

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΑΤΡΙΒΗ
ΣΤΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ



Μάθηση Βασικών Εννοιών Προγραμματισμού
Στη Δευτεροβάθμια Εκπαίδευση
Με Την Αξιοποίηση Του Λογισμικού MIT App Inventor

Γεώργιος Χατζηνικολάκης

Επιβλέπων Καθηγητής
Σπυρίδων Παπαδάκης

Μάιος 2014

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

**Μάθηση Βασικών Εννοιών Προγραμματισμού
Στη Δευτεροβάθμια Εκπαίδευση
Με Την Αξιοποίηση Του Λογισμικού MIT App Inventor**

Γεώργιος Χατζηνικολάκης

**Επιβλέπων Καθηγητής
Σπυρίδων Παπαδάκης**

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε
προς μερική εκπλήρωση των απαιτήσεων για απόκτηση

μεταπτυχιακού τίτλου σπουδών
στα Πληροφοριακά Συστήματα

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών
του Ανοικτού Πανεπιστημίου Κύπρου

Μάιος 2014

Περίληψη

Η αλγοριθμική και υπολογιστική σκέψη αναγνωρίζονται πλέον ως ιδιαίτερα σημαντικές δεξιότητες, απαραίτητες για όλους τους μαθητές, αυριανούς πολίτες και εργαζόμενους. Στην Ελλάδα, τα Προγράμματα Σπουδών για τα εισαγωγικά μαθήματα προγραμματισμού, τόσο στην πρωτοβάθμια, όσο και στη δευτεροβάθμια εκπαίδευση, συνιστούν, ιδιαίτερα στους μικρότερους μαθητές, την αξιοποίηση LOGO-like εργαλείων και περιβαλλόντων οπτικού προγραμματισμού. Παράλληλα, η μεγάλη διείσδυση των «έξυπνων» φορητών συσκευών στην παγκόσμια αγορά και η απήχηση που έχουν στις νεότερες γενιές, αποτελούν γεγονότα που μπορούν να αξιοποιηθούν για τη διευκόλυνση της διδασκαλίας και μάθησης βασικών εννοιών προγραμματισμού στους χρήστες τους.

Σκοπός της μεταπτυχιακής διατριβής είναι η διερεύνηση της δυνατότητας αξιοποίησης του App Inventor, ενός προγραμματιστικού περιβάλλοντος δημιουργίας εφαρμογών για ψηφιακές φορητές συσκευές με λειτουργικό σύστημα Android, για τη διευκόλυνση της διδασκαλίας και μάθησης βασικών εννοιών προγραμματισμού στη δευτεροβάθμια εκπαίδευση. Στο πλαίσιο αυτό, αναπτύχθηκαν μαθησιακές δραστηριότητες άμεσης και εύκολης δημιουργίας, πλήρως λειτουργικών και αυθεντικών εφαρμογών για συσκευές (smartphones και tablets) που οι μαθητές αγαπάνε και χρησιμοποιούν σε καθημερινή βάση. Οι δραστηριότητες αυτές χρησιμοποιήθηκαν ως υποστηρικτικά μαθησιακά εργαλεία για τις ανάγκες δυο εκπαιδευτικών εργαστηρίων για μαθητές δευτεροβάθμιας εκπαίδευσης που διοργανώθηκαν στη Χίο, εστιάζοντας στη διδασκαλία και μάθηση βασικών προγραμματιστικών εννοιών.

Με τη χρήση ερωτηματολογίων, μέσω συμμετοχικής παρατήρησης, αλλά και συνεντεύξεων, καταγράφηκαν και παρουσιάζονται τα ευρήματα ως προς τις στάσεις μαθητών και εκπαιδευτικών για το App Inventor και τα θετικά και αρνητικά του σημεία ως προς τη διευκόλυνση επίτευξης των μαθησιακών στόχων που τέθηκαν. Τα αποτελέσματα του πειράματος έδειξαν πως το App Inventor είναι εύκολο και ελκυστικό στη χρήση, συμβάλλει στην επίτευξη υψηλού επιπέδου δέσμευσης και δημιουργικότητας από τους μαθητές της δευτεροβάθμιας εκπαίδευσης, γεγονότα που το καθιστούν ικανό να διευκολύνει τη διδασκαλία και μάθηση θεμελιωδών εννοιών προγραμματισμού.

Summary

Algorithmic and computational thinking are now recognized as essential skills, necessary for all students, employees and citizens of tomorrow. In Greece, curricula for introductory programming courses in primary and secondary education, recommend the use of LOGO-like tools and visual programming environments, especially for the youngest students. At the same time, the massive penetration of "smart" mobile devices in the global market and the impact they have on new generations, both can be used to facilitate teaching and learning of basic programming concepts to their users.

The purpose of this M.Sc. dissertation is to investigate the possibility of using App Inventor, a programming environment for the development of applications for digital mobile devices running Android, to facilitate teaching and learning of basic concepts of programming in secondary education. In this context, we created several learning activities on easy and direct development of fully functional and original applications for devices (smartphones and tablets) that most students love and use on a daily basis. These activities were used as tools for supporting the learning needs of two educational workshops for secondary school students organized in Chios island, focusing on teaching and learning basic programming concepts.

Through the use of questionnaires, participant's observation and interviews, we recorded and presented the findings regarding the attitudes of both students and teachers towards App Inventor and the positive and negative points of it as to facilitate achievement of learning goals that were set. The results of the experiment showed that App Inventor is attractive and easy to use, it contributes to a high level of commitment and creativity of students in secondary education, which enable it to facilitate teaching and learning fundamental programming concepts.

Ευχαριστίες

Καταρχάς, ευχαριστώ βαθύτατα τον κύριο Σπυρίδωνα Παπαδάκη για την εμπιστοσύνη του, την καθοδήγηση, υποστήριξη και βοήθεια που μου προσέφερε σε όλα τα στάδια εκπόνησης της παρούσης Μεταπτυχιακής Διατριβής.

Επίσης, ευχαριστώ όλους τους διδάσκοντες των Θεματικών Ενοτήτων του Προγράμματος που παρακολούθησα, για τα πολύτιμα εφόδια που μου με βοήθησαν να αποκτήσω σε αυτή την πορεία, από την αρχή της μέχρι την κορύφωσή της, και ιδιαιτέρως τον κύριο Θανάση Χατζηλάκο για τις αξίες και τις γνώσεις που μου μετέδωσε.

Δεν μπορώ να παραλείψω να εκφράσω τις ευχαριστίες μου και στους συναδέλφους μου, ΔΣ και μέλη του Συλλόγου Εκπαιδευτικών Πληροφορικής Χίου για την αρωγή τους στην οργάνωση και υλοποίηση των εκπαιδευτικών εργαστηρίων, καθώς και για την ανάπτυξη του εκπαιδευτικού υλικού. Χωρίς τη βοήθειά τους σε όλους τους παραπάνω τομείς, και ιδιαιτέρως χωρίς την άψογη συνεργασία με τον εκλεκτό συνάδελφο Βασίλη Βασιλάκη, δεν θα μπορούσα να ολοκληρώσω την εργασία αυτή. Βεβαίως, τίποτα δεν θα μπορούσε να ολοκληρωθεί επίσης, χωρίς τη συνεργασία, υπομονή και καλή διάθεση των συμμετεχόντων μαθητών/μαθητριών και διευθυντών, τους οποίους και ευχαριστώ.

Πάνω απ' όλα όμως, οφείλω ένα τεράστιο ευχαριστώ στη σύζυγό μου Σοφία για την αμέριστη συμπαράσταση και υπομονή που επέδειξε καθ' όλο το χρονικό διάστημα εκπόνησης των σπουδών μου στο Ανοικτό Πανεπιστήμιο Κύπρου.

Στη Σοφία, λοιπόν, και στην κορούλα μας, την οποία αποκτήσαμε κατά τη διάρκεια του «ταξιδιού» αυτού...

Περιεχόμενα

1 Εισαγωγή	1
1.1 Αντικείμενο της Μεταπτυχιακής Διατριβής.....	2
1.2 Σκοπός έρευνας.....	2
1.2.1 Ερευνητικά ερωτήματα.....	3
1.2.2 Συμβολή της εργασίας.....	3
1.3 Η δομή της Μεταπτυχιακής Διατριβής.....	4
2 Ο Προγραμματισμός Υπολογιστών στην Π/θμια και Δ/θμια Εκπαίδευση	6
2.1 Η Πληροφορική ως γνωστικό αντικείμενο.....	6
2.1.1 Θέσεις της ACM για τη σημασία της πληροφορικής στην εκπαίδευση.....	7
2.1.2 Θέσεις της Ομάδας Εργασίας “Computing at School”.....	8
2.1.3 Αναφορά της CSTA για την πληροφορική στην Κ-8 βαθμίδα εκπαίδευσης.....	10
2.2 Υπολογιστική Σκέψη.....	10
2.2.1 Τι είναι η υπολογιστική σκέψη.....	11
2.2.2 Η σημασία της υπολογιστικής σκέψης στην εκπαίδευση.....	11
2.2.3 Το πιλοτικό εκπαιδευτικό πρόγραμμα “CS Principles”.....	12
2.2.4 Υπολογιστική σκέψη και πρακτική.....	13
2.3 Ο προγραμματισμός Η/Υ ως διδακτικό αντικείμενο.....	14
2.3.1 Αλγοριθμική και προγραμματισμός.....	15
2.3.2 “ACM K-12 CS Model Curriculum” και προγραμματισμός.....	15
2.3.4 Τα αρνητικά της χρήσης γλωσσών γενικού σκοπού.....	18
2.3.5 Τα οφέλη των εργαλείων οπτικού προγραμματισμού.....	18
2.3.6 Η γλώσσα Logo για την εκπαίδευση στον προγραμματισμό.....	19
2.3.7 Εποικοδομιστικά περιβάλλοντα προγραμματισμού.....	20
2.3.8 Η περίπτωση του Scratch.....	21
2.3.9 Η σημασία του «πλαισίου» και του μέσου.....	21
3 Διδασκαλία Προγραμματισμού στην ελληνική εκπαίδευση	24
3.1 ΤΠΕ και προγραμματισμός στο Δημοτικό σχολείο.....	24
3.1.1 Ε' Δημοτικού.....	27
3.1.2 ΣΤ' Δημοτικού.....	29
3.1.3 Ολοήμερα Δημοτικά Σχολεία με ΕΑΕΠ.....	30
3.2 Πληροφορική & προγραμματισμός στο Γυμνάσιο.....	32
3.2.1 Α' Γυμνασίου.....	34
3.2.2 Β' Γυμνασίου.....	36
3.2.3 Γ' Γυμνασίου.....	39

3.3	Πληροφορική και προγραμματισμός στο Λύκειο.....	40
3.3.1	Η Πληροφορική ως μάθημα επιλογής.....	41
3.3.2	Η πληροφορική στην Τεχνολογική Κατεύθυνση του Γενικού Λυκείου.....	42
3.3.3	Η Πληροφορική στο «νέο» Γενικό Λύκειο.....	44
3.3.4	Η Πληροφορική στο Επαγγελματικό Λύκειο.....	46
4	Προβλήματα στην εκμάθηση του προγραμματισμού υπολογιστών	47
4.1	Ο Προγραμματισμός στην ελληνική εκπαίδευση	47
4.1.1	Δημοτικό	48
4.1.2	Γυμνάσιο	48
4.1.3	Λύκειο	49
4.2	Δυσκολίες και παρανοήσεις των μαθητών.....	50
4.2.1	Γενικές δυσκολίες.....	51
4.2.2	Παρανοήσεις μαθητών σε βασικές έννοιες και δομές.....	53
5	Προγραμματισμός φορητών συσκευών.....	59
5.1	Κατηγορίες φορητών συσκευών.....	59
5.2	Έξυπνες φορητές συσκευές στην εκπαίδευση.....	65
5.2.1	Ιδιαίτερα χαρακτηριστικά των ψηφιακών φορητών συσκευών.....	65
5.2.2	Η πρακτική «Φέρε τη δική σου συσκευή» (BYOD)	66
5.3	Ανάπτυξη εφαρμογών για φορητές συσκευές.....	66
5.3.1	Το λειτουργικό σύστημα iOS της Apple.....	67
5.3.2	Το λειτουργικό σύστημα Android της Google.....	68
5.3.3	Αρχιτεκτονική του Android	69
5.3.4	Η διαδικασία ανάπτυξης εφαρμογών στο Android.....	72
5.3.5	Ανάπτυξη εφαρμογών που λειτουργούν και στις δυο πλατφόρμες.....	74
5.3.6	Προβληματισμοί – περιορισμοί για τον προγραμματισμό σε περιβάλλον Android.....	75
6	Η περίπτωση του App Inventor.....	77
6.1	Εισαγωγή.....	77
6.2	Απαιτήσεις.....	78
6.3	Το περιβάλλον δημιουργίας εφαρμογών	79
6.4	Διαθέσιμα συστατικά (components)	81
6.5	Διαθέσιμα πλακίδια (blocks)	83
6.5.1	Μεταβλητές.....	84
6.5.2	Λογικές εκφράσεις.....	84
6.5.3	Δομή επιλογής.....	85
6.5.4	Δομή επανάληψης.....	85
6.5.5	Υποπρογράμματα - Διαδικασίες.....	86

6.6	Ανάπτυξη εφαρμογών.....	86
6.7	Έλεγχος των εφαρμογών.....	88
6.7.1	Προσομοιωτής (on-screen Android emulator).....	89
6.7.2	Ενσύρματη σύνδεση (USB connection).....	90
6.7.3	Ασύρματη σύνδεση (Wi-Fi connection).....	91
6.8	Διανομή των εφαρμογών.....	92
6.9	Το ενδεχόμενο αξιοποίησης του AI στο σχολείο.....	93
6.9.1	Ένταξη του AI στα προτεινόμενα προγραμματιστικά περιβάλλοντα.....	93
6.9.2	AI και ομοιότητες με το Scratch.....	94
6.9.3	Σύνοψη των συμπερασμάτων από τη χρήση του AI στο εξωτερικό.....	97
7	Μεθοδολογία.....	99
7.1	Η μέθοδος.....	99
7.3	Δείγμα έρευνας.....	100
7.4	Η δομή του πειράματος.....	101
7.4.1	Περιγραφή του πειράματος.....	102
7.4.2	Ο ρόλος των εκπαιδευτικών.....	103
7.4.3	Ο ρόλος των μαθητών.....	104
7.4.4	Προετοιμασία των χώρων διεξαγωγής.....	105
7.5	Το εκπαιδευτικό υλικό.....	105
7.5.1	Ανάπτυξη εκπαιδευτικού υλικού.....	105
7.5.2	Στόχοι διδασκαλίας.....	106
7.5.3	Περιεχόμενο μαθημάτων.....	109
7.6	Μέσα και μέθοδοι συλλογής δεδομένων.....	110
7.6.1	Ερωτηματολόγια.....	111
7.6.2	Παρατήρηση.....	112
7.6.3	Συνεντεύξεις.....	113
8	Αποτελέσματα και συζήτηση.....	114
8.1	Εισαγωγή.....	114
8.2	Δημογραφικά στοιχεία.....	116
8.3	Πρότερες γνώσεις των μαθητών για τον προγραμματισμό.....	118
8.4	Ανίχνευση αλλαγών στις στάσεις των μαθητών για τον προγραμματισμό.....	121
8.5	Δεξιότητες που αναπτύχθηκαν.....	124
8.5.1	Ικανότητα δημιουργίας προσωπικών εφαρμογών με το AI.....	124
8.5.2	Βαθμός κατάκτησης δεξιοτήτων.....	125
8.6	Οι απόψεις των μαθητών για το App Inventor.....	130
8.6.1	Θετικά σημεία του AI.....	130
8.6.2	Αρνητικά σημεία του AI.....	131

8.6.3	Η συμβολή του ΑΙ στην ανάπτυξη του ενδιαφέροντος των μαθητών	132
8.7	Απόψεις των εκπαιδευτικών.....	134
8.7.1	Ευρήματα από τη συμμετοχική παρατήρηση του ερευνητή.....	134
8.7.1	Ευρήματα συνεντεύξεων.....	138
8.8	Αποτίμηση του πειράματος.....	140
9	Συμπεράσματα.....	142
9.1	Συμπεράσματα της έρευνας.....	142
9.2	Περιορισμοί της έρευνας	146
9.3	Προτάσεις για μελλοντική έρευνα.....	147
9.4	Επίλογος	147
	Βιβλιογραφικές αναφορές	148
	Κατάσταση ακρωνυμίων.....	157
Παράρτημα		
A	Σύνοψη Δραστηριοτήτων.....	A-1
A.1	Μάθημα 1	A-2
A.2	Μάθημα 2	A-4
A.3	Μάθημα 3	A-7
A.4	Ενδεικτικό παράδειγμα δραστηριότητας	A-10
A.5	Οδηγίες προετοιμασίας πριν το 1 ^ο μάθημα	A-13
B	Μέσα Συλλογής Δεδομένων	B-1
B.1	Δήλωση συγκατάθεσης συμμετοχής στην έρευνα	B-1
B.2	1ο ερωτηματολόγιο	B-3
B.3	2ο ερωτηματολόγιο	B-6
B.4	Φύλλα Παρατήρησης	B-13
Γ	Πίνακες Στατιστικής Ανάλυσης	Γ-1
Δ	Αποσπάσματα συνεντεύξεων	Δ-1

Κεφάλαιο 1

Εισαγωγή

Η αλγοριθμική και υπολογιστική σκέψη αναγνωρίζονται πλέον ως ιδιαίτερα σημαντικές δεξιότητες, απαραίτητες για όλους τους μαθητές, αυριανούς πολίτες και εργαζόμενους. Στην Ελλάδα, τα Αναλυτικά Προγράμματα Σπουδών (ΑΠΣ) για τα εισαγωγικά μαθήματα προγραμματισμού, τόσο στην πρωτοβάθμια, όσο και στη δευτεροβάθμια εκπαίδευση, συνιστούν, ιδιαίτερα στους μικρότερους μαθητές, την αξιοποίηση LOGO-like εργαλείων και περιβαλλόντων οπτικού προγραμματισμού.

Παράλληλα, είναι πραγματικότητα η μεγάλη διείσδυση των «έξυπνων» φορητών συσκευών στην παγκόσμια αγορά και η απήχηση που έχουν στις νεότερες γενιές, γεγονότα τα οποία μπορούν να αξιοποιηθούν για τη διευκόλυνση της διδασκαλίας και μάθησης βασικών εννοιών προγραμματισμού στους χρήστες τους.

Με την πεποίθηση ότι περιβάλλοντα κατάλληλα σχεδιασμένα για αρχάριους στον προγραμματισμό χρήστες και προσαρμοσμένα για εκπαιδευτική χρήση, δίνουν κίνητρα στους μαθητές, θεωρούμε ότι οι μαθητές μπορούν να δημιουργούν αυθεντικές εφαρμογές, άμεσα αξιοποιήσιμες από τους ίδιους αλλά από τον περίγυρό τους, για συσκευές (κινητά τηλέφωνα και ταμπλέτες), τις οποίες αγαπάνε και χρησιμοποιούν σε καθημερινή βάση.

1.1 Αντικείμενο της Μεταπτυχιακής Διατριβής

Αντικείμενο της μεταπτυχιακής διατριβής είναι η διερεύνηση της αξιοποίησης κινητών συσκευών για τη διδασκαλία και μάθηση θεμελιωδών εννοιών προγραμματισμού. Ειδικότεροι στόχοι ήταν:

- Η διερεύνηση της θέσης και του περιεχομένου διδασκαλίας του προγραμματισμού, σε σχέση με την επιστήμη της πληροφορικής, στην πρωτοβάθμια και δευτεροβάθμια εκπαίδευση της Ελλάδας.
- Η επιλογή κατάλληλου προγραμματιστικού περιβάλλοντος δημιουργίας εφαρμογών για κινητές συσκευές, για χρήση από μαθητές δευτεροβάθμιας εκπαίδευσης.
- Η ανάπτυξη εκπαιδευτικού υλικού για τη διδασκαλία και μάθηση συγκεκριμένων προγραμματιστικών εννοιών, με βάση τα Αναλυτικά Προγράμματα Σπουδών στο αντικείμενο του προγραμματισμού, αξιοποιώντας το επιλεγμένο περιβάλλον.
- Η πιλοτική εφαρμογή του παραγόμενου υλικού σε τάξη και η αξιολόγησή του από διδάσκοντες και μαθητές.
- Η ανάλυση των δεδομένων που προκύπτουν και η εξαγωγή συμπερασμάτων σχετικά με τις δυσκολίες που αντιμετώπισαν οι μαθητές, τις στάσεις που διαμορφώθηκαν, τις γνώσεις και δεξιότητες που αποκτήθηκαν.

1.2 Σκοπός έρευνας

Ο σκοπός της παρούσης μεταπτυχιακής διατριβής είναι η διερεύνηση της δυνατότητας αξιοποίησης του App Inventor, ενός προγραμματιστικού περιβάλλοντος δημιουργίας εφαρμογών για ψηφιακές φορητές συσκευές με λειτουργικό σύστημα Android, για τη

διευκόλυνση της διδασκαλίας και μάθησης βασικών εννοιών προγραμματισμού στη δευτεροβάθμια εκπαίδευση.

1.2.1 Ερευνητικά ερωτήματα

- Είναι εφικτή η αξιοποίηση των κινητών συσκευών για τη βελτίωση και διευκόλυνση της διδασκαλίας και μάθησης βασικών προγραμματιστικών εννοιών στη δευτεροβάθμια εκπαίδευση;
- Σε ποιους τομείς συμβάλλει το App Inventor στην αποτελεσματικότερη μάθηση βασικών προγραμματιστικών εννοιών από τους μαθητές και που όχι;
- Ποιες είναι οι απόψεις των μαθητών για προγραμματιστικά περιβάλλοντα οπτικού προγραμματισμού όπως το App Inventor;
- Πως κρίνεται από τους εκπαιδευτικούς η αξιοποίηση του App Inventor για τη βελτίωση του διδακτικού τους έργου στη διδασκαλία βασικών προγραμματιστικών εννοιών;

1.2.2 Συμβολή της εργασίας

Η εργασία αυτή δείχνει πως η μάθηση και διδασκαλία προγραμματιστικών εννοιών στην μέση εκπαίδευση της Ελλάδας μπορεί να επωφεληθεί, αξιοποιώντας τη μεγάλη διείσδυση και χρήση κινητών συσκευών (smartphones, tablets, κλπ.), από τους εκπαιδευτικούς και τους μαθητές.

Σε πολλές χώρες ¹ γίνονται προσπάθειες ένταξης της χρήσης των κινητών συσκευών που διαθέτουν μαθητές και εκπαιδευτικοί στη διαδικασία της μάθησης, τόσο εντός, όσο και εκτός της σχολικής τάξης, με θετικά αποτελέσματα. Μέχρι την έναρξη της παρούσης διατριβής, δεν εντοπίσαμε άλλες έρευνες αξιοποίησης των κινητών συσκευών από μαθητές για τη διδασκαλία και μάθηση εννοιών πληροφορικής στην Ελλάδα.

Τα προσδοκώμενα αποτελέσματα αναμένεται να δείξουν πως προγραμματιστικά περιβάλλοντα όπως το App Inventor επιτρέπουν την άμεση και εύκολη δημιουργία πλήρως λειτουργικών αυθεντικών εφαρμογών για Android συσκευές από τους μαθητές, οι οποίες έχουν νόημα για

¹ “Tablets for Schools” project: <http://www.tabletsforschools.org.uk/>

τους ίδιους, αφού μπορούν να αξιοποιηθούν άμεσα στην καθημερινότητά τους, αλλά και να διαμοιραστούν και χρησιμοποιηθούν και από άλλους.

Μέχρι την έναρξη της παρούσης διατριβής υπήρχαν επίσης ελάχιστες αναφορές για τη χρήση του AI στην ελληνική εκπαίδευση. Η δημιουργία και χρήση κατάλληλων μαθησιακών δραστηριοτήτων για βασικές έννοιες προγραμματισμού, όπως οι προτεινόμενες, με βασικό εργαλείο το App Inventor, είναι δυνατό να επιτύχουν υψηλότερα επίπεδα δέσμευσης και δημιουργικότητας από τους μαθητές, με αποτέλεσμα να διευκολύνεται η διδασκαλία και μάθηση θεμελιωδών εννοιών αλγοριθμικής.

1.3 Η δομή της Μεταπτυχιακής Διατριβής

Η μεταπτυχιακή διατριβή αναπτύσσεται σε 9 κεφάλαια.

Στο πρώτο κεφάλαιο γίνεται μια σύντομη εισαγωγή στο αντικείμενο της μεταπτυχιακής διατριβής, στο σκοπό της, στα ερευνητικά ερωτήματα που εξετάζει, καθώς και στη συμβολή της και στο ιδιαίτερο ενδιαφέρον που παρουσιάζει το εξεταζόμενο θέμα.

Στο δεύτερο κεφάλαιο, μέσα από μια επισκόπηση της διεθνούς και ελληνικής βιβλιογραφίας, παρουσιάζεται η σημασία της Πληροφορικής ως διδακτικό αντικείμενο στην Πρωτοβάθμια (Π/θμια) και Δευτεροβάθμια (Δ/θμια) εκπαίδευση. Γίνεται ιδιαίτερη αναφορά στην ανάγκη ανάπτυξης ικανοτήτων υπολογιστικής σκέψης από τους μαθητές και, κατ' επέκταση, στην ανάγκη οι μαθητές να γνωρίζουν βασικές αλγοριθμικές έννοιες, εξασκούμενοι με τον προγραμματισμό Η/Υ.

Στο τρίτο κεφάλαιο, παρουσιάζεται η τρέχουσα θέση του προγραμματισμού ως διδακτικό αντικείμενο στην ελληνική Πρωτοβάθμια και Δευτεροβάθμια εκπαίδευση.

Στο τέταρτο κεφάλαιο, με βάση τα προσδοκώμενα μαθησιακά αποτελέσματα που αφορούν στις βασικές έννοιες προγραμματισμού, όπως αυτά παρουσιάζονται στα αντίστοιχα Αναλυτικά Προγράμματα Σπουδών, γίνεται βιβλιογραφική ανασκόπηση των συνηθέστερων προβλημάτων που αντιμετωπίζουν οι μαθητές κατά τη μάθησή τους.

Στο πέμπτο κεφάλαιο γίνεται μια επισκόπηση των δημοφιλέστερων πλατφορμών για έξυπνες φορητές συσκευές στην παγκόσμια και ελληνική αγορά. Εξετάζεται η προοπτική αξιοποίησης έξυπνων φορητών συσκευών στο σχολείο και ειδικότερα για τη διδασκαλία και τη μάθηση βασικών εννοιών προγραμματισμού. Γίνεται αναφορά στα βασικά προγραμματιστικά εργαλεία ανάπτυξης εφαρμογών για τις δυο δημοφιλέστερες πλατφόρμες και διερευνάται η καταλληλότητά τους για χρήση και αξιοποίησή τους από μαθητές και εκπαιδευτικούς σε σχολικό περιβάλλον.

Στο έκτο κεφάλαιο παρουσιάζεται συνοπτικά το εργαλείο App Inventor, τα βασικά χαρακτηριστικά του, οι δυνατότητές του και το περιβάλλον λειτουργίας του.

Στο έβδομο κεφάλαιο παρουσιάζεται η μεθοδολογία της έρευνας (σχεδιασμός, χαρακτηριστικά δείγματος, μέθοδοι και μέσα συλλογής δεδομένων, τρόπος οργάνωσης εργαστηρίων πειράματος, ρόλοι εμπλεκόμενων) και γίνεται μια συνοπτική παρουσίαση των εκπαιδευτικών δραστηριοτήτων που αναπτύχθηκαν.

Στο όγδοο κεφάλαιο παρουσιάζονται τα ευρήματα και γίνεται συζήτηση των αποτελεσμάτων από την παρατήρηση, τις συνεντεύξεις των εκπαιδευτικών και τη στατιστική ανάλυση των απαντήσεων των μαθητών στα ερωτηματολόγια.

Στο ένατο κεφάλαιο παρατίθενται τα συμπεράσματα από την αξιοποίηση του App Inventor ως εργαλείου για τη μάθηση βασικών προγραμματιστικών εννοιών, μαζί με προτάσεις για μελλοντική έρευνα.

Η μεταπτυχιακή διατριβή ολοκληρώνεται με τις Βιβλιογραφικές Αναφορές και το Παράρτημα που περιλαμβάνει τα ερωτηματολόγια, αποσπάσματα συνεντεύξεων, πίνακες από τη στατιστική ανάλυση, καθώς και δείγματα από τα φύλλα παρατήρησης και το εκπαιδευτικό υλικό που αναπτύχθηκαν για τα εργαστήρια του πειράματος.

Κεφάλαιο 2

Ο Προγραμματισμός

Υπολογιστών

στην Π/θμια και Δ/θμια

Εκπαίδευση

Στο κεφάλαιο αυτό παρουσιάζεται η σημασία της πληροφορικής ως διδακτικό αντικείμενο στην Πρωτοβάθμια και Δευτεροβάθμια εκπαίδευση και γίνεται ιδιαίτερη αναφορά στην ανάγκη ανάπτυξης ικανοτήτων υπολογιστικής σκέψης από τους μαθητές. Ειδικότερα, τονίζεται η ανάγκη οι μαθητές να έρθουν σε επαφή με τον προγραμματισμό και να εμβαθύνουν σε βασικές αλγοριθμικές έννοιες.

2.1 Η Πληροφορική ως γνωστικό αντικείμενο

Η επιστήμη της Πληροφορικής (Computer Science) και οι εξαρτημένες από αυτήν τεχνολογίες βρίσκονται σήμερα στο επίκεντρο της παγκόσμιας οικονομίας και επηρεάζουν σε μεγάλο βαθμό τον τρόπο ζωής των πολιτών. Μάλιστα, οι έννοιες που σχετίζονται με την πληροφορική έχουν

τέτοιο βάθος και εύρος, που βρίσκουν αναρίθμητες εφαρμογές, τόσο στην καθημερινότητα, όσο και σε άλλες επιστήμες [12].

Προκειμένου οι πολίτες να είναι κατάλληλα εκπαιδευμένοι και προετοιμασμένοι για την επαγγελματική τους σταδιοδρομία στον 21^ο αιώνα, οφείλουν ως μαθητές να έχουν αποκτήσει σαφή εικόνα και να έχουν κατανοήσει τις βασικές αρχές και πρακτικές της επιστήμης της πληροφορικής, ανεξαρτήτως του πεδίου σπουδών ή του επαγγέλματος που επιθυμούν να ακολουθήσουν.

Προτού παρουσιάσουμε ορισμένες σημαντικές διεθνείς αναφορές για τη σημασία της πληροφορικής ως διδακτικό αντικείμενο στην εκπαίδευση, ενδεικτικά αναφέρουμε ότι ο πρόεδρος της Google, Eric Schmidt, σε πρόσφατη ομιλία του ² κατά την επίσκεψή του στην Αγγλία, επέκρινε το εκπαιδευτικό σύστημα της χώρας, λέγοντας: *«Με έκπληξη, διαπίστωσα ότι η επιστήμη υπολογιστών δεν αποτελεί μέρος της βασικής εκπαίδευσης στα σχολεία σας. Το πρόγραμμα σπουδών σας εστιάζει στην εκμάθηση χρήσης λογισμικού, αλλά καθόλου στο πως αυτό δημιουργείται...»*. Χαρακτηριστική είναι, επίσης, η δήλωση του Προέδρου των Η.Π.Α. Barack Obama, στα πλαίσια της πρωτοβουλίας «Hour of Code» ³, που μεταξύ άλλων αναφέρει: *«Μην παίζετε μόνο παιχνίδια στο τηλέφωνό σας, αλλά επιχειρήστε να το προγραμματίσετε...»*.

2.1.1 Θέσεις της ACM για τη σημασία της πληροφορικής στην εκπαίδευση

Στην αναφορά της διεθνούς επιστημονικής και εκπαιδευτικής ένωσης επιστημόνων πληροφορικής ACM (Association for Computing Machinery) για τη θέση της πληροφορικής στην εκπαίδευση των Η.Π.Α. [63], τονίζεται ότι η πληροφορική αποτελεί κινητήρια δύναμη ανάπτυξης και νέων επιστημονικών ανακαλύψεων, κάτι το οποίο καθιστά ακόμα σημαντικότερη τη διάκριση της επιστήμης της πληροφορικής από τη χρήση των εφαρμογών της. Παρόλα αυτά, η σύγχυση που επικρατεί γύρω από το θέμα αυτό, διαχέεται σε προγράμματα εκπαίδευσης πολλών σχολείων στις Η.Π.Α., αλλά και παγκοσμίως, με την πληροφορική να μην λογίζεται ως αυτόνομο διδακτικό αντικείμενο, παρά την εξέχουσα σημασία της στην ψηφιακή εποχή που ζούμε.

² Ομιλία Eric Schmidt στο Διεθνές Φεστιβάλ Τηλεόρασης (Εδιμβούργο, Αύγουστος 2011): <http://www.bbc.co.uk/news/uk-14683133>

³ Εβδομάδα Πληροφορικής – Ώρα Κώδικα (CSEdWeek, 9-15 Δεκεμβρίου 2013): <http://csedweek.org/>

Στο πλαίσιο της ορθότερης αντιμετώπισης της πληροφορικής, η ένωση ACM, από το 2003 (με μια αναθεώρηση το 2006) έχει αναπτύξει ένα πρότυπο πρόγραμμα σπουδών για την πληροφορική στην εκπαίδευση.

Συγκεκριμένα, το πρόγραμμα σπουδών “ACM K-12 Computer Science Model Curriculum”⁴ προτείνεται να χρησιμοποιηθεί διεθνώς στην πρωτοβάθμια και δευτεροβάθμια εκπαίδευση, εστιάζοντας στις θεμελιώδεις και διαχρονικές έννοιες της επιστήμης της πληροφορικής και στο πως αυτές πρέπει να διδάσκονται.

Ανάμεσα στους βασικούς σκοπούς της εκπαίδευσης για τους μαθητές, βρίσκεται η ανάπτυξη ικανοτήτων αξιοποίησης της αλγοριθμικής σκέψης για την επίλυση προβλημάτων διαφόρων αντικειμένων. Εξάλλου, η ACM ορίζει την πληροφορική ως *«την επιστήμη που μελετά τους υπολογιστές και τις αλγοριθμικές διαδικασίες, συμπεριλαμβανομένων των αρχών τους, της σχεδίασης του υλικού και λογισμικού τους, των εφαρμογών τους και της επίδρασής τους στην κοινωνία»*.

2.1.2 Θέσεις της Ομάδας Εργασίας “Computing at School”

Η ομάδα εργασίας “Computing at School”⁵ στοχεύει στην προώθηση και αναβάθμιση της διδασκαλίας της πληροφορικής στα σχολεία. Στο πλαίσιο αυτό, έχει εκδώσει πλήθος αναφορών, μεταξύ των οποίων και ένα ιδιαίτερος δημοφιλές προτεινόμενο Πρόγραμμα Σπουδών⁶ για το τι πρέπει να περιλαμβάνει το μάθημα της πληροφορικής στο σχολείο.

Σύμφωνα με αυτό το ΠΣ, οι συνιστώσες της πληροφορικής, με την ευρύτερη έννοια, είναι: η επιστήμη της πληροφορικής (CS: Computer Science), οι τεχνολογίες της πληροφορίας (IT: Information Technology), ο ψηφιακός εγγραμματισμός (dl: digital literacy) και η μάθηση υποβοηθούμενη από την τεχνολογία (TEL: Technology Enhanced Learning).

Οι τεχνολογίες της πληροφορίας ή οι τεχνολογίες πληροφορίας και επικοινωνιών (ICT: Information and Communication Technology), ως διδακτικό αντικείμενο, στοχεύουν στην ορθολογική και αποτελεσματική χρήση των βασικών εργαλείων που σχετίζονται με την πληροφορική, κατά αντίστοιχο τρόπο με τη δημοφιλή ρήση που αποδίδεται στον Edsger Dijkstra

⁴ ACM K-12 CS Model Curriculum, 2nd Edition: <http://csta.acm.org/Curriculum/sub/CurrResources.html>

⁵ Computing at School (CAS) Working Group: <http://www.computingatschool.org.uk>

⁶ Computer Science: A curriculum for schools, March 2012: <http://www.computingatschool.org.uk/index.php?id=cacfs>

πως «*Η επιστήμη της πληροφορικής έχει τέτοια σχέση με τους υπολογιστές, όπως έχει η αστρονομία με το τηλεσκόπιο*» [33]. Αυτή η αντιπαράθεση επικεντρώνεται στην αποκλειστική θεώρηση και χρήση των υπολογιστών ως τεχνολογίας και στην ικανότητα ψηφιακού εγγραμματισμού, ενώ οι υπολογιστές συνιστούν κάτι ευρύτερο από αυτό.

Επιπλέον, στη συγκεντρωτική συγκριτική αναφορά του “Computing at School” για το πώς προσεγγίζεται η πληροφορική ως μάθημα στη δευτεροβάθμια εκπαίδευση [47], ο Payton Jones διαπιστώνει ότι σε πολλές χώρες, η πληροφορική αναγνωρίζεται ως αυτόνομο αντικείμενο, διακριτό από τις δεξιότητες της χρήσης των υπολογιστών. Είναι, επίσης, ευρέως αποδεκτό ότι η επιστήμη της πληροφορικής έχει τεράστια εκπαιδευτικά οφέλη (ικανότητες σκέψης επίλυσης προβλημάτων), καθώς και οικονομικά (ανάγκες αγοράς εργασίας). Καταλήγει λοιπόν στο συμπέρασμα ότι θα έπρεπε να αποτελεί μέρος της εκπαίδευσης κάθε μαθητή.

Παρά τις ιδιαιτερότητες των εκπαιδευτικών συστημάτων κάθε χώρας, η αναφορά επισημαίνει ορισμένα χαρακτηριστικά «μαθήματα» που μπορούμε να πάρουμε, από τη διεθνή εμπειρία:

- Είναι ζωτικής σημασίας ο διαχωρισμός μεταξύ Επιστήμης Υπολογιστών (Computer Science) από τη μια και Εφαρμογών Πληροφορικής (IT applications) ή ψηφιακού γραμματισμού (Digital Literacy) από την άλλη.
- Πολλές χώρες έχουν εντάξει στα σχολικά προγράμματα, μαθήματα περί Επιστήμης Υπολογιστών, ήδη από την πρωτοβάθμια εκπαίδευση, επιτρέποντας στους μαθητές τη γνωριμία με αυτήν, από πολύ μικρή ηλικία.
- Υπάρχει ολοένα και μεγαλύτερη σαφήνεια ως προς το ότι η πληροφορική δεν ταυτίζεται με την εκμάθηση μιας γλώσσας προγραμματισμού, όπως η Java ή η C++. Ναι μεν ο προγραμματισμός κατέχει σημαντική θέση, αλλά οι βασικές αρχές αλγοριθμικής, δομών δεδομένων, καθώς και οι ικανότητες υπολογιστικής σκέψης, αποτελούν πιο διαχρονικά και πιο ουσιώδη στοιχεία.
- Η σύγχυση μεταξύ Πληροφορικής και δεξιοτήτων χρήσης των Τεχνολογιών Πληροφοριών και Επικοινωνίας (ΤΠΕ) συνδυάζεται πολύ συχνά με τη διδασκαλία του αντικειμένου από άτομα ανειδίκευτα και χωρίς ουσιαστικό υπόβαθρο. Παράλληλα, η μεγάλη απορρόφηση επιστημόνων πληροφορικής στην αγορά, μειώνει τη δεξαμενή

διαθέσιμων, ικανών και προσοντούχων, εκπαιδευτικών. Τα παραπάνω γεγονότα συμβάλουν στη γενικότερη υποβάθμιση του αντικειμένου στην εκπαίδευση.

