 2.** Βαθμός δυσκολίας κατά τη μετάβαση από το περιβάλλον με πλακίδια σε περιβάλλον Logo

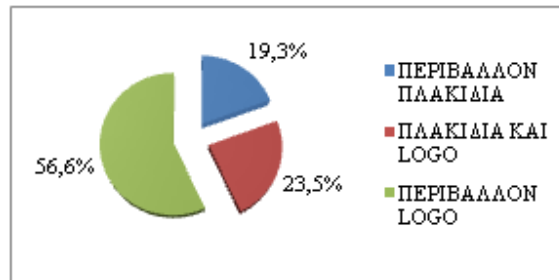
Στην ίδια διαδικασία μετατροπής κώδικα από το ένα περιβάλλον στο άλλο οι μαθητές κλήθηκαν να απαντήσουν αν η πρότερη χρήση του περιβάλλοντος με πλακίδια βοήθησε στη δημιουργία ενός προγράμματος στο περιβάλλον της Logo. Οι απαντήσεις έδειξαν ότι ένα ποσοστό 9% (15 μαθητές) πιστεύει πως δεν βοήθησε καθόλου, ένα ποσοστό 14,5% (24 μαθητές) βοήθησε λίγο, το 20,5% (34 μαθητές) έτσι και έτσι, το 38% (63 μαθητές) αρκετά και τέλος ένα ποσοστό 17,5% (29 μαθητές) ότι βοήθησε πολύ.

Σε ερώτηση σχετικά με την επίδραση που είχε η εμπειρία ενός περιβάλλοντος με πλακίδια στο χρόνο συγγραφής ενός προγράμματος σε Logo σχεδόν οι μισοί μαθητές 52,4% (87 μαθητές) απάντησαν ότι θα χρειάζονταν τον ίδιο χρόνο είτε είχαν εργαστεί σε ένα περιβάλλον προγραμματισμού με πλακίδια είτε όχι. Αντίστοιχα το 16,9% των μαθητών (28 μαθητές) θεώρησε πως θα χρειάζονταν λιγότερο και ένα σημαντικό ποσοστό 30,7% (51 μαθητές) δήλωσε ότι θα χρειαζόταν περισσότερο χρόνο.

Αντίστοιχα σε ερώτηση αν η δημιουργία των προγραμμάτων κατευθείαν στο περιβάλλον Logo θα ήταν ευκολότερη η δυσκολότερη, χωρίς προηγούμενη χρήση του περιβάλλον με πλακίδια, το 42,2% (70 μαθητές) των μαθητών απάντησε πως θα ήταν το ίδιο δύσκολη, ενώ, το 47,6% (79 μαθητές) ότι θα ήταν δυσκολότερη.

Τέλος, ζητήθηκε από τους μαθητές να απαντήσουν ποια γλώσσα πιστεύουν ότι είναι πιο κοντά σε μια γλώσσα προγραμματισμού. Όπως φαίνεται στο Σχήμα 3, πάνω από τους μισούς μαθητές, το 56,6 % (94 μαθητές) πιστεύει ότι πιο κοντά σε μια γλώσσα προγραμματισμού είναι το περιβάλλον της Logo ενώ το 23,5% (39 μαθητές) πιστεύει ότι και τα δύο περιβάλλοντα είναι το ίδιο κοντά και 19,3% (32 μαθητές) ότι πιο κοντά είναι το περιβάλλον με τα πλακίδια.





**Σχήμα 3.** Απόψεις μαθητών για το πιο περιβάλλον είναι πιο κοντά σε μια γλώσσα προγραμματισμού

Αξίζει να σημειωθεί ότι από τους μαθητές που πίστευαν πως πιο κοντά σε μια γλώσσα προγραμματισμού είναι το περιβάλλον της Logo το 47,6% πιστεύει πως θα είχε δυσκολευτεί στη συγγραφή των εντολών, αν δεν είχε χρησιμοποιήσει το περιβάλλον με τα πλακίδια. Το ίδιο όμως ισχύει και για όσους θεωρούν το περιβάλλον με πλακίδια πιο κοντά σε μια γλώσσα προγραμματισμού, όπου το αντίστοιχο ποσοστό είναι 53,1%.

## 5. Συμπεράσματα

Τα συμπεράσματα της έρευνας επιβεβαίωσαν τις προϋπάρχουσες έρευνες. Στο πρώτο ερευνητικό ερώτημα διαπιστώθηκε ότι υπάρχει μια μικρή διαφορά υπέρ των απόψεων ότι ο προγραμματισμός με πλακίδια είναι πιο γρήγορος και πιο εύκολος από τον προγραμματισμό σε περιβάλλον κειμένου. Όσον αφορά στο δεύτερο ερευνητικό ερώτημα, διαπιστώθηκε ότι η σημαντική πλειοψηφία των μαθητών δε δυσκολεύτηκε στη μετατροπή του κώδικα από το ένα περιβάλλον στο άλλο. Επιπρόσθετα, κατά τη διερεύνηση του τρίτου ερευνητικού ερωτήματος διαπιστώθηκε ότι στην πλειοψηφία τους οι μαθητές θεωρούν ότι η πρότερη εργασία τους στο περιβάλλον με πλακίδια βοήθησε στη δημιουργία του προγράμματος σε Logo ενώ το ένα τρίτο των μαθητών θεωρούν πως βοηθήθηκαν ελάχιστα. Το συμπέρασμα αυτό συμφωνεί με την έρευνα των Dorling και White (2015) για τα πλεονεκτήματα της παράλληλης χρήσης περιβαλλόντων καθώς και με αυτά των Dann et al. (2012) και Wagner et al. (2013). Τέλος στο ερευνητικό ερώτημα που αφορά ποιο περιβάλλον οι μαθητές θεωρούν πλησιέστερα σε μια πραγματική γλώσσα προγραμματισμού στην πλειοψηφία τους απάντησαν για τα περιβάλλοντα συγγραφής εντολών. Αυτό επιβεβαιώνει τις έρευνες των Powers et al., (2004), Lewis, (2010) και Lewis et al., (2014), οι οποίοι διαπιστώνουν ότι οι μαθητές θεωρούν «παιχνίδι» τον προγραμματισμό σε περιβάλλον πλακιδίων.

