

Από το Scratch στην Python. Μια έρευνα σε μαθητές Γυμνασίου

Ευριπίδης Βραχνός, Σοφία Ντούσκα

Ζάννειο Πειραματικό Γυμνάσιο Πειραιά
{evrachnos, sodouska}@gmail.com

Περίληψη

Η επιλογή της εισαγωγικής γλώσσας προγραμματισμού παίζει σημαντικό ρόλο στη μαθησιακή διαδικασία. Στο γυμνάσιο οι μαθητές έρχονται σε επαφή με περιβάλλοντα όπως η Logo και το Scratch. Ωστόσο πολλοί μαθητές θέλουν να δουν μια πραγματική γλώσσα προγραμματισμού με την οποία να μπορούν να σχεδιάσουν σύγχρονες εφαρμογές όπως είναι η Python, η οποία τα τελευταία χρόνια κερδίζει σημαντικό έδαφος στην εκπαίδευση. Στην εργασία αυτή παρουσιάζουμε κάποια συμπεράσματα από την παράλληλη διδασκαλία του Scratch και της Python που δοκιμάσαμε στους μαθητές του Ζαννείου Πειραματικού Γυμνασίου Πειραιά. Πρόκειται για μια έρευνα πεδίου η οποία υλοποιήθηκε καθ' όλη τη διάρκεια της χρονιάς, με σκοπό τη μελέτη της αποτελεσματικότητας εναλλακτικών διδακτικών προσεγγίσεων και της αξιολόγησής τους από τους μαθητές. Τα αποτελέσματα ανέδειξαν τη συμβολή του Scratch και της Python σχετικά με την αντίληψη που έχουν οι μαθητές για τον προγραμματισμό και γενικότερα για την επιστήμη της πληροφορικής. Επίσης τα αποτελέσματα ανέδειξαν τη διδακτική χρησιμότητα του Scratch ως μια σκαλωσιά μάθησης η οποία θα διευκολύνει την εισαγωγή μιας γλώσσας προγραμματισμού όπως είναι η Python.

Λέξεις κλειδιά: Python, Scratch, προγραμματισμός, έρευνα

1. Εισαγωγή

Οι βασικές προγραμματιστικές δομές που συναντώνται σε ένα εισαγωγικό μάθημα προγραμματισμού όπως είναι η μεταβλητή, η δομή επιλογής και η δομή επανάληψης, δυσκολεύουν αρκετά τους μαθητές (Robins et. al. 2003, Soloway 1989).

Εκτός από την εγγενή δυσκολία που παρουσιάζουν αυτές οι έννοιες ένας ακόμα παράγοντας δυσκολίας που υπεισέρχεται στη διαδικασία της μάθησης έχει να κάνει με τη γλώσσα προγραμματισμού που χρησιμοποιείται. Η επιλογή της γλώσσας προγραμματισμού με την οποία οι μαθητές και οι φοιτητές θα έχουν την πρώτη επαφή με την πληροφορική καθορίζει σε μεγάλο βαθμό τις πρώτες αναπαραστάσεις που θα δημιουργήσουν οι μαθητές για τα προγραμματιστικά αντικείμενα και τις αλγοριθμικές δομές, όπως επίσης και την στάση των μαθητών απέναντι στον προγραμματισμό (Kelleher & Pausch, 2005, McIver 1996).

Αρκετές έρευνες έχουν γίνει για το ποια γλώσσα προγραμματισμού είναι η κατάλληλη (Kaplan 2010, Jayal 2011, Stefik, & Hanenberg, 2014, Koulouri, 2014). Τα τελευταία χρόνια υπάρχει ένα σημαντικό ρεύμα υπέρ της Python (Grandell, et. al., 2006, Agarwal, et. al., 2008, Goldwasser, & Letscher, 2008, Shein, 2015, Guo, 2014).

Στην ελληνική δευτεροβάθμια εκπαίδευση, διδάσκεται ακόμα στο Γυμνάσιο η γλώσσα Logo και στο Λύκειο ένα μείγμα της Pascal και της Basic. Αυτό έχει σαν αποτέλεσμα το μειωμένο ενδιαφέρον των μαθητών αλλά και τη μικρή ευελιξία των εκπαιδευτικών. Ωστόσο τα τελευταία χρόνια αρκετοί εκπαιδευτικοί έχουν στραφεί και προς άλλα εκπαιδευτικά περιβάλλοντα προγραμματισμού όπως το Scratch στο γυμνάσιο και το App Inventor (Meerbaum-Salant, 2010, Uludag, 2011) στο νέο μάθημα *Εφαρμογές Πληροφορικής* της Α' Λυκείου.