2.1.3 Αναφορά της CSTA για την πληροφορική στην Κ-8 βαθμίδα εκπαίδευσης

Η Ένωση Εκπαιδευτικών Πληροφορικής CSTA (Computer Science Teacher Association) με έδρα τις Η.Π.Α. και συνεργαζόμενα παραρτήματα σε πολλές ακόμα χώρες, δραστηριοποιείται στην ανάπτυξη και πρόταση καινοτόμων πρακτικών και στρατηγικών για την εισαγωγή της επιστήμης υπολογιστών στην εκπαίδευση.

Σε αναφορά της για την πρωτοβάθμια εκπαίδευση ⁷ (Κ-8: ηλικίες 5 μέχρι 12), αναφέρεται στην πρόταση του Papert [41], από το 1980 ακόμα, για ευρεία χρήση υπολογιστών στην εκπαίδευση, εξαιρώντας τη δυνατότητά τους *«να πάρουν χιλιάδες μορφές και να προσφέρουν χιλιάδες λειτουργίες, οπότε και να ταιριάζουν σε χιλιάδες διαφορετικά γούστα»*.

Στην πρότασή τους, λοιπόν, διαφαίνονται πως η δύναμη των υπολογιστών έγκειται στην δυνατότητά τους να αποτελέσουν κάτι σημαντικό για τον οποιονδήποτε. Η επιστήμη υπολογιστών, επομένως, αποτελεί το μέσο που επιτρέπει στους μαθητές να πάρουν τον έλεγχο, να επιλύσουν προβλήματα και να χτίσουν το μέλλον, με εργαλεία τη φαντασία και τη δημιουργικότητά τους.

Θεωρούν, εν τέλει, αδιανόητο να θέλει κανείς να χαθεί αυτή η ευκαιρία, του να παρέχουμε δηλαδή στους μαθητές τη δύναμη που ανοίγει το δρόμο για πολλά επιτεύγματα, που δεν μπορούμε καν να κατονομάσουμε ακόμα.

2.2 Υπολογιστική Σκέψη

Η επιστήμη της πληροφορικής συνδέεται βαθύτατα με τη μελέτη του τρόπου λειτουργίας των υπολογιστικών συστημάτων, καθώς και του σχεδιασμού και προγραμματισμού τους. Η ικανότητα αυτή αναφέρεται συχνά ως υπολογιστική ευχέρεια [78]. Παρόλο που ο τεχνολογικός γραμματισμός και η υπολογιστική ευχέρεια είναι απαραίτητες ως ικανότητες, δεν επαρκούν για να μπορεί ο άνθρωπος να κατανοήσει σε βάθος τη δυναμική που αναπτύσσεται από τον ανθρώπινο εγκέφαλο κατά τη διαδικασία των υπολογισμών, σκοπεύοντας στην αύξηση της

⁷ CSTA "Computer Science K-8: Building a Strong Foundation."
<http://csta.acm.org/Curriculum/sub/CSK8.html>

δημιουργικότητάς του. Εξάλλου, η ουσία βρίσκεται στον τρόπο με τον οποίο σκέπτονται οι άνθρωποι και όχι οι υπολογιστές, καθώς οι άνθρωποι είναι δημιουργικοί, ενώ οι υπολογιστές όχι.

2.2.1 Τι είναι η υπολογιστική σκέψη

Η πρώτη επίσημη αναφορά στον όρο υπολογιστική σκέψη (computational thinking) έγινε από την Wing το 2006 [64]. Όπως αναφέρει η ίδια, η υπολογιστική σκέψη είναι ένα σύνολο κρίσιμων διανοητικών και λογικών ικανοτήτων, ισοδύναμης σημασίας με τη γραφή, την ανάγνωση και την αριθμητική [65]. Συνιστά ένα ζωτικής σημασίας συστατικό για τη μάθηση της επιστήμης, της τεχνολογίας, της μηχανικής και των μαθηματικών (STEM: Science, Technology, Engineering and Mathematics).

Σε νεότερο άρθρο της το 2010, η Wing [66] ορίζει την υπολογιστική σκέψη ως «τις νοητικές διεργασίες που περιλαμβάνονται στη διατύπωση και επίλυση προβλημάτων, κατά την οποία η αποτύπωση των λύσεων γίνεται σε μορφή που να μπορούν να εκτελεστούν από κάποιον πράκτορα επεξεργασίας πληροφορίας».

Ο Aho, το 2012 [01], απλοποιεί ακόμα περισσότερο τον παραπάνω ορισμό, ορίζοντας την υπολογιστική σκέψη ως «τις νοητικές διεργασίες που περιλαμβάνονται στη διατύπωση και επίλυση προβλημάτων, κατά την οποία η αποτύπωση της λύσης μπορεί να αναπαρασταθεί με υπολογιστικά βήματα σε μορφή αλγορίθμου».

2.2.2 Η σημασία της υπολογιστικής σκέψης στην εκπαίδευση

Η πρόοδος στην επιστήμη υπολογιστών, κατά την Wing [65], επιτρέπει στους ερευνητές, ανεξαρτήτως πεδίου, να οραματιστούν νέες στρατηγικές επίλυσης προβλημάτων και να ελέγξουν τις νέες λύσεις, τόσο στον πραγματικό, όσο και στον εικονικό κόσμο. Αυτή η πρόοδος, με τη σειρά της, καθιστά αναγκαία την εκπαίδευση που θα ενισχύσει και θα επεκτείνει τη θέση της υπολογιστικής σκέψης στην επίλυση προβλημάτων, σε ακόμα πιο ευρεία κλίμακα [03].

Οι μαθητές, λοιπόν, στο μάθημα της επιστήμης της πληροφορικής έρχονται σε επαφή με υπολογιστικά συστήματα όλων των ειδών, είτε περιέχουν υπολογιστές, είτε όχι, τα εξερευνούν και τα μοντελοποιούν με υπολογιστικούς όρους και αφαιρετική προσέγγιση. Αυτή ακριβώς η προσέγγιση επιτρέπει και τη διαχείριση της πολυπλοκότητας των συστημάτων, καθώς η διαδικασία της αφαίρεσης μπορεί να πάρει πολλές μορφές, όπως μοντελοποίηση, ανάλυση και

γενίκευση. Οι Κοτίνη και Τζελέπη [78] θεωρούν ότι η ανάπτυξη και καλλιέργεια αυτών των ικανοτήτων θα πρέπει να διακατέχει όλο το φάσμα της δευτεροβάθμιας εκπαίδευσης.

2.2.3 Το πιλοτικό εκπαιδευτικό πρόγραμμα “CS Principles”

Ειδικά σε ότι αφορά τα Προγράμματα Σπουδών στη Δευτεροβάθμια Εκπαίδευση, η μη κερδοσκοπική οργάνωση “College Board” και το Εθνικό Ίδρυμα Ερευνών NSF (National Science Foundation) των Η.Π.Α., μέσα από το πρόγραμμα “CS Principles”⁸ αναπτύσσουν ένα νέο πιλοτικό μάθημα, το οποίο εστιάζει στις πρακτικές της υπολογιστικής σκέψης και βασίζεται στις επτά κυριότερες ιδέες (big ideas) της επιστήμης υπολογιστών:

1. Η υπολογιστική είναι μια δημιουργική ανθρώπινη δραστηριότητα που προωθεί την καινοτομία, την εξερεύνηση και την παραγωγή γνώσης.
2. Η αφαίρεση (abstraction) μειώνει τον όγκο των πληροφοριών και των λεπτομερειών, για να δοθεί βαρύτητα στις έννοιες που σχετίζονται με την κατανόηση και την επίλυση των προβλημάτων.
3. Τα δεδομένα και οι πληροφορίες διευκολύνουν την παραγωγή γνώσης.
4. Οι αλγόριθμοι είναι εργαλεία για την ανάπτυξη και τη διατύπωση λύσεων σε υπολογιστικά προβλήματα.
5. Ο προγραμματισμός είναι μια δημιουργική διαδικασία που οδηγεί στην παραγωγή υπολογιστικών τεχνουργημάτων.
6. Οι ψηφιακές συσκευές και συστήματα, καθώς και τα δίκτυα για τη διασύνδεσή τους, επιτρέπουν και διευκολύνουν την υπολογιστική προσέγγιση για την επίλυση προβλημάτων.
7. Η πληροφορική προάγει την καινοτομία σε πολλά πεδία, όπως στις κοινωνικές και ανθρωπιστικές επιστήμες, στις τέχνες, στην ιατρική, στη μηχανική και στις επιχειρήσεις.

⁸ The College Board “Computer Science: Principles”, CT Practices: <http://www.csprinciples.org/>

Η υπολογιστική σκέψη επηρεάζει επιστημονικά πεδία, όπως τη βιολογία, τη χημεία, τη γλωσσολογία, τη ψυχολογία, τα οικονομικά και τη στατιστική. Επιτρέπει την επίλυση προβλημάτων, το σχεδιασμό συστημάτων, καθώς και την κατανόηση της ισχύος αλλά και των ορίων, τόσο της ανθρώπινης, όσο και της υπολογιστικής νοημοσύνης, ικανότητες απαραίτητες για να ανταπεξέλθουν οι νέοι στις απαιτήσεις της σύγχρονης κοινωνίας.

2.2.4 Υπολογιστική σκέψη και πρακτική

Ένας από τους πρωταρχικούς στόχους της εκπαίδευσης πρέπει να είναι η διαμόρφωση του ρόλου των μαθητών από παθητικοί καταναλωτές μιας αδιαφανούς και μυστηριώδους τεχνολογίας, σε γνώστες, δημιουργοί και συνδιαμορφωτές του ψηφιακού κόσμου που τους περιβάλλει.

Η υπολογιστική σκέψη αποτελεί μια προσέγγιση επίλυσης προβλημάτων, με τέτοιο τρόπο, ώστε να μπορεί να υλοποιηθεί από υπολογιστή. Έτσι, οι μαθητές μετατρέπονται από απλοί χρήστες εργαλείων, σε δημιουργοί τους. Εξοικειώνονται με τη χρήση ενός συνόλου από έννοιες, όπως αφαίρεση, αναδρομή και επανάληψη, για την επεξεργασία και ανάλυση δεδομένων, και για την ανάπτυξη πραγματικών ή εικονικών τεχνημάτων/τεχνουργημάτων.

Το πρακτικό μέρος του αντικείμενου ενθαρρύνει την επινοητικότητα και την εφευρετικότητα. Αναμένεται από τους μαθητές η πρακτική εφαρμογή των ακαδημαϊκών αρχών που μαθαίνουν περί συστημάτων, για την παραγωγή υπολογιστικών έργων με νόημα και για συγκεκριμένους σκοπούς. Αυτός ο συνδυασμός θεωρίας, πρακτικής και παραγωγικότητας, καθιστά την πληροφορική στην εκπαίδευση ως ένα εξαιρετικά χρήσιμο αντικείμενο, εμποτισμένο με τον ενθουσιασμό και τη χαρά της δημιουργίας, τόσο σε υλικό, όσο και σε πνευματικό επίπεδο.

Είναι χαρακτηριστικό της φιλοσοφίας περί υπολογιστικής σκέψης, το ότι ο Fletcher [14] τάσσεται υπέρ των προσπαθειών, οι μαθητές να αποκτούν θεμελιώδεις γνώσεις περί υπολογιστικής σκέψης, πολύ πριν έρθουν σε επαφή με κάποια γλώσσα προγραμματισμού, κάτι το οποίο μπορεί να επιτευχθεί ακόμα και μέσω του μαθήματος των μαθηματικών.

Μόνο τότε, οι μαθητές θα είναι κατάλληλα προετοιμασμένοι για να εντρυφήσουν στην επιστήμη των υπολογιστών, όχι μόνο για τις ευκαιρίες στον επαγγελματικό χώρο, αλλά και για το διανοητικό περιεχόμενο σε ακαδημαϊκό επίπεδο και να σταματήσει να ταυτίζεται η πληροφορική με τον προγραμματισμό. Παραλληλίζει τον προγραμματισμό και τη σχέση του με

την επιστήμη υπολογιστών, με τη μέθοδο της απόδειξης και τη σχέση της με τα μαθηματικά ή την λογοτεχνική ανάλυση και τη σχέση της με τα αγγλικά, αντίστοιχα.

Παράλληλα, οι Barr & Stephenson [03] ισχυρίζονται ότι όλοι οι μαθητές πρέπει να ξεκινήσουν από νωρίς την ενασχόλησή τους με αλγοριθμική επίλυση προβλημάτων και υπολογιστικές μεθόδους και εργαλεία. Οι μεθοδολογίες αυτές μπορούν μάλιστα να αυτοματοποιηθούν, να μεταφερθούν και να εφαρμοστούν σε πολλά ακόμα πεδία.

2.3 Ο προγραμματισμός Η/Υ ως διδακτικό αντικείμενο

Οι νέοι σήμερα θεωρούνται ψηφιακοί ιθαγενείς (digital natives), επειδή από τη γέννησή τους υπάρχει η ψηφιακή τεχνολογία και εξοικειώνονται πολύ γρήγορα με αυτή, ενώ οι μεγαλύτεροι ενήλικες θεωρούνται ψηφιακοί μετανάστες (digital immigrants), επειδή η ψηφιακή τεχνολογία εμφανίζεται ως νέα σε αυτούς, αφού δεν είχαν ήδη γεννηθεί [48]. Πράγματι, η πλειοψηφία των νέων σήμερα έχει ευχέρεια στην αποστολή μηνυμάτων κειμένου, στο παίξιμο online παιχνιδιών και στην πλοήγηση στον Παγκόσμιο Ιστό. Ο Resnick [53] θέτει τον προβληματισμό κατά πόσο αυτή η άνεση, είναι αρκετή για να θεωρηθεί ότι οι νέοι κατέχουν τις απαραίτητες γνώσεις περί των ψηφιακών αυτών τεχνολογιών. Μπορεί τα παιδιά να αλληλοεπιδρούν διαρκώς με τα ψηφιακά μέσα, αλλά ελάχιστα είναι ικανά να δημιουργήσουν π.χ. τα δικά τους παιχνίδια, κάτι που παρομοιάζεται με την κατοχή της ικανότητας ανάγνωσης, αλλά όχι γραφής.

Κρίνεται λοιπόν ιδιαιτέρως σημαντική η ικανότητα σχεδιασμού, δημιουργίας και επινόησης με νέα ψηφιακά μέσα [53]. Για την επίτευξη του σκοπού αυτού, απαιτείται η εκμάθηση κάποιου είδους προγραμματισμού. Η ικανότητα προγραμματισμού προσφέρει μερικά πολύ σημαντικά οφέλη, όπως ότι διευρύνει το τι μπορεί κανείς να δημιουργήσει (και να εκφραστεί) με τη χρήση υπολογιστή, καθώς και το τι μπορεί να μάθει.

Ειδικότερα, ο προγραμματισμός υποστηρίζει την ανάπτυξη υπολογιστικής σκέψης, βοηθώντας στην μάθηση στρατηγικών σχεδιασμού και επίλυσης προβλημάτων, όπως π.χ. τμηματοποίηση (modularization) και επαναληπτικότητα, που μεταφέρονται και σε άλλα πεδία, πέραν του προγραμματισμού υπολογιστών.

Επιπρόσθετα, λόγω του ότι ο προγραμματισμός περιλαμβάνει τη δημιουργία εξωτερικών αναπαραστάσεων περί διαδικασιών επίλυσης προβλημάτων από τον καθένα, παρέχεται η δυνατότητα αναπαράστασης της ατομικής σκέψης, αλλά και του προβληματισμού για αυτήν.

2.3.1 Αλγοριθμική και προγραμματισμός

Η πλέον θεμελιώδης έννοια σε σχέση με τις υπολογιστικές διαδικασίες είναι η έννοια του αλγορίθμου, όπως διαπιστώνεται και μετά από πρόσφατη βιβλιογραφική ανασκόπηση περί βασικών εννοιών πληροφορικής [82], στο σύνολο των 18 πηγών του ερευνητή από τη διεθνή βιβλιογραφία.

Ο προγραμματισμός φαίνεται να κατέχει κεντρική θέση στα προγράμματα σπουδών παγκοσμίως [49], με κρίσιμη συμβολή στην κατανόηση των υπολογιστικών διαδικασιών, επειδή παρέχει τη γλώσσα εκείνη που επιτρέπει την ανάπτυξη του αντίστοιχου νοητικού μοντέλου.

Εξάλλου, κάποιιοι τολμούν να παρουσιάσουν την αλγοριθμική (algorithms) σαν το 4^ο “R” των βασικών γνώσεων που πρέπει να παρέχει η εκπαίδευση [10]. Τα άλλα τρία R είναι η ανάγνωση, η γραφή και η αριθμητική (Reading, wRiting, aRithmetic) [64].

Όπως έχει δηλώσει άλλωστε και ο πρώτος επιστήμονας βραβευμένος με ACM Turing Award, Alan Perlis [46], *«όλοι θα έπρεπε να μαθαίνουν να προγραμματίζουν, ως μέρος της βασικής δωρεάν εκπαίδευσης»*.

2.3.2 “ACM K-12 CS Model Curriculum” και προγραμματισμός

Στο πρότυπο Πρόγραμμα Σπουδών της ACM για την Επιστήμη Υπολογιστών στην βασική εκπαίδευση (K-12: ηλικίες 7 μέχρι 18), ο προγραμματισμός κατέχει σημαντική θέση.

Από παιδαγωγικής σκοπιάς, ο προγραμματισμός σχετίζεται με τη μελέτη της πληροφορικής, όπως σχετίζεται η εκμάθηση ενός μουσικού οργάνου με τη μελέτη της μουσικής και όπως σχετίζεται η ζωγραφική με τη μελέτη της τέχνης. Σε κάθε περίπτωση, η οποιαδήποτε πρακτική εμπειρία ενισχύει τη μακροπρόθεσμη εκτίμηση και κατανόηση του αντικειμένου, ακόμα κι αν ο μαθητής δεν ασχοληθεί με τον προγραμματισμό, το μουσικό όργανο ή τη ζωγραφική ως

ενήλικος. Παρά το γεγονός ότι για να γίνει κάποιος έμπειρος και ικανός προγραμματιστής, πιανίστας ή ζωγράφος απαιτείται πολύς χρόνος και ταλέντο, είναι πολύ σημαντικό να παρέχεται σε όλους τους μαθητές η δυνατότητα και η χαρά του να αισθάνονται δημιουργικοί.

Στόχος, επομένως της διδασκαλίας της πληροφορικής πρέπει να είναι η ανάθεση και ενασχόληση των μαθητών με ελκυστικές πρακτικές δραστηριότητες. Τέτοιου είδους δραστηριότητες, που να προκαλούν το ενδιαφέρον των μαθητών, είναι ο σχεδιασμός και η συγγραφή προγραμμάτων για τον έλεγχο των φορητών τους συσκευών ή κάποιας ρομποτικής κατασκευής, η σύνθεση μουσικής κ.α.. Με αυτό τον τρόπο οι μαθητές θα επιδιώξουν οι ίδιοι να μάθουν να χρησιμοποιούν μεταβλητές, συνθήκες, επαναληπτικές δομές και διάφορες άλλες θεμελιώδεις για την πληροφορική έννοιες, προκειμένου να επιτύχουν τους σκοπούς τους.

2.3.3 Προβληματισμοί για τη θέση του προγραμματισμού στο σχολείο

Παρά τον αρχικό ενθουσιασμό που διακατείχε τη χρήση των γλωσσών προγραμματισμού στην εκπαίδευση, δεν είναι λίγοι αυτοί που θεωρούν τον προγραμματισμό, μια εξειδικευμένη, τεχνική δραστηριότητα που απευθύνεται σε λίγους [48]. Μια συνηθισμένη, λοιπόν, παρανόηση των παιδιών είναι η θεώρηση του προγραμματισμού ως δύσκολη και ανιαρή δραστηριότητα, που απαιτεί υψηλού επιπέδου μαθηματικές ικανότητες.

Ερευνητές παρατηρούν επίσης το γενικότερο πρόβλημα της μειωμένης προθυμίας των μαθητών να εγγράφονται σε προγράμματα σπουδών πληροφορικής και θεωρούν ότι αυτό επιδεινώνεται γιατί οι μαθητές βρίσκουν βαρετά τα μαθήματα που αφορούν σε θεμελιώδεις έννοιες πληροφορικής [31].

Ο Repenning [51], μάλιστα, δικαιολογεί τον σκεπτικισμό γύρω από τον προγραμματισμό στο σχολείο, αναφέροντας το χαρακτηριστικό παράδειγμα μαθητή, ο οποίος, αν και αρχικά ενθουσιασμένος στη θέα των LEGO blocks, μόλις αντιλήφθηκε ότι επρόκειτο να χρησιμοποιηθούν για εκμάθηση προγραμματισμού, δυσανασχέτησε, θεωρώντας ότι θα αντιγράψει κώδικα από τον πίνακα στον υπολογιστή, για να μη λειτουργεί τελικά.

Στην προσπάθεια του να εξηγήσει τους παράγοντες που οδήγησαν σε αυτή την κατάσταση, ο Prensky [48] κατέληξε στα εξής:

- Οι παλιότερες γλώσσες προγραμματισμού ήταν δύσχρηστες και πολλά παιδιά αντιμετώπιζαν δυσκολίες με το συντακτικό τους.
- Η εισαγωγή στον προγραμματισμό γινόταν μέσα από δραστηριότητες που δεν συνδέονταν με τα ενδιαφέροντα και τις εμπειρίες των παιδιών.
- Το πλαίσιο υπό το οποίο διδάσκονταν οι μαθητές δεν ήταν έτοιμο για την παροχή κατάλληλης καθοδήγησης όταν υπήρχαν δυσκολίες, ούτε για την ενθάρρυνση βαθύτερης εξερεύνησης όταν όλα κυλούσαν ομαλά και επιτυχώς.

Ο Guzdial [22] αναφέρθηκε στα μεγάλα ποσοστά μη συμμετοχής ή αποτυχίας των κοριτσιών σε μαθήματα επιστήμης υπολογιστών που καταγράφηκαν στην Αμερική⁹, λόγω της αντίληψης ότι τα μαθήματα αυτά ήταν υπερβολικά τεχνικά και δεν άφηναν περιθώριο δημιουργικότητας σε προσωπικό επίπεδο από τους μαθητές. Ισχυρίστηκε ότι αυτή η απομάκρυνση των παιδιών από τον τομέα της επιστήμης υπολογιστών οφείλεται στον απαρχαιωμένο τρόπο προσέγγισης του θέματος και των μαθητών. Οι εκπαιδευτικοί, όταν μάθαιναν οι ίδιοι να προγραμματίζουν, μπορεί να ενθουσιάζονταν απλά και μόνο επειδή κατάφερναν να εμφανίσουν μια πρόταση του στυλ “Hello, World” ως έξοδο των προγραμμάτων τους, αλλά τα πράγματα πλέον έχουν αλλάξει. Όπως αναφέρει χαρακτηριστικά, η «γενιά του Nintendo» απαιτείται να παράγει περισσότερο δημιουργικά έργα για να δεσμευτεί και να συμμετάσχει ενεργά στη διαδικασία της μάθησης. Οι σύγχρονες δυνατότητες των υπολογιστικών συσκευών, από την άποψη της υποστήριξης πολυμέσων και διαδραστικότητας διευκολύνουν κατά πολύ μια διαφορετική προσέγγιση των μαθημάτων πληροφορικής. Αντί η επεξεργασία να αφορά σε δεδομένα κειμένου, μπορεί κάλλιστα να σχετίζεται με πολυμεσικό υλικό, όπως μουσική, γραφικά και βίντεο.

Επιπλέον, η δημιουργία παιχνιδιών και η ρομποτική εμφανίζονται συχνά ως δραστηριότητες σε σχολικά προγράμματα σπουδών και προτείνονται τόσο ως μέσα που παρέχουν κίνητρα και λειτουργούν θετικά για τη δέσμευση των μαθητών, αλλά και ως ιδανική εισαγωγή στην επιστήμη της πληροφορικής και σαν σκαλοπάτι για τη μετάβαση σε γλώσσες υψηλότερων δυνατοτήτων, όπως η Java και η Python.

⁹ Αναφορά της AAUW (American Association of University Women) για την σχέση των κοριτσιών με την πληροφορική: <http://www.aauw.org/resource/tech-savvy-educating-girls-in-the-new-computer-age/>

2.3.4 Τα αρνητικά της χρήσης γλωσσών γενικού σκοπού

Οι Μάργαρης και Παπαστεργίου [80] παρατηρούν ότι παρά την κοινή παραδοχή για τα αναμφισβήτητα οφέλη του προγραμματισμού για τους μαθητές, στην πράξη έχει αποδειχθεί ότι η εκμάθησή του είναι μια δύσκολη υπόθεση για αρχάριους όλων των ηλικιών. Η κλασσική προσέγγιση, με τη σταδιακή παρουσίαση των δομών μιας προγραμματιστικής γλώσσας γενικού σκοπού και η επίλυση προβλημάτων αυξανόμενης δυσκολίας, κρίνεται αναποτελεσματική, ιδιαίτερα για μαθητές μικρότερης ηλικίας, καθώς θέτει μια σειρά από εμπόδια στους αρχάριους προγραμματιστές. Σύμφωνα και με τους Brusilovsky [06] και Reichert [51]:

- Απαιτείται από το μαθητή να εξοικειωθεί ταυτόχρονα τόσο με την αυστηρή σύνταξη και τη σημασιολογία της ίδιας της γλώσσας όσο και με τις βασικές αρχές του προγραμματισμού.
- Η διδασκαλία και χρήση μιας πλήρους γλώσσας προγραμματισμού στο σχολείο μπορεί να αποβεί πολύ χρονοβόρα.
- Παρέχεται συνήθως περιορισμένη υποστήριξη όσον αφορά στην κατανόηση των βασικών εντολών και δομών ελέγχου της γλώσσας αφού η διαδικασία της εκτέλεσης του προγράμματος παραμένει κρυμμένη από το μαθητή, ενώ η έλλειψη οπτικής ανάδρασης εμποδίζει την κατανόηση της σημασιολογίας της γλώσσας.
- Οι μαθητές δύσκολα μπορούν να εντοπίσουν και να διορθώσουν λάθη στα προγράμματά τους.
- Τα πρώτα προβλήματα που τίθενται στους μαθητές αφορούν κατά κανόνα στην επεξεργασία αριθμών ή συμβόλων και αποτυγχάνουν να κινηθούν το ενδιαφέρον των μαθητών, ενώ η ανάπτυξη πιο ελκυστικών εφαρμογών απαιτεί την εκμάθηση ενός μεγάλου υποσυνόλου της γλώσσας.

2.3.5 Τα οφέλη των εργαλείων οπτικού προγραμματισμού

Πολλές εργασίες έχουν αναφερθεί στους κρίσιμους παράγοντες που επηρεάζουν την επιτυχή εκμάθηση εννοιών προγραμματισμού. Πλήθος ερευνητών [11] [26] [54], συμφωνούν για την κρισιμότητα του απαιτούμενου γνωστικού φορτίου για τη χρήση του προγραμματιστικού

περιβάλλοντος. Σε αρχάριους στον προγραμματισμό είναι σημαντικό να μειώσουμε την ποσότητα της πληροφορίας που απαιτείται για να επιτευχθεί η επίλυση του προβλήματος, κάτι το οποίο διευκολύνεται από περιβάλλοντα οπτικού προγραμματισμού. Έτσι, οι χρήστες τους δεν είναι υποχρεωμένοι να θυμούνται μεγάλο αριθμό εντολών για τη συγγραφή κώδικα, αφού συνήθως αυτές επιλέγονται από λίστες έτοιμων οπτικών αντικειμένων που αντιπροσωπεύουν εντολές, έτοιμα κομμάτια κώδικα ή δομές. Συνήθως, με την τεχνική «σύρε κι άφησε» (drag n' drop) οι χρήστες συνδέουν τουβλάκια τύπου Lego μεταξύ τους και τα παραμετροποιούν, όπου επιτρέπεται. Έτσι, διευκολύνεται ο πειραματισμός με προγραμματιστικές δομές και η βαρύτερη δίνεται στη δομή των λύσεων και όχι στη σύνταξη των εντολών. Η ευκολία αυτή ευνοεί τη χρήση τέτοιων εργαλείων από άτομα όλων των ηλικιών, εκπαιδευτικού υπόβαθρου και ενδιαφερόντων. Επιτρέπει επίσης την άμεση ανατροφοδότηση στο χρήστη για το αποτέλεσμα των ενεργειών του, ενώ ταυτόχρονα καθιστά πιο ευχάριστη και εύκολη την εκσφαλμάτωση.

Δημοφιλή προγραμματιστικά περιβάλλοντα που πληρούν τα παραπάνω κριτήρια (σε διαφορετικό βαθμό το καθένα) είναι τα γραφικού περιβάλλοντος εργαλεία Scratch, Alice, Game Maker, Kodu και Greenfoot. Το σημαντικότερο πλεονέκτημα αυτών των εργαλείων είναι ότι είναι σχετικά ευκολότερα στη χρήση τους και επιτρέπουν στους αρχάριους να εστιάζουν στη σχεδίαση και στην υλοποίηση, χωρίς να προβληματίζονται για τον κίνδυνο συντακτικών λαθών [21].

2.3.6 Η γλώσσα Logo για την εκπαίδευση στον προγραμματισμό

Ο Seymour Papert ήδη από το 1980 ήταν υπέρμαχος της ιδέας του να αναπτύξουν οι μαθητές ικανότητες αναλυτικής και αλγοριθμικής σκέψης, μέσα από τον προγραμματισμό στο περιβάλλον της Logo [41][42].

Η ονομασία Logo, από την ελληνική λέξη «λόγος», επιλέχθηκε για να δώσει έμφαση στο ιδιαίτερο χαρακτηριστικό του να δίνεται έμφαση στη χρήση λέξεων και σύνταξη προτάσεων, σε αντίθεση με τα προγραμματιστικά περιβάλλοντα της τότε εποχής, που εστίαζαν στην αριθμητική. Στα 30 χρόνια που μεσολάβησαν μέχρι σήμερα, έχουν αναπτυχθεί εκατοντάδες εκδόσεις και παραλλαγές της γλώσσας Logo, με πολλές και ποικίλες δυνατότητες. Έτσι, πέρα από τη μετακίνηση χελωνών στην οθόνη ή την επίλυση μαθηματικών προβλημάτων, σήμερα διατίθεται ένα μεγάλο εύρος αντίστοιχων εργαλείων, που επιτρέπουν τη δημιουργία διαδραστικών ιστοριών και παιχνιδιών, ακόμα και τον έλεγχο ρομποτικών κατασκευών με μοτέρ και διάφορους αισθητήρες.

Την τελευταία δεκαετία, λοιπόν, αναπτύσσεται μεγάλο ερευνητικό και εκπαιδευτικό ενδιαφέρον για περιβάλλοντα προγραμματισμού με βάση τη Logo, περιβάλλοντα εκπαιδευτικής ρομποτικής, προγραμματιστικούς μικρόκοσμους, περιβάλλοντα προσομοίωσης και οπτικοποίησης αλγορίθμων, κλπ..

2.3.7 Εποικοδομιστικά περιβάλλοντα προγραμματισμού

Περιβάλλοντα, όπως τα ανωτέρω, παρέχουν νέες δυνατότητες για την οικοδόμηση γνώσεων και την ανάπτυξη δεξιοτήτων στον προγραμματισμό. Παράλληλα, διαπιστώνεται μια έντονη τάση για το σχεδιασμό μαθησιακών δραστηριοτήτων που βασίζονται στις αρχές του εποικοδομισμού [32][50].

Κύριοι άξονες των εποικοδομιστικών διδακτικών προσεγγίσεων είναι η εκτίμηση των προϋπαρχουσών γνώσεων και αντιλήψεων των μαθητών και η οργάνωση διδακτικών-μαθησιακών δραστηριοτήτων που να ευνοούν τη διερευνητική, ανακαλυπτική και συνεργατική μάθηση. Οι προσεγγίσεις αυτές δίνουν έμφαση στον παιδαγωγικό σχεδιασμό της διδασκαλίας του προγραμματισμού και στη μετατόπιση από το συντακτικό στην καλλιέργεια δεξιοτήτων επίλυσης προβλημάτων (αναλυτική-συνθετική σκέψη, αφαιρετική ικανότητα, μοντελοποίηση λύσεων).

Ο Resnick [54] υποστηρίζει ότι οι μαθητές μπορούν να αναπτύξουν ικανότητες υπολογιστικής σκέψης, συμμετέχοντας ενεργά σε αυθεντικές, διερευνητικές, ομαδοσυνεργατικές δραστηριότητες, που έχουν νόημα για τους ίδιους και αξιοποιούν το γνωστικό και πολιτιστικό τους υπόβαθρο. Η εποικοδομιστική αυτή προσέγγιση, δίνει μεγαλύτερη βαρύτητα στο πως μαθαίνουν οι μαθητές, παρά στο τι μαθαίνουν και ενισχύει το ρόλο των μαθητών ως δημιουργοί και αυτό-εκφραστές παρά χρήστες και καταναλωτές της τεχνολογίας.

Όπως αναφέρει και ο Lee [36], σε αρκετά εισαγωγικά μαθήματα προγραμματισμού με τη χρήση τέτοιων εργαλείων, η συνήθης πρακτική είναι η εξέλιξη των μαθητών σε τρία στάδια: «χρήση – τροποποίηση – δημιουργία».

2.3.8 Η περίπτωση του Scratch

Το 2006, η ομάδα Lifelong Kindergarten Group του ερευνητικού εργαστηρίου MIT Media Lab ανέπτυξε το Scratch ¹⁰, μια έκδοση της Logo, βασισμένη σε προγραμματισμό blocks, που ανέδειξε και επανέφερε στο προσκήνιο τον οπτικό προγραμματισμό στην εκπαίδευση, καθώς επέτρεψε τη σχεδίαση και ανάπτυξη προγραμμάτων με τη χρήση μεταφοράς και απόθεσης (drag and drop) αντικειμένων και αποσπώμενων κομματιών κώδικα που μοιάζουν με κομμάτια από παζλ.

Με το Scratch, οι μαθητές μπορούν να φτιάχνουν τις δικές τους διαδραστικές ιστορίες και παιχνίδια, μουσική κι άλλες μορφές τέχνης, εύκολα και γρήγορα, ενώ παράλληλα εξοικειώνονται με τις βασικές αρχές του προγραμματισμού. Το Scratch συγκεντρώνει ενδιαφέροντα στοιχεία για την εισαγωγή και εμβάθυνση στον προγραμματισμό συγκριτικά με άλλα περιβάλλοντα. Ειδικότερα, διαθέτει γραφική γλώσσα προγραμματισμού, αποτρέπει τα συντακτικά λάθη, επιτρέπει μερική και άμεση εκτέλεση, υιοθετεί το σκηνοθετικό υπόδειγμα για τη διαδικασία του προγραμματισμού, και υποστηρίζει ταυτόχρονο προγραμματισμό. Με αυτή τη πλατφόρμα προγραμματισμού και αυτόν τον τρόπο διδακτικής προσέγγισης, οι μαθητές έρχονται σε επαφή με σημαντικές μαθηματικές και υπολογιστικές ιδέες, ενώ παράλληλα κατανοούν καλύτερα τη γενική διαδικασία του σχεδιασμού αλγορίθμων. Παράλληλα αναπτύσσεται η αναλυτική σκέψη των μαθητών και καλλιεργείται η δημιουργικότητά τους.

Στα πλαίσια αυτά, ομάδα ερευνητών [54] έχουν σχεδιάσει και προτείνουν την αξιοποίηση μιας σειράς από προγραμματιστικές δραστηριότητες σε Scratch για το σχεδιασμό και την ανάπτυξη αλληλεπιδραστικών πολυμεσικών εφαρμογών. Οι δραστηριότητες αυτές εμπεριέχουν τη διάσταση της υπολογιστικής σκέψης, σε τρία επίπεδα: τις υπολογιστικές έννοιες (δομές ακολουθίας, επιλογής και επανάληψης, γεγονότα, ταυτόχρονη εκτέλεση, δεδομένα, τελεστές), τις υπολογιστικές πρακτικές (αξιολόγηση και ανατροφοδότηση, δοκιμή και εκσφαλμάτωση, επαναχρησιμοποίηση και συνδυασμός, αφαίρεση και μοντελοποίηση) και τις υπολογιστικές προοπτικές (δυνατότητες έκφρασης, συνεργασίας και προβληματισμών).

2.3.9 Η σημασία του «πλαisiού» και του μέσου

¹⁰ MIT Scratch: <http://scratch.mit.edu/about/>

Οι Cooper και Cunningham [07] εξαίρουν τη σημασία του «πλαίσιου» (βλ. contextualism¹¹), ορίζοντάς το ως τις συνθήκες υπό τις οποίες λαμβάνει χώρα κάτι. Έτσι, το «πλαίσιο» για ένα μάθημα πληροφορικής δεν περιλαμβάνει μόνο τις βασικές αρχές και διαδικασίες της επιστήμης υπολογιστών, αλλά και τις πηγές, καθώς και τις εφαρμογές αυτών των αρχών και διαδικασιών. Αυτό το «πλαίσιο» θα αποτελεί πηγή παραδειγμάτων, κινήτρων και ιδεών για εργασίες, στα πλαίσια του μαθήματος.

Παρατηρούν, ότι πολύ συχνά, ζητείται από τους μαθητές και σπουδαστές η μάθηση εννοιών πληροφορικής, υπό συνθήκες, οι οποίες δεν μοιάζουν να έχουν νόημα για τους ίδιους. Επηρεασμένη από το φορμαλισμό των μαθηματικών, η εκπαίδευση στην επιστήμη υπολογιστών εστιάζει υπερβολικά στη θεωρία, χωρίς να τη συνδέει με την αντίστοιχη εφαρμογή της.

Ακόμη και το Συμβούλιο Εθνικών Ερευνών NRC (National Resource Council) των Η.Π.Α. επισημαίνει ότι η παροχή ελκυστικών εργαστηριακών δραστηριοτήτων είναι μια κρίσιμη πρόκληση για την εκπαίδευση σε όλα τα επιστημονικά πεδία [15].

Εξάλλου, ερευνητές, όπως οι Cooper και Margulieux [07][38] επιχειρηματολογούν υπέρ της προσεκτικής επιλογής κατάλληλων δραστηριοτήτων και για τη γεφύρωση του χάσματος των δυο φύλων στο πεδίο της πληροφορικής.