Μελλοντικά οι συγγραφείς θα επιδιώξουν την επαλήθευση των συμπερασμάτων με χρήση ενός ευρύτερου δείγματος, καθώς και τη μελέτη αποτελεσματικότητας μεικτών περιβαλλόντων που συνδυάζουν τα πλακίδια με την εμφάνιση εντολών.

### **Αναφορές**

- Armoni, M., Meerbaum-Salant, O., & Ben-Ari, M. (2015). From Scratch to “Real” Programming. *Trans. Comput. Educ.* 14(4), Article 25.
- Brown, N.C.C., Mönig, J., Bau, A., & Weintrop, D. (2016). Panel: Future Directions of Block-based Programming. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE '16)*. ACM, New York, USA, 315-316.
- Chetty, J., & Barlow-Jones, G. (2012). Bridging the Gap: the Role of Mediated Transfer for Computer Programming. *International Proceedings of Computer Science & Information Technology*, 43.
- Dann, W., Cosgrove, D., Slater, D., Culyba, D., & Cooper, S. (2012). Mediated transfer: Alice 3 to Java. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education (SIGCSE '12)*. ACM, New York, USA, 141-146.
- DiSalvo, B. (2014). Graphical Qualities of Educational Technology: Using Drag-and-Drop and Text-Based Programs for Introductory Computer Science. *IEEE Computer Graphics and Applications*, (6), 12-15.
- Dorling, M. & White D. (2015). Scratch: A Way to Logo and Python. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15)*. ACM, New York, USA, 191-196.
- Feurzeig, W., Papert, S., Bloom, M., Grant, R., & C. Solomon, C. (1970). Programming-languages as a conceptual framework for teaching mathematics. *SIGCUE Outlook* 4, 2 (April 1970), 13-17.
- Lewis, C. M. (2010). How programming environment shapes perception, learning and goals: logo vs. scratch. In *Proceedings of the 41st ACM technical symposium on Computer science education (SIGCSE '10)*. ACM, New York, NY, USA, 346-350.
- Lewis, C., Esper, S., Bhattacharyya, V., Fa-Kaji, N., Dominguez, N. & Schlesinger, A. (2014). Children's perceptions of what counts as a programming language. *J. Comput. Sci. Coll.* 29, 4 (April 2014), 123-133.
- Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, Inc.
- Parsons, D., & Haden, P. (2007). Programming osmosis: Knowledge transfer from imperative to visual programming environments. In *Conference of the National Advisory Committee on Computing Qualifications*.

- Perkins, D. H., & Salomon, G. (1988). Teaching for Transfer. *Educational Leadership*, 46, 22-32.
- Powers, K., Ecott, S. & Hirshfield L. M. (2007). Through the looking glass: teaching CS0 with Alice. In *Proceedings of the 38th SIGCSE technical symposium on Computer science education* (SIGCSE '07). ACM, New York, NY, USA, 213-217.
- Price, T. W. & Barnes, T. (2015). Comparing Textual and Block Interfaces in a Novice Programming Environment. In *Proceedings of the eleventh annual International Conference on International Computing Education Research* (ICER '15). ACM, New York, NY, USA, 91-99.
- Wagner, A., Gray, J., Corley, J. & Wolber, D. (2013). Using app inventor in a K-12 summer camp. In *Proceeding of the 44th ACM technical symposium on Computer science education* (SIGCSE '13). ACM, New York, NY, USA, 621-626.
- Weintrop, D. & Wilensky, U. (2015). To Block or not to Block, That is the Question: Students' Perceptions of Blocks-based Programming. In *Proceedings of the 14th International Conference on Interaction Design and Children*. (IDC '15). ACM New York, NY, USA: 199-208.
- Weintrop, D. & Wilensky, U. (2016). Bringing Blocks-based Programming into High School Computer Science Classrooms. *Paper to be presented at the Annual Meeting of the American Educational Research Association (AERA 2016)*, Washington DC, USA.
- Wolz, U., Leitner, H., Malan, D. & Maloney, J. (2009). Starting with Scratch in CS 1. In *Proceedings of the 40th ACM technical symposium on Computer science education* (SIGCSE '09). ACM, New York, NY, USA, 2-3.
- Καραλιοπούλου, Μ. & Κανίδης, Ε. (2016). Σύγκριση και συνδυασμός προγραμματιστικών περιβαλλόντων για την εισαγωγή στον Προγραμματισμό. *10<sup>o</sup> Πανελλήνιο Συνέδριο Καθηγητών Πληροφορικής*, Ναύπλιο.

### Abstract

Programming environments can be divided into block-based programming and text-based programming environments. In this paper, we present the results of a research on the ease of each category. We also try to answer the questions whether the use of a block-based programming environment ease the transition to a text based programming environment and which of the them is closer to “real” programming, based on the recent bibliography and our results by questionnaire.

**Keywords:** Programming environment, “real” programming, Block based programming, text-based programming