Στην εργασία αυτή παρουσιάζονται κάποια συμπεράσματα από την διδασκαλία του Scratch και της Python στους μαθητές της Γ' τάξης του Ζαννείου πειραματικού γυμνασίου το σχολικό έτος 2014–2015. Πρόκειται για μια έρευνα πεδίου η οποία υλοποιήθηκε καθ' όλη τη διάρκεια της χρονιάς, με σκοπό τη μελέτη και αξιολόγηση των διδακτικών παρεμβάσεων που υλοποιήσαμε και της αποδοχής τους από τους μαθητές.

Στο πρώτο τρίμηνο οι μαθητές ασχοληθήκαν με τη γεωμετρία της χελώνας μέσα από το Scratch. Όταν οι μαθητές εξοικειώθηκαν με το περιβάλλον προγραμματισμού Scratch και με τις βασικές αλγοριθμικές δομές έγινε η μετάβαση στη γλώσσα προγραμματισμού Python. Αυτό συνέβη στα μέσα του 2ου τριμήνου. Η έρευνα πεδίου διεξήχθη καθ' όλη τη διάρκεια της χρονιάς μέσω της παρατήρησης των μαθητών κατά την επίλυση προβλημάτων και μέσω συνεντεύξεων/συζητήσεων με τους μαθητές όπου αυτό κρίθηκε απαραίτητο και είχε καθαρά ποιοτικά χαρακτηριστικά.

Τα αποτελέσματα έδειξαν ότι οι μαθητές με υψηλές επιδόσεις προτιμούσαν την Python για την επίλυση προβλημάτων, ειδικά αν αυτά είχαν μαθηματική φύση και απαιτούσαν τη χρήση μεταβλητών. Υπήρξαν μαθητές που έδειξαν μεγάλο ενδιαφέρον για τον προγραμματισμό εφαρμογών σε Python, ωστόσο αρκετοί μαθητές είχαν καλύτερη απόδοση στην ενότητα του Scratch για το οποίο έδειξαν μεγαλύτερο ενδιαφέρον, ιδιαίτερα για τον προγραμματισμό απλών παιχνιδιών.

2. Γλώσσες Προγραμματισμού στην Εκπαίδευση

Η γλώσσα με την οποία θα έχει την πρώτη του επαφή ένας φοιτητής ή μαθητής αποτελεί ακόμα και σήμερα ένα ενδιαφέρον θέμα συζήτησης μεταξύ επιστημόνων που ακολουθούν διαφορετικές προσεγγίσεις (Kaplan 2010, Jayal 2011, Stefik, & Hanenberg, 2014, Koulouri, 2014). Οι επικρατέστερες από αυτές αφορούν είτε την εισαγωγή μιας αντικειμενοστρεφούς γλώσσας όπως η Java είτε μιας διαδικασιακής γλώσσας όπως η C. Ωστόσο τα τελευταία χρόνια νέες προσεγγίσεις κερδίζουν έδαφος όπως είναι ο προγραμματισμός με πλακίδια, ο οποίος δεν προτιμάται πλέον μόνο στις

μικρές ηλικίες αλλά και σε πανεπιστημιακό επίπεδο. Ένα χαρακτηριστικό παράδειγμα αυτής της προσέγγισης ακολουθείται στο δημοφιλές μάθημα *Εισαγωγή στην Επιστήμη Υπολογιστών* του πανεπιστημίου του Harvard, γνωστού και ως CS50 (Wolz et al., 2009). Στο μάθημα αυτό οι φοιτητές ξεκινούν μαθαίνοντας τα βασικά του προγραμματισμού στο Scratch και στη συνέχεια μεταβαίνουν σε μια γλώσσα προγραμματισμού όπως είναι η C και η Javascript. Αυτή την προσέγγιση ακολουθήσαμε και εμείς με τη διαφορά ότι η μετάβαση έγινε στην γλώσσα Python.