Κάνουν μάλιστα ιδιαίτερη αναφορά στο προγραμματιστικό περιβάλλον για φορητές συσκευές Android, με την ονομασία “MIT App Inventor”¹², παρομοιάζοντάς το με το Scratch, αλλά το θεωρούν πιο ολοκληρωμένο και πιο ουδέτερο ως προς την καταλληλότητά του για χρήση και από τα δυο φύλα.

Ο Holz [24] έχει μελετήσει το θέμα της επιλογής του κατάλληλου μέσου, που να προσελκύει εξίσου και αρκετά, και τα δυο φύλα, για την αντιμετώπιση του φαινομένου του μειωμένου ενδιαφέροντος προς τα μαθήματα πληροφορικής. Μελέτη που διεξήχθη στη Γερμανία το 2008 (JIM Study¹³) και εξέτασε τη χρήση των διαφόρων μέσων από εφήβους (12 - 19 ετών), έδειξε ότι τα κινητά τηλέφωνα είναι οι μόνες συσκευές τις οποίες κατέχουν εξίσου αγόρια και κορίτσια, με ένα μικρό προβάδισμα, μάλιστα, για τα κορίτσια. Η αντίστοιχη μελέτη, δυο χρόνια μετά, το 2010 έδειξε την αντικατάσταση των κινητών τηλεφώνων από έξυπνα τηλέφωνα, τα οποία έχουν

¹¹ Contextualism: <http://en.wikipedia.org/wiki/Contextualism>

¹² MIT App Inventor: <http://appinventor.mit.edu/>

¹³ MPFS JIM Study 2010: <http://www.mpfs.de/index.php?id=181>

ακόμα περισσότερες δυνατότητες και επιτρέπουν πιο ευρεία χρήση τους, για την κάλυψη ακόμα περισσότερων ενδιαφερόντων των μαθητών.

Οι Repenning και Resnick [52][54] επισημαίνουν τα χαρακτηριστικά που οφείλει να έχει ένα προγραμματιστικό περιβάλλον για να προάγει την υπολογιστική σκέψη στα παιδιά. Η ιδέα αναφερόμενη ως “low floor - high ceiling” σημαίνει ότι το εργαλείο αυτό πρέπει να είναι εύκολο στη χρήση από έναν αρχάριο για να ξεκινήσει να δημιουργεί λειτουργικά προγράμματα, αλλά και αρκετά ισχυρό και με εκτενείς δυνατότητες για να ικανοποιήσει τις ανάγκες και πιο έμπειρων και ικανών προγραμματιστών. Παράλληλα, απαιτούνται “wide walls”, ώστε να υποστηρίζονται πολλών ειδών έργα, για να προσελκύει άτομα με διαφορετικά ενδιαφέροντα και μαθησιακά στυλ.

Θεωρείται επίσης ότι το App Inventor ευνοεί και πληροί την απαίτηση “low floor - high ceiling”, καθώς όλα τα παιδιά, ανεξαρτήτως φύλου, μπορούν να δημιουργήσουν (απλές ή πιο σύνθετες) εφαρμογές για Android, ενώ ταυτόχρονα υποστηρίζει σύνθετες υπολογιστικές λειτουργίες, όπως η αφαιρετικότητα δεδομένων και διαδικασιών, συνθήκες, επαναληπτικότητα και αναδρομή, δομημένη ανάλυση και εκσφαλμάτωση [07][38].

Ο Kurkovsky [34] μάλιστα τονίζει ότι η ανάπτυξη παιχνιδιών για κινητές συσκευές είναι και πιο απλή και πιο δελεαστική, σε σχέση με τους κλασσικούς υπολογιστές, επομένως και καταλληλότερη για την επαφή των μαθητών με τον προγραμματισμό και τις βασικές του έννοιες.

Στη συνέχεια, εξετάζουμε τη θέση του αντικειμένου του προγραμματισμού υπολογιστών με βάση τα Προγράμματα σπουδών για την Πληροφορική στην Π/θμια και Δ/θμια εκπαίδευση της Ελλάδας.

Κεφάλαιο 3

Διδασκαλία Προγραμματισμού στην ελληνική εκπαίδευση

Στο κεφάλαιο αυτό, γίνεται αναφορά της τρέχουσας θέσης του διδακτικού αντικειμένου του προγραμματισμού στην πρωτοβάθμια και δευτεροβάθμια εκπαίδευση της Ελλάδας, όπως αυτή διαμορφώνεται με τις αλλαγές που ξεκίνησε να προωθεί το Υπουργείο Παιδείας από το 2010 μέχρι και σήμερα. Οι σημαντικότερες αλλαγές αφορούν στο πρόγραμμα σπουδών των νέων διδακτικών αντικειμένων που έχουν εισαχθεί στα ολόημερα δημοτικά σχολεία που λειτουργούν με Ενιαίο Αναμορφωμένο Εκπαιδευτικό Πρόγραμμα (ΕΑΕΠ)¹⁴ το 2010, στην Πιλοτική Εφαρμογή του διδακτικού πεδίου «Τεχνολογίες Πληροφορίας και Επικοινωνιών» στα Δημοτικά και Γυμνάσια από το 2011¹⁵, καθώς και στο «Σχέδιο Νόμου για την Αναδιάρθρωση της Β' Βάθμιας Εκπαίδευσης»¹⁶ για το «Νέο Λύκειο» από το 2013.

3.1 ΤΠΕ και προγραμματισμός στο Δημοτικό σχολείο

¹⁴ Πρόγραμμα Σπουδών Δημοτικών Σχολείων με ΕΑΕΠ: ΦΕΚ 1139/ 28-8-2010.

¹⁵ Υπουργικές Αποφάσεις 104868/Γ2/13-09-2011 (Β' 2240) και 115475/Γ2/21-08-2013 (Β' 2121).

¹⁶ ΦΕΚ Νόμου 4186-2013 (Α 193) 17/09/2013 για την αναδιάρθρωση της Δευτεροβάθμιας Εκπαίδευσης και λοιπές διατάξεις.

Το ΑΠΣ Πληροφορικής για το Δημοτικό που εκδόθηκε το 2003 ¹⁷ προβλέπει τη διδασκαλία προγραμματισμού στον άξονα «Ελέγχω και Προγραμματίζω», στις Ε' και ΣΤ' τάξεις. Πιο συγκριμένα, επιδιώκεται οι μαθητές να χρησιμοποιήσουν μια απλή γλώσσα προγραμματισμού (Logo like) για τον έλεγχο και τον προγραμματισμό του υπολογιστή, κυρίως για τη δημιουργία απλών γεωμετρικών σχημάτων.

Το 2001, το Παιδαγωγικό Ινστιτούτο ανέπτυξε το νέο «Πρόγραμμα Σπουδών για τις ΤΠΕ στην Πρωτοβάθμια Εκπαίδευση» ¹⁸. Σύμφωνα με αυτό, το σημερινό σχολείο οφείλει να προετοιμάσει αποτελεσματικά τον αυριανό πολίτη της Κοινωνίας της Γνώσης, προκειμένου να είναι σε θέση να αντιμετωπίσει τις προκλήσεις αλλά και να αξιοποιήσει τις ευκαιρίες της νέας εποχής. Λαμβάνοντας υπόψη ότι οι ΤΠΕ θα συνεχίσουν να αναπτύσσονται και να διεισδύουν στο κοινωνικό πεδίο με ταχύτατους ρυθμούς, το Νέο Πρόγραμμα Σπουδών για τις ΤΠΕ και τον πληροφορικό γραμματισμό στο Δημοτικό Σχολείο προσδιορίζει και εξειδικεύει τις διαστάσεις του πληροφορικού γραμματισμού, δηλαδή τις ικανότητες (**γνώσεις, δεξιότητες, στάσεις και αξίες** για τις ΤΠΕ) που θα πρέπει να αναπτύξουν όλοι οι μαθητές και είναι απαραίτητες για τη συνέχιση των σπουδών τους στο Γυμνάσιο και την παραπέρα ζωή τους. Απώτερος στόχος είναι οι ΤΠΕ να συμβάλουν με **νέα μέσα και νέες πρακτικές** στη βελτίωση του εκπαιδευτικού αποτελέσματος και, τελικά, στη διαμόρφωση ενός νέου σχολείου.

Ο γενικός σκοπός του Π.Σ. του μαθήματος των ΤΠΕ στο Δημοτικό Σχολείο είναι όλοι οι μαθητές να έχουν τις ευκαιρίες να αναπτύξουν τουλάχιστον τις προτεινόμενες ικανότητες (γνώσεις, δεξιότητες και στάσεις) που σχετίζονται με τις ΤΠΕ. Στον ακόλουθο πίνακα παρουσιάζονται οι άξονες γύρω από τους οποίους διαρθρώνονται οι μαθησιακοί στόχοι του Προγράμματος Σπουδών για τις ΤΠΕ στο Δημοτικό Σχολείο.

Άξονες μαθησιακών στόχων
Γνωρίζω, δημιουργώ και εκφράζομαι με τις ΤΠΕ <ul style="list-style-type: none">• Γνωρίζω και χειρίζομαι τον υπολογιστή• Δημιουργώ και εκφράζομαι με πολυμέσα και παρουσιάσεις• Δημιουργώ με τον κειμενογράφο
Επικοινωνώ και συνεργάζομαι με ΤΠΕ <ul style="list-style-type: none">• Γνωρίζω το Διαδίκτυο

¹⁷ Διαθεματικό Ενιαίο Πλαίσιο Προγραμμάτων Σπουδών (ΔΕΠΠΣ) και Αναλυτικά Προγράμματα Σπουδών (ΑΠΣ) Υποχρεωτικής Εκπαίδευσης για την Πληροφορική στο Δημοτικό (ΦΕΚ 304Β/13-03-2003).

¹⁸ Πρόγραμμα Σπουδών για τις ΤΠΕ στην Πρωτοβάθμια Εκπαίδευση. ΕΣΠΑ 2007-2013/Ε.Π.Ε. & ΔΒΜ/Α.Π. 1-2-3 «ΝΕΟ ΣΧΟΛΕΙΟ (Σχολείο 21^{ου} αιώνα) – Οριζόντια Πράξη» MIS: 295450.

<ul style="list-style-type: none"> • Αναζητώ πληροφορίες • Επικοινωνώ και συνεργάζομαι
<p>Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ</p> <ul style="list-style-type: none"> • Μοντελοποιώ με εννοιολογικούς χάρτες • Λύνω προβλήματα με Υπολογιστικά Φύλλα • Προγραμματίζω τον υπολογιστή • Υλοποιώ σχέδια έρευνας
<p>Οι ΤΠΕ ως κοινωνικό φαινόμενο</p> <ul style="list-style-type: none"> • Ο ρόλος των ΤΠΕ στη σύγχρονη εποχή • Ψηφιακή κουλτούρα (στάσεις, συμπεριφορές, αξίες)

Πίνακας 3.1: Άξονες μαθησιακών στόχων του Π.Σ. για τις ΤΠΕ στο Δημοτικό

Το Νέο Πρόγραμμα Σπουδών δίνει ιδιαίτερη έμφαση στη διαμόρφωση μαθησιακών καταστάσεων που θα επιτρέπουν την ολοκλήρωση των γνώσεων και των τεχνικών δεξιοτήτων, την καλλιέργεια μαθησιακών δεξιοτήτων και ικανοτήτων μεθοδολογικού χαρακτήρα στις ΤΠΕ και, τελικά, την αυτόνομη ανάπτυξη όλων των μαθητών. Για το λόγο αυτό έχει ενταχθεί μια σημαντική ενότητα στο Π.Σ., κατά την οποία οι μαθητές υλοποιούν μαθησιακά σχέδια εργασίας (projects) χρησιμοποιώντας ποικίλα εργαλεία των ΤΠΕ. Ενδεικτικά αναφέρονται λογισμικά γενικής χρήσης, επεξεργασίας και ανάπτυξης πολυμέσων, εκπαιδευτικά περιβάλλοντα προγραμματισμού και ρομποτικής, εκπαιδευτικά λογισμικά (εννοιολογική χαρτογράφηση, προσομοιώσεις κλπ.), πηγές στον Παγκόσμιο Ιστό πληροφοριών (ιστοεξερευνήσεις), υπηρεσίες και εφαρμογές Web 2.0 (wikis, blogs, ηλεκτρονικές συζητήσεις, εργαλεία διαμοίρασης υλικού).

Οι θεματικές ενότητες του Π.Σ. έχουν σχεδιαστεί με τρόπο ώστε οι μαθησιακοί στόχοι να ανταποκρίνονται στο βαθμό ετοιμότητας και ανάπτυξης των μαθητών. Δεν είναι όμως απαραίτητο να διδαχθούν σειριακά. Η προτεινόμενη κατανομή του διδακτικού χρόνου είναι ενδεικτική. Ο εκπαιδευτικός θα πρέπει να κάνει τον χρονοπρογραμματισμό και το σχεδιασμό της διάρθρωσης της ύλης με βάση τα μαθησιακά χαρακτηριστικά, τα ενδιαφέροντα και τις προϋπάρχουσες γνώσεις και εμπειρίες των μαθητών της τάξης του. Επίσης, θα πρέπει να χρησιμοποιεί τη σπειροειδή προσέγγιση και να επανέρχεται, με κάθε ευκαιρία και ειδικά στα πλαίσια της υλοποίησης σχεδίων έρευνας, σε βασικές ενότητες που απαιτούν τη χρήση ποικίλων εργαλείων λογισμικού. Εν γένει, πρόκειται για ένα ανοικτό Αναλυτικό Πρόγραμμα Σπουδών, το οποίο ο εκπαιδευτικός αξιοποιεί σύμφωνα με τις εκπαιδευτικές ανάγκες και τα μέσα που διαθέτει.

Ειδικότερα, στο επιστημονικό πεδίο «Τεχνολογίες Πληροφορίας & Επικοινωνιών στην Πρωτοβάθμια Εκπαίδευση» περιλαμβάνεται στους βασικούς άξονες μαθησιακών στόχων του Π.Σ., ο προγραμματισμός του υπολογιστή για τις Ε' και ΣΤ' τάξεις, με βασικό στόχο την σταδιακή εξοικείωση των μαθητών με τον προγραμματισμό μέσα από την αξιοποίηση διαθέσιμων εκπαιδευτικών περιβαλλόντων οπτικού προγραμματισμού.

3.1.1 Ε' Δημοτικού

Σε σύνολο 60 διδακτικών ωρών, στην ενδεικτική κατανομή ανά ενότητα και στον άξονα «Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ» προτεινόμενης διάρκειας 36 ωρών, η υποενότητα «Προγραμματίζω τον υπολογιστή» κατέχει 10 ώρες.

Άξονες μαθησιακών στόχων	Προτεινόμενες ώρες διδασκαλίας
Δημιουργώ και εκφράζομαι με τις ΤΠΕ <ul style="list-style-type: none"> • Δημιουργώ με τον κειμενογράφο • Δημιουργώ και εκφράζομαι με πολυμέσα και παρουσιάσεις 	16
Επικοινωνώ και συνεργάζομαι με ΤΠΕ <ul style="list-style-type: none"> • Γνωρίζω το Διαδίκτυο • Επικοινωνώ και συνεργάζομαι 	8
Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ <ul style="list-style-type: none"> • Λύνω προβλήματα με Υπολογιστικά Φύλλα • Προγραμματίζω τον υπολογιστή • Υλοποιώ σχέδια έρευνας 	36

Πίνακας 3.2: Ενδεικτική κατανομή διδακτικών ωρών ΤΠΕ Ε' Δημοτικού

Τα προσδοκώμενα μαθησιακά αποτελέσματα, τα βασικά θέματα, καθώς και το προτεινόμενο εκπαιδευτικό υλικό και μαθησιακές δραστηριότητες αναφέρονται αναλυτικά παρακάτω:

Προγραμματίζω τον υπολογιστή

Προσδοκώμενα Μαθησιακά Αποτελέσματα	Βασικά θέματα	Εκπαιδευτικό υλικό
Ο μαθητής/τρια πρέπει να είναι ικανός/ή <ul style="list-style-type: none"> • να αναγνωρίζει τις βασικές συνιστώσες ενός εκπαιδευτικού περιβάλλοντος οπτικού προγραμματισμού • να εκτελεί έτοιμα προγράμματα που θα του δοθούν • να περιγράφει με λεκτικό τρόπο τα βήματα 	Εκπαιδευτικά περιβάλλοντα οπτικού προγραμματισμού Αλγόριθμος Πρόγραμμα Γεγονότα (events) Ανάπτυξη και εκτέλεση απλών εφαρμογών	Εκπαιδευτικά περιβάλλοντα οπτικού προγραμματισμού Προσομοιώσεις αλγορίθμων EasyLogo Scratch BYOB

<p>απλών αλγορίθμων που καλείται να υλοποιήσει στο εκπαιδευτικό περιβάλλον οπτικού προγραμματισμού</p> <ul style="list-style-type: none"> • να διατυπώνει απλές εντολές στο περιβάλλον οπτικού προγραμματισμού • να ορίζει ενέργειες και σενάρια που πρέπει να εκτελεστούν για να επιτευχθούν επιθυμητά γεγονότα • να εξηγεί γιατί ένα αντικείμενο του προγραμματιστικού περιβάλλοντος συμπεριφέρεται με συγκεκριμένο τρόπο • να κωδικοποιεί έναν αλγόριθμο σε προγραμματιστικό περιβάλλον και να αναπτύσσει μικρές εφαρμογές χρησιμοποιώντας ένα εκπαιδευτικό περιβάλλον οπτικού προγραμματισμού • να αναλύει ένα πρόβλημα σε επιμέρους απλούστερα • να συνθέτει ένα έργο από τα επιμέρους στοιχεία του (που έχουν προκύψει από την ανάλυση) • να διακρίνει διάφορα γεγονότα (events) στο περιβάλλον οπτικού προγραμματισμού • να εξοικειωθεί με τεχνικές διόρθωσης σφαλμάτων και βελτιστοποίησης των προγραμμάτων που αναπτύσσουν εκπαιδευτικό περιβάλλον προγραμματισμού • να δημιουργεί σύνθετα έργα που βασίζονται στη σύνθεση επιμέρους απλούστερων έργων 	<p>Διόρθωση σφαλμάτων Βελτιστοποίηση προγραμμάτων</p>	<p>Kodu Microworlds Pro gameMaker K-turtle Turtle Art openStarlogo Εκπαιδευτική ρομποτική</p>
--	---	---

Δραστηριότητες

Βασικός στόχος της ενότητας αυτής είναι η σταδιακή εξοικείωση των μαθητών με τον προγραμματισμό μέσα από την αξιοποίηση διαθέσιμων εκπαιδευτικών περιβαλλόντων οπτικού προγραμματισμού. Οι μαθητές σε κατάλληλα εκπαιδευτικό περιβάλλοντα προγραμματισμού, χειρίζονται και διερευνούν έτοιμα προγράμματα και εισάγονται στην έννοια του αλγορίθμου, έχοντας ως γενικό προσανατολισμό τη μετάβαση από την ψηφιακή ζωγραφική στα προγραμματιζόμενα πολυμέσα. Ο εκπαιδευτικός, προκαλεί το ενδιαφέρον των μαθητών, ώστε να δημιουργήσουν μικρές εφαρμογές, στις οποίες θα προκαλείται η δράση αντικειμένων στη σκηνή ή στο χώρο επισκόπησης με χρήση χειριστηρίων (πληκτρολόγιο, ποντίκι).

Ενδεικτικά προτείνεται να χρησιμοποιηθεί παιχνίδι ρόλων με στόχο οι μαθητές να μνηθούν στον προγραμματισμό (π.χ. λογική της γεωμετρίας της χελώνας, αυστηρότητα διατύπωσης των εντολών κλπ.). Ένας μαθητής παίζει το ρόλο της χελώνας (του ρομπότ) και ένας άλλος είναι ο προγραμματιστής που τον κατευθύνει στο χώρο της τάξης (με τις εντολές Μπροστά, Αριστερά, Δεξιά).

Ο εκπαιδευτικός παρουσιάζει το προγραμματιστικό περιβάλλον στους μαθητές αρχικά ως μια επέκταση του προγράμματος ζωγραφικής. Σχεδιάζουν εκτελώντας κατάλληλες εντολές απλά σχήματα.

Στη συνέχεια, καλούνται να σχεδιάσουν γράμματα της αλφαβήτου όπως τα Ι, Γ, Π, Ξ, Τ, Ε, Η. Αναλύουν κάθε φορά και σχεδιάζουν στο χαρτί τα βήματα που χρειάζεται να πραγματοποιηθούν, μελετούν τα λάθη στις κωδικοποιήσεις τους και τα διορθώνουν. Στη συνέχεια, οι μαθητές τροποποιούν τα προγράμματά τους σχεδιάζοντας τα παραπάνω σχήματα με διαφορετικά χαρακτηριστικά (χρώμα, πάχος γραμμής κλπ.).

Ακολουθεί συζήτηση σχετικά με κοινά σημεία που υπάρχουν στους διαφορετικούς αλγορίθμους που σχεδίασαν και στο πως ότι κάποια από αυτά θα μπορούσαν να επαναχρησιμοποιούνται.

Στη συνέχεια, καλούνται να σχεδιάσουν σύνθετα σχήματα, αφού προηγηθεί η ανάλυσή τους σε απλούστερα γεωμετρικά σχήματα, όπως

- ένα τετράγωνο ως η σύνθεση τεσσάρων ευθύγραμμων τμημάτων (ευθύγραμμης κίνησης και στροφής 90°)
- ένα σπίτι ως η σύνθεση ενός ορθογωνίου και ενός τριγώνου
- ένα δένδρο ως σύνθεση ενός ορθογωνίου και ενός κύκλου (έλλειψης)
- μια σκάλα, ως η σύνθεση διαδοχικών Γ.

Ενδεικτικός διδακτικός χρόνος: 10 ώρες

Πίνακας 3.3: Ενότητα "Προγραμματίζω τον υπολογιστή" ΤΠΕ Ε' Δημοτικού

3.1.2 ΣΤ' Δημοτικού

Σε σύνολο 60 διδακτικών ωρών, στην ενδεικτική κατανομή ανά ενότητα και στον άξονα «Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ» προτεινόμενης διάρκειας 38 ωρών, η υποενότητα «Προγραμματίζω τον υπολογιστή» κατέχει 12 ώρες.

Άξονες μαθησιακών στόχων	Προτεινόμενες ώρες διδασκαλίας
Δημιουργώ και εκφράζομαι με τις ΤΠΕ <ul style="list-style-type: none"> • Δημιουργώ με τον κειμενογράφο • Δημιουργώ και εκφράζομαι με πολυμέσα και παρουσιάσεις 	14
Επικοινωνώ και συνεργάζομαι με ΤΠΕ <ul style="list-style-type: none"> • Γνωρίζω το Διαδίκτυο • Επικοινωνώ και συνεργάζομαι 	8
Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ <ul style="list-style-type: none"> • Λύνω προβλήματα με Υπολογιστικά Φύλλα • Προγραμματίζω τον υπολογιστή • Υλοποιώ σχέδια έρευνας 	38

Πίνακας 3.4: Ενδεικτική κατανομή διδακτικών ωρών ΤΠΕ ΣΤ' Δημοτικού

Τα προσδοκώμενα μαθησιακά αποτελέσματα, τα βασικά θέματα, καθώς και το προτεινόμενο εκπαιδευτικό υλικό και μαθησιακές δραστηριότητες αναφέρονται αναλυτικά παρακάτω:

Προγραμματίζω τον υπολογιστή

Προσδοκώμενα Μαθησιακά Αποτελέσματα	Βασικά θέματα	Εκπαιδευτικό υλικό
Ο μαθητής/τρια πρέπει να είναι ικανός/ή <ul style="list-style-type: none"> • να εξηγεί γιατί ένα αντικείμενο του προγραμματιστικού περιβάλλοντος συμπεριφέρεται με συγκεκριμένο τρόπο • να ορίζει ενέργειες και σενάρια που πρέπει να εκτελεστούν για να επιτευχθούν επιθυμητά γεγονότα • να αντιλαμβάνεται την αναγκαιότητα και τη χρησιμότητα της δομής επανάληψης • να χρησιμοποιεί εντολές επανάληψης στα προγράμματα που αναπτύσσει • να αναλύει ένα πρόβλημα σε επιμέρους απλούστερα • να συνθέτει ένα έργο προγραμματισμού από τα επιμέρους στοιχεία της ανάλυσης • να αντιλαμβάνεται τη χρησιμότητα το ρόλο της διαδικασίας σε ένα πρόγραμμα • να χρησιμοποιεί διαδικασίες στα έργα του 	Εκπαιδευτικά περιβάλλοντα οπτικού προγραμματισμού Αλγόριθμος Πρόγραμμα Γεγονότα (events) Επαναληπτική δομή Η δομή επανάληψης Διαδικασία (υποπρόγραμμα) Κλήση διαδικασιών Διόρθωση σφαλμάτων	EasyLogo Scratch BYOB Microworlds Pro Kodu gameMaker K-turtle Turtle Art openStarlogo Εκπαιδευτική ρομποτική Προσομοιώσεις αλγορίθμων java applets flash animations

<ul style="list-style-type: none"> • να εφαρμόζει τεχνικές ελέγχου και διόρθωσης σφαλμάτων στα προγράμματα που δημιουργεί • να κατανοεί τη λειτουργία έτοιμων προγραμμάτων που του δίνονται 		
Δραστηριότητες		
<p>Οι μαθητές υλοποιούν κατάλληλες δραστηριότητες ώστε να επεκτείνουν και να ενισχύσουν τις προγραμματιστικές τους δεξιότητες. Στόχος είναι, μέσα από την ενεργητική συμμετοχή τους, η διερευνητική προσέγγιση της γνώσης, η συνεργασία, η αυτενέργεια, η ανάπτυξη της δημιουργικότητας και της φαντασίας των μαθητών.</p> <p>Με τη χρήση κατάλληλων παραδειγμάτων αναδεικνύεται η ανάγκη της επαναχρησιμοποίησης τμήματος εντολών αλλά και της επαναληπτικής εκτέλεσης εντολών που είχαν τεθεί στην προηγούμενη τάξη. Παρουσιάζονται στους μαθητές οι έννοιες της διαδικασίας και της επανάληψης και καλούνται να επανασχεδιάσουν προγράμματα που έχουν ήδη υλοποιήσει. Οι μαθητές καθοδηγούνται να αναλύσουν τα σχήματα σε απλούστερα, να εντοπίσουν και να διορθώσουν λάθη στα προγράμματά τους.</p> <p>Ενδεικτικά παραδείγματα είναι:</p> <ul style="list-style-type: none"> • δημιουργία τρένου (σύνθεση βαγονιών, όπου κάθε βαγόνι είναι ένα ορθογώνιο με δύο κυκλικές ρόδες) • δημιουργία πίνακα ζωγραφικής (με απλά γεωμετρικά σχήματα σε διαφορετικά μεγέθη και χρώματα) • δημιουργία και κίνηση ανεμόμυλου (ως σύνθεση ορθογωνίου, τριγώνων και γραμμών). • ο προγραμματισμός της κίνησης ενός αντικειμένου (π.χ. μπάλα) στο χώρο ή σε λαβύρινθο. • προγραμματισμός διαδραστικών παιχνιδιών και ιστοριών. <p>Είναι δυνατή η χρήση προσομοιώσεων (java applets, flash animations), με στόχο οι μαθητές να προβληματιστούν για το πώς μπορούν να καταγράψουν τα βήματα επίλυσης ενός προβλήματος και πώς μπορούν να τα περιγράψουν στο προγραμματιστικό περιβάλλον.</p> <p>Ενδεικτικός διδακτικός χρόνος: 12 ώρες</p>		

Πίνακας 3.5: Ενότητα "Προγραμματίζω τον υπολογιστή" ΤΠΕ ΣΤ' Δημοτικού

3.1.3 Ολοήμερα Δημοτικά Σχολεία με ΕΑΕΠ

Το 2010 επανεξετάστηκαν και επικαιροποιήθηκαν τα Αναλυτικά Προγράμματα και οι οδηγίες για τα διδακτικά αντικείμενα του προγράμματος των ολοήμερων Δημοτικών Σχολείων με ΕΑΕΠ (Ενιαίο Αναμορφωμένο Πρόγραμμα Σπουδών)¹⁹. Με βάση το ΠΣ για το διδακτικό αντικείμενο των ΤΠΕ, οι μαθητές των Ε' και ΣΤ' τάξεων στη Θεματική Ενότητα «Προγραμματίζω και Ελέγχω» έρχονται σε επαφή με τον προγραμματισμό, όπως φαίνεται στον πίνακα που ακολουθεί.

Ε' Τάξη	
Στόχοι	Ενδεικτικές Δραστηριότητες
<p>Οι μαθητές επιδιώκεται να:</p> <ul style="list-style-type: none"> • Να σχεδιάζουν και να επεξεργάζονται δικές τους ζωγραφιές σε ένα Logo like προγραμματιστικό περιβάλλον. <p>Να εισάγουν και να επεξεργάζονται έτοιμα σχήματα.</p>	<p>Οι μαθητές ανακαλύπτουν τις δυνατότητες σχεδίασης ενός Logo like προγραμματιστικού περιβάλλοντος δημιουργώντας τις δικές τους ζωγραφιές.</p> <p>Εμπλουτίζουν τις εργασίες τους εισάγοντας χελώνες που τις κινούν με απλές εντολές. Αλλάζουν το σχήμα της χελώνας χρησιμοποιώντας έτοιμα γραφικά της εφαρμογής.</p>

¹⁹ Διδασκαλία – Πρόγραμμα Σπουδών των νέων διδακτικών αντικειμένων στα ολοήμερα δημοτικά σχολεία με Ενιαίο Αναμορφωμένο Εκπαιδευτικό Πρόγραμμα (ΕΑΕΠ). Απόφαση Φ.12/879/88413/Γ1.

<p>Να εισάγουν έτοιμες εικόνες και ζωγραφιές.</p> <p>Να εισάγουν χελώνα και να αλλάζουν τις διαστάσεις της.</p> <p>Να αλλάζουν τη μορφή της χελώνας χρησιμοποιώντας τα έτοιμα σχήματα.</p> <p>Να γνωρίζουν τις βασικές εντολές κίνησης της χελώνας (μπροστά, πίσω δεξιά, αριστερά, περίμενε, πάνω –pen up–, κάτω – pen down– κ.λπ.).</p> <p>Να σχεδιάζουν απλά γεωμετρικά σχήματα (τετράγωνο, παραλληλόγραμμο, κ.ά.) με τη γραφίδα της χελώνας.</p>	<p>Προγραμματίζουν τη χελώνα ώστε αυτή να κινείται όταν επιλέγεται με το ποντίκι. Αναλαμβάνουν τη δημιουργία ολοκληρωμένων σελίδων με κίνηση γραφικών όπως: η κίνηση στο δρόμο, τα ζώα στην εξοχή κ.ά.</p> <p>Μαθαίνουν για το στυλό της χελώνας.</p> <p>Πειραματίζονται με την κίνηση της χελώνας και την περιστροφή της. Θέτουν ερωτήματα, σχεδιάζουν, εφαρμόζουν και ανακαλύπτουν τους κανόνες σχεδίασης βασικών γεωμετρικών σχημάτων. Αναγνωρίζουν την αναγκαιότητα της εντολής επανάληψης.</p> <p>Μαθαίνουν την εντολή και τη χρησιμοποιούν στη σχεδίαση των σχημάτων τους.</p> <p>Πειραματίζονται με τη γωνία περιστροφής της χελώνας και ανακαλύπτουν πόσες μοίρες περιστρέφεται μια χελώνα για να ολοκληρώσει τη διαδρομή της γύρω από ένα κλειστό σχήμα.</p> <p>Ετοιμάζουν ερωτήσεις αντιστοιχίας, όπου οι συμμαθητές τους καλούνται να αντιστοιχίσουν ένα σύνολο εντολών με το αντίστοιχο παραγόμενο σχήμα και αντίστροφα. Συζητούν.</p> <p>Παρατήρηση: Οι δραστηριότητες προσαρμόζονται στις δυνατότητες που προσφέρει το προγραμματιστικό περιβάλλον που χρησιμοποιεί ο εκπαιδευτικός.</p>
ΣΤ' Τάξη	
Στόχοι	Ενδεικτικές Δραστηριότητες
<p>Οι μαθητές επιδιώκεται να:</p> <ul style="list-style-type: none"> • Να συντάξουν απλές διαδικασίες σε ένα Logo like προγραμματιστικό περιβάλλον. • Να κατανοήσουν την έννοια της μεταβλητής. • Να χρησιμοποιούν διαδικασίες με συνθήκη, παραμετρικές και αναδρομικές διαδικασίες. 	<p>Οι μαθητές συντάξουν διαδικασίες για τη σχεδίαση γεωμετρικών σχημάτων στο προγραμματιστικό περιβάλλον της Logo.</p> <p>Συνδυάζουν τις διαδικασίες για το σχεδιασμό σύνθετων σχεδίων π.χ. κάστρο, χωριό, βάρκα, πλοίο κ.λπ.</p> <p>Εμπλουτίζουν τα σχέδια τους με χρώματα και με άλλα σχέδια από τα έτοιμα γραφικά.</p> <p>Μέσα από τα παραδείγματα, τα παιδιά κατανοούν ότι ο υπολογιστής εκτελεί τις οδηγίες που του δίνει ο άνθρωπος.</p> <p>Παρατήρηση: Οι δραστηριότητες προσαρμόζονται στις δυνατότητες που προσφέρει το προγραμματιστικό περιβάλλον που χρησιμοποιεί ο εκπαιδευτικός.</p>

Πίνακας 3.6: Ενότητα "Προγραμματίζω και ελέγχω" ΤΠΕ Ε' και ΣΤ' Ολοήμερου Δημοτικού με ΕΑΕΠ.

3.2 Πληροφορική & προγραμματισμός στο Γυμνάσιο

Το ΑΠΣ Πληροφορικής για το Γυμνάσιο που εκδόθηκε το 2003 ²⁰ και εξακολουθεί να ισχύει μέχρι σήμερα, προβλέπει τη διδασκαλία προγραμματισμού στον άξονα «Γνωρίζω τον υπολογιστή ως ενιαίο σύστημα» για 14 ώρες, μόνο για τη Γ' τάξη. Πιο συγκριμένα, επιδιώκεται οι μαθητές να αναγνωρίζουν την έννοια της γλώσσας προγραμματισμού και την αναγκαιότητα της χρήσης της, να σχεδιάζουν τη λύση ενός απλού προβλήματος και να την υλοποιούν σε ένα προγραμματιστικό περιβάλλον. Παρόλο που στο ΑΠΣ δεν γίνεται αναφορά σε κάποιο προγραμματιστικό περιβάλλον, το σχολικό βιβλίο εστιάζει στη χρήση της γλώσσας Logo και συγκεκριμένα στο λογισμικό Microworlds Pro ²¹.

Πλέον, σε αρκετά Γυμνάσια της χώρας εφαρμόζεται πιλοτικά το νέο «Πρόγραμμα Σπουδών για τον Πληροφορικό Γραμματισμό στο Γυμνάσιο» (4^η έκδοση, Μάιος 2011) ²². Ο γενικός σκοπός του Π.Σ. του μαθήματος της Πληροφορικής είναι όλοι οι μαθητές να έχουν τις ευκαιρίες να αναπτύξουν τουλάχιστον τις προτεινόμενες ικανότητες (γνώσεις, δεξιότητες, και στάσεις) που σχετίζονται με τις ΤΠΕ. Στον ακόλουθο πίνακα παρουσιάζονται οι άξονες γύρω από τους οποίους διαρθρώνονται οι μαθησιακοί στόχοι του πιλοτικού (νέου) Π.Σ. Πληροφορικής.

Άξονες μαθησιακών στόχων
Χειρίζομαι και δημιουργώ με τα εργαλεία των ΤΠΕ <ul style="list-style-type: none">• Θεμελιώδεις γνώσεις και δεξιότητες ΤΠΕ• Δημιουργικότητα με χρήση εργαλείων ΤΠΕ
Αναζητώ πληροφορίες, επικοινωνώ και συνεργάζομαι με ΤΠΕ <ul style="list-style-type: none">• Αναζήτηση, αξιολόγηση και διαχείριση πληροφοριών• Επικοινωνία-συνεργασία
Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ <ul style="list-style-type: none">• Κριτική ικανότητα, μοντελοποίηση, καινοτομία
Οι ΤΠΕ ως κοινωνικό φαινόμενο <ul style="list-style-type: none">• Ο ρόλος των ΤΠΕ στη σύγχρονη εποχή• Ψηφιακή κουλτούρα (στάσεις, συμπεριφορές, αξίες)

Πίνακας 3.7: Άξονες μαθησιακών στόχων του Π.Σ. Πληροφορικής για το Γυμνάσιο

²⁰ Διαθεματικό Ενιαίο Πλαίσιο Προγραμμάτων Σπουδών (ΔΕΠΠΣ) και Αναλυτικά Προγράμματα Σπουδών (ΑΠΣ) Υποχρεωτικής Εκπαίδευσης για την Πληροφορική στο Γυμνάσιο (ΦΕΚ 304Β/13-03-2003).

²¹ MicroWorld Pro: <http://odysseia.cti.gr/kirki/1stProductGroup/Microworlds%20Pro.htm>

²² Ψηφιακό Σχολείο / Διαδραστικά Σχολικά Βιβλία και Προγράμματα Σπουδών: <http://ebooks.edu.gr/>

Στο ΑΠΣ αναφέρεται ρητά ότι με τη διδασκαλία της Πληροφορικής στο Γυμνάσιο, επιδιώκονται οι παρακάτω επιμέρους ειδικοί σκοποί για τους μαθητές:

- Να αποκτήσουν γνώσεις σχετικά με τη διαδικασία επίλυσης απλών προβλημάτων με τη χρήση του υπολογιστή. Να διαπιστώσουν και να αντιληφθούν ότι μια απλή μηχανή ελέγχεται και προγραμματίζεται από τον άνθρωπο.
- Να αναπτύξουν κριτικές δεξιότητες για την αντιμετώπιση προβλημάτων με τη χρήση του υπολογιστή και να επιλύσουν απλά προβλήματα σε προγραμματιστικό περιβάλλον.
- Να συνεργασθούν για την εκτέλεση συγκεκριμένης εργασίας, να αναπτύξουν πρωτοβουλίες, να σχεδιάσουν, να θέσουν στόχους, να διαλεχτούν, να υπερβούν τις αντιθέσεις τους, να διατηρήσουν την ανεξαρτησία τους με σεβασμό στην άποψη των άλλων, να αναγνωρίσουν τη συμβολή της ομαδικής εργασίας στην παραγωγή έργου, να συζητήσουν και να κρίνουν την εργασία τους και την εργασία των άλλων.

Ακόμα μια βασική συνιστώσα διάρθρωσης του προτεινόμενου πλαισίου ένταξης του πληροφορικού γραμματισμού και των ΤΠΕ στο Γυμνάσιο είναι και η αξιοποίησή τους ως μεθοδολογία επίλυσης προβλημάτων. Αναλυτικότερα, οι μαθητές εμπλέκονται σε δραστηριότητες επίλυσης προβλημάτων που έχουν ως σκοπό την καλλιέργεια δεξιοτήτων μεθοδολογικού χαρακτήρα (επεξεργασία δεδομένων, σχεδιασμός και υλοποίηση αλγορίθμων, μοντελοποίηση λύσεων, προγραμματισμός υπολογιστών, δημιουργικότητα και καινοτομία) και δεξιοτήτων υψηλού επιπέδου (διερεύνηση, κριτική και αναλυτική σκέψη, συνθετική ικανότητα, ικανότητες επικοινωνίας και συνεργασίας).

Κατά παρόμοιο τρόπο με το αντίστοιχο ΠΣ στο Δημοτικό, και στο Γυμνάσιο δίνεται ιδιαίτερη έμφαση στη διαμόρφωση μαθησιακών καταστάσεων που θα επιτρέπουν την ολοκλήρωση των γνώσεων και την εμβάθυνση σε έννοιες των ΤΠΕ, την καλλιέργεια μαθησιακών δεξιοτήτων και ικανοτήτων μεθοδολογικού χαρακτήρα στις ΤΠΕ και, τελικά, την αυτόνομη ανάπτυξη όλων των μαθητών. Για το λόγο αυτό έχει ενταχθεί μια σημαντική ενότητα στο Π.Σ., συνολικής διάρκειας 14-16 διδακτικών ωρών, κατά την οποία οι μαθητές υλοποιούν μαθησιακά σχέδια εργασίας (projects) χρησιμοποιώντας ποικίλα εργαλεία των ΤΠΕ. Ενδεικτικά αναφέρονται λογισμικά γενικής χρήσης, επεξεργασίας και ανάπτυξης πολυμέσων, εκπαιδευτικά περιβάλλοντα προγραμματισμού και ρομποτικής, εκπαιδευτικά λογισμικά (εννοιολογική χαρτογράφηση, προσομοιώσεις κλπ.), πηγές στον Παγκόσμιο Ιστό πληροφοριών (ιστοεξερευνήσεις), υπηρεσίες

και εφαρμογές Web 2.0 (wikis, blogs, ηλεκτρονικές συζητήσεις, εργαλεία διαμοίρασης, επικοινωνίας, κλπ.).

Οι θεματικές ενότητες του Π.Σ. έχουν σχεδιαστεί με τρόπο ώστε οι μαθησιακοί στόχοι να ανταποκρίνονται στο βαθμό ετοιμότητας και ανάπτυξης των μαθητών. Δεν είναι όμως απαραίτητο να διδαχθούν σειριακά. Η προτεινόμενη κατανομή του διδακτικού χρόνου είναι ενδεικτική. Ο εκπαιδευτικός θα πρέπει να κάνει τον χρονοπρογραμματισμό και το σχεδιασμό της διάρθρωσης της ύλης με βάση τα μαθησιακά χαρακτηριστικά, τα ενδιαφέροντα και τις προϋπάρχουσες γνώσεις και εμπειρίες των μαθητών της τάξης του. Επίσης, θα πρέπει να χρησιμοποιεί τη **σπειροειδή προσέγγιση** και να επανέρχεται, με κάθε ευκαιρία και ειδικά στα πλαίσια της υλοποίησης σχεδίων έρευνας, σε βασικές ενότητες που απαιτούν τη χρήση ποικίλων εργαλείων λογισμικού.

Στο παραπάνω πλαίσιο, προβλέπεται και προτείνεται η ενασχόληση των μαθητών με προγραμματισμό υπολογιστών και στις τρεις τάξεις του Γυμνασίου, με μεγαλύτερη εμβάθυνση στο αντικείμενο, στη Γ' τάξη.

Βεβαίως, το ΑΠΣ Πληροφορικής για το Γυμνάσιο που εκδόθηκε το 2003 ²³ και εξακολουθεί να ισχύει μέχρι σήμερα για όσα σχολεία δεν εφαρμόζουν το προαναφερόμενο νέο Πιλοτικό ΠΣ, προβλέπει τη διδασκαλία προγραμματισμού στον άξονα «Γνωρίζω τον υπολογιστή ως ενιαίο σύστημα» για 14 ώρες, μόνο για τη Γ' τάξη. Πιο συγκεκριμένα, επιδιώκεται οι μαθητές να αναγνωρίζουν την έννοια της γλώσσας προγραμματισμού και την αναγκαιότητα της χρήσης της, να σχεδιάζουν τη λύση ενός απλού προβλήματος και να την υλοποιούν σε ένα προγραμματιστικό περιβάλλον. Παρόλο που στο ΑΠΣ δεν γίνεται αναφορά σε κάποιο προγραμματιστικό περιβάλλον, το σχολικό βιβλίο εστιάζει στη χρήση της γλώσσας Logo και συγκεκριμένα στο λογισμικό Microworlds Pro ²⁴.

3.2.1 Α' Γυμνασίου

²³ Διαθεματικό Ενιαίο Πλαίσιο Προγραμμάτων Σπουδών (ΔΕΠΠΣ) και Αναλυτικά Προγράμματα Σπουδών (ΑΠΣ) Υποχρεωτικής Εκπαίδευσης για την Πληροφορική στο Γυμνάσιο (ΦΕΚ 304Β/13-03-2003).

²⁴ ΚΙΡΚΗ: MicroWorld Pro της εταιρίας Logo Computer Systems Inc.
<http://odysseia.cti.gr/kirki/1stProductGroup/Microworlds%20Pro.htm>

Σε σύνολο 54 διδακτικών ωρών, στην ενδεικτική κατανομή ανά ενότητα και στον άξονα «Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ» προτεινόμενης διάρκειας 32 ωρών, η υποενότητα «Προγραμματίζω τον υπολογιστή» κατέχει 12 ώρες.

Άξονες μαθησιακών στόχων	Προτεινόμενες ώρες διδασκαλίας
Χειρίζομαι και δημιουργώ με τα εργαλεία των ΤΠΕ <ul style="list-style-type: none"> • Βασικές έννοιες ΤΠΕ • Λειτουργικά περιβάλλοντα • Δημιουργώ με τον κειμενογράφο 	20
Αναζητώ πληροφορίες, επικοινωνώ και συνεργάζομαι με ΤΠΕ <ul style="list-style-type: none"> • Δημιουργώ και εκφράζομαι με πολυμέσα και παρουσιάσεις • Γνωρίζω το Διαδίκτυο, επικοινωνώ και συνεργάζομαι 	12
Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ <ul style="list-style-type: none"> • Προγραμματίζω τον υπολογιστή • Υλοποιώ σχέδια έρευνας 	32

Πίνακας 3.8: Ενδεικτική κατανομή διδακτικών ωρών ΤΠΕ Α' Γυμνασίου

Τα προσδοκώμενα μαθησιακά αποτελέσματα, τα βασικά θέματα, καθώς και το προτεινόμενο εκπαιδευτικό υλικό και μαθησιακές δραστηριότητες αναφέρονται αναλυτικά παρακάτω:

Προσδοκώμενα Μαθησιακά Αποτελέσματα	Βασικά θέματα	Εκπαιδευτικό υλικό
Ο μαθητής/τρια πρέπει να είναι ικανός/ή <ul style="list-style-type: none"> • να αναγνωρίζει τις βασικές συνιστώσες ενός εκπαιδευτικού περιβάλλοντος οπτικού προγραμματισμού • να εκτελεί έτοιμα προγράμματα που θα του δοθούν • να περιγράφει με λεκτικό τρόπο απλούς αλγορίθμους που καλείται να υλοποιήσει στο περιβάλλον οπτικού προγραμματισμού • να διακρίνει διάφορα γεγονότα (events) στο περιβάλλον οπτικού προγραμματισμού • να καθορίζει/συντάσσει απλές εντολές στο περιβάλλον οπτικού προγραμματισμού • να ορίζει ενέργειες και σενάρια που πρέπει να εκτελεστούν για επιθυμητά γεγονότα • να εξηγεί γιατί ένα αντικείμενο του προγραμματιστικού περιβάλλοντος συμπεριφέρεται με συγκεκριμένο τρόπο • να αναπτύσσει μικρές εφαρμογές χρησιμοποιώντας ένα εκπαιδευτικό περιβάλλον οπτικού προγραμματισμού 	Εισαγωγή στον Προγραμματισμό Υπολογιστών Εκπαιδευτικά περιβάλλοντα οπτικού προγραμματισμού Οδηγούμενος από γεγονότα προγραμματισμός Ανάπτυξη και εκτέλεση απλών εφαρμογών	Σχολικό εγχειρίδιο Scratch BYOB Microworld pro Starlogo TNG Turtle Art Kodu Storytelling Alice, Mama gameMaker GreenFoot e-toys
Δραστηριότητες		

<p>Βασικός στόχος της ενότητας αυτής είναι η σταδιακή εξοικείωση των μαθητών με τον προγραμματισμό μέσα από την αξιοποίηση διαθέσιμων εκπαιδευτικών περιβαλλόντων οπτικού προγραμματισμού.</p> <p>Ο εκπαιδευτικός, μέσα από την παρουσίαση και συζήτηση έτοιμων προγραμμάτων, προκαλεί το ενδιαφέρον των μαθητών και διαμορφώνει ένα πλαίσιο ενεργητικής συμμετοχής. Οι μαθητές εισάγονται σε εκπαιδευτικά περιβάλλοντα προγραμματισμού και εξοικειώνονται με τις βασικές συνιστώσες τους μέσα από κατάλληλες δραστηριότητες, οι οποίες έχουν ως στόχο τη διερευνητική προσέγγιση της γνώσης, την αλληλεπίδραση με τον υπολογιστή, τη συνεργασία, την αυτενέργεια, τη δημιουργικότητα και την ενεργοποίηση της φαντασίας των μαθητών.</p> <p>Οι μαθητές εισάγονται στην έννοια του αλγορίθμου, έχοντας ως γενικό προσανατολισμό τη μετάβαση από την ψηφιακή ζωγραφική στα προγραμματιζόμενα πολυμέσα. Στόχος είναι η δημιουργία μικρών εφαρμογών, στις οποίες θα προκαλείται η δράση αντικειμένων στη σκηνή ή στο χώρο επισκόπησης με χρήση χειριστηρίων (πληκτρολόγιο, ποντίκι). Ενδεικτικά παραδείγματα έργων είναι ο προγραμματισμός της κίνησης ενός αντικειμένου (π.χ. μπάλα, ζώακι, αερόστατο) στο χώρο ή σε λαβύρινθο.</p>		
<p>Ενδεικτικός διδακτικός χρόνος: 4 ώρες</p>		
<p>Προσδοκώμενα Μαθησιακά Αποτελέσματα</p>	<p>Βασικά θέματα</p>	<p>Εκπαιδευτικό υλικό</p>
<p>Ο μαθητής/τρια πρέπει να είναι ικανός/ή</p> <ul style="list-style-type: none"> • να υλοποιεί λειτουργίες και σενάρια με βάση προκαθορισμένα γεγονότα χρησιμοποιώντας περιβάλλον οπτικού προγραμματισμού • να συσχετίζει γεγονότα με τις αντίστοιχες εντολές • να επιλέγει αντικείμενα από το προγραμματιστικό περιβάλλον και να ορίζει τις ιδιότητες και τη συμπεριφορά τους • να αναπτύσσει απλές εφαρμογές σε περιβάλλον οπτικού προγραμματισμού • να εφαρμόζει τεχνικές ελέγχου και διόρθωσης σφαλμάτων στα προγράμματα που δημιουργεί 	<p>Σύνταξη εντολών ελέγχου αντικειμένων</p> <p>Προγραμματισμός απλών πολυμεσικών εφαρμογών (πλοήγηση, αλληλεπίδραση)</p> <p>Τεχνικές ελέγχου και διόρθωσης προγραμμάτων</p>	<p>Σχολικό εγχειρίδιο</p> <p>Scratch</p> <p>BYOB</p> <p>Starlogo TNG</p> <p>Turtle Art</p> <p>Kodu</p> <p>Story telling Alice, Mama gameMaker</p> <p>GreenFoot</p> <p>e-toys</p>
<p>Δραστηριότητες</p>		
<p>Με την καθοδήγηση του εκπαιδευτικού, οι μαθητές συνεργάζονται και προγραμματίζουν/επεκτείνουν απλές εφαρμογές σε εκπαιδευτικό περιβάλλον οπτικού προγραμματισμού.</p> <p>Ενδεικτικά παραδείγματα είναι η δημιουργία</p> <ul style="list-style-type: none"> • πολυμεσικών καρτών με σύνθεση στοιχείων που αντιδρούν σε ενέργειες του χρήστη. • απλών ψηφιακών διαδραστικών αφηγήσεων ή ιστοριών που χρησιμοποιούν πλαίσια διαλόγου για τους χαρακτήρες. <p>Οι μαθητές παρουσιάζουν τα έργα τους στην τάξη και ακολουθεί συζήτηση με στόχο τη βελτίωσή τους.</p>		
<p>Ενδεικτικός διδακτικός χρόνος: 4 ώρες</p>		

Πίνακας 3.9: Ενότητα "Προγραμματίζω τον υπολογιστή" ΤΠΕ Α' Γυμνασίου

3.2.2 Β' Γυμνασίου

Σε σύνολο 54 διδακτικών ωρών, στην ενδεικτική κατανομή ανά ενότητα και στον άξονα «Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ» προτεινόμενης διάρκειας 32 ωρών, η υποενότητα «Προγραμματίζω τον υπολογιστή» κατέχει 10 ώρες.

Άξονες μαθησιακών στόχων	Προτεινόμενες ώρες διδασκαλίας
Χειρίζομαι και δημιουργώ με τα εργαλεία των ΤΠΕ <ul style="list-style-type: none"> • Βασικές έννοιες ΤΠΕ • Λειτουργικά περιβάλλοντα • Δημιουργώ με τον κειμενογράφο 	10
Αναζητώ πληροφορίες, επικοινωνώ και συνεργάζομαι με ΤΠΕ <ul style="list-style-type: none"> • Δημιουργώ και εκφράζομαι με πολυμέσα και παρουσιάσεις • Διερευνώ και συνεργάζομαι μέσω του Διαδικτύου 	12
Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ <ul style="list-style-type: none"> • Λύνω προβλήματα με υπολογιστικά φύλλα • Προγραμματίζω τον υπολογιστή • Υλοποιώ σχέδια έρευνας 	32

Πίνακας 3.10: Ενδεικτική κατανομή διδακτικών ωρών ΤΠΕ Β' Γυμνασίου

Τα προσδοκώμενα μαθησιακά αποτελέσματα, τα βασικά θέματα, καθώς και το προτεινόμενο εκπαιδευτικό υλικό και μαθησιακές δραστηριότητες αναφέρονται αναλυτικά παρακάτω:

Προσδοκώμενα Μαθησιακά Αποτελέσματα	Βασικά θέματα	Εκπαιδευτικό υλικό
Ο μαθητής/τρια πρέπει να είναι ικανός/ή <ul style="list-style-type: none"> • να περιγράφει έναν αλγόριθμο με φυσική γλώσσα κατά βήματα • να κωδικοποιεί έναν αλγόριθμο σε προγραμματιστικό περιβάλλον • να αναλύει ένα πρόβλημα σε επιμέρους απλούστερα • να κατανοεί το ρόλο της διαδικασίας σε ένα πρόγραμμα • να δημιουργεί διαδικασίες στα έργα του • να κατανοεί τα πλεονεκτήματα από τη χρήση παραμέτρων σε διαδικασίες • να ορίζει τις κατάλληλες παραμέτρους στις διαδικασίες που αναπτύσσει • να κατανοεί και να εξηγεί τη λειτουργία ολοκληρωμένων προγραμμάτων 	Κατανόηση και ανάλυση προβλήματος Η έννοια του αλγορίθμου Η έννοια του προγράμματος Η έννοια της διαδικασίας (υποπρόγραμμα) Η έννοια της μεταβλητής ως παραμέτρου σε διαδικασία	Σχολικό εγχειρίδιο Προσομοιώσεις αλγορίθμων Applets και προσομοιώσεις στον Παγκόσμιο Ιστό Scratch BYOB K-turtle msw logo Microworld pro Starlogo TNG Εκπαιδευτική ρομποτική
Δραστηριότητες		
Ο εκπαιδευτικός αναφέρει παραδείγματα αλγορίθμων από την καθημερινή ζωή, όπως είναι οι συνταγές μαγειρικής, το μενού χειρισμού μιας ψηφιακής συσκευής (π.χ. κινητό, ψηφιακή κάμερα, GPS). Οι μαθητές περιγράφουν τη διαδικασία επίλυσης ενός προβλήματος (σύλληψη, διερεύνηση, σχεδιασμός, επίλυση, αποτέλεσμα) και εισάγονται στην έννοια του αλγορίθμου και στον προγραμματισμό. Καλούνται να μελετήσουν προβλήματα των οποίων η λύση είναι μια ακολουθία βημάτων. Ενδεικτικά παραδείγματα αλγορίθμων: <ul style="list-style-type: none"> • Επιστροφή από το σχολείο στο σπίτι • Εύρεση της εξόδου από έναν λαβύρινθο • Σχεδίαση της ελληνικής σημαίας 		

<ul style="list-style-type: none"> • Αλφαβητική ταξινόμηση καρτών με ονόματα μαθητών • Μέτρηση 4 λίτρων νερού με χρήση δύο δοχείων των 3 και 5 λίτρων • Το πρόβλημα των πύργων του Ανόι • Το παιχνίδι της τριλιζας. <p>Τα προβλήματα αυτά μπορούν να προσεγγιστούν με παιχνίδι ρόλων ή/και με χρήση προσομοιώσεων (java applets, flash animations). Στόχος είναι οι μαθητές να προβληματιστούν για το πώς μπορούν να χρησιμοποιήσουν τον υπολογιστή ως εργαλείο επίλυσης προβλημάτων. Διαπιστώνουν την αναγκαιότητα μιας γλώσσας επικοινωνίας με τον υπολογιστή.</p> <p>Στη συνέχεια εξοικειώνονται με το προγραμματιστικό περιβάλλον που θα χρησιμοποιήσουν. Περιγράφουν σε φυσική γλώσσα με βήματα αλγορίθμους σχεδίασης απλών σχημάτων. Τέλος, κωδικοποιούν τους αλγορίθμους στο προγραμματιστικό περιβάλλον. Ενδεικτικά παραδείγματα δραστηριοτήτων που αναθέτει ο εκπαιδευτικός είναι η σχεδίαση απλών σχημάτων (π.χ. τετράγωνο, ορθογώνιο, σκάλα) ή/και γραμμμάτων (π.χ. Ι, Γ, Π, Τ, Η, Ε, Ξ). Ακολούθως, οι μαθητές τροποποιούν τα προγράμματά τους σχεδιάζοντας τα παραπάνω σχήματα με διαφορετικά χαρακτηριστικά (πάχος, χρώμα γραμμής και χρώμα γεμίματος), ώστε να προκύψει η αναγκαιότητα της χρήσης παραμέτρων στις διαδικασίες.</p> <p>Στη συνέχεια, οι μαθητές καλούνται να σχεδιάσουν κανονικά πολύγωνα (τετράγωνο, εξάγωνο, οκτάγωνο) συνδυάζοντας μεμονωμένες εντολές. Ο εκπαιδευτικός καθοδηγεί τους μαθητές να εντοπίσουν επαναλαμβανόμενες ομάδες εντολών στην κωδικοποίηση που έχουν αναπτύξει (π.χ. στον αλγόριθμο σχεδίασης του οκταγώνου).</p> <p>Ενδεικτικός διδακτικός χρόνος: 4 ώρες</p>		
Προσδοκώμενα Μαθησιακά Αποτελέσματα		
<p>Ο μαθητής/τρια πρέπει να είναι ικανός/ή</p> <ul style="list-style-type: none"> • να αντιλαμβάνεται την αναγκαιότητα και τη χρησιμότητα των δομών επανάληψης και επιλογής • να χρησιμοποιεί εντολές επανάληψης και επιλογής στα προγράμματα που αναπτύσσει • να οικοδομεί σύνθετες διαδικασίες βασιζόμενος σε προϋπάρχουσες • να κατανοεί το μηχανισμό μεταβίβασης ελέγχου από μια διαδικασία σε άλλη, κατά την κλήση της • να δημιουργεί διαδικασίες που καλούν άλλες διαδικασίες • να εφαρμόζει τεχνικές ελέγχου και διόρθωσης σφαλμάτων στα προγράμματα που δημιουργεί • να κατανοεί τη λειτουργία ολοκληρωμένων προγραμμάτων 	<p>Βασικά θέματα</p> <p>Δομή επανάληψης</p> <p>Εντολές εισόδου και εξόδου</p> <p>Κλήση διαδικασιών από διαδικασίες</p> <p>Σύνθετες διαδικασίες</p> <p>Στρατηγικές ανάπτυξης και διόρθωσης προγραμμάτων</p>	<p>Εκπαιδευτικό υλικό</p> <p>Σχολικό εγχειρίδιο</p> <p>Scratch</p> <p>BYOB</p> <p>K-turtle</p> <p>msw logo</p> <p>Microworld pro</p> <p>Starlogo TNG</p> <p>Turtle Art</p>
Δραστηριότητες		
<p>Σε συνέχεια της προηγούμενης ενότητας, οι μαθητές τροποποιούν την κωδικοποίηση των διαδικασιών σχεδίασης σχημάτων χρησιμοποιώντας επαναληπτική δομή. Καθοδηγούνται ώστε να φτάσουν στη γενίκευση δημιουργώντας διαδικασία που σχεδιάζει κανονικά πολύγωνα με χρήση παραμέτρου (το πλήθος των γωνιών). Ο καθορισμός των τιμών των παραμέτρων μπορεί να γίνεται από το χρήστη της εφαρμογής με εντολή εισόδου.</p> <p>Οι μαθητές πειραματίζονται εκτελώντας το πρόγραμμα πολλές φορές και μεταβάλλοντας κάθε φορά το πλήθος των γωνιών. Η δραστηριότητα ολοκληρώνεται με τη σχεδίαση του κύκλου και τη συσχέτιση γνώσεων που έχουν από τα μαθηματικά.</p> <p>Ενδεικτικός διδακτικός χρόνος: 2 ώρες</p> <p>Στην ενότητα αυτή οι μαθητές αναπτύσσουν σύνθετα έργα με ιεραρχική χρήση διαδικασιών. Οι μαθητές καλούνται να υλοποιήσουν διαδικασία σχεδιασμού σπιτιού που θα χρησιμοποιεί τη διαδικασία πολύγωνο για να σχεδιάσει το τετράγωνο (κύριο κτίριο) και το τρίγωνο (σκεπή). Αναδεικνύεται με τον τρόπο αυτό η σημασία της κλήσης διαδικασίας από διαδικασία και αναλύονται οι έννοιες του ιεραρχικού σχεδιασμού και του τμηματικού προγραμματισμού. Εναλλακτικό παράδειγμα μπορεί να είναι η σχεδίαση τραίνου ή το πλάνο της τάξης.</p>		

Πρόσθετες δραστηριότητες που μπορούν να υλοποιηθούν από τους μαθητές είναι ο σχεδιασμός σύνθετων σχημάτων, όπως κάστρο, δένδρο, λουλούδι, αυτοκίνητο, ρολόι με δείκτες, ηλιακό σύστημα κ.λπ. Οι μαθητές καθοδηγούνται να αναλύσουν τα σχήματα σε απλούστερα, να προτείνουν κωδικοποιήσεις, να εντοπίσουν και να διορθώσουν λάθη στο πρόγραμμά τους χρησιμοποιώντας το προγραμματιστικό περιβάλλον.

Ενδεικτικός διδακτικός χρόνος: 4 ώρες

Πίνακας 3.11: Ενότητα "Προγραμματίζω τον υπολογιστή" ΤΠΕ Β' Γυμνασίου

3.2.3 Γ' Γυμνασίου

Σε σύνολο 54 διδακτικών ωρών, στην ενδεικτική κατανομή ανά ενότητα και στον άξονα «Διερευνώ, σχεδιάζω και λύνω προβλήματα με ΤΠΕ» προτεινόμενης διάρκειας 36 ωρών, η υποενότητα «Προγραμματίζω τον υπολογιστή» κατέχει 8 ώρες.

Άξονες μαθησιακών στόχων	Προτεινόμενες ώρες διδασκαλίας
Δημιουργώ, επικοινωνώ και συνεργάζομαι με ΤΠΕ <ul style="list-style-type: none"> • Δημιουργώ παρουσιάσεις • Δημιουργώ σε συνεργατικά περιβάλλοντα • Επικοινωνώ & συνεργάζομαι σε διαδικτυακά περιβάλλοντα 	18
Διερευνώ, σχεδιάζω και λύνω προβλήματα με ΤΠΕ <ul style="list-style-type: none"> • Λύνω προβλήματα με υπολογιστικά φύλλα • Προγραμματίζω τον υπολογιστή • Υλοποιώ σχέδια έρευνας 	36

Πίνακας 3.12: Ενδεικτική κατανομή διδακτικών ωρών ανά ενότητα ΤΠΕ Γ' Γυμνασίου

Τα προσδοκώμενα μαθησιακά αποτελέσματα, τα βασικά θέματα, καθώς και το προτεινόμενο εκπαιδευτικό υλικό και μαθησιακές δραστηριότητες αναφέρονται αναλυτικά παρακάτω:

Προσδοκώμενα Μαθησιακά Αποτελέσματα	Βασικά θέματα	Εκπαιδευτικό υλικό
Ο μαθητής/τρια πρέπει να είναι ικανός/ή <ul style="list-style-type: none"> • να διακρίνει και να κατονομάζει τις έννοιες δεδομένα και πληροφορία στο πλαίσιο πραγματικών προβλημάτων • να περιγράφει τον κύκλο επεξεργασίας δεδομένων • να αναγνωρίζει τον υπολογιστή ως μηχανή επεξεργασίας που δέχεται δεδομένα και παράγει πληροφορίες • να διακρίνει τους διαφορετικούς τύπους δεδομένων (αριθμητικά, αλφαριθμητικά) • να ορίζει τις μεταβλητές που απαιτούνται για την ανάπτυξη των προγραμμάτων που 	Δεδομένα Πληροφορία Κύκλος επεξεργασίας δεδομένων Πρόγραμμα Τύποι δεδομένων - Μεταβλητές Απόδοση τιμής σε μεταβλητή Τελεστές Σύνταξη εκφράσεων Εντολές εισόδου εξόδου Δομή επιλογής	Αλγοριθμική ΓΛΩΣΣΟΜΑΘΕΙΑ Scratch BYOB K-turtle Starlogo TNG Turtle Art Story telling Alice, Mama gameMaker Kodu GreenFoot e-toys

<p>σχεδιάζει</p> <ul style="list-style-type: none"> • να χρησιμοποιεί εντολές εισόδου/εξόδου στα προγράμματα που αναπτύσσει • να συντάσσει αριθμητικές και λογικές εκφράσεις • να αντιλαμβάνεται την αναγκαιότητα και τη χρησιμότητα της δομής επιλογής • να χρησιμοποιεί σύνθετες εντολές επιλογής (εμφωλευμένα ΑΝ) στα προγράμματα που αναπτύσσει • να χρησιμοποιεί σύνθετες εντολές επανάληψης στα προγράμματα που αναπτύσσει (εμφωλευμένη επανάληψη) • να ελέγχει την ορθότητα των προγραμμάτων του εκτελώντας τα με διαφορετικά δεδομένα εισόδου • να εφαρμόζει τεχνικές ελέγχου και διόρθωσης σφαλμάτων στα προγράμματα που δημιουργεί 		
<p>Δραστηριότητες</p>		
<p>Ο εκπαιδευτικός συνδέει την ενότητα αυτή με την αντίστοιχη ενότητα της προηγούμενης τάξης. Αναδεικνύεται η ανάγκη υλοποίησης αλγορίθμων που θα επιτελούν αριθμητικές και λογικές πράξεις. Παρατίθενται προβλήματα που απαιτούν την πραγματοποίηση υπολογισμών και συσχετίζεται η σύνταξη αριθμητικών εκφράσεων με τον αντίστοιχο τρόπο σύνταξης στα υπολογιστικά φύλλα.</p> <p>Ενδεικτικά υπολογιστικά προβλήματα που επιλύουν οι μαθητές σε προγραμματιστικό περιβάλλον είναι:</p> <ul style="list-style-type: none"> • Διενέργεια απλών υπολογισμών (π.χ. μέσος όρος βαθμολογίας, περίμετρος και εμβαδό τετραγώνου) • Υλοποίηση αριθμομηχανής με μνήμη • Υπολογισμός της τιμής μιας συνάρτησης • Αντιμετάθεση περιεχομένων δύο μεταβλητών. <p>Οι μαθητές σχεδιάζουν τον αλγόριθμο στο χαρτί και περιγράφουν τα δεδομένα, την επεξεργασία που πρέπει να υλοποιηθεί και την πληροφορία που προκύπτει από τη συγκεκριμένη επεξεργασία. Τέλος, υλοποιούν πρόγραμμα σε προγραμματιστικό περιβάλλον.</p> <p>Οι μαθητές σχεδιάζουν και υλοποιούν προγράμματα που απαιτούν χρήση δομής επιλογής. Ενδεικτικά παραδείγματα προβλημάτων είναι τα εξής:</p> <ul style="list-style-type: none"> • Σχεδιασμός και υλοποίηση ενός γύρου παιχνιδιού (π.χ. Πέτρα - Ψαλίδι - Χαρτί, Κορώνα - Γράμματα, Βρες τον αριθμό, Κρεμάλα) • Εύρεση του μεγίστου/ελαχίστου τριών αριθμών • Ενημέρωση υπολοίπου χρόνου ομιλίας καρτοκινητού με αποστολή μηνύματος • Προσομοίωση λειτουργίας ATM κατά την ανάληψη χρημάτων. <p>Ενδεικτικός διδακτικός χρόνος: 8 ώρες</p>		

Πίνακας 3.13: Ενότητα "Προγραμματίζω τον υπολογιστή" ΤΠΕ Γ' Γυμνασίου

3.3 Πληροφορική και προγραμματισμός στο Λύκειο

Το μάθημα της Πληροφορικής εισήχθη το 1998 στο Ενιαίο Λύκειο, στα πλαίσια του Ενιαίου Πλαισίου Προγραμμάτων Σπουδών Πληροφορικής (ΕΠΠΣΠ, Δεκέμβριος 1997). Το συγκεκριμένο ΠΣ αντιμετώπιζε την Πληροφορική ως αυτόνομο γνωστικό αντικείμενο, με ιδιαίτερη αξία για την ολοκληρωμένη προετοιμασία των μαθητών, σε ότι αφορά στην πορεία τους μετά το Λύκειο.

Σε γενικές γραμμές, τα μαθήματα επιλογής εστίαζαν κυρίως σε εφαρμογές πληροφορικής, εμβαθύνοντας σε θέματα που γνώρισαν οι μαθητές στο Γυμνάσιο, στα πρότυπα του σπειροειδούς μοντέλου μάθησης.

Ο προγραμματισμός υπολογιστών δεν έλειπε ως ενότητα από τα μαθήματα επιλογής, αλλά η ουσιαστικότερη επαφή των μαθητών με θέματα αλγοριθμικής σκέψης, βασικών αρχών πληροφορικής και εννοιών προγραμματισμού γινόταν στο μάθημα «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον» της Τεχνολογικής κατεύθυνσης της Γ' Λυκείου.

Υπήρχε, βέβαια, το ενδεχόμενο, μαθητές να αποφοιτήσουν από το Λύκειο, χωρίς να έχουν διδαχθεί ούτε μια ώρα πληροφορικής, πόσο μάλλον προγραμματισμού.

Τον Απρίλιο του 2014 ανακοινώθηκαν, από το Υπουργείο Παιδείας και Θρησκευμάτων, αλλαγές στη διάρθρωση των μαθημάτων πληροφορικής και του περιεχομένου τους, οι οποίες αναμένεται να τεθούν σε εφαρμογή από το σχολικό έτος 2014-2015. Τα κύρια σημεία τους παρουσιάζονται στη συνέχεια, αφού πρώτα γίνει αναφορά στο τρέχον πλαίσιο των αντίστοιχων μαθημάτων, όπως αυτό ισχύει μέχρι σήμερα.

3.3.1 Η Πληροφορική ως μάθημα επιλογής

Η προσέγγιση αυτή συνοδεύτηκε από την ένταξη των μαθημάτων επιλογής «Εφαρμογές Πληροφορικής» και «Εφαρμογές Υπολογιστών» στο ωρολόγιο πρόγραμμα, των Α' και Β'/Γ' τάξεων αντίστοιχα, με γενικό σκοπό:

- *την επέκταση της γενικής πληροφορικής παιδείας των μαθητών με έμφαση στην ανάπτυξη ικανοτήτων και δεξιοτήτων στη χρήση και αξιοποίηση των υπολογιστικών και δικτυακών τεχνολογιών ως εργαλείων μάθησης και σκέψης*
- *την ενημέρωση των μαθητών για τις εφαρμογές της πληροφορικής στο σύγχρονο κόσμο και ειδικότερα για τις δυνατότητες που προσφέρει και τις προοπτικές που δημιουργεί στον κλάδο/κατεύθυνση που επέλεξαν (ή πρόκειται να επιλέξουν) για να σπουδάσουν*
- *την ευαισθητοποίηση, τον προβληματισμό και την ανάπτυξη κριτικής ικανότητας εκ μέρους των μαθητών, στα κοινωνικά, ηθικά, πολιτισμικά, κ.ά. ζητήματα που τίθενται με την*

«εισβολή» των υπολογιστικών και δικτυακών τεχνολογιών σε όλους τους τομείς της ανθρώπινης δραστηριότητας.

Ειδικότερα σε σχέση με την ενότητα του προγραμματισμού και την απαίτηση υλοποίησης του παραπάνω γενικού σκοπού, ο άξονας υλοποίησης «Διερευνώ - Δημιουργώ - Ανακαλύπτω», προτείνει *«οι μαθητές να δραστηριοποιούνται στο πλαίσιο πιο σύνθετων και ολοκληρωμένων εργασιών, χρησιμοποιώντας λογισμικό εφαρμογών γενικής χρήσης, εκπαιδευτικό λογισμικό, προγραμματιστικά εργαλεία, λογισμικό ανάπτυξης εφαρμογών πολυμέσων και λογισμικό δικτύων».*

Παρομοίως, οι μαθητές που θα έχουν παρακολουθήσει με επιτυχία τα μαθήματα Εφαρμογές Πληροφορικής και Εφαρμογές Υπολογιστών, πρέπει, μεταξύ άλλων:

- *Να μπορούν να περιγράψουν τα χαρακτηριστικά και τις δυνατότητες των σύγχρονων προγραμματιστικών εργαλείων.*
- *Να μπορούν να επιλύουν απλά προβλήματα με χρήση προγραμματιστικών εργαλείων.*
- *Να μπορούν να αναπτύσσουν απλές εφαρμογές πολυμέσων.*

3.3.2 Η πληροφορική στην Τεχνολογική Κατεύθυνση του Γενικού Λυκείου

Οι μαθητές του κύκλου «Πληροφορικής και Υπηρεσιών» της Τεχνολογικής Κατεύθυνσης της Γ' τάξης Ενιαίου/Γενικού Λυκείου (ΓΕΛ), διδάσκονται το μάθημα «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον», συχνά αναφερόμενο και συντομογραφικά ως ΑΕΠΠ.

Ο γενικός σκοπός του μαθήματος είναι, να αναπτύξουν οι μαθητές αναλυτική και συνθετική σκέψη, να αποκτήσουν ικανότητες μεθοδολογικού χαρακτήρα και να μπορούν να επιλύουν απλά προβλήματα σε προγραμματιστικό περιβάλλον.

Άξονες υλοποίησης του γενικού σκοπού

Η προσέγγιση των εννοιών και η καλλιέργεια δεξιοτήτων που απαιτούνται για την υλοποίηση του γενικού σκοπού ταξινομούνται σε τρεις άξονες:

- **Ανάλυση-Σχεδίαση:** οι μαθητές κατανοούν το πρόβλημα, το αναλύουν, προσεγγίζουν με αυστηρότητα την έννοια του αλγορίθμου και περιγράφουν την αλγοριθμική διαδικασία επίλυσής του.
- **Υλοποίηση σε προγραμματιστικό περιβάλλον:** οι μαθητές μαθαίνουν να χρησιμοποιούν προγραμματιστικά εργαλεία, να εφαρμόζουν προγραμματιστικές τεχνικές, να γράφουν το πρόγραμμα, να το εκτελούν, να το διορθώνουν και να το βελτιώνουν.
- **Τεκμηρίωση-Αξιολόγηση:** οι μαθητές τεκμηριώνουν την εργασία τους και αξιολογούν την ποιότητά της.

Βασικός πυρήνας γνώσεων και δεξιότητες που πρέπει να αποκτηθούν

Οι μαθητές που θα έχουν παρακολουθήσει με επιτυχία το μάθημα, πρέπει:

- να μπορούν να διακρίνουν και να αναγνωρίζουν προβλήματα και καταστάσεις που επιλύονται/αντιμετωπίζονται σε προγραμματιστικό περιβάλλον
- να μπορούν να αποφασίζουν σχετικά με την πολυπλοκότητα προβλημάτων και καταστάσεων
- να μπορούν να αναλύουν ένα απλό πρόβλημα και να σχεδιάζουν τη λύση του
- να έχουν αναπτύξει ικανότητες μοντελοποίησης και αλγοριθμικής επίλυσης προβλημάτων
- να μπορούν να χρησιμοποιούν συμβολικές μεθόδους για την επίλυση προβλημάτων και την επεξεργασία δεδομένων
- να μπορούν να χρησιμοποιούν σύγχρονα προγραμματιστικά εργαλεία για την υλοποίηση αλγορίθμων
- να μπορούν να προσδιορίζουν τους απαιτούμενους πόρους του συστήματος.

Προτεινόμενα προγραμματιστικά εργαλεία

Στα πλαίσια του έργου «Πλειάδες» / Ενότητας «Νηρηίδες» του ΥΠΕΠΘ έχει αναπτυχθεί το ολοκληρωμένο προγραμματιστικό περιβάλλον διδακτικής υποστήριξης μαθημάτων Πληροφορικής Γυμνασίου-Λυκείου με την ονομασία «Αλγοριθμική-Προγραμματισμός»²⁵. Όπως αναφέρεται στην επίσημη ιστοσελίδα του: «Το προτεινόμενο ολοκληρωμένο εκπαιδευτικό πακέτο αποσκοπεί – μέσω της εντρύφησης με την αλγοριθμική και τον προγραμματισμό - στην καλλιέργεια της αναλυτικής σκέψης και συνθετικής ικανότητας των μαθητών, στην ανάπτυξη ικανοτήτων μεθοδολογικού χαρακτήρα, στην καλλιέργεια της αυστηρότητας της διατύπωσης και της έκφρασης. Στόχος δεν είναι η εκμάθηση κάποιας συγκεκριμένης γλώσσας προγραμματισμού και για το λόγο αυτό το προτεινόμενο ψηφιακό περιβάλλον δεν αντιστοιχεί σε κάποιο από αυτά των γλωσσών».