2.1 Το περιβάλλον προγραμματισμού Scratch

Ο προγραμματισμός με πλακίδια είναι αρκετά διαδεδομένος στη δευτεροβάθμια εκπαίδευση και ειδικότερα στις πολύ μικρές ηλικίες, με πιο δημοφιλές προγραμματιστικό περιβάλλον το Scratch. Ενώ ο προγραμματισμός με Scratch παρουσιάζει σημαντικά διδακτικά οφέλη στις μικρές ηλικίες (Meerbaum-Salant, Armoni και Ben-Ari (2010), έχει και ένα σοβαρό μειονέκτημα (Lewis et. al., 2014). Μετά τον αρχικό ενθουσιασμό οι μαθητές αντιλαμβάνονται ότι δεν προγραμματίζουν σε μια “πραγματική” γλώσσα προγραμματισμού οπότε το ενδιαφέρον κάποιων μειώνεται αρκετά. Ο Lewis (2010) έφτασε στο συμπέρασμα ότι οι μαθητές πρωτοβάθμιας και δευτεροβάθμιας εκπαίδευσης που προγραμματίζουν σε γλώσσες όπου χρειάζεται να γράψουν κώδικα, όπως είναι η Logo, παρουσιάζουν μεγαλύτερη αυτοπεποίθηση στην επίλυση προβλημάτων από αυτούς που προγραμματίζουν σε Scratch.

2.2 Η γλώσσα προγραμματισμού Python

Αρκετές έρευνες έχουν γίνει σχετικά με την επιλογή της γλώσσας για ένα εισαγωγικό μάθημα προγραμματισμού (Kaplan 2010, Jayal 2011, Stefik, & Hanenberg, 2014, Kouliouri, 2014). Στις περισσότερες από αυτές συγκρίνονται οι δυο επικρατέστερες τάσεις, η πιο παλιά που υποστηρίζει την εισαγωγή της Java και η σύγχρονη που υιοθετεί την εισαγωγή της Python. Στις περισσότερες περιπτώσεις τα αποτελέσματα είναι αρκετά θετικά για την Python κυρίως όσον αφορά την απόδοση και τα ποσοστά επιτυχίας των φοιτητών στα μαθήματα αυτά.

Ένα πρόγραμμα σε Python δεν απέχει πολύ από την περιγραφή ενός αλγορίθμου σε ψευδογλώσσα, αφού η σύνταξή του είναι εξαιρετικά απλή. Δεν υπάρχει τμήμα δήλωσης μεταβλητών αφού η γλώσσα χρησιμοποιεί ένα δυναμικό σύστημα τύπων το οποίο σε συνδυασμό με τη χρήση του διερμηνευτή, διευκολύνει τον πειραματισμό των μαθητών. Έτσι μπορούμε να ορίσουμε μια συνάρτηση η οποία μπορεί να χρησιμοποιηθεί για οποιοδήποτε τύπο της γλώσσας, δεδομένου ότι οι πράξεις που περιέχει ορίζονται για μεταβλητές του τύπου αυτού, όπως φαίνεται παρακάτω:

```
def Επίλυση(Συνάρτηση):
    for x in range(-100000, 100000):
        if Συνάρτηση(x) == 0 :
```

```
print("x =", x)
```

```
>>> Επίλυση( lambda x: x*x - 5*x + 6 )
```

Ένα άλλο πλεονέκτημα της Python είναι η υποστήριξη τριών διαφορετικών προγραμματιστικών υποδειγμάτων, του διαδικασιακού, του αντικειμενοστρεφούς και του συναρτησιακού, κάτι που δίνει πολλές επιλογές στον καθηγητή. Επίσης η Python έχει πολύ μεγάλη κοινότητα εκπαιδευτικών και προγραμματιστών, οι οποίοι διαθέτουν αρκετό υλικό (βιβλία, ασκήσεις, φύλλα εργασίας, tutorials, σημειώσεις) στο διαδίκτυο.

Συμπερασματικά η Python φαίνεται σαν μια εξαιρετική επιλογή για την εκπαίδευση και έχει υιοθετηθεί από πολλά πανεπιστήμια κυρίως στην Αμερική (Grandell, et. al., 2006, Agarwal, et. al., 2008, Goldwasser, & Letscher, 2008, Shein, 2015, Guo, 2014). Πρόσφατα έχει αρχίσει να μπαίνει και στον χώρο της δευτεροβάθμιας εκπαίδευσης.