Το ολοκληρωμένο πακέτο περιλαμβάνει αλληλεπιδραστικό περιβάλλον δημιουργίας, εκτέλεσης και ελέγχου, τόσο διαγραμμάτων ροής, όσο και προγραμμάτων σε ψευδοκώδικα, πλήρως συμβατού με τη ΓΛΩΣΣΑ (το περιβάλλον που περιγράφεται στο χρησιμοποιούμενο διδακτικό πακέτο του μαθήματος «Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον» της Γ' τάξης Τεχνολογικής Κατεύθυνσης Ενιαίου Λυκείου).

Αξίζει να σημειωθεί ότι διατίθενται και άλλα αξιόλογα περιβάλλοντα που υποστηρίζουν τη δημιουργία, εκτέλεση και οπτικοποίηση αλγορίθμων και προγραμμάτων, κατάλληλα για χρήση στη Β/θμια εκπαίδευση και ειδικότερα για το μάθημα της «ΑΕΠΠ»²⁶.

3.3.3 Η Πληροφορική στο «νέο» Γενικό Λύκειο

Η σχολική χρονιά 2013-2014, σύμφωνα και με το «Σχέδιο Νόμου για την Αναδιάρθρωση της Β' Βάθμιας Εκπαίδευσης»²⁷ του Υπουργείου Παιδείας και Θρησκευμάτων για το «Νέο Λύκειο», είναι η τελευταία χρονιά διδασκαλίας τόσο των προαναφερομένων μαθημάτων επιλογής πληροφορικής, όπως αυτά είχαν καθοριστεί από το 1998, όσο και του αντίστοιχου υποχρεωτικού μαθήματος της Τεχνολογικής Κατεύθυνσης.

²⁵ Το έργο «Αλγοριθμική-Προγραμματισμός»: <http://www.ecedu.upatras.gr/algorithmics/index.htm>

²⁶ Περιβάλλοντα για τη διδασκαλία της ΑΕΠΠ: <http://goo.gl/y6Yd2>

²⁷ ΦΕΚ Νόμου 4186-2013 (Α 193) 17/09/2013 για την αναδιάρθρωση της Δευτεροβάθμιας Εκπαίδευσης και λοιπές διατάξεις.

Συγκεκριμένα, προβλέπεται πλέον το μάθημα επιλογής «Εφαρμογές Πληροφορικής» στην Α' τάξη Λυκείου και το μάθημα γενικής παιδείας «Εισαγωγή στις Αρχές της Επιστήμης των Η/Υ» στις Β' και Γ' τάξεις Λυκείου, ενώ παύει να υφίσταται το αντίστοιχο μάθημα Κατεύθυνσης/Προσανατολισμού στη Γ' Λυκείου. Τα Προγράμματα Σπουδών των δυο ανωτέρω μαθημάτων ανακοινώθηκαν τον Απρίλιο του 2014 και αναμένεται να εφαρμοστούν από το σχολικό έτος 2014-2015.

Σύμφωνα με το ΠΣ του μαθήματος «Εφαρμογές Πληροφορικής» της Α' τάξης Γενικού Λυκείου ²⁸, ένας βασικός στόχος είναι *«οι μαθητές να αναλύουν προβλήματα, να σχεδιάζουν και να αναπτύσσουν μικροεφαρμογές των ηλεκτρονικών υπολογιστών, “έξυπνων” κινητών συσκευών»*. Συγκεκριμένα, προβλέπονται 12 διδακτικές ώρες στην θεματική ενότητα «Προγραμματιστικά Περιβάλλοντα – Δημιουργία Εφαρμογών». Μάλιστα, στις ενδεικτικές δραστηριότητες γίνεται αναφορά στο εκπαιδευτικό προγραμματιστικό περιβάλλον του App Inventor, μαζί με τα Game Maker και Alice.

Το ΠΣ του μαθήματος «Εισαγωγή στις Αρχές της Επιστήμης των Η/Υ» της Β' τάξης Γενικού Λυκείου ²⁹, περιλαμβάνει δυο θεματικές ενότητες που σχετίζονται με τον προγραμματισμό:

α) Στην ενότητα «Αλγόριθμοι», για την οποία προβλέπονται 10 ώρες, οι μαθητές έρχονται σε επαφή με βασικές εντολές και δομές που χρησιμοποιούνται σε έναν αλγόριθμο, μέσω αυθεντικών παραδειγμάτων από τον πραγματικό κόσμο.

β) Στην ενότητα «Προγραμματισμός», για την οποία προβλέπονται 6 ώρες, οι μαθητές συνδυάζουν αλγοριθμικές δομές και δεδομένα/δομές δεδομένων για να δημιουργήσουν κώδικα/πρόγραμμα.

Το μάθημα εξακολουθεί να γίνεται σε εργαστηριακό περιβάλλον, ακολουθώντας μεθόδους αναζήτησης και ανακάλυψης. Ως κοινωνική οργάνωση της τάξης ακολουθείται η ομαδοσυνεργατική προσέγγιση. Ο καθηγητής λειτουργεί ως διευκολυντής για την οικοδόμηση της γνώσης από τον μαθητή κινούμενος με τεχνικές του «πλαισίου στήριξης» (scaffolding).

²⁸ ΦΕΚ 932Β-2014, Απόφαση 53248/Γ2.

²⁹ ΦΕΚ 934Β-2014, Απόφαση 53235/Γ2.

3.3.4 Η Πληροφορική στο Επαγγελματικό Λύκειο

Το περιεχόμενο και ο χαρακτήρας των μαθημάτων πληροφορικής στα Τεχνολογικά Επαγγελματικά Εκπαιδευτήρια (ΤΕΕ), μετονομαζόμενα αργότερα σε Επαγγελματικά Λύκεια (ΕΠΑΛ), καθορίστηκε το 1999, μαζί με τα Προγράμματα Σπουδών των μαθημάτων όλων των τομέων και ειδικοτήτων για όλες τις τάξεις ³⁰ των ΤΕΕ.

Οι μαθητές των ΕΠΑΛ διδάσκονται το μάθημα «Εφαρμογές Πληροφορικής» για όλες τις Ομάδες Προσανατολισμού στην Α' τάξη. Στη Β' τάξη και στον τομέα Πληροφορικής προσφέρονται έξι μαθήματα Πληροφορικής, εκ των οποίων μόνο το μάθημα «Προγραμματιστικά Εργαλεία για το Διαδίκτυο» σχετίζεται με τον προγραμματισμό, καθώς παρουσιάζονται επεκτάσεις της γλώσσας HTML με έμφαση στη χρήση εφαρμογών Διαδικτύου (applets) και οι γλώσσες Javascript και Vbscript. Στην ειδικότητα «Υποστήριξης Συστημάτων, Εφαρμογών και Δικτύων Η/Υ» της Γ' τάξης, από το έξι προσφερόμενα μαθήματα, το μάθημα «Δομημένος Προγραμματισμός» εξετάζεται σε πανελλαδικό επίπεδο, ενώ σχετικό με τον προγραμματισμό είναι και το μάθημα «Στοιχεία Προγραμματισμού σε Γραφικό Περιβάλλον». Σκοπός του πρώτου μαθήματος είναι *«να αποκτήσει ο μαθητής ικανότητες μεθοδολογικού χαρακτήρα, να μπορεί να επιλύει απλά προβλήματα και να αναπτύσσει απλές εφαρμογές σε προγραμματιστικό περιβάλλον»*, ενώ το προτεινόμενο προγραμματιστικό περιβάλλον είναι αυτό της Turbo Pascal. Σκοπός του δεύτερου μαθήματος είναι *«να καταστήσει το μαθητή ικανό να δημιουργεί απλά προγράμματα»*, με τη χρήση Visual Basic.

Στο πλαίσιο υλοποίησης του Νέου Τεχνολογικού Λυκείου, το 2011, ορίστηκαν μεταβατικά ωρολόγια προγράμματα στην Α' τάξη των Ημερησίων ΕΠΑΛ, με στόχο οι μαθητές σταδιακά να ακολουθήσουν το νέο πρόγραμμα. Από τότε, το Υπουργείο έχει ανακοινώσει ότι θα ακολουθήσουν σημαντικές αλλαγές στη δομή του Επαγγελματικού/Τεχνολογικού Λυκείου, αλλά μέχρι την ολοκλήρωση της Μεταπτυχιακής Διατριβής δεν έχει διευκρινιστεί επ' ακριβώς το περιεχόμενο των νέων μαθημάτων πληροφορικής, καθώς αναμένεται η έκδοση των νέων ΠΣ. Σε κάθε περίπτωση, θεωρούμε πως το App Inventor μπορεί κάλλιστα να ενταχθεί στα προτεινόμενα προγραμματιστικά περιβάλλοντα των σχετικών μαθημάτων.

Η μάθηση του προγραμματισμού υπολογιστών, σε όλες τις τάξεις και ηλικίες, δεν στερείται δυσκολιών και παρανοήσεων από τους μαθητές. Στη συνέχεια γίνεται μια επισκόπηση των σημαντικότερων προβλημάτων που έχουν καταγραφεί στην ελληνική και διεθνή βιβλιογραφία.

³⁰ ΦΕΚ 2327-1999, Απόφαση 4219/Γ2.

Κεφάλαιο 4

Προβλήματα στην εκμάθηση του προγραμματισμού υπολογιστών

Στο κεφάλαιο αυτό παρουσιάζονται τα προσδοκώμενα μαθησιακά αποτελέσματα από τη διδασκαλία της πληροφορικής και των ΤΠΕ και ειδικότερα του προγραμματισμού στο ελληνικό εκπαιδευτικό σύστημα και γίνεται βιβλιογραφική επισκόπηση των προβλημάτων που αντιμετωπίζονται κατά τη διδασκαλία και μάθηση βασικών προγραμματιστικών εννοιών διεθνώς.

4.1 Ο Προγραμματισμός στην ελληνική εκπαίδευση

Ειδικά σε ότι αφορά τη διδασκαλία και μάθηση του προγραμματισμού στη σχολική εκπαίδευση, το ενδιαφέρον έχει απομακρυνθεί από τη στείρα εκμάθηση της γραμματικής και του συντακτικού κάποιας γλώσσας προγραμματισμού γενικού σκοπού, για να εστιάσει περισσότερο στην ποικιλία συλλογισμών που κάνουν οι μαθητές. Η εδραιωμένη τεχνική προγραμματισμού είναι ο δομημένος προγραμματισμός, ενώ σε πολλά σχολεία, ειδικά σε μικρότερες ηλικίες,

αξιοποιείται η γλώσσα Logo, η οποία στηρίχθηκε σε ένα διαφορετικό παιδαγωγικό πλαίσιο, τον εποικοδομισμό [41][43]. Σε κάθε περίπτωση, βασικός στόχος είναι η απόκτηση δεξιοτήτων σχεδίασης και υλοποίησης λύσεων. Στη συνέχεια, παρουσιάζονται συνοπτικά, τα προσδοκώμενα αποτελέσματα ανά βαθμίδα εκπαίδευσης, για το μέσο μαθητή στο ελληνικό σχολείο, χωρίς να λαμβάνονται υπόψη μαθήματα ειδικοτήτων ή κατευθύνσεων.

4.1.1 Δημοτικό

Οι μαθητές, τελειώνοντας το Δημοτικό, θα πρέπει να είναι σε θέση να αναλύουν ένα πρόβλημα, να εφαρμόζουν σωστά τις διάφορες προγραμματιστικές δομές, και να μπορούν έτσι να επιλύουν το πρόβλημα αυτό, με τη βοήθεια προγραμματιστικών εργαλείων. Τα προσδοκώμενα μαθησιακά αποτελέσματα για το μαθητή στο Δημοτικό συνοψίζονται στον ακόλουθο πίνακα:

Προσδοκώμενα μαθησιακά αποτελέσματα
<ul style="list-style-type: none">• Διατύπωση απλών εντολών στο περιβάλλον οπτικού προγραμματισμού.• Ορισμός ενεργειών και σεναρίων που πρέπει να εκτελεστούν για την επίτευξη επιθυμητών γεγονότων.• Διάκριση διαφόρων γεγονότων (events) στο περιβάλλον οπτικού προγραμματισμού.• Εντοπισμός και διόρθωση σφαλμάτων.• Εξήγηση γιατί ένα αντικείμενο του προγραμματιστικού περιβάλλοντος συμπεριφέρεται με συγκεκριμένο τρόπο.• Κωδικοποίηση αλγορίθμου σε προγραμματιστικό περιβάλλον και ανάπτυξη μικρών εφαρμογών χρησιμοποιώντας ένα εκπαιδευτικό περιβάλλον οπτικού προγραμματισμού.• Ανάλυση ενός προβλήματος σε επιμέρους απλούστερα.• Σύνθεση ενός έργου προγραμματισμού από τα επιμέρους στοιχεία του (που έχουν προκύψει από την ανάλυση).• Αντίληψη της χρησιμότητας και του ρόλου των διαδικασιών (υποπρογραμμάτων).• Χρήση (κλήση) διαδικασιών στα έργα.

Πίνακας 4.1: Προσδοκώμενα μαθησιακά αποτελέσματα για τον προγραμματισμό στο Δημοτικό

4.1.2 Γυμνάσιο

Στο Γυμνάσιο, γίνεται ιδιαίτερη αναφορά στον οδηγούμενο από γεγονότα (event-driven) προγραμματισμό και τα προσδοκώμενα μαθησιακά αποτελέσματα περιλαμβάνουν επιπρόσθετα τα παρακάτω:

Προσδοκώμενα μαθησιακά αποτελέσματα

- Κατανόηση και εξήγηση της λειτουργίας ολοκληρωμένων προγραμμάτων.
- Επιλογή αντικειμένων από το προγραμματιστικό περιβάλλον και ορισμός των ιδιοτήτων και της συμπεριφοράς τους.
- Αντίληψη της αναγκαιότητας και της χρησιμότητας των δομών επανάληψης και επιλογής.
- Χρήση εντολών επανάληψης και επιλογής.
- Χρήση εντολών εισόδου/εξόδου.
- Κατανόηση της έννοιας της μεταβλητής.
- Ορισμός των μεταβλητών που απαιτούνται για την ανάπτυξη προγραμμάτων.
- Απόδοση τιμής σε μεταβλητή.
- Σύνταξη αριθμητικών και λογικών εκφράσεων.
- Αντίληψη της αναγκαιότητας και της χρησιμότητας της δομής επιλογής.
- Χρήση σύνθετων εντολών επιλογής (εμφωλευμένων ΑΝ).
- Κατανόηση του ρόλου της διαδικασίας σε ένα πρόγραμμα.
- Η έννοια της μεταβλητής ως παραμέτρου σε διαδικασία.
- Δημιουργία διαδικασιών (υποπρογραμμάτων).
- Κατανόηση των πλεονεκτημάτων από τη χρήση παραμέτρων σε διαδικασίες.
- Ορισμός των κατάλληλων παραμέτρων σε διαδικασίες.

Πίνακας 4.2: Προσδοκώμενα μαθησιακά αποτελέσματα για τον προγραμματισμό στο Γυμνάσιο

4.1.3 Λύκειο

Στο Γενικό Λύκειο, σε ότι αφορά τους μαθητές των Α' και Β' τάξεων, στα μαθήματα επιλογής, βασικός σχετικός μαθησιακός στόχος είναι η επίλυση προβλημάτων με χρήση προγραμματιστικών εργαλείων. Γενικά παρέχεται ευελιξία στον εκπαιδευτικό να χρησιμοποιήσει όποιο εργαλείο κρίνει καταλληλότερο για την επίτευξη των μαθησιακών στόχων. Σημαντικό είναι επίσης, οι μαθητές να εισαχθούν σε εκπαιδευτικά περιβάλλοντα προγραμματισμού και να εξοικειωθούν με τις βασικές συνιστώσες τους μέσα από κατάλληλες δραστηριότητες, οι οποίες έχουν ως στόχο τη διερευνητική προσέγγιση της γνώσης, την αλληλεπίδραση με τον υπολογιστή, τη συνεργασία, την αυτενέργεια, τη δημιουργικότητα και την ενεργοποίηση της φαντασίας τους. Συνεπώς, ο καθορισμός των επιμέρους διδακτικών στόχων, η επιλογή του προγραμματιστικού εργαλείου, όπως επίσης και η δημιουργία δραστηριοτήτων είναι στην κρίση του εκπαιδευτικού.

Σύμφωνα με τις αποφάσεις που αφορούν στα νέα μαθήματα «Εφαρμογές Πληροφορικής» και «Εισαγωγή στις Αρχές της Επιστήμης των Η/Υ», τα προσδοκώμενα μαθησιακά αποτελέσματα για το αντικείμενο του προγραμματισμού συνοψίζονται στα εξής:

Προσδοκώμενα μαθησιακά αποτελέσματα

- Περιγραφή του πως από το πρόβλημα φτάνουμε στην εφαρμογή.
- Αναγνώριση εκπαιδευτικών προγραμματιστικών περιβαλλόντων.
- Επιλογή προγραμματιστικού περιβάλλοντος ανάλογα με τις ανάγκες σχεδιασμού κάθε εφαρμογής.
- Διάκριση λειτουργιών και απαιτήσεων μιας εφαρμογής.
- Ανάπτυξη μικροεφαρμογών με εκπαιδευτικά προγραμματιστικά περιβάλλοντα.
- Διάκριση των βασικών εντολών και δομών που χρησιμοποιούνται σε έναν αλγόριθμο.
- Προσδιορισμός του τρόπου λειτουργίας των δομών δεδομένων.
- Εντοπισμός και διόρθωση λογικών λαθών ενός αλγορίθμου.
- Συνδυασμός αλγοριθμικών δομών και δεδομένων/δομών δεδομένων για τη δημιουργία κώδικα/προγράμματος.
- Διαπίστωση της πολυπλοκότητας των εφαρμογών και της ανάγκης η δημιουργία τους να ακολουθεί συγκεκριμένα μοντέλα ανάπτυξης εφαρμογών λογισμικού που εξελίσσονται σε συγκεκριμένες φάσεις.

Πίνακας 4.3: Προσδοκώμενα μαθησιακά αποτελέσματα για τον προγραμματισμό στο Λύκειο.

Σημειώνεται ότι τα νέα ΠΣ των μαθημάτων του Λυκείου ανακοινώθηκαν κατά τη διάρκεια εκπόνησης της Μεταπτυχιακής Διατριβής, ενώ με την ολοκλήρωσή της, δεν είχαν ακόμα ανακοινωθεί οι αναμενόμενες αλλαγές στα Επαγγελματικά Λύκεια.

4.2 Δυσκολίες και παρανοήσεις των μαθητών

Στην παράγραφο 2.3 αναφερθήκαμε στους παράγοντες που δυσχεραίνουν την αποτελεσματική ένταξη του αντικειμένου του προγραμματισμού στην εκπαίδευση, στα αρνητικά της χρήσης γλωσσών γενικού σκοπού σε σχέση με τη χρήση εργαλείων οπτικού προγραμματισμού, καθώς και στην ανάγκη υιοθέτησης μιας περισσότερο εποικοδομιστικής προσέγγισης, που να δίνει ιδιαίτερη σημασία στο πλαίσιο που λαμβάνει χώρα η μάθηση και να έχει νόημα για τους μαθητές. Τέλος, έχει γίνει ήδη μια πρώτη αναφορά στα πλεονεκτήματα της αξιοποίησης των δυνατοτήτων των φορητών συσκευών των μαθητών.

Στη συνέχεια, γίνεται μια πιο λεπτομερής επισκόπηση των συνήθων προβλημάτων που αντιμετωπίζουν οι αρχάριοι στον προγραμματισμό, όπως έχει καταγραφεί στην ελληνική και διεθνή βιβλιογραφία.

4.2.1 Γενικές δυσκολίες

Ο προγραμματισμός δεν αποτελεί ένα εύκολο αντικείμενο για τους μαθητές, ανεξαρτήτως βαθμίδας εκπαίδευσης. Οι δυσκολίες πηγάζουν τόσο από παρανοήσεις των μαθητών, όσο και από τον τρόπο διδασκαλίας και μάθησης, καθώς και από το ίδιο το προγραμματιστικό περιβάλλον που χρησιμοποιείται [85].

Ο Du Boulay [13] διακρίνει πέντε περιοχές δυσκολίας στην εκμάθηση προγραμματισμού:

α) Προσανατολισμός (orientation)

Οι μαθητές θα πρέπει να ανακαλύψουν ποια είναι η χρησιμότητα του προγραμματισμού, που και πως μπορεί να εφαρμοστεί στην καθημερινότητά τους, τι είδους προβλήματα ενδέχεται να κληθούν να αντιμετωπίσουν, καθώς και τι θα αποκομίσουν, αφιερώνοντας χρόνο και κόπο για την εκμάθησή του.

β) Εννοιολογική μηχανή (notational machine)

Ένα πρόγραμμα που εκτελείται στον υπολογιστή αποτελεί έναν μηχανισμό, τη συμπεριφορά του οποίου, καλείται να αντιληφθεί και να ελέγξει ο μαθητής. Η έρευνα έχει δείξει ότι οι μαθητές δεν αναπτύσσουν αποτελεσματικά εννοιολογικά μοντέλα που αφορούν στον υπολογιστή και στο πως λειτουργεί κατά την εκτέλεση του προγράμματος. Είναι σημαντικό, οι μαθητές να κατασκευάζουν αναλυτικά εννοιολογικά μοντέλα λειτουργίας, καθώς θα τους βοηθήσουν να καταλάβουν τη λειτουργία των προγραμμάτων και να κατανοήσουν καλύτερα τις νέες έννοιες που διδάσκονται.

γ) Συντακτικό και σημασιολογία της γλώσσας προγραμματισμού (notation)

Οι μαθητές πολύ συχνά υποπίπτουν σε συντακτικά και σημασιολογικά σφάλματα. Οι αρχάριοι προγραμματιστές, κατά τη συγγραφή κώδικα, μπερδεύονται από τον τρόπο έκφρασης της φυσικής γλώσσας και διατυπώνουν λανθασμένα τις εντολές ή κάνουν συντακτικές γενικεύσεις που δεν υποστηρίζονται από τη γλώσσα προγραμματισμού.

δ) Δομές (structures)

Οι μαθητές μαθαίνουν επιφανειακά και προσωρινά τις βασικές αλγοριθμικές δομές, με αποτέλεσμα να μην είναι σε θέση να τις ανακαλέσουν και να τις ενσωματώσουν στα νέα προγράμματα που δημιουργούν.

ε) Διαδικασίες ανάπτυξης και εκσφαλμάτωσης προγραμμάτων (pragmatics)

Συχνά ένας αρχάριος δεν είναι σε θέση να διακρίνει τα σφάλματα στον κώδικα, να τα εντοπίσει και να τα αντιμετωπίσει. Ενδεχομένως, μάλιστα, να προτιμήσει να διαγράψει ολόκληρο τον κώδικα και να τον ξαναγράψει από την αρχή, παρά να προσπαθήσει να αλλάξει το προβληματικό τμήμα του.

Εν γένει, πέρα από τις παραπάνω περιοχές δυσκολίας, δεν υπάρχει άμεσος χειρισμός και οι έννοιες είναι αφηρημένες [05], ενώ εξίσου σημαντικό πρόβλημα εντοπίζεται στον επαρκή συντονισμό των διαφορετικών προβλημάτων [45].

Στην διεθνή έρευνα επισκόπησης του Lahtinen [35], στην οποία συμμετείχαν περισσότεροι από 500 μαθητές και εκπαιδευτικοί, διερευνήθηκαν οι δυσκολίες που αντιμετωπίζουν οι εμπλεκόμενοι τόσο στη διδασκαλία, όσο και στη μάθηση του προγραμματισμού. Στη βιβλιογραφική του επισκόπηση αναφέρει ότι οι αρχάριοι προγραμματιστές περιορίζονται στην επιφανειακή και γραμμή προς γραμμή κατανόηση του κώδικα των προγραμμάτων και αδυνατούν να εφαρμόσουν τις έννοιες που γνωρίζουν και να τις συνδυάσουν για να παράγουν δικά τους έγκυρα προγράμματα.

Ο Pea [44] διεξήγαγε μακροχρόνια έρευνα σε παιδιά 8-12 και 14-17 ετών που μάθαιναν να προγραμματίζουν με τη γλώσσα Logo και περιγράφει τις παρανοήσεις τους στην κατανόηση του τρόπου εκτέλεσης των προγραμμάτων. Χαρακτηριστικά, αναφέρει ότι οι αρχάριοι προγραμματιστές έχουν την ψευδαίσθηση ότι ο υπολογιστής μπορεί να επεξεργαστεί διάφορες εντολές ταυτόχρονα και δεν αντιλαμβάνονται πως η εκτέλεση των εντελών γίνεται ακολουθιακά.

Όπως έχει ήδη αναφερθεί στο κεφάλαιο 2.3.3, οι μαθητές δεν δείχνουν ιδιαίτερο ζήλο για το αντικείμενο του προγραμματισμού, κάτι το οποίο επιδεινώνεται από την κλασική διδακτική

προσέγγιση, με τη διδασκαλία μιας συγκεκριμένης γλώσσας προγραμματισμού γενικού σκοπού. Οι μαθητές, κατά αυτόν τον τρόπο, αναπτύσσουν την πεποίθηση ότι η επίλυση ενός προβλήματος μέσω υπολογιστή σχετίζεται κυρίως με την κωδικοποίηση της λύσης στη γλώσσα αυτή [70][76][81][85].

Γλώσσες γενικού σκοπού, όπως η C και η Pascal, δυσχεραίνουν τους αρχάριους προγραμματιστές, που πρέπει να μάθουν μια πληθώρα εντολών, με πολύ συγκεκριμένο συντακτικό και δομή. Ο Παπαευθυμίου [86] προσθέτει ότι ακόμα και για τη δημιουργία ενός απλού προγράμματος, οι μαθητές απαιτείται να έχουν μάθει τη γλώσσα σε μεγάλο βαθμό, κάτι το οποίο τους κουράζει και τους αποθαρρύνει ήδη από την αρχή. Όσα εργαλεία και εξελιγμένες προγραμματιστικές δυνατότητες κι αν διαθέτει ένα προγραμματιστικό περιβάλλον, η πολυπλοκότητά του αποτελεί ένα επιπλέον πρόβλημα για τους αρχάριους.

Η στόχευση, αντιθέτως, πρέπει να βρίσκεται στην ποικιλία συλλογισμών που κάνουν οι μαθητές, στη μεθοδολογία επίλυσης προβλημάτων και στην εκμάθηση βασικών προγραμματιστικών εννοιών και όχι στο συντακτικό μιας συγκεκριμένης γλώσσας.

Ο Gomes [17] θεωρεί ότι η σημαντικότερη αιτία προβλημάτων είναι η έλλειψη γενικευμένων ικανοτήτων επίλυσης προβλημάτων και η δυσκολία κατανόησης και εφαρμογής αφηρημένων προγραμματιστικών εννοιών, όπως π.χ. των δομών ελέγχου.

4.2.2 Παρανοήσεις μαθητών σε βασικές έννοιες και δομές

Τα μαθήματα πληροφορικής, όπως παρουσιάστηκαν στο κεφάλαιο 3, στοχεύουν, μεταξύ άλλων, στη γνωριμία και εξοικείωση των μαθητών με βασικές προγραμματιστικές έννοιες και δομές. Βασικός στόχος είναι η απόκτηση δεξιοτήτων σχεδίασης και υλοποίησης λύσεων [73]. Οι μαθητές, δηλαδή, θα πρέπει να είναι σε θέση να αναλύουν ένα πρόβλημα, να εφαρμόζουν σωστά τις διάφορες προγραμματιστικές δομές, και να μπορούν έτσι να επιλύουν το πρόβλημα αυτό, με τη βοήθεια προγραμματιστικών εργαλείων. Μέσα από αυτή τους την ενασχόληση, οι μαθητές καλλιεργούν και αναπτύσσουν τη νοητική δεξιότητα να σκέφτονται δομημένα για την επίλυση ενός προβλήματος, ανεξαρτήτως πεδίου και γνωστικού αντικειμένου.

Σύμφωνα με την κλασική διδακτική προσέγγιση, συνήθως παρουσιάζονται κλιμακωτά από τον εκπαιδευτικό οι βασικές προγραμματιστικές έννοιες και δομές, και σταδιακά οι μαθητές έρχονται σε επαφή με τη δομή ακολουθίας, τις μεταβλητές, τη δομή επιλογής, τη δομή επανάληψης, κ.ο.κ.

Συνδυάζοντας και εφαρμόζοντας κατάλληλα τα παραπάνω, οι μαθητές είναι σε θέση να σχεδιάσουν και να αναπτύξουν λύσεις για διάφορα προβλήματα.

Άσχετα με τον τύπο προγραμματισμού που έρχονται σε επαφή οι μαθητές, είτε αυτός είναι διαδικαστικός, είτε αντικειμενοστρεφής, κάποια στιγμή οι μαθητές θα συναντήσουν τις έννοιες των μεταβλητών, των λογικών εκφράσεων, των δομών επιλογής, των επαναληπτικών δομών, των υποπρογραμμάτων και της μεταβίβασης παραμέτρων, έννοιες που σχετίζονται με τις περισσότερες παρανοήσεις εκ μέρους τους.

Όπως έχουν δείξει αρκετές έρευνες στην Ελλάδα [84][68][76][81][87], οι αρχάριοι προγραμματιστές έχουν δυσκολίες στην κατανόηση βασικών εννοιών, όπως οι μεταβλητές, καθώς και των πρωτογενών δομών του προγραμματισμού.

Ακόμα και ο αντικειμενοστρεφής προγραμματισμός, λόγω της αφαίρεσης που διέπει τη σχεδίαση της προγραμματιστικής λύσης και λόγω του διαφορετικού τρόπου σκέψης κατά την ανάλυση και τη σχεδίαση της λύσης αυτής, δεν ευνοεί τους αρχάριους στο να κατανοήσουν ευκολότερα τις βασικές έννοιες προγραμματισμού [84].

Μεταβλητές

Η μεταβλητή, σύμφωνα με τον Knuth [32] ορίζεται ως μια ποσότητα που μπορεί να λαμβάνει διαφορετικές τιμές κατά την εκτέλεση του προγράμματος. Τα χαρακτηριστικά της, εκτός από την τιμή, είναι ένα όνομα, ένα σύνολο ιδιοτήτων και μια αναφορά, δηλαδή η θέση μνήμης που την προσδιορίζει [25].

Η οικοδόμηση της έννοιας της μεταβλητής από τους μαθητές κατέχει πρωταρχική σημασία στον προγραμματισμό, καθώς τους βοηθάει να οικοδομήσουν πιο σύνθετες δομές [89].

Για να λυθεί ένα υπολογιστικό πρόβλημα χρειάζονται συνήθως διάφορες εντολές χειρισμού μεταβλητών, όπως είναι η εκχώρηση, η ανάγνωση και η εγγραφή τιμής, οι οποίες παραπάνω εντολές απαιτείται να συνδυαστούν με εντολές ακολουθίας, επιλογής και επανάληψης, στις οποίες και ενσωματώνονται.

Οι μαθητές, έχοντας διδαχθεί περί μεταβλητών στο μάθημα των μαθηματικών, είναι σε θέση να κατανοήσουν τις μεταβλητές σταθερής τιμής, του υπολογισμού και της αντιγραφής, δυσκολεύονται όμως να κατανοήσουν την περίπτωση της συσσώρευσης. Για παράδειγμα η έκφραση $x = x + 5$ αποτελεί εκχώρηση νέας τιμής στη μεταβλητή x , λαμβάνοντας υπόψη την προηγούμενη της τιμή, αλλά δεν έχει νόημα στα μαθηματικά. Σαν λύση, ο Du Boulay [13] προτείνει τη χρήση πιο ουδέτερων συμβολισμών από το σύμβολο της ισότητας, όπως π.χ. του βέλους προς τη μεριά της εκχώρησης <- .

Ένα ακόμα συνηθισμένο σφάλμα είναι η παρανόηση των μαθητών ότι οι μεταβλητές συγκρατούν στη «μνήμη» τους τις τιμές που παίρνουν και δεν αντιλαμβάνονται πως κάθε νέα τιμή καταχωρείται στη θέση της προηγούμενης, την οποία αντικαθιστά.

Στην έρευνά τους σε μαθητές δευτεροβάθμιας εκπαίδευσης, οι Τζιμογιάννης και Κόμης [88] καταγράφουν τη σύγχυση των προγραμματιστικών μεταβλητών με τις μαθηματικές, την ερμηνεία της εκχώρησης τιμής με την μαθηματική ισότητα, καθώς και την αντίληψη περί διαδοχικών εκχωρήσεων τιμών ως μαθηματικές σχέσεις που ισχύουν ταυτόχρονα (σύστημα εξισώσεων).

Ο Φεσάκης [90] προσθέτει στα παραπάνω τη δυσκολία των μαθητών να καταλάβουν ότι ο τύπος των μεταβλητών παραμένει σταθερός σε ένα πρόγραμμα.

Ο Du Boulay [13] βρίσκει προβληματική την παρομοίωση της μεταβλητής σαν ένα κουτί αποθήκευσης, καθώς αρκετοί μαθητές θεωρούν ότι θα μπορούσε να αποθηκεύσει περισσότερες από μία τιμές.

Μια συνηθισμένη παρερμηνεία της εντολής ανάθεσης τιμής και της θεώρησής της σαν μαθηματικής ισότητας ότι τα δυο μέρη συνδέονται μεταξύ τους, έτσι ώστε όποιες αλλαγές γίνονται στο αριστερό μέρος, επηρεάζουν άμεσα και το δεξί μέρος.

Παρατηρείται επίσης ότι ένα συχνό λάθος είναι η μη απόδοση αρχικής τιμής σε μετρητές βρόχων. Στην περίπτωση αυτή οι μαθητές μπερδεύονται με την αναλογία μεταβλητής – κουτιού, όπου το κουτί είναι άδειο έως ότου τοποθετήσουμε κάτι μέσα του, οπότε ανάλογα θεωρούν ως 0 και το περιεχόμενο της μεταβλητής.

Δομή επιλογής

Η Δομή Επιλογής ή ελέγχου είναι μια από τις βασικότερες δομές που εμφανίζονται σε ένα πρόγραμμα. Η δομή αυτή δίνει τη δυνατότητα σε έναν αλγόριθμο να εκτελέσει συγκεκριμένα σετ εντολών, ανάλογα με τις περιστάσεις. Ο αλγόριθμος επιλέγει ουσιαστικά ποιες εντολές θα εκτελεστούν, ανάλογα με το αν μια συνθήκη είναι αληθής ή ψευδής [74].

Η δομή επιλογής, κατά την εκτέλεση ενός αλγορίθμου, ελέγχει μια συνθήκη. Η συνθήκη είναι μια λογική έκφραση που μπορεί να πάρει τις τιμές αληθής ή ψευδής. Ανάλογα με την τιμή της συνθήκης εκτελείται η κατάλληλη ομάδα εντολών, που αντιστοιχούν σε κάθε περίπτωση. Μια συνθήκη ή λογική έκφραση δύναται να περιλαμβάνει συγκριτικούς, αριθμητικούς ή λογικούς τελεστές.

Ο Εφόπουλος [71] αναφέρει ότι η δομή επιλογής προκαλεί αρκετές δυσκολίες στους μαθητές, τόσο ως προς τη διδασκαλία της, όσο και ως προς την κατανόησή της. Ο Κόμης 2005 [74] εντοπίζει τις δυσκολίες αυτές στο λογικό περιεχόμενο των συνθηκών, στις συμβολικές τους αναπαραστάσεις, τις σημασιολογικές και συντακτικές ιδιότητες της δομής επιλογής σε κάθε γλώσσα προγραμματισμού και στις αλληλεπιδράσεις με τις αναπαραστάσεις της ακολουθιακής μορφής της εκτέλεσης.

Δυσκολίες στην κατανόηση εμφανίζονται και σε άλλα σημεία, όπως είναι ο βαθμός εμφώλευσης [55], οι πολύπλοκες λογικές εκφράσεις που έχουν να κάνουν με τη συνθήκη ελέγχου [09] και ο εντοπισμός του χώρου εμβέλειας μετά το ΤΟΤΕ και το ΑΛΛΙΩΣ [58]. Η βασική δυσκολία στην κατανόηση της δομής ελέγχου έγκειται στο γεγονός ότι οι μαθητές έχουν μια ισχυρή αναπαράσταση για τη σειριακή εκτέλεση όλων των εντολών ενός προγράμματος [74]. Η Δομή Ελέγχου είναι η πρώτη περίπτωση που συναντούν οι μαθητές, όπου διακόπτεται αυτή η σειριακή ακολουθία και το τι θα εκτελεστεί δεν ταυτίζεται απόλυτα με την αλληλουχία των εντολών του προγράμματος, αλλά εξαρτάται από μια λογική συνθήκη που η τιμή της καθορίζεται δυναμικά κατά την εκτέλεση του προγράμματος και μπορεί να διαφέρει κάθε φορά (αληθής ή ψευδής).

Ο Δαγδιλέλης [70] αναφέρει ότι όταν δημιουργούνται συνδυασμοί προτάσεων με τους λογικούς τελεστές AND, OR, NOT, εμφανίζονται δυσκολίες κατανόησης, οι οποίες, όμως, είναι λιγότερες, εάν οι μαθητές έχουν ανεπτυγμένο μαθηματικό υπόβαθρο.

Ένα ακόμα σημείο δυσκολίας είναι στη λειτουργία των δομών επιλογής, όπως για παράδειγμα, στην περίπτωση της AN <συνθήκη>, όταν η συνθήκη είναι ψευδής, οι μαθητές συχνά νομίζουν ότι το πρόγραμμα τερματίζει και εμφανίζει μήνυμα λάθους. Εάν πάλι, η δομή επιλογής περιέχει και τμήμα AN και τμήμα ΑΛΛΙΩΣ, οι μαθητές περιμένουν να εκτελεστούν οι εντολές και των δυο τμημάτων [70].

Δομή επανάληψης

Η δομή επανάληψης χρησιμοποιείται για την επαναληπτική εκτέλεση μιας ακολουθίας εντολών (βρόχος). Η έξοδος από αυτή την επαναληπτική εκτέλεση ελέγχεται από μια συνθήκη. Ο αριθμός των επαναλήψεων, σύμφωνα με τον Soloway [60], μπορεί να είναι εκ των προτέρων γνωστός και καθορισμένος από εμάς ή να εξαρτάται από μια κατάσταση που να λαμβάνει χώρα κατά τη διάρκεια της επεξεργασίας.

Οι επαναληπτικές δομές παρουσιάζουν ιδιαίτερα διδακτικά προβλήματα [69] [72]. Οι Rogalski και Samurçay [55] διακρίνουν τρεις τύπους διεργασιών μεταβλητών σε μια δομή επανάληψης:

α) Την αρχικοποίηση, δηλαδή τον προσδιορισμό των αρχικών τιμών των μεταβλητών του βρόχου. Οι μαθητές δυσκολεύονται περισσότερο όταν πρόκειται για μεταβλητές μετρητές ή αθροιστές και απαιτείται να χρησιμοποιηθεί εντολή ανάθεσης τιμής.

β) Τη συνθήκη ελέγχου για τον τερματισμό του βρόχου, είτε περιλαμβάνει μια σταθερή τιμή, είτε προκύπτει από υπολογισμό εντός της επανάληψης.

γ) Την αναπροσαρμογή των τιμών των μεταβλητών, την οποία οι αρχάριοι συχνά παραλείπουν να προσθέσουν στον κώδικα του βρόχου.

Επίσης, έχουν παρατηρηθεί δυσκολίες στον καθορισμό του βήματος, της τιμής κατά την οποία αλλάζει η μεταβλητή μετρητής.

Εν γένει, παρατηρείται οι αρχάριοι να προτιμούν να ξαναγράψουν το ίδιο κομμάτι κώδικα, από το να χρησιμοποιήσουν ένα βρόχο [71].

Τα προβλήματα στην εκμάθηση του προγραμματισμού υπολογιστών είναι ποικίλα, τόσο γενικής φύσεως, αλλά και ειδικά σε ότι αφορά συγκεκριμένες προγραμματιστικές δομές. Συνδέονται με παρανοήσεις και άλλες δυσκολίες που δημιουργούν εμπόδια στους μαθητές, αλλά

και στους εκπαιδευτικούς, οι οποίοι συχνά ακολουθούν λάθος προσέγγιση. Λαμβάνοντας υπόψη τα αρνητικά της χρήσης γλωσσών γενικού σκοπού έναντι των εργαλείων οπτικού προγραμματισμού και τα πλεονεκτήματα της αξιοποίησης των δυνατοτήτων των φορητών συσκευών των μαθητών, αξίζει να εξετάσουμε κατά πόσο θα μπορούσαν όλα αυτά να συνδυαστούν για τη διευκόλυνση της διδασκαλίας και μάθησης βασικών προγραμματιστικών εννοιών.

Κεφάλαιο 5

Προγραμματισμός φορητών συσκευών

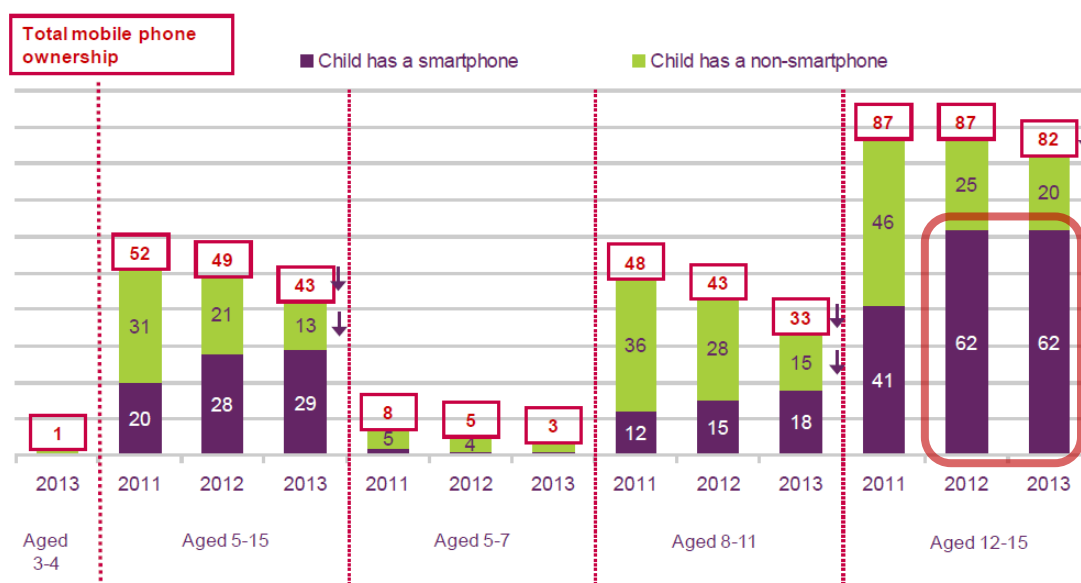
Στο κεφάλαιο αυτό, αρχικά γίνεται μια επισκόπηση των δημοφιλέστερων πλατφορμών για «έξυπνες» φορητές ψηφιακές συσκευές στην παγκόσμια και ελληνική αγορά. Στη συνέχεια εξετάζεται η προοπτική αξιοποίησης των συσκευών αυτών στο σχολείο και ειδικότερα στη διδασκαλία και μάθηση βασικών εννοιών προγραμματισμού. Γίνεται αναφορά στα βασικά προγραμματιστικά εργαλεία ανάπτυξης εφαρμογών για τις δυο δημοφιλέστερες πλατφόρμες και διερευνάται η καταλληλότητά τους για χρήση και αξιοποίησή τους από μαθητές και εκπαιδευτικούς σε σχολικό περιβάλλον.

5.1 Κατηγορίες φορητών συσκευών

Στις ψηφιακές φορητές συσκευές περιλαμβάνονται τα «έξυπνα» τηλέφωνα (smartphones) και οι ταμπλέτες (tablets), συσκευές οι οποίες έχουν διεισδύσει σε πολύ μεγάλο βαθμό στην παγκόσμια αγορά. Η μεγάλη αποδοχή και χρήση τους από τους νέους, μας οδήγησε να διερευνήσουμε τη δυνατότητα αξιοποίησής τους και για εκπαιδευτικούς σκοπούς, αλλά πριν από

αυτό, ως δούμε ορισμένα χρήσιμα στατιστικά στοιχεία, τα οποία θα μας κατευθύνουν στη συνέχεια.

Στην Αγγλία, ο κυβερνητικός φορέας Ofcom (Office of Communications) σε μελέτη ³¹ του (Οκτώβριος 2013) για τη χρήση ψηφιακών φορητών συσκευών από παιδιά και εφήβους την περίοδο 2007 - 2013 αναφέρει πολλά και ενδιαφέροντα στατιστικά ευρήματα και τάσεις.



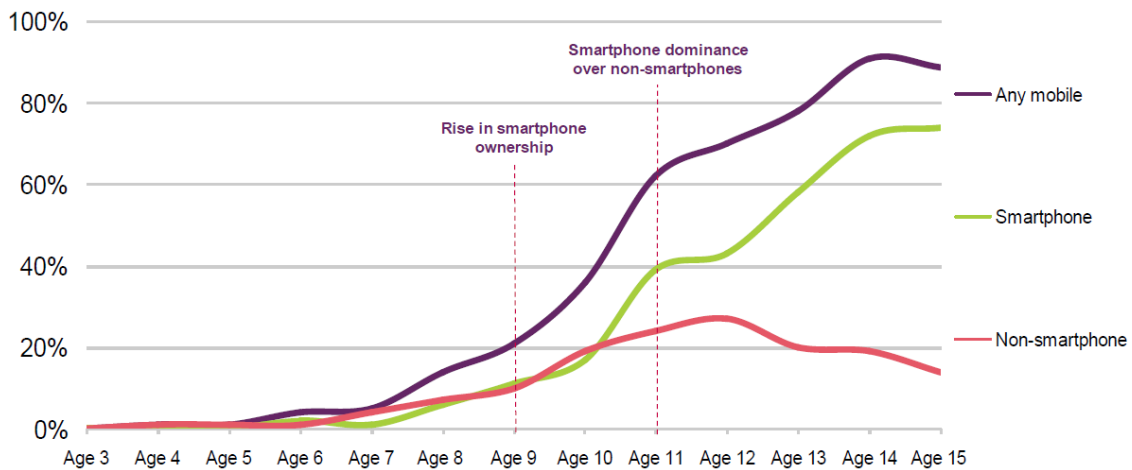
Εικόνα 5.1: Κατοχή κινητών τηλεφώνων ανά ηλικία για τα έτη 2011, 2012 και 2013).

Πηγή: Έρευνα του Ofcom, Ιούνιος 2013

Στην εικόνα 5.1, παρατηρούμε την αυξανόμενη χρήση smartphone σε όλες τις ηλικιακές ομάδες, η οποία φτάνει και στο 62%.

Ειδικότερα για το έτος 2013, τα αντίστοιχα ποσοστά φαίνονται στο επόμενο διάγραμμα (εικόνα 5.2), όπου φαίνεται η έντονη διείσδυση των έξυπνων κινητών τηλεφώνων στην καθημερινότητα των εφήβων.

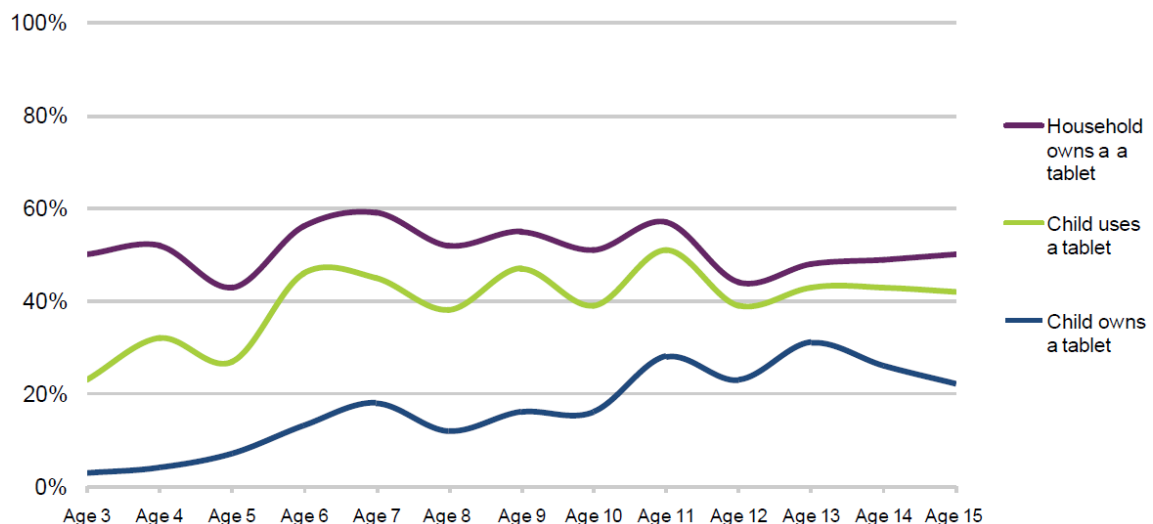
³¹ Ofcom "Children and Parents: Use and Attitudes Report" Research Document, 3 October 2013. <http://stakeholders.ofcom.org.uk/market-data-research/media-literacy-pubs/>



Εικόνα 5.2: Κατοχή smartphone, ανά ηλικία για το 2013.

Πηγή: Έρευνα του Ofcom, Απρ.-Ιουν. 2013

Παρομοίως, τα αντίστοιχα στοιχεία για την χρήση tablet δείχνουν υψηλά ποσοστά χρήσης, για όλες τις ηλικιακές ομάδες (εικόνα 5.3):



Εικόνα 5.3: Χρήση tablet ανά ηλικιακή ομάδα για το έτος 2013.

Πηγή: Έρευνα του Ofcom, Απρ.-Ιουν. 2013

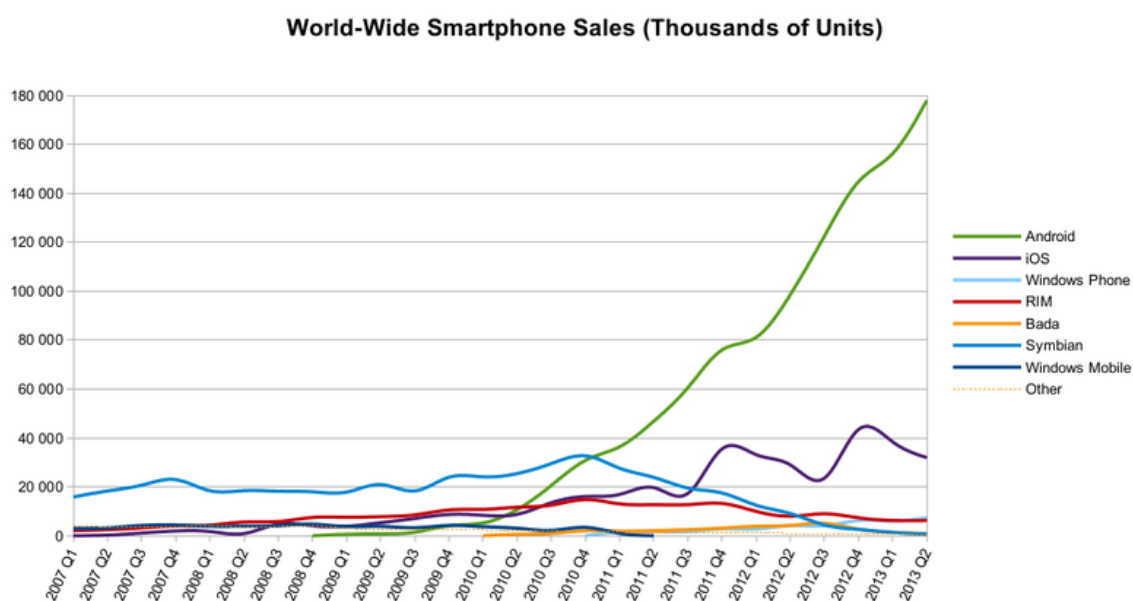
Τα αποτελέσματα του ερευνητικού προγράμματος “EU Kids Online II” για την Ελλάδα ³², το 2009, δείχνουν ότι, στα παιδιά ηλικίας 10-14 ετών, 43% των αγοριών και 51% των κοριτσιών κατέχουν κινητό τηλέφωνο, με ένα ποσοστό της τάξης του 10% να κατέχει 3G κινητό τηλέφωνο. Αντιστοίχως, στα παιδιά ηλικίας 15-18 ετών, 57% των αγοριών και 64% των κοριτσιών κατέχουν κινητό τηλέφωνο, με ένα ποσοστό της τάξης του 27% να είναι 3G κινητό τηλέφωνο.

³² Τσαλίκη Λ., Χρονάκη Δ., «Παιδιά και Διαδίκτυο στην Ελλάδα», GRKGO, Ιούνιος 2012

Ως προς το λειτουργικό σύστημα που χρησιμοποιείται στις κινητές συσκευές, τα δημοφιλέστερα είναι: α) το Android της Google και β) το iOS της Apple.

Οι μελέτες της Gartner ³³ και της IDC ³⁴ για τις παγκόσμιες πωλήσεις smartphone φανερώνουν την τεράστια διείσδυση του λειτουργικού συστήματος Android στην αγορά.

Στην εικόνα 5.4 φαίνεται η ανοδική πορεία των πωλήσεων κινητών τηλεφώνων με Android, οι οποίες έχουν ξεπεράσει τις 180.000.000 μονάδες. Παρατηρούμε επίσης ότι οι πωλήσεις συσκευών με iOS είναι σημαντικά χαμηλότερες, ενώ ακόμα χαμηλότερες είναι και οι πωλήσεις Windows Phone κινητών τηλεφώνων.



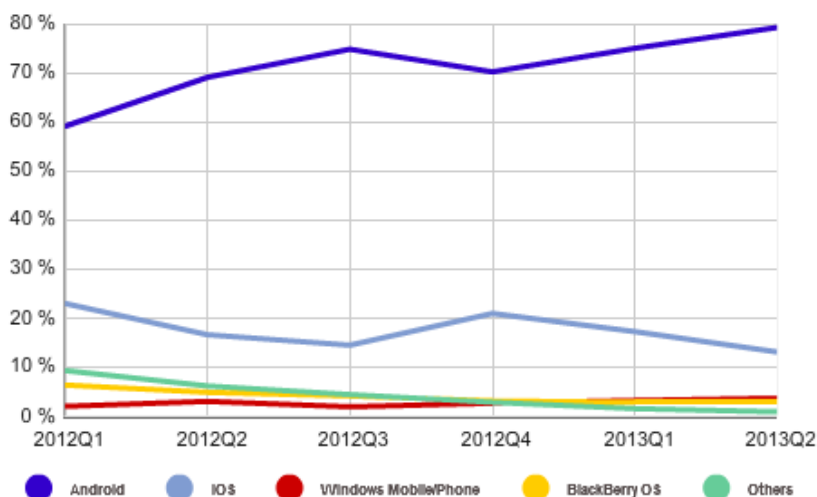
Εικόνα 5.4: Πωλήσεις smartphones σε παγκόσμιο επίπεδο (σε χιλιάδες).
Πηγή: Gartner Smart Phone Marketshare 2007 Q1 – 2013 Q2.

Σε αντίστοιχη έρευνα της αμερικάνικης εταιρείας IDC (International Data Corporation) φαίνεται το μερίδιο αγοράς των συσκευών με λειτουργικό σύστημα Android να αγγίζει το 80% (εικόνα 5.5).

³³ Gartner press release, Αύγουστος 2013: <http://www.gartner.com/newsroom/id/2573415>

³⁴ Top Smartphone Operating Systems, Shipments and Market Share, 2013 Q3. IDC Worldwide Mobile Phone Tracker, Aug. 2013. <http://goo.gl/1rMD4r>

Worldwide Smartphone Platforms Market Share, 2013 Q2 (Unit Share)



Εικόνα 5.5: Ποσοστιαίο μερίδιο αγοράς των λειτουργικών συστημάτων smartphone για τα έτη 2012-2013.

Πηγή: IDC Worldwide Mobile Phone Tracker, August 7, 2013

Από τα παραπάνω, είναι φανερό ότι, σήμερα, η πλατφόρμα με τη μεγαλύτερη δημοτικότητα παγκοσμίως είναι αυτή του Android του Google.

Ενεργοποιήσεις συσκευών με Android σε παγκόσμιο επίπεδο

Τον Ιούλιο του 2011, η Google δήλωσε ότι κατά μέσο όρο 550.000 συσκευές με Android ενεργοποιούνται καθημερινά και μάλιστα με αυξανόμενο ρυθμό της τάξης του 4.4% ανά εβδομάδα. Το Σεπτέμβριο του 2012 είχαν ενεργοποιηθεί συνολικά 500 εκ. συσκευές, με τις ημερήσιες ενεργοποιήσεις να είχαν αυξηθεί σε 1.3 εκ.. Το Μάιο του 2013, στη σύσκεψη Google I/O, ο επικεφαλής των Android και Chrome, Sundar Pichai, δήλωσε ότι στα τέσσερα χρόνια, από τη διάθεση του λειτουργικού τους συστήματος στο κοινό μέχρι σήμερα, έχουν γίνει 900 εκ. ενεργοποιήσεις, ενώ ο πρόεδρος Eric Schmidt προέβλεψε ότι θα ξεπεράσουν το δισεκατομμύριο μέχρι το τέλος του έτους ³⁵. Η πρόβλεψη αυτή φαίνεται ότι επιβεβαιώθηκε, με δήλωση του Vic Gundotra, αντιπροέδρου του τμήματος μηχανικής της Google, ήδη από τον Σεπτέμβριο του 2013 ³⁶.

Διαθέσιμες εφαρμογές για Android

³⁵ Δηλώσεις στο Google I/O για τις ενεργοποιήσεις συσκευών Android. The Verge, Μάιος 2013.

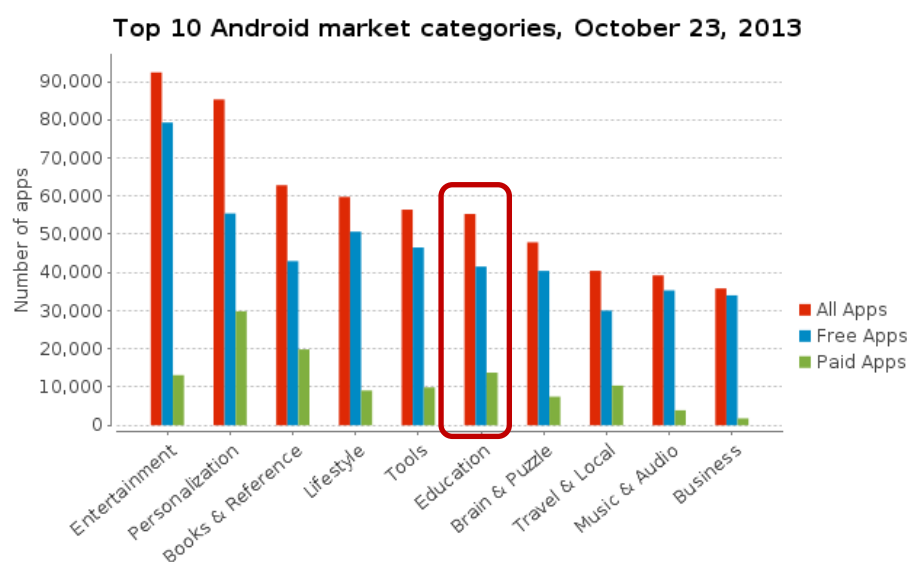
<http://www.theverge.com/2013/5/15/4333584/total-android-activations-900-million>

³⁶ Ανακοίνωση του Vic Gundotra για 1.000.000.000 ενεργοποιήσεις Android συσκευών:

<https://plus.google.com/+VicGundotra/posts/8CVI79nPQwN>

Η επίσημη πλατφόρμα διανομής εφαρμογών (mobile applications ή απλά “apps”) για το Android ονομάζεται Google Play (πρώην Android Market) και επιτρέπει στους χρήστες να περιηγηθούν και να κατεβάσουν δωρεάν ή επί πληρωμή εφαρμογές για τις συσκευές τους. Τον Ιούλιο του 2013, ξεπέρασε τις ένα εκατομμύριο διαθέσιμες εφαρμογές, με περισσότερες από 50 δισεκατομμύρια συνολικές λήψεις ³⁷.

Με βάση στατιστικά στοιχεία της εταιρείας Appbrain, ενδιάμεσου παρόχου εφαρμογών του «καταστήματος» Google Play, οι εκπαιδευτικές εφαρμογές στο Android ξεπερνάνε τις 55.000, όπως φαίνεται και στην παρακάτω εικόνα ³⁸. Από αυτές, το 80% είναι δωρεάν.



Εικόνα 5.6: Οι 10 δημοφιλέστερες κατηγορίες εφαρμογών στο Android (Οκτ. 2013).

Σημείωση: εμφανίζονται μόνο εφαρμογές με περισσότερες από 50.000 λήψεις.

Πηγή: <http://www.appbrain.com> (23 Οκτ. 2013).

Εξάλλου, η ομάδα διαχείρισης της πλατφόρμας Google Play έχει ξεκινήσει πιλοτικά ειδικό τομέα για εκπαιδευτικές εφαρμογές ³⁹, ώστε να διευκολύνει τόσο τους προγραμματιστές να προωθήσουν τα έργα τους, όσο και το εκπαιδευτικό προσωπικό και μαθητές να εντοπίζουν ευκολότερα το περιεχόμενο και τα εργαλεία που χρειάζονται. Η επίσημη έναρξη λειτουργίας του τομέα αναμένεται το φθινόπωρο του 2013 και αρχικά θα στοχεύει στην αγορά K-12 των Η.Π.Α (ηλικίες 6 μέχρι 18 ετών).

³⁷ Εφαρμογές στο Google Play: http://en.wikipedia.org/wiki/Google_Play#Applications

³⁸ Appbrain Android statistics: <http://www.appbrain.com/stats/android-market-app-categories>

³⁹ Google Play for education: <http://developer.android.com/distribute/googleplay/edu/about.html>

5.2 Έξυπνες φορητές συσκευές στην εκπαίδευση

Η NMC (The New Media Consortium)⁴⁰, μια διεθνής κοινότητα ειδικών στην εκπαιδευτική τεχνολογία, εκδίδει ετήσιες αναφορές ανά βαθμίδα εκπαίδευσης (NMC Horizon Reports - K-12 Editions), στις οποίες περιγράφονται οι νέες τεχνολογίες που αναμένεται να έχουν σημαντικό αντίκτυπο στη διδασκαλία, τη μάθηση και την έρευνα στην εκπαίδευση, για το τρέχον έτος, αλλά και για τα αμέσως επόμενα. Στις αναφορές αυτές [28][29][30], γίνεται ιδιαίτερη μνεία στη σημασία των φορητών συσκευών και στον ισχυρό ρόλο που μπορούν να διαδραματίσουν στο χώρο της εκπαίδευσης.

5.2.1 Ιδιαίτερα χαρακτηριστικά των ψηφιακών φορητών συσκευών

Πρόκειται για συσκευές, κατά κανόνα, με οθόνη αφής, κάμερα, δέκτη GPS, αισθητήρες, πολλαπλές δυνατότητες συνδεσιμότητας με άλλες συσκευές και δίκτυα, καθώς και λειτουργικό σύστημα που πέραν των εφαρμογών που ενσωματώνει, δέχεται την εγκατάσταση ενός μεγάλου αριθμού επιπλέον εφαρμογών (mobile applications ή απλά “apps”), κατά βούληση των χρηστών τους [28].

Ορισμένα χαρακτηριστικά, λοιπόν, των συσκευών αυτών, τους προσδίδουν ιδιότητες που μπορούν κάλλιστα να αξιοποιηθούν στην εκπαίδευση. Τα σημαντικότερα από αυτά τα χαρακτηριστικά [26] είναι τα παρακάτω:

α) φορητότητα: το μέγεθος των συσκευών ευνοεί την κινητικότητα των χρηστών τους.

β) υπολογιστική ισχύς: η αναλογικά αυξημένη υπολογιστική ισχύς επιτρέπει την εκτέλεση εργασιών σε ικανοποιητικό βαθμό και παρομοίως με τις μεγαλύτερες και λιγότερο φορητές συσκευές.

γ) συνδεσιμότητα: η συνεχής, σταθερή και υψηλής ταχύτητας σύνδεση στο διαδίκτυο, επιτρέπει την άμεση πρόσβαση σε πληροφορίες και την επικοινωνία, από οπουδήποτε.

Ο Kurkovsky [34] διαπιστώνει ότι οι νέοι σήμερα αδυνατούν να φανταστούν την καθημερινότητά τους, χωρίς να αφιερώνουν ένα σεβαστό χρονικό διάστημα στη χρήση του κινητού τηλεφώνου τους, είτε για επικοινωνία, είτε για πρόσβαση στο διαδίκτυο, είτε για

⁴⁰ The NMC Horizon Project and Reports: <http://www.nmc.org/horizon-project>

διασκέδαση. Οι περισσότεροι μαθητές, συνδέουν πλέον όλο και περισσότερο την ψηφιακή φορητή τους συσκευή με την έννοια της υπολογιστικής μηχανής, παρά με κάποιο επιτραπέζιο υπολογιστή, ο οποίος είναι, παραδοσιακά, το βασικότερο μέσο διδασκαλίας και μάθησης που χρησιμοποιείται στην πλειοψηφία των μαθημάτων πληροφορικής [37].

5.2.2 Η πρακτική «Φέρε τη δική σου συσκευή» (BYOD)

Στις αναφορές της NMC, γίνεται εκτενής αναφορά στον όρο BYOD (Bring Your Own Device) στο χώρο της εκπαίδευσης. Σύμφωνα με την πρακτική αυτή, κάθε μαθητής δύναται να χρησιμοποιεί στην τάξη την προσωπική του φορητή συσκευή (laptop, tablet, smartphone, κλπ.) [18].

Παράλληλα, σε πολλές ανεπτυγμένες χώρες, οι μαθητές φέρνουν στο σχολείο τις φορητές τους ηλεκτρονικές συσκευές, παρά το γεγονός ότι δεν επιτρέπεται η χρήση τους στην τάξη, όπως και στα ελληνικά σχολεία ⁴¹.

Η τάση αυτή, λοιπόν, εξακολουθεί να έρχεται σε αντίθεση με τους κανονισμούς λειτουργίας πολλών σχολικών μονάδων σε διάφορες χώρες ή να μην τυγχάνει ευρείας αποδοχής από τους εκπαιδευτικούς, κάτι το οποίο όμως, σταδιακά, φαίνεται να αλλάζει [18].

Εξάλλου, έρευνες [02][23][27] έδειξαν ότι όταν οι μαθητές χρησιμοποιούν τις προσωπικές τους συσκευές για μάθηση, παρατηρείται αύξηση της παραγωγικότητας και του αισθήματος δέσμευσης ως προς τις υποχρεώσεις τους, με ταυτόχρονη ενίσχυση της μαθησιακής εμπειρίας, τόσο στην τάξη, όσο και εκτός σχολείου.

5.3 Ανάπτυξη εφαρμογών για φορητές συσκευές

Ένα περιβάλλον ανάπτυξης εφαρμογών για κινητές συσκευές με προοπτικές να αξιοποιηθεί σε σχολικό περιβάλλον οφείλει να πληροί κάποιες προϋποθέσεις που να καθιστούν δυνατή και κατάλληλη τη χρήση του από όσο το δυνατόν περισσότερους μαθητές. Οφείλουμε να αναλογιστούμε το κόστος, τόσο του απαιτούμενου υλικού και λογισμικού για την κτήση και χρήση του, όσο και των ίδιων των συσκευών για τις οποίες αναπτύσσονται οι εφαρμογές.

⁴¹ Αποφάσεις Αρ.Πρωτ.100553/Γ2/04-09-2012/ΥΠΑΙΘΠΑ και Αρ.Πρωτ.132328/Γ2/07-12-2006/ΥΠΕΠΘ

Οφείλουμε να εξετάσουμε την καταλληλότητά του, από παιδαγωγικής σκοπιάς, τόσο ως προς τη δυνατότητα αξιοποίησης σύγχρονων διδακτικών προσεγγίσεων, όσο και ως προς τις ικανότητες και δεξιότητες που απαιτούνται από τους μαθητές.

Στη συνέχεια εξετάζουμε τις δυο δημοφιλέστερες πλατφόρμες έξυπνων τηλεφώνων και ταμπλετών, οι οποίες είναι το Android της Google και το iOS της Apple.

5.3.1 Το λειτουργικό σύστημα iOS της Apple

Το iOS ⁴² είναι ένα λειτουργικό σύστημα για ψηφιακές φορητές συσκευές, το οποίο έχει αναπτυχθεί και διανέμεται από την Apple Inc.. Πρωτοεμφανίστηκε το 2007 στο iPhone, ενώ πλέον υποστηρίζει κι άλλες συσκευές της εταιρίας, όπως το iPod touch και το iPad.

Αξίζει να σημειωθεί ότι η Apple δεν δίνει άδεια για εγκατάσταση του λογισμικού iOS σε συσκευές άλλων κατασκευαστών. Παράλληλα, η ανάπτυξη εφαρμογών με τα επίσημα προγραμματιστικά εργαλεία της εταιρίας απαιτεί ετήσια καταβολή συνδρομής (με εξαίρεση κάποια ιδρύματα τριτοβάθμιας εκπαίδευσης), καθώς και υπολογιστές Macintosh βασισμένους σε Intel επεξεργαστή, με λειτουργικό σύστημα Mac OS X Leopard ή νεότερο ⁴³, κάτι που δεν συνηθίζεται στα ελληνικά σχολικά εργαστήρια Η/Υ. Επίσης, οι τιμές των ανωτέρω ψηφιακών φορητών συσκευών, αλλά και υπολογιστών, είναι αναλογικά υψηλότερες από τις αντίστοιχων τεχνικών προδιαγραφών, συσκευές της πλατφόρμας Android. Αυτό οφείλεται σε μεγάλο βαθμό στην διαφορετική άδεια χρήσης του Android, που επιτρέπει την παραμετροποίηση και διανομή του λογισμικού από οποιονδήποτε ⁴⁴, με αποτέλεσμα να το προτιμούν και να το υιοθετούν όλο και περισσότεροι κατασκευαστές.

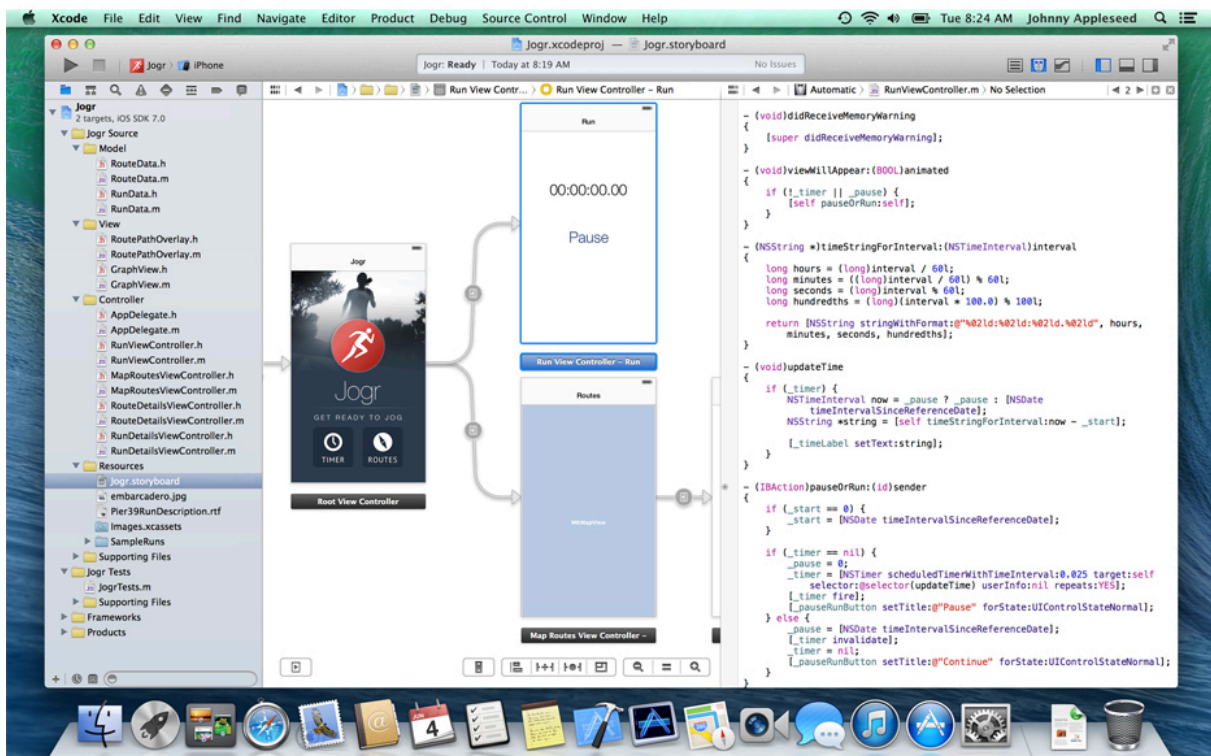
Το επίσημο περιβάλλον ανάπτυξης εφαρμογών για iOS και OS X ονομάζεται Xcode ⁴⁵. Αποτελεί ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) και περιέχει όλα τα απαραίτητα SDKs και εργαλεία ανάλυσης, καθώς και εξομοιωτή (βλ. εικόνα 5.7), ενώ η βασική γλώσσα προγραμματισμού είναι η Objective-C.

⁴² Επίσημος ιστότοπος του Apple iOS για την Ελλάδα: <http://www.apple.com/gr/ios/what-is/>

⁴³ Προγράμματα ανάπτυξης εφαρμογών για προγραμματιστές από την Apple: <https://developer.apple.com/programs/>

⁴⁴ Android Open Handset Alliance: http://www.openhandsetalliance.com/android_overview.html

⁴⁵ Xcode IDE από την Apple: <https://developer.apple.com/xcode/>



Εικόνα 5.7: Στιγμιότυπο της οθόνης του Apple Xcode.
Πηγή: <https://developer.apple.com/xcode/>

5.3.2 Το λειτουργικό σύστημα Android της Google

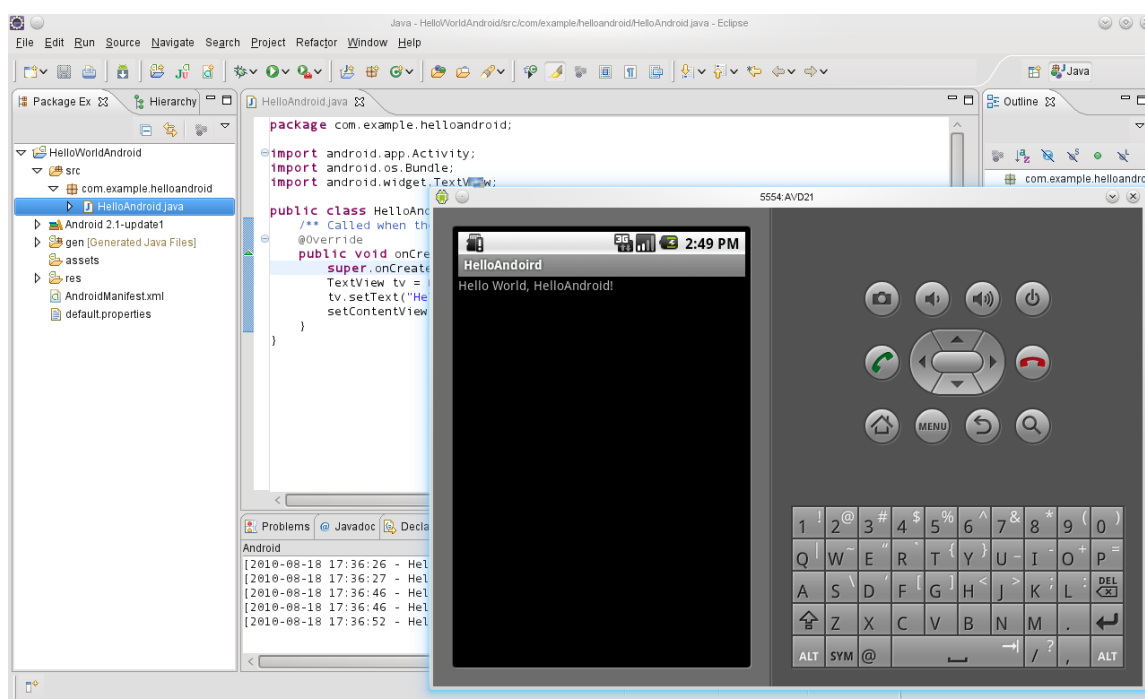
Το Android είναι ένα λειτουργικό σύστημα για ψηφιακές φορητές συσκευές, το οποίο αναπτύχθηκε αρχικά από την Google και στη συνέχεια από την Open Handset Alliance. Πρωτοπαρουσιάστηκε το 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance ¹², μιας κοινοπραξίας 48 εταιρειών τηλεπικοινωνιών, υλικού και λογισμικού, οι οποίες είναι αφιερωμένες στην ανάπτυξη και εξέλιξη ανοικτών προτύπων στις συσκευές κινητής τηλεφωνίας. Το μεγαλύτερο μέρος του κώδικα του Android διατίθεται υπό τους όρους της Apache License ⁴⁶, μιας ελεύθερης άδειας λογισμικού. Τρέχει τον πυρήνα του λειτουργικού συστήματος Linux, ενώ επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με τη χρήση της γλώσσας προγραμματισμού Java.

Η ανάπτυξη εφαρμογών γίνεται σε οποιοδήποτε σύγχρονο λειτουργικό σύστημα (ελάχιστη απαίτηση για Windows XP, Mac OS X 10.5.8 ή Linux kernel 2.6 και άνω) και χωρίς συγκεκριμένες απαιτήσεις σε προδιαγραφές υλικού [16]. Αυτό παρέχει μεγαλύτερη ευελιξία και διευκολύνει τόσο την εργασία από το σπίτι, όσο και στο σχολικό εργαστήριο Η/Υ.