3.Η Έρευνα

3.1 Στόχος της Έρευνας

Ο βασικός στόχος της έρευνας ήταν να διερευνήσουμε τις αντιλήψεις και τις απόψεις των μαθητών σχετικά με τις δυο γλώσσες προγραμματισμού με τις οποίες ασχολήθηκαν, δηλαδή το Scratch και την Python.

3.2 Δείγμα

Το δείγμα της έρευνας ήταν ένα τμήμα 23 μαθητών της Γ' Γυμνασίου του Ζαννείου Πειραματικού Γυμνασίου Πειραιά. Οι περισσότεροι μαθητές δεν είχαν προγραμματίσει ξανά σε κάποια γλώσσα προγραμματισμού εκτός από 4 μαθητές που ήταν μέλη του ομίλου Αλγοριθμικής του σχολείου ο οποίος λειτουργήσε στο πλαίσιο των ομίλων αριστείας και δημιουργικότητας των πρότυπων πειραματικών σχολείων. Το υλικό που δόθηκε στους μαθητές της τάξης αλλά και το πρόγραμμα σπουδών του ομίλου μπορείτε να το βρείτε στη διεύθυνση <http://envripides.mysch.gr>.

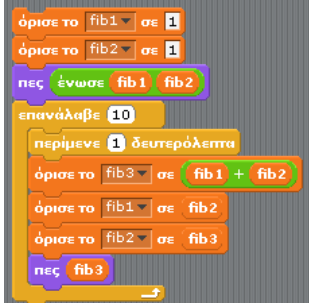
3.3 Μεθοδολογία

Η έρευνα διεξήχθη μέσω παρατήρησης των μαθητών κατά τη διάρκεια όλης της χρονιάς. Τα κομβικά σημεία στα οποία δώσαμε περισσότερη προσοχή ήταν στο δεύτερο τρίμηνο όπου έλαβε χώρα η μετάβαση από το Scratch στην Python και λίγο αργότερα όταν οι μαθητές αντιμετώπισαν απαιτητικά προβλήματα στην γλώσσα Python όπου τους επιτράπηκε αν θέλουν να αναπτύξουν τις λύσεις τους σε Scratch.

Στο τέλος της χρονίας δόθηκε και ένα ερωτηματολόγιο στους μαθητές για την αποτύπωση των απόψεών τους σχετικά με τα δυο περιβάλλοντα προγραμματισμού που χρησιμοποιήσαν.

3.4 Διδακτική Προσέγγιση

Σύμφωνα με την προσέγγιση που ακολουθήσαμε οι μαθητές ξεκίνησαν να προγραμματίζουν με το Scratch στο πρώτο τρίμηνο και αφού καλύψαμε τις βασικές αλγοριθμικές δομές ξεκινήσαμε στο δεύτερο τρίμηνο την εισαγωγή στη γλώσσα Python. Προσπαθήσαμε να εκμεταλλευτούμε το Scratch ως σημείο αναφοράς και στα πρώτα μαθήματα δίναμε πάντα δυο υλοποιήσεις δίπλα – δίπλα σε Scratch και σε Python, όπως φαίνεται στα παρακάτω τμήματα κώδικα:

Python	Scratch
<pre> fib1 = 1 fib2 = 1 print(fib1, fib2) for i in range(10): fib3 = f1 + f2 f1 = f2 f2 = f3 print(f3) </pre>	 <p>The Scratch code blocks are as follows: <ul style="list-style-type: none"> Set fib1 to 1 Set fib2 to 1 Loop: join fib1 and fib2 Repeat 10 times: <ul style="list-style-type: none"> Wait 1 second Set fib3 to fib1 + fib2 Set fib1 to fib2 Set fib2 to fib3 Loop: fib3 </p>

Εικόνα 1. Οι αριθμοί Fibonacci σε Python και Scratch

Η προσέγγιση αυτή υιοθετείται από πολλούς ερευνητές (Wolz et al., 2009, Wagner et. al., 2013, Dorling, 2015), με πολύ θετικά αποτελέσματα. Ο σκοπός της εισαγωγής σε ένα περιβάλλον προγραμματισμού με πλακίδια δεν είναι η εκμάθησή του σε βάθος, αλλά η αξιοποίησή του σαν σκαλωσιά μάθησης για την πρώτη πραγματική γλώσσα προγραμματισμού που θα μάθουν οι μαθητές. Τα πιο γνωστά περιβάλλοντα με πλακίδια είναι το Scratch, το Snap ή ακόμα και το App Inventor ενώ η πρώτη γλώσσα προγραμματισμού στις περισσότερες περιπτώσεις είναι η Java ή η Python. Σε ελάχιστες περιπτώσεις όπως στο μάθημα CS50 του Harvard η δεύτερη γλώσσα είναι η C.