⁴⁶ Άδεια ελεύθερου λογισμικού Apache License: <http://www.apache.org/licenses/LICENSE-2.0>

Συγκεκριμένα, το Android SDK παρέχει ελεύθερα και δωρεάν τα απαραίτητα εργαλεία ⁴⁷ και APIs (Application Programming Interfaces) για την ανάπτυξη εφαρμογών, με την αξιοποίηση ενός plugin της Google (ADT: Android Developer Tools) για το εργαλείο Eclipse για Java. Το Eclipse είναι το συνιστώμενο και ιδιαίτερος δημοφιλές ολοκληρωμένο περιβάλλον ανάπτυξης (IDE: Integrated Development Environment) που παρέχει πολλές διευκολύνσεις στον προγραμματιστή, ενώ στην επίσημη ιστοσελίδα του Android παρέχεται ένα πακέτο λογισμικού με την ονομασία “Android SDK ADT Bundle” που παρέχει όλα τα απαραίτητα εργαλεία ανάπτυξης, εξομοιωτή, καθώς και την τελευταία έκδοση της πλατφόρμας Android ⁴⁸.

Στην παρακάτω εικόνα φαίνεται ένα στιγμιότυπο από το περιβάλλον του Eclipse.



Εικόνα 5.8: Στιγμιότυπο της κεντρικής οθόνης του Eclipse IDE με ενεργό τον εξομοιωτή.

Τα παραπάνω, απαιτούν από τους χρήστες, γνώση της γλώσσας προγραμματισμού Java, εξοικείωση με το περιβάλλον Eclipse και γνώση των κανόνων σύνταξης αρχείων XML.

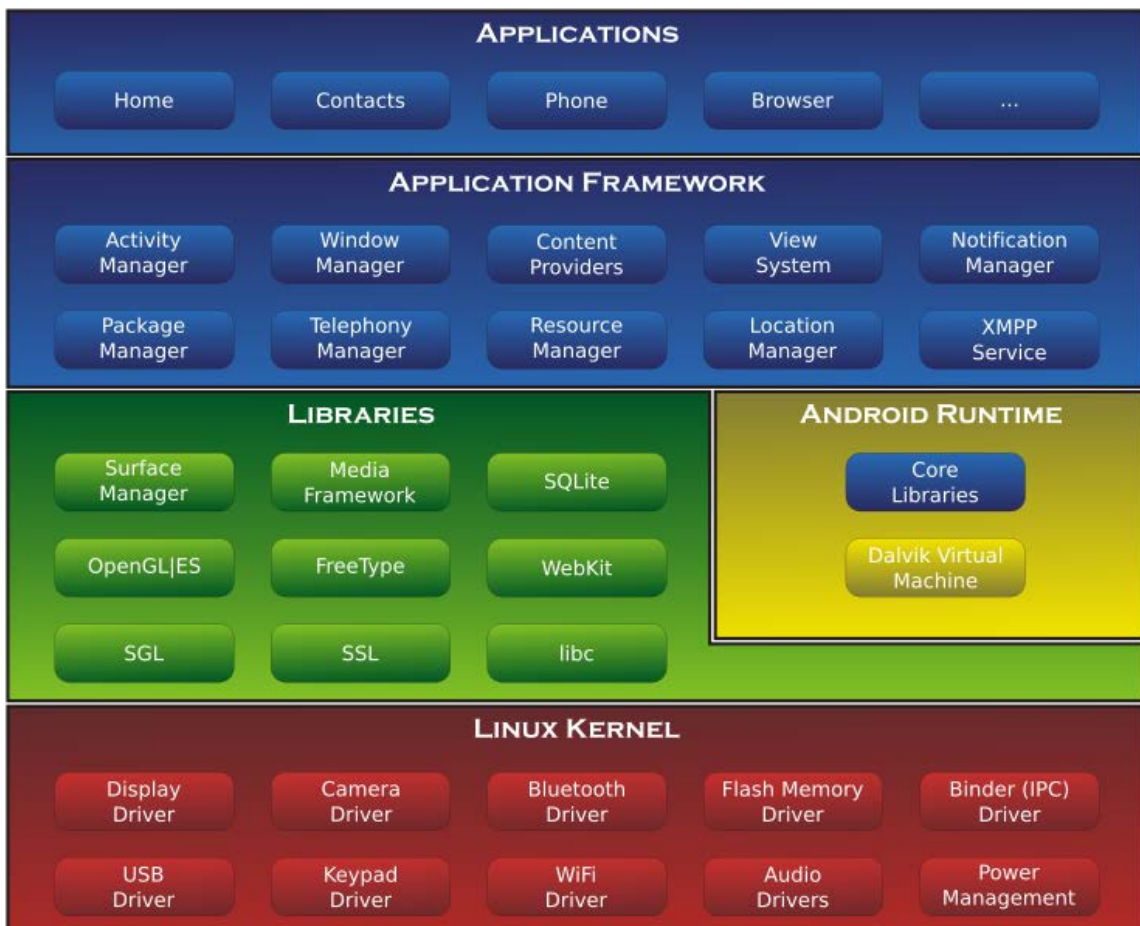
5.3.3 Αρχιτεκτονική του Android

⁴⁷ Android Developer Tools: <http://developer.android.com/tools/index.html>

⁴⁸ Android SDK ADT Bundle download: <http://developer.android.com/sdk/index.html>

Σκοπός του Android, ως λειτουργικού συστήματος, είναι η παροχή επιπέδων αφαιρετικότητας ανάμεσα στο υλικό και στον χρήστη. Μπορούμε λοιπόν να θεωρήσουμε το Android ως στοίβα ομάδων λογισμικού σε διάφορα επίπεδα, όπου κάθε επίπεδο έχει συγκεκριμένους ρόλους και παρέχει διαφορετικές υπηρεσίες στο ανώτερο του επίπεδο.

Αυτή η αρχιτεκτονική φαίνεται στο σχήμα (εικόνα 5.9) που ακολουθεί.



Εικόνα 5.9: Η αρχιτεκτονική του Android.

Στο κατώτερο επίπεδο βρίσκεται ο **πυρήνας** του Linux, με μικρές αλλαγές στην αρχιτεκτονική του από την Google. Αποτελεί τη βάση για το Android και περιλαμβάνει όλους τους απαραίτητους οδηγούς συσκευών (drivers). Οι drivers είναι προγράμματα που είναι υπεύθυνα για τον έλεγχο και την επικοινωνία με το υλικό. Για παράδειγμα, απαιτείται WiFi Driver για την αλληλεπίδραση με την κάρτα ασύρματου δικτύου, Bluetooth Driver για τον πομποδέκτη Bluetooth, κ.ο.κ..

Ο πυρήνας λειτουργεί επίσης ως ενδιάμεσο και αφαιρετικό επίπεδο μεταξύ του υλικού και των ανώτερων επιπέδων λογισμικού. Το Android χρησιμοποιεί τον πυρήνα για όλες τις καίριες λειτουργίες του, όπως διαχείριση μνήμης, διαχείριση διεργασιών, δικτύωση, ρυθμίσεις ασφάλειας, κ.α..

Ο πυρήνας του Linux είναι συμβατός με μεγάλη ποικιλία αρχιτεκτονικών σε επίπεδο υλικού, οπότε εξασφαλίζεται η υποστήριξη και συμβατότητα με μια ευρεία γκάμα συσκευών, από κινητά τηλέφωνα μέχρι φωτογραφικές μηχανές και τηλεοράσεις. Το Android λοιπόν έχει «χτιστεί» στα γερά θεμέλια του καταξιωμένου και δημοφιλούς πυρήνα του Linux, οπότε η προσαρμογή του για χρήση σε μια ποικιλία υλικού είναι σχετικά απλή διαδικασία.

Στο επόμενο επίπεδο βρίσκονται οι ενσωματωμένες **βιβλιοθήκες** (native libraries) του Android, οι οποίες είναι γραμμένες κυρίως σε γλώσσα C και C++. Το επίπεδο αυτό είναι υπεύθυνο για το χειρισμό των διαφόρων τύπων δεδομένων. Υπάρχουν υλοποιήσεις βιβλιοθηκών για διαχείριση γραφικών, για συστήματα βάσεων δεδομένων, για πρωτόκολλα ασφάλειας, για κωδικοποίηση και αποκωδικοποίηση βίντεο, κ.α..

Η πρόσβαση στις βιβλιοθήκες αυτές και η εκτέλεση των εφαρμογών που τις χρησιμοποιούν, επιτυγχάνεται μέσω του μηχανισμού **Android Runtime**. Το Android Runtime περιλαμβάνει την εικονική μηχανή Dalvic (VM: Virtual Machine) που διερμηνεύει τις Android εφαρμογές και τις βασικές βιβλιοθήκες υψηλότερου επιπέδου του πυρήνα της Java.

Αναλυτικότερα, η εικονική μηχανή Dalvic VM⁴⁹ χρησιμοποιείται για την εκτέλεση των εφαρμογών. Οι εφαρμογές είναι γραμμένες σε Java, οπότε απαιτούν την ύπαρξη ενός διερμηνευτή Java για την εκτέλεσή τους. Στο Android δεν έχει χρησιμοποιηθεί η τυπική Java, αλλά μια ειδική έκδοση, βελτιστοποιημένη για περιβάλλοντα χαμηλής υπολογιστικής ισχύος και μνήμης. Σε αντίθεση με τη Java VM, η Dalvic δεν «τρέχει» .class αλλά .dex αρχεία (dalvic executable format), τα οποία παράγονται κατά τη διάρκεια μεταγλώττισης και είναι καλύτερα από την άποψη των απαιτούμενων πόρων. Επιτρέπει επίσης την ύπαρξη πολλαπλών στιγμιότυπων εικονικών μηχανών για υποστήριξη πολυνηματικότητας, δυνατότητας απομόνωσης, καλύτερης διαχείρισης μνήμης και μεγαλύτερης ασφάλειας. Τα αρχεία εγκατάστασης των εφαρμογών έχουν την επέκταση .apk και περιέχουν τα αρχεία .dex για

⁴⁹ Η Dalvic Virtual Machine είναι ελεύθερο και ανοικτού κώδικα λογισμικό. Δημιουργήθηκε από τον Dan Bornstein και πήρε το όνομά της από το ομώνυμο χωριό της Ισλανδίας.
<http://android-developers.blogspot.gr/2010/05/dalvik-jit.html>

εκτέλεση από τον διερμηνευτή της Dalvik, καθώς και διάφορα συνοδευτικά αρχεία (εικόνες, ήχους, κλπ.) που χρειάζεται η εκάστοτε εφαρμογή.

Στο αμέσως επόμενο επίπεδο (**Application Framework**) βρίσκεται το Android SDK που περιέχει όλες τις απαραίτητες βιβλιοθήκες για τη συγγραφή εφαρμογών. Παρέχει μια πληθώρα από APIs που δίνουν τη δυνατότητα στον προγραμματιστή να αξιοποιήσει δυνατότητες όπως δημιουργία διεπαφών χρήστη, πρόσβαση σε αποθηκευτικά μέσα της συσκευής, συνδεσιμότητα στο διαδίκτυο, κ.α. Για κάθε υπηρεσία που προσφέρει το επίπεδο αυτό, υπάρχουν διαχειριστές που καλούνται για την παροχή της στις εφαρμογές, κάτι που επιτρέπει τον καθορισμό δικαιωμάτων ή περιορισμών των λειτουργιών ανά εφαρμογή.

Το ανώτερο επίπεδο, είναι το **επίπεδο εφαρμογών** του Android. Όλες οι υπηρεσίες που έχουν αναφερθεί στα κατώτερα επίπεδα διατίθενται στο επίπεδο εφαρμογών και είναι στην ευχέρεια του προγραμματιστή το αν και πως θα τις χρησιμοποιήσει, μέσα από το Android SDK. Στο επίπεδο αυτό περιλαμβάνονται και οι προεγκατεστημένες εφαρμογές του Android, όπως οι εφαρμογές κλήσεων, διαχείρισης επαφών και μηνυμάτων, κ.α..

Αξίζει να σημειωθεί πως πολλοί κατασκευαστές παραμετροποιούν το περιβάλλον διεπαφής ή το βελτιστοποιούν με βάση τα ιδιαίτερα χαρακτηριστικά των συσκευών τους και συχνά προσθέτουν δικές τους εφαρμογές (βλ. Samsung TouchWiz⁵⁰, HTC Sense⁵¹, κ.α.). Η «βάση» του Android λοιπόν διατίθεται ελεύθερα προς αξιοποίηση από κατασκευαστές συσκευών και προγραμματιστές ανά τον κόσμο, και μόνο αν επιθυμούν τα προϊόντα τους να φέρουν και την πιστοποίηση της Google, τίθενται από την ίδια, κάποιιοι περιορισμοί⁵².

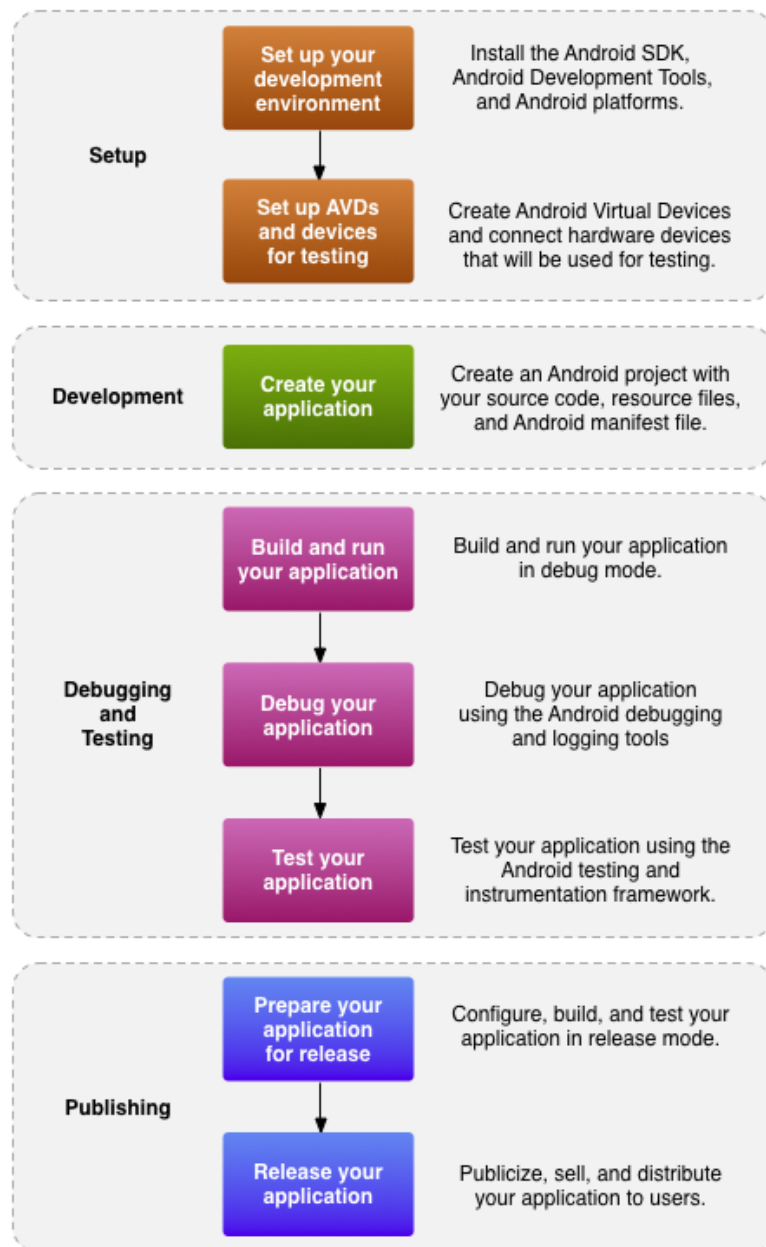
5.3.4 Η διαδικασία ανάπτυξης εφαρμογών στο Android

Η ανάπτυξη εφαρμογών από τους χρήστες λαμβάνει χώρα και αφορά στο επίπεδο εφαρμογών του Android, αξιοποιώντας τις υπηρεσίες που του προσφέρουν τα κατώτερα επίπεδα.

⁵⁰ Samsung TouchWiz: <http://en.wikipedia.org/wiki/TouchWiz>

⁵¹ HTC Sense: http://en.wikipedia.org/wiki/HTC_Sense

⁵² Android Open Source FAQ: <http://source.android.com/faqs.html>



Εικόνα 5.10: Η πορεία ανάπτυξης εφαρμογών στο Android.

Πηγή: <http://developer.android.com/tools/workflow/index.html>

Τα βασικά βήματα για την ανάπτυξη εφαρμογών, με βάση το επίσημο διάγραμμα από τον ιστότοπο του Android, όπως φαίνεται και στην παραπάνω εικόνα (5.10) είναι τα παρακάτω τέσσερα:

- Εγκατάσταση των απαραίτητων εργαλείων, δημιουργία εικονικών συσκευών (AVD: Android Virtual Devices) και πραγματοποίηση σύνδεσης των πραγματικών συσκευών στις οποίες θα εγκατασταθούν οι παραγόμενες εφαρμογές.

- Ανάπτυξη του project, που περιλαμβάνει τον πηγαίο κώδικα και τα σχετικά αρχεία πόρων που περιλαμβάνονται στην εφαρμογή.
- Έλεγχος σφαλμάτων και καλής λειτουργίας του παραγόμενου αρχείου .apk, το οποίο εγκαθίσταται στον εξομοιωτή ή στην πραγματική συσκευή. Το ίδιο το Android SDK διαθέτει εργαλεία εκσφαλμάτωσης και καταγραφής συμβάντων.
- Τελική παραμετροποίηση και δημοσίευση της εφαρμογής, με συνηθέστερη πρακτική τη διάθεσή της στο κοινό μέσω της πλατφόρμας Google Play. Σημειώνεται ότι στην παρούσα φάση, απαιτείται εγγραφή των χρηστών ως developers με καταβολή ποσού εγγραφής της τάξης των \$25 ⁵³.

5.3.5 Ανάπτυξη εφαρμογών που λειτουργούν και στις δυο πλατφόρμες

Θα ήταν ιδανικό να μπορούσαμε να αναπτύσσουμε εφαρμογές που να λειτουργούν σε όλες τις πλατφόρμες και να μην εξαρτώνται από τη βασική γλώσσα της κάθε πλατφόρμας, βλ. Objective-C για το iOS και Java για το Android. Η διαλειτουργικότητα αυτή, είναι δυνατό να επιτευχθεί, αν δεν προγραμματίσουμε στην native γλώσσα κάθε πλατφόρμας, αλλά αντί αυτού, δημιουργήσουμε web εφαρμογές (web apps) [59].

Στα πλαίσια αυτά έχουν δημιουργηθεί διάφορα εργαλεία ⁵⁴, αξίζει όμως να κάνουμε ιδιαίτερη αναφορά στο PhoneGap. Το PhoneGap ⁵⁵ αποτελεί ένα ανοικτού κώδικα framework λογισμικού που έχει σχεδιαστεί έτσι ώστε να υποστηρίζει την ανάπτυξη «υβριδικών» εφαρμογών. Με τον όρο «υβριδικές» εννοούμε ότι οι παραγόμενες εφαρμογές δεν είναι ούτε καθαρά βασισμένες στον ιστό (web-based), ούτε και εξ' ολοκλήρου native, καθώς έχουν πρόσβαση στα native APIs κάθε πλατφόρμας, ενώ διανέμονται και ως πακέτα, ξεχωριστά για κάθε πλατφόρμα, αλλά η εκτέλεση και η διεπαφή προς τους χρήστες γίνεται τοπικά σε επίπεδο web. Το πλήρες σετ που αποτελεί τη βάση λειτουργίας του PhoneGap είναι το Apache Cordova ⁵⁶, ενώ υπάρχει και υποστήριξη χρήσης του και μέσα από εργαλεία όπως το Adobe DreamWeaver ⁵⁷.

⁵³ Εγγραφή ως Google Play Developer (Οκτώβριος 2013): <https://play.google.com/apps/publish/signup/>

⁵⁴ Multiple phone web-based application frameworks: http://en.wikipedia.org/wiki/Multiple_phone_web-based_application_framework

⁵⁵ PhoneGap: <http://phonegap.com/about/feature/>

⁵⁶ Apache Cordova: <http://cordova.apache.org/>

⁵⁷ Adobe Dreamweaver CS6 supports PhoneGap Build: <http://phonegap.com/2012/04/24/adobe-dreamweaver-cs6-supports-phonegap-build/>

Οι προγραμματιστές που επιθυμούν να αναπτύξουν εφαρμογές στο PhoneGap χρειάζονται γνώσεις τεχνολογιών Web και συγκεκριμένα HTML5, CSS3 και JavaScript. Το γεγονός αυτό, ναί μεν καθιστά τη δημιουργία εφαρμογών ανεξάρτητη πλατφόρμας, αλλά το ερώτημα της καταλληλότητας τέτοιων λύσεων για χρήση από μαθητές, αρχάριους στον προγραμματισμό, παραμένει.

5.3.6 Προβληματισμοί - περιορισμοί για τον προγραμματισμό σε περιβάλλον Android

Όπως παρατηρεί και ο Tigrek [61], το βασικό πρόβλημα της εκμάθησης προγραμματισμού Android εφαρμογών είναι ο απαιτούμενος χρόνος για να φτάσει κανείς σε ικανοποιητικό επίπεδο ευχέρειας δημιουργίας λειτουργικών εφαρμογών. Στην τριτοβάθμια εκπαίδευση, συνήθως ακολουθείται η διαδικασία της μάθησης σε δυο στάδια, που διαρκεί δυο ακαδημαϊκά εξάμηνα. Το πρώτο στάδιο αφιερώνεται στην εκμάθηση της γλώσσας Java και ακολουθεί το επόμενο στάδιο για τον προγραμματισμό εφαρμογών για κινητές συσκευές.

Εξάλλου, όπως διαπιστώνουν και οι Κόμης και Τζιμογιάννης [75] τα συνήθη προγραμματιστικά περιβάλλοντα και οι γλώσσες προγραμματισμού που χρησιμοποιούνται για εκπαιδευτικούς σκοπούς έχουν σχεδιαστεί για την ανάπτυξη εφαρμογών και όχι για τη διδασκαλία του προγραμματισμού. Είναι, συνεπώς, προσαρμοσμένα στο πλαίσιο γνώσεων και δεξιοτήτων των έμπειρων προγραμματιστών, γεγονός που ενισχύει τις δυσκολίες και τα εμπόδια που συναντούν οι μαθητές και οι αρχάριοι στον προγραμματισμό.

Έχουμε ήδη αναφερθεί (κεφάλαιο 2.3) στα πλεονεκτήματα της αξιοποίησης εργαλείων οπτικού προγραμματισμού, έναντι της χρήσης γλωσσών γενικού σκοπού, για την εισαγωγή των παιδιών σε προγραμματιστικές έννοιες.

Έχουμε επίσης καταλήξει στην πεποίθηση ότι η πλατφόρμα του Android βρίσκεται πιο κοντά στα δεδομένα της ελληνικής δευτεροβάθμιας εκπαίδευσης, τόσο από την πλευρά των απαιτήσεων σε υλικό-λογισμικό και του κόστους, όσο και από την μεριά της δημοτικότητάς της, όπως δείχνει το μερίδιο που κατέχει στην ευρωπαϊκή και ελληνική αγορά.

Στο επόμενο κεφάλαιο, λοιπόν, παρουσιάζουμε ένα προγραμματιστικό εργαλείο για ανάπτυξη εφαρμογών σε Android που τυγχάνει ευρείας υποδοχής από εκπαιδευτικούς και μαθητές όλων των βαθμίδων εκπαίδευσης στο εξωτερικό, για την ανάπτυξη υπολογιστικής σκέψης από τους μαθητές και τη διδασκαλία και μάθηση βασικών εννοιών προγραμματισμού, αλλά και για την αξιοποίησή του σε άλλα επιστημονικά πεδία και για άλλους σκοπούς. Πρόκειται για το MIT App Inventor, το οποίο διατηρεί τα πλεονεκτήματα της χρήσης της πλατφόρμας Android και περιορίζει τα αρνητικά της, συνδυάζοντας ορισμένα ιδιαίτερος κρίσιμα χαρακτηριστικά για την επιτυχή χρήση του στα σχολεία, τα οποία παρουσιάζονται και στη συνέχεια.

Κεφάλαιο 6

Η περίπτωση του App Inventor

Στο κεφάλαιο αυτό παρουσιάζεται συνοπτικά το εργαλείο App Inventor (AI), τα βασικά χαρακτηριστικά του και το περιβάλλον λειτουργίας του. Το AI είναι ένα Ελεύθερο Λογισμικό Ανοικτού Κώδικα (ΕΛ/ΛΑΚ) εργαλείο οπτικού προγραμματισμού, το οποίο λειτουργεί ως cloud-based υπηρεσία και παρέχει τη δυνατότητα ανάπτυξης εφαρμογών για συσκευές Android, ακόμα και από αρχάριους στον προγραμματισμό. Μοιάζει σε αρκετά σημεία με το Scratch, όπου γίνεται ιδιαίτερη αναφορά, και το κεφάλαιο ολοκληρώνεται με τους λόγους που θεωρούμε πως το καθιστούν ένα χρήσιμο εργαλείο για την εκπαιδευτική αξιοποίησή του.

6.1 Εισαγωγή

Τον Ιούλιο του 2010, η Google, σε συνεργασία με μια ομάδα ερευνητών από το Πανεπιστήμιο Massachusetts Institute of Technology (MIT), με επικεφαλής τον Hal Abelson, διέθεσε στο κοινό το εργαλείο Google App Inventor, ώστε ο οποιοσδήποτε να μπορεί να δημιουργήσει εφαρμογές για τηλέφωνα Android, ακόμα κι αν δεν διαθέτει ιδιαίτερες γνώσεις προγραμματισμού.

Στα μέσα του 2011, όμως, η Google αποφάσισε τον τερματισμό ορισμένων πρωτότυπων πειραματικών δράσεων της (Google Labs), μεταξύ των οποίων και το App Inventor. Έτσι, επισήμως από το Μάρτιο του 2012 μέχρι και σήμερα, την ευθύνη για τη συντήρηση και ανάπτυξη του AI, έχει αναλάβει το Κέντρο Κινητής Μάθησης του MIT ⁵⁸, το οποίο διαθέτει το AI ως ελεύθερο και ανοικτό λογισμικό.

Μάλιστα, το Δεκέμβριο του 2013, ταυτόχρονα με την έναρξη της παγκόσμιας καμπάνιας της «Ωρας Κώδικα» (Hour Of Code), το MIT ανακοίνωσε τη διάθεση της νέας έκδοσης (App Inventor 2), η οποία έχει τις παρακάτω κύριες διαφορές σε σχέση με την προηγούμενη έκδοση (App Inventor Classic):

- Το περιβάλλον συγγραφής προγραμμάτων (Blocks Editor) δεν βασίζεται πλέον στη Java βιβλιοθήκη Open Blocks, αλλά στο Blockly ⁵⁹. Πιο συγκεκριμένα, αρχικά τα πλακίδια του AI βασίστηκαν σε εργασία της Ricarose Roque, η οποία ανέπτυξε μια γενικευμένη προγραμματιστική διεπαφή (interface) της StarLogo ⁶⁰, η οποία απαιτούσε την ύπαρξη της Java στον υπολογιστή. Απεναντίας, το Blockly αποτελεί ένα οπτικό περιβάλλον προγραμματισμού, το οποίο εκτελείται εξ ολοκλήρου στο πρόγραμμα περιήγησης στον Παγκόσμιο Ιστό (browser) του χρήστη.
- Προστέθηκε η δυνατότητα παραμετροποίησης ορισμένων πλακιδίων, ώστε να επεκτείνεται, να περιορίζεται ή να τροποποιείται το σχήμα και η λειτουργικότητά τους.
- Οι μεταβλητές διαχωρίστηκαν σε τοπικές (local) και καθολικές (global), ώστε να παρέχεται μεγαλύτερη ευελιξία στον προγραμματιστή.

6.2 Απαιτήσεις

⁵⁸ MIT Center for Mobile Learning @ The Media Lab: <http://mitmobilelearning.org/>

⁵⁹ Blockly. A visual programming editor: <http://code.google.com/p/blockly/>

⁶⁰ The MIT roots of Google App Inventor, MITnews, Aug. 2010: <http://web.mit.edu/newsoffice/2010/android-abelson-0819.html>

Το εργαλείο σχεδιασμού και ανάπτυξης εφαρμογών εκτελείται διαδικτυακά από το νέφος (cloud) και δεν απαιτείται η εγκατάσταση επιπλέον λογισμικού για τη χρήση του. Οι ελάχιστες απαιτήσεις συστήματος είναι:

Εφαρμογή περιήγησης στον Ιστό (web browser):

- Mozilla Firefox 3.6 ή νεότερος.
- Apple Safari 5.0 ή νεότερος.
- Google Chrome 4.0 ή νεότερος.

Ο Microsoft Internet Explorer δεν υποστηρίζεται προς το παρόν (Ιανουάριος 2014).

Λειτουργικό Σύστημα (Operating system):

- Macintosh (με επεξεργαστή Intel): Mac OS X 10.5 ή νεότερο.
- Windows: Windows XP, Windows Vista, Windows 7 ή Windows 8.
- GNU/Linux: Ubuntu 8 ή νεότερο, Debian 5 ή νεότερο.

Τηλέφωνο (smartphone) ή ταμπλέτα (tablet) ή προσομοιωτής (emulator):

- Λειτουργικό σύστημα Android έκδοσης 2.3 ("Gingerbread") ή νεότερης.

Για την πρόσβαση στο περιβάλλον δημιουργίας εφαρμογών και για την αποθήκευση των παραγόμενων έργων στο νέφος, απαιτείται αυθεντικοποίηση του χρήστη μέσω λογαριασμού Google (Google account).

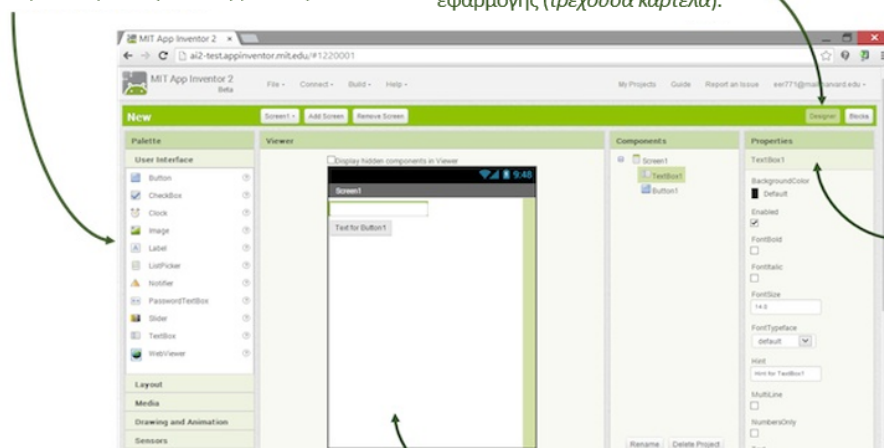
6.3 Το περιβάλλον δημιουργίας εφαρμογών

Η σχεδίαση και ανάπτυξη εφαρμογών στο περιβάλλον του AI, γίνεται εξ ολοκλήρου μέσα από κάποιο σύγχρονο πρόγραμμα πλοήγησης στον Παγκόσμιο Ιστό. Το περιβάλλον αυτό, όπως εμφανίζεται στο πρόγραμμα περιήγησης, χωρίζεται σε δυο διακριτά και αλληλένδετα μέρη: την καρτέλα σχεδίασης και την καρτέλα συγγραφής των εντολών.

Στην καρτέλα σχεδίασης (Design Editor), ο χρήστης σχεδιάζει τη διεπαφή (interface) της εφαρμογής, τοποθετώντας τα απαραίτητα συστατικά (components), στις θέσεις που επιθυμεί και καθορίζοντας τις βασικές τους ιδιότητες.

Παλέτα (Palette): Εντοπίζουμε το συστατικό (component) που θέλουμε και το προσθέτουμε στην εφαρμογή μας, σέρνοντάς το στην Οθόνη (Viewer).

Κουμπί Designer: Πατώντας το, μεταβαίνουμε στην καρτέλα σχεδίασης της οθόνης της εφαρμογής (τρέχουσα καρτέλα).



Ιδιότητες (Properties): Επιλέγουμε συστατικό από τη λίστα (Components List) και καθορίζουμε/τροποποιούμε τις ιδιότητές του (λ.χ. χρώμα, μέγεθος, κ.α.).

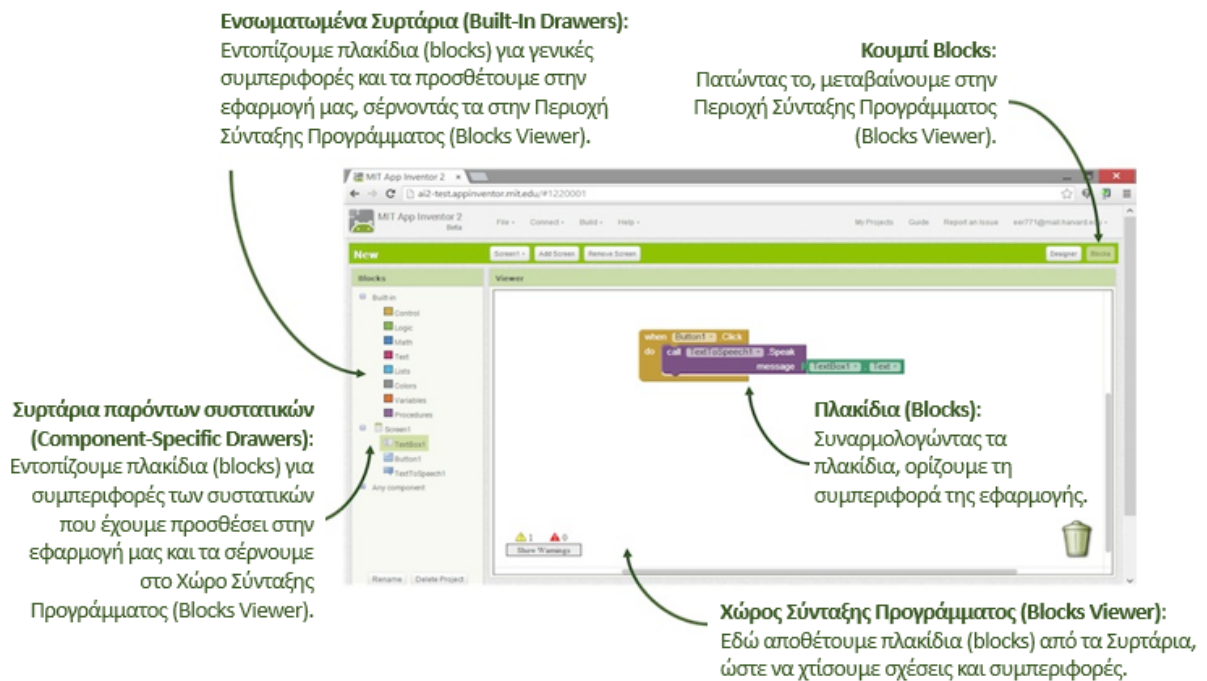
Οθόνη (Viewer): Στην περιοχή της οθόνης αποθέτουμε τα συστατικά από την Παλέτα και σχεδιάζουμε τη διεπαφή της εφαρμογής.

Εικόνα 6.1: Η καρτέλα σχεδίασης (Designer Editor) του AI.

Προσαρμογή από: <http://appinventor.mit.edu/explore/designer-blocks.html>

Εδώ, προστίθενται και συστατικά που δεν είναι ορατά στην οθόνη των αναπτυσσόμενων εφαρμογών, αλλά προσθέτουν λειτουργικότητα σε αυτές, όπως π.χ. πρόσβασης στην κάμερα, αποστολής και λήψης SMS και αξιοποίησης των διάφορων αισθητήρων της συσκευής.

Στην καρτέλα επεξεργασίας των πλακιδίων (Blocks Editor), ο χρήστης προγραμματίζει τη συμπεριφορά των αντικειμένων της εφαρμογής, συναρμολογώντας πλακίδια.



Εικόνα 6.2: Η καρτέλα επεξεργασίας πλακιδίων (Blocks Editor) του AI.

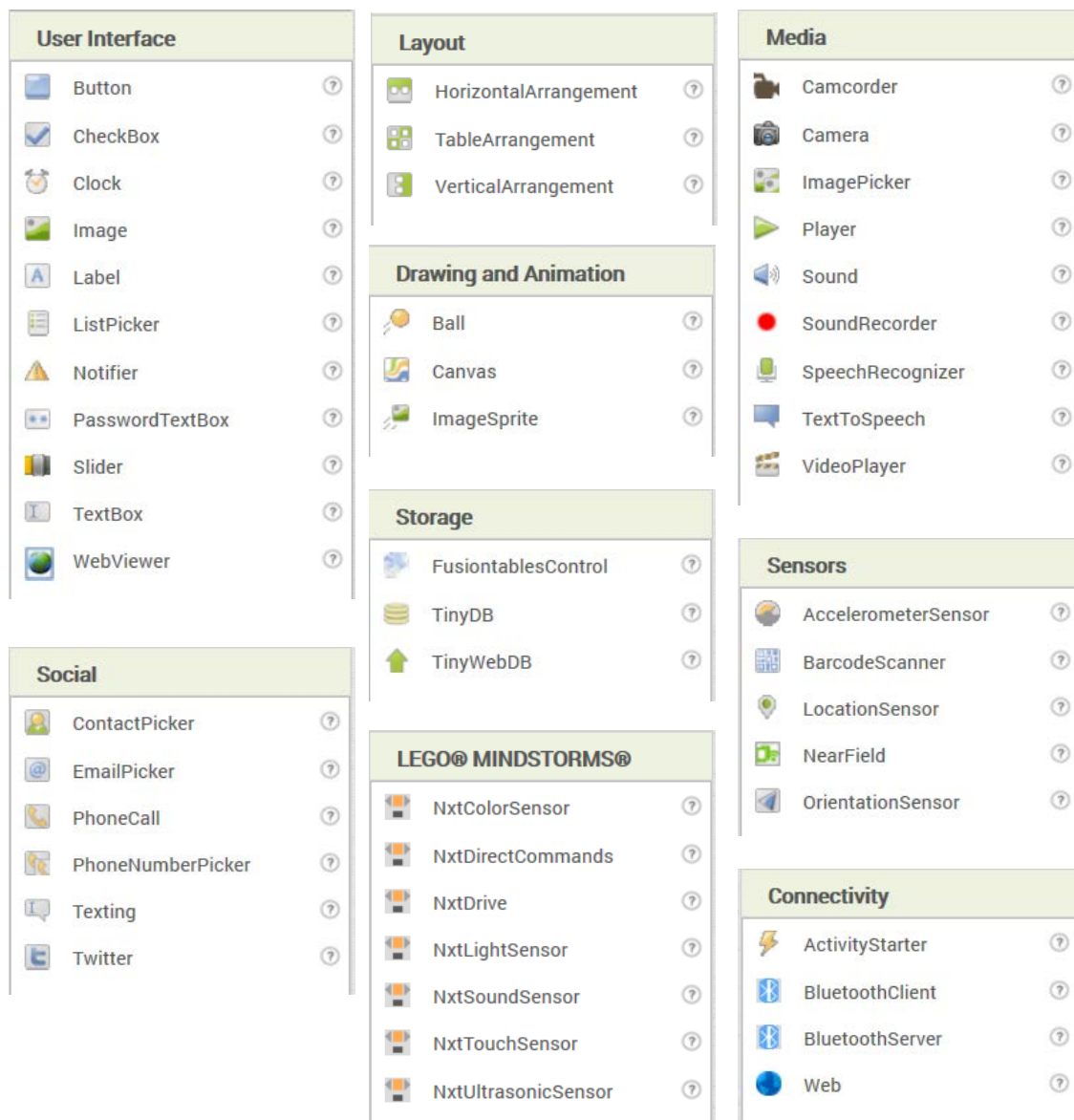
Προσαρμογή από: <http://appinventor.mit.edu/explore/designer-blocks.html>

Τα διαθέσιμα πλακίδια χωρίζονται σε εκείνα που αφορούν στα συγκεκριμένα συστατικά που έχει προσθέσει ο χρήστης κατά τη σχεδίαση της εφαρμογής και στα ενσωματωμένα πλακίδια γενικής συμπεριφοράς, που είναι ομαδοποιημένα σε διάφορες κατηγορίες, κάθε μια με το δικό της χρώμα.