Η έκδοση της Python που προτιμήθηκε ήταν η 3 κυρίως επειδή επιτρέπει την συγγραφή προγραμμάτων στα ελληνικά αλλά και επειδή υποστηρίζει αρκετά καλά τον προγραμματισμό με τη γεωμετρία της χελώνας. Έτσι η γεωμετρία της χελώνας λειτούργησε ως γέφυρα μεταξύ του Scratch και της Python.

Τα προγράμματα που προκύπτουν δεν είναι τόσο διαφορετικά μεταξύ τους αλλά ούτε και με τα αντίστοιχα της Logo, όπως φαίνεται στο παρακάτω πρόγραμμα σχεδιασμού ενός τετραγώνου.

Python	Scratch	Logo
<pre>for times in range(4): franklin.forward(100) franklin.right(90)</pre>		<pre>επανάλαβε 4 [μπροστά 100 δεξιά 90]</pre>

Εικόνα 2. Σχεδιασμός τετραγώνου σε Python, Scratch και Logo

Έτσι στην αρχή οι μαθητές υλοποίησαν σε Python αρκετά από τα προγράμματα που είχαν γράψει στο πρώτο τρίμηνο σε Scratch για να δουν την αντιστοιχία μεταξύ των δομών επιλογής και επανάληψης και των εντολών της γεωμετρίας της χελώνας. Η μόνη δυσκολία στην python ήταν ότι έπρεπε να εξηγήσουμε στους μαθητές ότι έχουμε δημιουργήσει ένα αντικείμενο – χελώνα, τον Φράνκλιν στον οποίο στέλνουμε μηνύματα για το πώς θα μετακινηθεί.

Όλα τα μαθήματα έγιναν στο εργαστήριο πληροφορικής με φύλλα εργασίας μέσω των οποίων οι μαθητές ήρθαν σε επαφή με τις βασικές δομές της Python ακολουθώντας μια διερευνητική – ανακαλυπτική προσέγγιση.

Στη συνέχεια έγινε μια μετάβαση από εφαρμογές της γεωμετρίας της χελώνας σε προγράμματα με καθαρά υπολογιστή φιλοσοφία όπως οι αριθμοί Fibonacci που δείξαμε παραπάνω. Στο τέλος της χρονιάς δόθηκαν διάφορα προβλήματα στους μαθητές και τους ζητήθηκε να αναπτύξουν προγράμματα για την επίλυσή τους είτε σε Scratch είτε σε Python.

5. Ανάλυση των αποτελεσμάτων – Παρατηρήσεις

Το σημαντικότερο σημείο αυτής της έρευνας ήταν στο δεύτερο τρίμηνο όταν δοκιμάσαμε τη μετάβαση από το Scratch στην Python. Το Scratch όπως αναφέρθηκε παραπάνω λειτούργησε ως σημείο αναφοράς για τις δομές επιλογής και επανάληψης, μειώνοντας τον χρόνο εξοικείωσης των μαθητών με τις δομές αυτές. Ένα σημαντικό πλεονέκτημα του Scratch αποτελεί η εισαγωγή της δομής επανάληψης χωρίς προηγούμενος να έχουμε μιλήσει για μεταβλητές, όπως συμβαίνει και στη Logo.

Η πληκτρολόγηση των εντολών στην Python δεν ήταν πρόβλημα για τους μαθητές λόγω της απλής της σύνταξης. Ωστόσο χρειάστηκε να τους επιστήσουμε την προσοχή πολλές φορές στο θέμα του ορισμού των μπλοκ εντολών με βάση τις εσοχές.

Οι μαθητές αναγνώρισαν την εκφραστική δύναμη της Python όταν ασχολήθηκαν με μαθηματικές εφαρμογές στις οποίες χρειάστηκε η χρήση μεταβλητών. Ένα τέτοιο παράδειγμα είναι η επίλυση της δευτεροβάθμιας εξίσωσης. Η ενότητα αυτή είναι στην ύλη των μαθηματικών της Γ' γυμνασίου και δείχνει πολύ καλά ότι τα προγράμ-

ματα σε Scratch δεν είναι πάντα πιο απλά από ότι σε μια πραγματική γλώσσα προγραμματισμού όπως η Python.

Python	Scratch
<pre> if D>0 : x1 = (-b + sqrt(D)) / (2*a) x2 = (-b - sqrt(D)) / (2*a) print ('x1 = ', x1) print ('x2 = ', x2) elif D==0: x1 = (-b)/(2*a) print ('x1 = ', x1) else: print('Δεν έχει λύσεις')</pre>	

Εικόνα 3. Επίλυση δευτεροβάθμιας εξίσωσης σε Python και Scratch

Παρατηρήσαμε ότι η ανάπτυξη του παραπάνω προγράμματος σε Scratch πήρε περισσότερο χρόνο από ότι σε Python. Η ένωση όλων των προγραμματιστικών δομών με οπτικό τρόπο πήρε περισσότερο χρόνο από την πληκτρολόγησή λόγω της πολυπλοκότητας των εκφράσεων και της εμπλοκής μεταβλητών σε αυτές.

Ήταν αναμενόμενο ότι το Scratch θα είχε ένα μειονέκτημα στη διαχείριση μεταβλητών σε σχέση με την Python. Ωστόσο ένα πλεονέκτημα στη διαχείριση των μεταβλητών για το Scratch αφορά τον τρόπο αύξησης μιας μεταβλητής, κάτι που είναι απαραίτητο σε μια δομή επανάληψης. Έτσι μπορούμε να παρουσιάσουμε στους μαθητές τον παρακάτω τρόπο αύξησης αν θεωρήσουμε ότι η εντολή *δείκτης = δείκτης + 1*, θα προκαλέσει σύγχυση στους μαθητές.

Python	Scratch
<pre> δείκτης += 1</pre>	

Εικόνα 4. Αύξηση μεταβλητής κατά 1 σε Python και Scratch

Θα πρέπει να σημειωθεί ότι οι περισσότεροι μαθητές με καλές επιδόσεις στα μαθηματικά έδειξαν προτίμηση στην Python όπως και οι μαθητές που συμμετείχαν στον όμιλο αλγοριθμικής του σχολείου. Αυτό προέκυψε από ένα ερωτηματολόγιο που δόθηκε στους μαθητές στο τέλος της χρονιάς.

Αξίζει να σημειωθεί ότι οι μαθητές που έδειξαν ιδιαίτερο ενδιαφέρον και έλυσαν όλα τα προβλήματα μέσω της πλατφόρμας interactivpython.org που χρησιμοποιήσαμε

δεν ήταν άριστοι μαθητές. Οι μαθητές με πολύ υψηλούς βαθμούς σε άλλα μαθήματα δεν έκαναν πολλές προαιρετικές ασκήσεις παρά μόνο αυτές που ήταν υποχρεωμένοι.

6. Συμπεράσματα

Στην εργασία αυτή παρουσιάσαμε μια έρευνα που έλαβε χώρα την σχολή χρονιά 2014 – 2015 σε 23 μαθητές του Ζαννείου πειραματικού γυμνασίου Πειραιά σχετικά με την εκμάθηση του προγραμματισμού με τη χρήση του Scratch και της Python.

Τα αποτελέσματα ήταν πολύ ενθαρρυντικά, και έδειξαν όχι μόνο το υψηλό ενδιαφέρον των μαθητών για τον προγραμματισμό με τη γλώσσα Python, αλλά και την αξία του Scratch ως ένα εργαλείο για τη μετάβαση σε μια πραγματική γλώσσα προγραμματισμού όπως η Python. Το γεγονός ότι η Python είναι απλή και επεξηγηματική σαν ψευδογλώσσα, βοήθησε αρκετά στη μετάβαση αυτή. Υπήρξαν μαθητές που έδειξαν μεγάλο ενδιαφέρον για τον προγραμματισμό εφαρμογών σε Python, ωστόσο αρκετοί μαθητές είχαν καλύτερη απόδοση στην ενότητα του Scratch για το οποίο έδειξαν μεγαλύτερο ενδιαφέρον, ιδιαίτερα για τον προγραμματισμό απλών παιχνιδιών. Σε ερωτηματολόγιο που δόθηκε στο τέλος της χρονιάς στους μαθητές, οι μαθητές απάντησαν ότι το Scratch τους βοήθησε πολύ στα πρώτα τους βήματα με την Python. Επίσης η πλειονότητα των μαθητών απάντησε ότι η Python είναι μια γλώσσα που μπορεί να διδάσκεται στο Γυμνάσιο και έχει ενδιαφέρον.

Αναφορές

- Agarwal, K., Agarwal, A., & Celebi, E. (2008). Python puts a squeeze on java for CS0 and beyond. *J. Comput. Sci. Coll.* 23, 6, 49-57
- 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, NY, USA, 191-196
- Goldwasser, M., & Letscher, D. (2008). Teaching an object-oriented CS1 -: with Python. *SIGCSE Bull.* 40, 3 (June 2008), 42-46.
- Grandell, L., Peltomäki, M., Back, R., & Salakoski, T. (2006). Why complicate things?: introducing programming in high school using Python. In *Proceedings of the 8th Australasian Conference on Computing Education - Volume 52 (ACE '06)*, Vol. 52. Australian Computer Society
- Guo, P., (2014). *Python is now the Most Popular Introductory Teaching Language at Top U.S. Universities*. Survey published at the Communications of the ACM blog (CACM blog).
- Jayal, A., Lauria, S., Tucker, A., & Swift, S. (2011). Python for teaching introductory programming: A quantitative evaluation, *ITALICS* , vol. 10, no. 1, pp. 86–90, 2011.

- Kaplan, R. (2010). Choosing a first programming language. In *Proceedings of the 2010 ACM conference on Information technology education (SIGITE '10)*. ACM, New York, NY, USA, 163-164
- Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Comput. Surv.* 37, 2, 83-137.
- Lewis, C. (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., and Schlesinger, A. (2014). Children's perceptions of what counts as a programming language. *J. Comput. Sci. Coll.* 29, 4, 123-133.
- Koulouri, T., Lauria, S., & Macredie, R. (2014). Teaching Introductory Programming: A Quantitative Evaluation of Different Approaches. *Trans. Comput. Educ.* 14, 4, Article 26 .
- McIver, L., & Conway, D. (1996). Seven Deadly Sins of Introductory Programming Language Design. In *Proceedings of the 1996 International Conference on Software Engineering: Education and Practice (SE:EP '96)*. IEEE Computer Society.
- Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (2010). Learning computer science concepts with scratch. In *Proceedings of the Sixth international workshop on Computing education research (ICER '10)*. ACM, New York, NY, USA, 69-76
- Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education*, 13(2), 137-172.
- Shein, E. (2015). Python for beginners. *Commun. ACM* 58, 3, 19-21.
- Soloway, E., & Spohrer, J. C. (1989) (eds.). *Studying the Novice Programmer*. Hillsdale, NJ: Lawrence Erlbaum.
- Stefik, A., & Hanenberg, S. (2014). The Programming Language Wars: Questions and Responsibilities for the Programming Language Community. In *Proceedings of the 2014 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming & Software*. ACM, New York, , NY, USA 283-299.
- Uludag, S., Karakus, M., & Turner, S. (2011). Implementing IT0/CS0 with scratch, app inventor for android, and lego mindstorms. In *Proceedings of the 2011 conference on Information technology education (SIGITE '11)*. ACM, New York, NY, USA, 183-190

- Wagner, A., Gray, J., Corley, J., and 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
- Wolz, U., Leitner, H., Malan, D., and 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

Abstract

Computer programming is perceived as an important competence for the development of problem solving skills in addition to logical reasoning. The choice of the introductory programming language plays an important role in the learning process. Secondary education students learn programming using logo-like environments such as Logo and Scratch. However many students want to see a real programming language with which they can design modern applications such as Python, which in recent years is gaining significant ground in education. In this paper we present the results of an exploratory effort to investigate the effect of Scratch and Python programming on high school students. This is a field research which was carried out throughout the course of the year to study the effectiveness of alternative teaching approaches and their evaluation by the students. The qualitative results revealed important information about the contribution of Scratch and Python on students' interest about programming and computer science.

Keywords: Python, Scratch, programming, experimental study