Ο χρήστης χτίζει σενάρια σχέσεων και συμπεριφοράς, συναρμολογώντας και παραμετροποιώντας τα προαναφερόμενα πλακίδια, τα οποία δεν «κουμπώνουν» μεταξύ τους εάν η πρόταση που σχηματίζεται δεν είναι συντακτικά σωστή.

6.4 Διαθέσιμα συστατικά (components)

Η παλέτα με τα συστατικά που διατίθενται για προσθήκη και αξιοποίηση στις αναπτυσσόμενες εφαρμογές, τα διαχωρίζει σε ομάδες, όπως φαίνεται παρακάτω:



Εικόνα 6.3: Τα διαθέσιμα συστατικά (components) στο περιβάλλον σχεδίασης (Designer). Φεβ 2014

Ορισμένα από αυτά τα συστατικά είναι ορατά στην οθόνη της εφαρμογής, ενώ κάποια άλλα, χωρίς να εμφανίζονται, προσθέτουν επιπλέον δυνατότητες στην εφαρμογή.

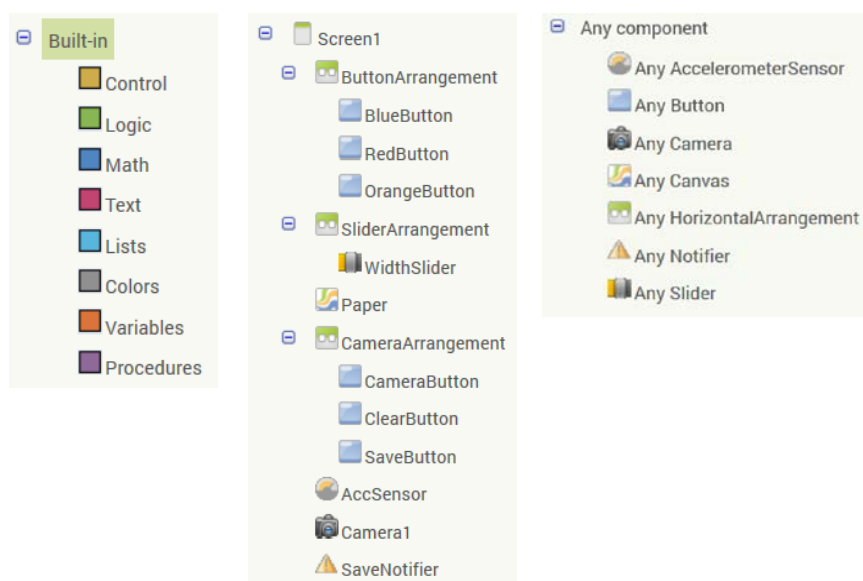
Σε κάθε περίπτωση, τα αντικείμενα που προστίθενται στην εφαρμογή έχουν ιδιότητες που μπορεί να τροποποιήσει άμεσα ο χρήστης ή να προγραμματίσει σενάρια για την αλλαγή τους.

Οι χρήστες μπορούν να αξιοποιήσουν τις μετρήσεις διαφόρων αισθητήρων (επιτάχυνσης, προσανατολισμού, θέσης, κ.α.) και τις πολυμεσικές λειτουργίες της συσκευής (ηχογράφηση, λήψη φωτογραφιών και βίντεο, αναπαραγωγή ήχων και δόνηση, μετατροπή κειμένου σε φωνή και αντίστροφα, κ.α.). Μπορούν επίσης να χειριστούν την αποστολή και λήψη σύντομων

μηνυμάτων κειμένου (SMS), τις επαφές του τηλεφώνου, την επικοινωνία με το Twitter και άλλες web υπηρεσίες, ενώ μέσω Bluetooth ή web αιτήσεων μπορούν να αλληλεπιδράσουν με άλλες συσκευές και διακομιστές. Τέλος, αξιοσημείωτη είναι η δυνατότητα χειρισμού ρομπότ Mindstorms NXT της LEGO ⁶¹.

6.5 Διαθέσιμα πλακίδια (blocks)

Στο περιβάλλον επεξεργασίας εντολών (Blocks Editor) εμφανίζονται ομαδοποιημένα σε κατηγορίες τα προγραμματιστικά πλακίδια. Σε ξεχωριστή ομάδα (built-in blocks) ανήκουν οι εντολές και τα πλακίδια που δεν σχετίζονται με μεμονωμένα συστατικά, ενώ σε άλλη ομάδα, ανάλογα με τα συστατικά που διαθέτει η κάθε εφαρμογή, εμφανίζονται και τα σχετικά μπλοκ εντολών. Τέλος, σε μια τρίτη ομάδα, περιλαμβάνονται πλακίδια που αφορούν μεν σε συστατικά που έχουν εισαχθεί από το χρήστη στην εφαρμογή, αλλά δεν συνδέονται άμεσα με κάποιο συγκεκριμένο αντικείμενο. Στην παρακάτω εικόνα εμφανίζονται οι ομάδες πλακιδίων για ένα ενδεικτικό έργο. Αριστερά εμφανίζονται οι ενσωματωμένες ομάδες που είναι πάντα διαθέσιμες στο χρήστη, στη μέση εμφανίζονται τα συστατικά - αντικείμενα που έχει προσθέσει ο χρήστης στο περιβάλλον σχεδίασης και δεξιά εμφανίζονται τα πλακίδια που αφορούν γενικευμένα όλα τα συστατικά που έχει προσθέσει ο χρήστης.



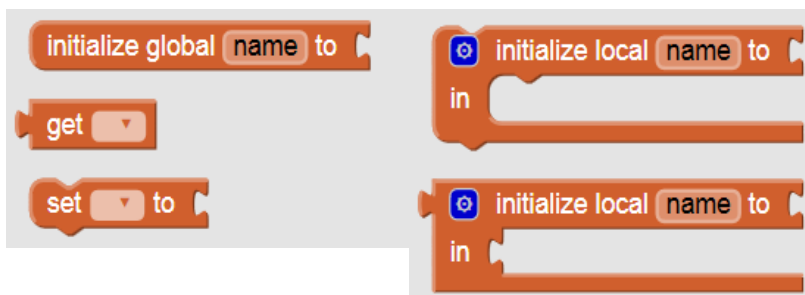
Εικόνα 5.4: Ενδεικτικές ομάδες πλακιδίων στο περιβάλλον συγγραφής του προγράμματος (Blocks Editor).

⁶¹ App Inventor και LEGO MINDSTORMS:

<http://ai2.appinventor.mit.edu/reference/components/legomindstorms.html>

Στη συνέχεια παρουσιάζονται τα πλακίδια εντολών, τα οποία αφορούν σε βασικές προγραμματιστικές έννοιες και δομές: α) μεταβλητές, β) λογικές εκφράσεις, γ) δομή επιλογής, δ) δομή επανάληψης και ε) υποπρογράμματα – διαδικασίες.

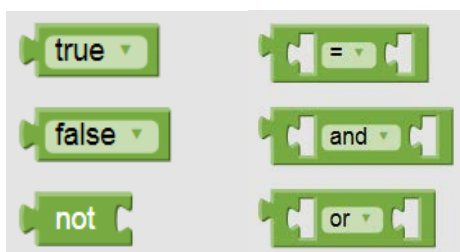
6.5.1 Μεταβλητές



Εικόνα 6.5: Πλακίδια για τον ορισμό και χειρισμό μεταβλητών στο AI.

Ο ορισμός μεταβλητών δεν γίνεται με ταυτόχρονη δήλωση του τύπου τους, ενώ οι μεταβλητές διακρίνεται σε τοπικές (local) και καθολικές (global). Η εντολή εκχώρησης τιμής σε μεταβλητή γίνεται με την εντολή `set ... to ...`, ενώ η πρόσβαση στο περιεχόμενό της γίνεται με την εντολή `get ...`

6.5.2 Λογικές εκφράσεις

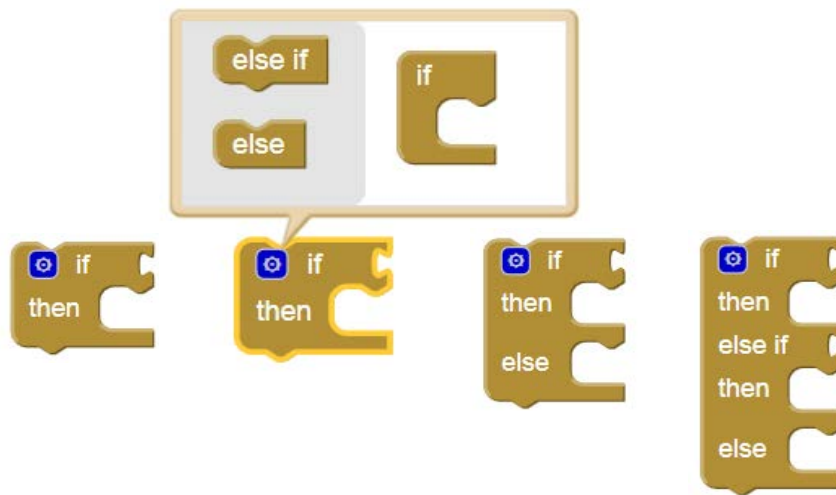


Εικόνα 6.6: Πλακίδια για τη σύνταξη λογικών εκφράσεων.

Για τη δημιουργία σύνθετων λογικών συνθηκών και προτάσεων διατίθενται πλακίδια των λογικών τιμών ΑΛΗΘΗΣ και ΨΕΥΔΗΣ, οι τελεστές ΌΧΙ, ΚΑΙ, Ή και ΙΣΟ ΜΕ/ΔΙΑΦΟΡΟ ΑΠΟ, τα οποία μπορούν να συνδυαστούν με τα πλακίδια αριθμών (math) και κειμένου (text).

6.5.3 Δομή επιλογής

Στην ομάδα Control περιέχονται εντολές που σχετίζονται με τη δομή επιλογής:

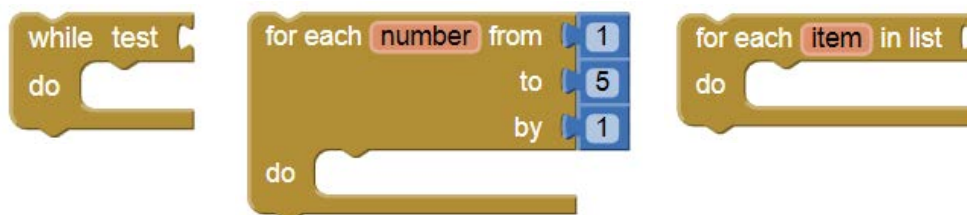


Εικόνα 6.7: Πλακίδια για την υλοποίηση δομών ελέγχου.

Η δομή απλής επιλογής (AN ... ΤΟΤΕ ...), με τη χρήση μεταλλαγέα (mutator) επεκτείνεται και μετατρέπεται σε δομή σύνθετης επιλογής (AN ... ΤΟΤΕ ... ΑΛΛΙΩΣ ...) ή πολλαπλών επιλογών (AN ... ΤΟΤΕ ... ΑΛΛΙΩΣ_ΑΝ ... ΤΟΤΕ ... ΑΛΛΙΩΣ ...), ενώ συνδυάζοντας τα παραπάνω πλακίδια, παράγονται και εμφωλευμένες δομές επιλογής.

6.5.4 Δομή επανάληψης

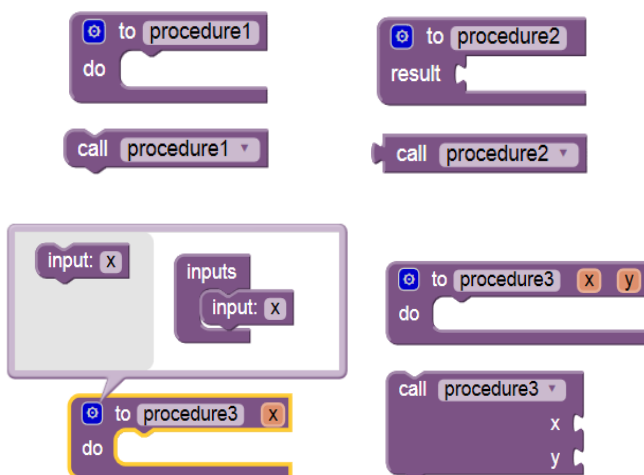
Τα πλακίδια για την υλοποίηση δομών επανάληψης βρίσκονται επίσης στην ομάδα πλακιδίων Control:



Εικόνα 6.8: Πλακίδια για την υλοποίηση δομών επανάληψης.

Η δομή επανάληψης υλοποιείται με την 'ΟΣΟ ... ΕΠΑΝΕΛΑΒΕ ή με τη ΓΙΑ ... ΑΠΟ ... ΜΕΧΡΙ ... ΜΕ_ΒΗΜΑ ... , ενώ διατίθεται και εκδοχή της ΓΙΑ, ειδικά για προσπέλαση λιστών.

6.5.5 Υποπρογράμματα - Διαδικασίες

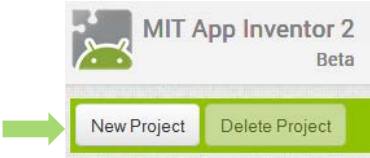
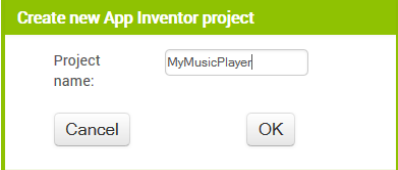


Εικόνα 6.9: Πλακίδια για τη δημιουργία και κλήση διαδικασιών.

Υποστηρίζεται η δημιουργία διαδικασιών με παραμέτρους ή χωρίς, ενώ οι διαδικασίες μπορούν να επιστρέφουν ή όχι κάποιο αποτέλεσμα, μετά την κλήση τους.

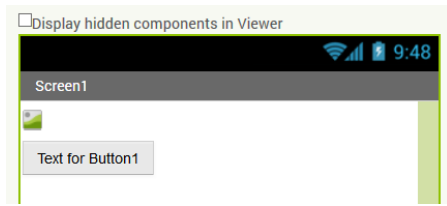
6.6 Ανάπτυξη εφαρμογών

Προκειμένου να δοθεί μια εικόνα για το πώς δημιουργείται μια εφαρμογή στο AI, στον πίνακα που ακολουθεί, παρουσιάζεται συνοπτικά ένα ενδεικτικό παράδειγμα ανάπτυξης μιας απλής εφαρμογής αναπαραγωγής μουσικής. Στην εφαρμογή αυτή, θα υπάρχει μια εικόνα και ένα κουμπί, το οποίο ξεκινάει την αναπαραγωγή ενός μουσικού κομματιού, ενώ με ταρακούνημα της συσκευής, η αναπαραγωγή σταματάει. Θα αναπτύξουμε, λοιπόν, σταδιακά την εφαρμογή μας στον υπολογιστή και στη συνέχεια θα τη δοκιμάσουμε στην κινητή μας συσκευή.

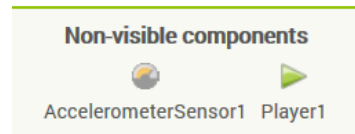
Εφαρμογή “My Music Player”	
Βήμα 1: Δημιουργία νέου project	
α) Μεταβαίνουμε στη δ/νση http://ai2.appinventor.mit.edu και επιλέγουμε τη δημιουργία ενός νέου project.	β) Δίνουμε ένα όνομα στο έργο, φροντίζοντας να αποτελείται από λατινικούς χαρακτήρες, χωρίς κενά (π.χ. MyMusicPlayer).
	

Βήμα 2: Σχεδίαση της οθόνης της εφαρμογής

α) Σέρνουμε τα συστατικά (components) μιας εικόνας (Image) και ενός κουμπιού (Button) από την ομάδα User Interface της παλέτας και το αποθέτουμε εντός της οθόνης (Screen1 στην περιοχή Viewer).



β) Σέρνουμε και αποθέτουμε στην οθόνη, το συστατικό Player από την ομάδα Media και το συστατικό AccelerometerSensor από την ομάδα Sensors, τα οποία αποτελούν μη-ορατά αντικείμενα.

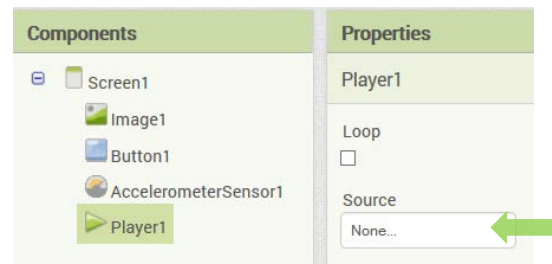


Βήμα 3: Προσθήκη των αρχείων πολυμέσων και σύνδεσή τους με τα αντικείμενα της οθόνης

α) Επιλέγουμε από τη λίστα Components το αντικείμενο Image1 και από την περιοχή Properties δεξιά του, για την ιδιότητα Picture, επιλέγουμε μια εικόνα της αρεσκείας μας, μεταφορτώνοντάς την από τον H/Y μας στο έργο μας. Φροντίζουμε τα ονόματα των αρχείων να αποτελούνται από λατινικούς χαρακτήρες.



β) Ομοίως, επιλέγουμε από τη λίστα Components το αντικείμενο Player1 και από την περιοχή Properties, για την ιδιότητα Source, επιλέγουμε και μεταφορτώνουμε στο έργο μας ένα αρχείο ήχου της αρεσκείας μας. Μπορούμε επίσης να ορίσουμε την ένταση του ήχου ή/και αυτόματη επανάληψη της αναπαραγωγής όταν αυτή ολοκληρωθεί.



Βήμα 4: Συγγραφή προγράμματος

α) Μεταβαίνουμε στην καρτέλα Σύνταξης Προγράμματος, πατώντας στο κουμπί Blocks, ώστε να ορίσουμε τη συμπεριφορά της εφαρμογής, συσχετίζοντας ενέργειες με γεγονότα.



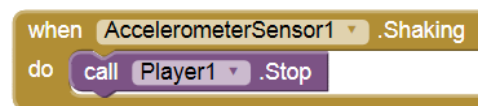
β) Ανοίγουμε το συρτάρι των πλακιδίων που σχετίζονται με το αντικείμενο Button1, επιλέγουμε το πλακίδιο when Button1.Click do και το αποθέτουμε στον Viewer.



γ) Από το συρτάρι του Player1 επιλέγουμε το πλακίδιο call Player1.Start και το «κουμπώνουμε» εσωτερικά του προαναφερόμενου πλακιδίου, ώστε να σχηματιστεί η εντολή που ξεκινάει την αναπαραγωγή της μουσικής με το άγγιγμα του κουμπιού Button1.

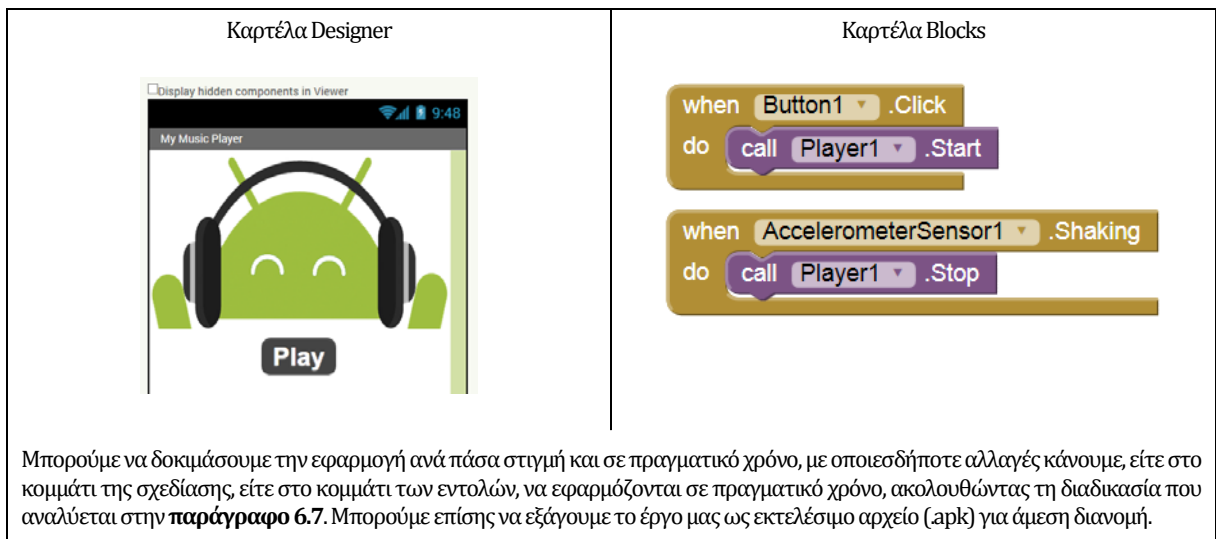


δ) Από το συρτάρι του AccelerometerSensor1, επιλέγουμε το πλακίδιο when AccelerometerSensor1.Shaking και το αποθέτουμε στον Viewer, ώστε να συνδέσουμε με αυτό το γεγονός την εκτέλεση της εντολής call Player1.Stop του συρταριού Player1.



Βήμα 5: Έλεγχος και διανομή της εφαρμογής

Μέχρι στιγμής, τα δυο τμήματα της εφαρμογής μας είναι, ενδεικτικά, όπως τα παρακάτω και απομένει να ελέγξουμε τη λειτουργία της.

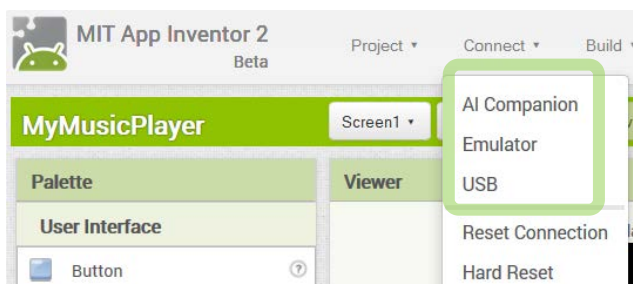


Πίνακας 6.1: Ενδεικτικό παράδειγμα ανάπτυξης εφαρμογής κατά βήματα στο AI.

6.7 Έλεγχος των εφαρμογών

Ο έλεγχος και η αποσφαλμάτωση των παραγόμενων εφαρμογών, όπως φαίνεται και στην εικόνα 6.3, γίνεται σε πραγματικό χρόνο με τρεις εναλλακτικούς τρόπους:

- Με τον ενσωματωμένο προσομοιωτή Android (emulator) που μπορεί να εγκατασταθεί επιπρόσθετα στον υπολογιστή του χρήστη.
- Με ενσύρματη USB σύνδεση μεταξύ υπολογιστή και Android συσκευής.
- Ασύρματα, μέσω Wi-Fi σύνδεσης της συσκευής σε κοινό δίκτυο με τον υπολογιστή (προτεινόμενη λύση).



Εικόνα 6.10: Οι τρεις δυνατές επιλογές για έλεγχο των εφαρμογών στο AI.

Στη συνέχεια γίνεται αναφορά σε καθέναν από τους τρεις αυτούς τρόπους.

6.7.1 Προσομοιωτής (on-screen Android emulator)

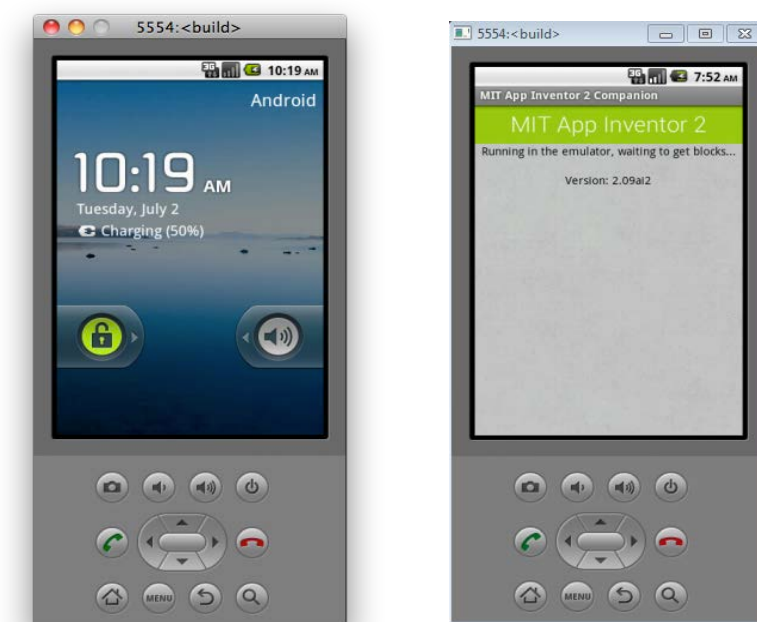


Εικόνα 6.11: Η περίπτωση του προσομοιωτή Android για έλεγχο των εφαρμογών στο AI.

Πηγή: <http://appinventor.mit.edu/explore/ai2/setup-emulator.html>

Εγκαθιστώντας ένα πακέτο λογισμικού (App Inventor Setup Software) από την ιστοσελίδα του AI, ο χρήστης μπορεί να ελέγχει σε πραγματικό χρόνο τις εφαρμογές που αναπτύσσει σε μια εικονική μηχανή Android, εντός του υπολογιστή του. Προς το παρόν (Ιανουάριος 2014), διατίθενται εκδόσεις για Windows και Mac OS X, με την έκδοση για GNU/Linux να αναμένεται σύντομα.

Με την εγκατάσταση του παραπάνω λογισμικού, ο χρήστης εκτελεί στον υπολογιστή μια εφαρμογή με την ονομασία aiStarter, η οποία επιτρέπει στον web browser να επικοινωνήσει με τον προσομοιωτή. Έτσι, όποτε θελήσει να δοκιμάσει την εφαρμογή, μπορεί σε μερικά δευτερόλεπτα να τη δει στην οθόνη της εικονικής συσκευής.



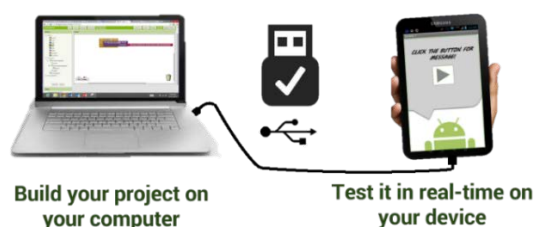
Εικόνα 6.12: Στιγμιότυπα του προσομοιωτή Android πριν και κατά τη σύνδεσή του για έλεγχο έργου.

Πηγή: <http://appinventor.mit.edu/explore/ai2/setup-emulator.html>

Η λύση του προσομοιωτή προσφέρει το πλεονέκτημα της μη υποχρεωτικής διάθεσης πραγματικής συσκευής, που μπορεί να φανεί χρήσιμη σε σχολικά εργαστήρια ή υπό οποιεσδήποτε συνθήκες δεν διατίθεται αρκετός αριθμός συσκευών Android.

Ο έλεγχος σε προσομοιωτή, βέβαια, έχει το σημαντικό μειονέκτημα ότι ο χρήστης δεν μπορεί να εκμεταλλευτεί αρκετές από τις ιδιαίτερες δυνατότητες μιας πραγματικής συσκευής Android, π.χ. αισθητήρες προσανατολισμού ή επιτάχυνσης και GPS, ενώ πρακτικά, παρατηρείται κάποια καθυστέρηση σε σχετικά αργούς υπολογιστές.

6.7.2 Ενσύρματη σύνδεση (USB connection)



Εικόνα 6.13: Η περίπτωση της USB σύνδεσης για έλεγχο των εφαρμογών στο AI.

Πηγή: <http://appinventor.mit.edu/explore/ai2/setup-device-usb.html>

Για την επιτυχή επικοινωνία της Android συσκευής με τον υπολογιστή του χρήστη, απαιτείται η ύπαρξη του προαναφερόμενου λογισμικού στον υπολογιστή, όπως επίσης και η εγκατάσταση και εκτέλεση μιας επιπλέον εφαρμογής στην Android συσκευή, με ονομασία MIT AI2 Companion. Αξίζει να σημειωθεί ότι, ενώ σε περιβάλλον Mac και Linux δεν απαιτείται η εγκατάσταση οδηγών (drivers), σε περιβάλλον Windows η εύρεση και εγκατάσταση των απαραίτητων drivers για ορισμένες συσκευές Android, μπορεί να ταλαιπωρήσει το μέσο χρήστη.

Λήψη και εγκατάσταση της εφαρμογής MIT AI2 Companion στην Android συσκευή

Η εφαρμογή MIT AI2 Companion μπορεί να εγκατασταθεί, είτε από το επίσημο Play Store ⁶² της Google, είτε χειροκίνητα (ως .apk αρχείο εγκατάστασης) από την ιστοσελίδα του AI. Συνιστάται η

⁶² MIT AI2 Companion app στο Play Store:

<https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3>

εγκατάσταση μέσω του Play Store, ενώ εναλλακτική λύση αποτελεί η εγκατάσταση του .apk αρχείου είναι η λύση στην περίπτωση αδυναμίας πρόσβασης στο Play Store.

6.7.3 Ασύρματη σύνδεση (Wi-Fi connection)

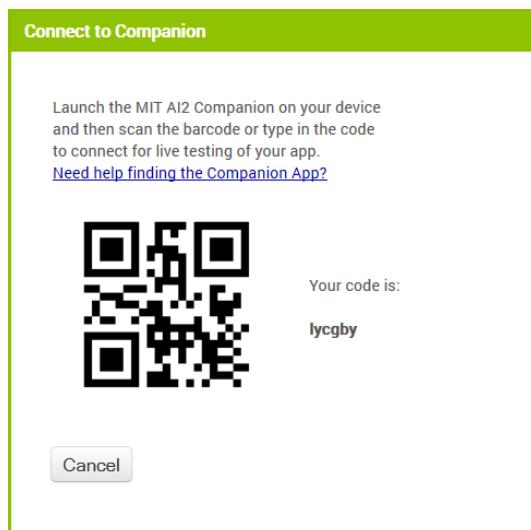
Ο πιο απλός τρόπος ελέγχου των εφαρμογών είναι μέσω ασύρματης σύνδεσης με τον υπολογιστή. Στην περίπτωση αυτή, δεν απαιτείται η εγκατάσταση επιπλέον λογισμικού στον υπολογιστή του χρήστη, παρά μόνο η εγκατάσταση και εκτέλεση της προαναφερόμενης εφαρμογής MIT AI2 Companion στην Android συσκευή στην οποία θα γίνει η δοκιμή.



Εικόνα 6.14: Η περίπτωση της ασύρματης σύνδεσης για έλεγχο των εφαρμογών στο AI.

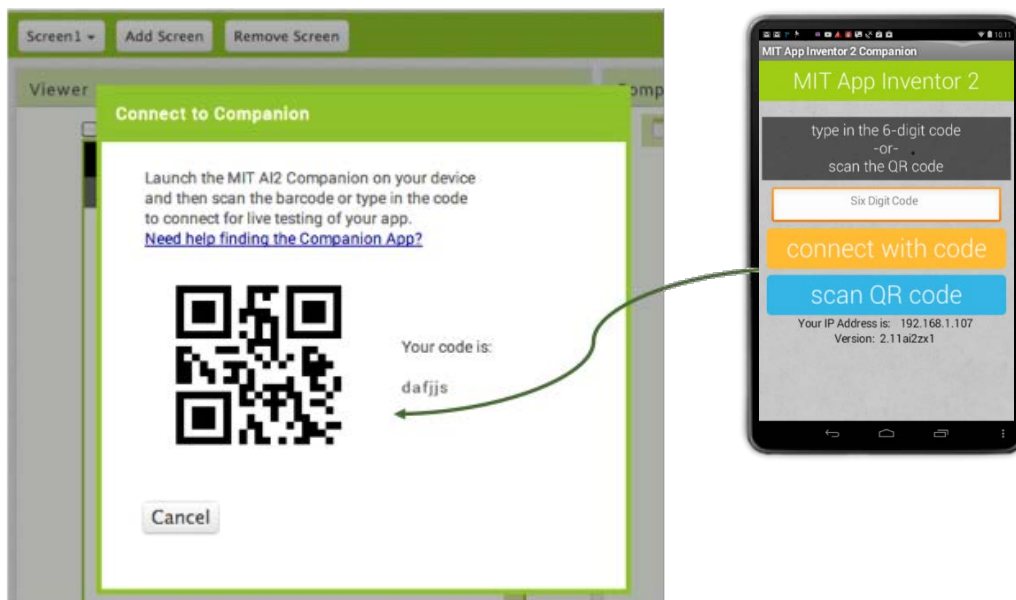
Πηγή: <http://appinventor.mit.edu/explore/ai2/setup-device-wifi.html>

Πιο συγκεκριμένα, με την προϋπόθεση ότι υπολογιστής και Android συσκευή είναι συνδεδεμένα στο ίδιο τοπικό δίκτυο, ο χρήστης επιλέγει τη σύνδεση μέσω του AI Companion από το περιβάλλον του AI στο πρόγραμμα πλοήγησης που χρησιμοποιεί και εμφανίζεται ένα παράθυρο με έναν μοναδικό κωδικό μήκους έξι χαρακτήρων και το ισοδύναμο QR code του.



Εικόνα 6.15: Εμφάνιση του κωδικού για σύνδεση της συσκευής με το AI.

Εκτελώντας την εφαρμογή MIT AI2 Companion στην Android συσκευή, ο χρήστης καλείται να πληκτρολογήσει τον κωδικό αυτό ή να σαρώσει με την κάμερα της συσκευής το QR code, ώστε να αναγνωριστεί αυτόματα ο κωδικός και να επιτευχθεί η ασύρματη σύνδεση.



Εικόνα 6.16: Σάρωση του κωδικού για ασύρματη σύνδεση της συσκευής με το AI.

Στη συνέχεια, ο χρήστης μπορεί να δοκιμάσει σε πραγματικό χρόνο την εφαρμογή του. Μπορεί μάλιστα να πραγματοποιεί αλλαγές σε αυτή, είτε στο τμήμα της σχεδίασης της οθόνης, είτε στο τμήμα των εντολών και αυτές να εφαρμόζονται άμεσα στη φορητή του συσκευή.

Σημειώνεται ότι στο Παράρτημα A.5 διατίθεται ένας σύντομος οδηγός για την εγκατάσταση των απαραίτητων εφαρμογών για τη δοκιμή των αναπτυσσόμενων εφαρμογών μέσω ασύρματης σύνδεσης.

6.8 Διανομή των εφαρμογών

Οι παραγόμενες εφαρμογές μπορούν να εξαχθούν ως εκτελέσιμα αρχεία (.apk) για άμεση εγκατάσταση σε οποιαδήποτε συσκευή ή τη διάθεσή τους στο Google Play Store.

Είναι επίσης δυνατή η εξαγωγή (export) του πηγαίου κώδικα, συνοδευόμενου με τα αρχεία πολυμέσων, εφόσον υπάρχουν τέτοια, με τη μορφή ενός ενιαίου αρχείου ανά έργο (.aia) για την τήρηση αντιγράφων ασφαλείας ή τη διανομή του. Η αντίστοιχη δυνατότητα εισαγωγής (import) των παραπάνω αρχείων δίνει τη δυνατότητα εύκολου διαμοιρασμού του πηγαίου κώδικα και των συνοδευτικών αρχείων των εφαρμογών με άλλους, εν προκειμένω με τους μαθητές.

Σημειώνεται ότι σε ορισμένες περιπτώσεις δεν είναι δυνατή η δοκιμή σε πραγματικό χρόνο, οπότε αναγκαστικά γίνεται εξαγωγή του εκτελέσιμου αρχείου και εγκατάστασή του σε κάποια συσκευή για έλεγχο.

6.9 Το ενδεχόμενο αξιοποίησης του AI στο σχολείο

Με δεδομένη την κατάσταση που τείνει να διαμορφωθεί με τις αναπροσαρμογές στα Αναλυτικά Προγράμματα Σπουδών, ειδικά στο Λύκειο, είναι ευκαιρία να εξεταστεί το ενδεχόμενο να ενταχθεί ο προγραμματισμός ψηφιακών φορητών συσκευών στη δευτεροβάθμια εκπαίδευση, με ένα οπτικό περιβάλλον προγραμματισμού, κατάλληλο για αρχάριους, όπως είναι το App Inventor ή τουλάχιστον να συμπεριληφθεί στα προτεινόμενα προγραμματιστικά περιβάλλοντα.

6.9.1 Ένταξη του AI στα προτεινόμενα προγραμματιστικά περιβάλλοντα

Στο κεφάλαιο 3, παρουσιάστηκε η τάση να προτείνονται και να προτιμώνται περιβάλλοντα οπτικού προγραμματισμού, από τη βαθμίδα του Δημοτικού μέχρι και του Λυκείου. Στα προτεινόμενα προγραμματιστικά περιβάλλοντα περιλαμβάνονται ενδεικτικά τα ακόλουθα: EasyLogo, Scratch, BYOB, Kodu, GreenFoot, Storytelling Alice.

Παρουσιάστηκαν επίσης, οι προγραμματιστικές έννοιες και ικανότητες που αναμένεται να γνωρίσουν και να κατακτήσουν οι μαθητές. Συνυπολογίζοντας το γεγονός ότι τα νεότερα ΑΠΣ πρεσβεύουν και συστήνουν τη σπειροειδή προσέγγιση, με τα πολλαπλά περάσματα ανά γνωστικό αντικείμενο, καθώς και το γεγονός ότι, κατά πάσα πιθανότητα, οι μαθητές έχουν εξοικειωθεί με παρόμοια περιβάλλοντα, όπως το Scratch, προετοιμάζουν ευνοϊκά το έδαφος για

την ένταξη και χρήση ενός παρόμοιου σε λογική και λειτουργία προγραμματιστικού περιβάλλοντος στην εκπαιδευτική διαδικασία.

Σε πρόσφατη αναφορά [79] στα κριτήρια που πρέπει να λαμβάνονται υπόψη από τον εκπαιδευτικό για την επιλογή του κατάλληλου προγραμματιστικού περιβάλλοντος αναφέρονται τα εξής:

- Να ικανοποιεί τους στόχους του Προγράμματος Σπουδών.
- Να ανήκει σε ομάδα λογισμικών με κοινή φιλοσοφία.
- Να διαθέτει διεπαφή Οπτικού Προγραμματισμού.
- Να είναι ελεύθερο λογισμικό.
- Να είναι εξελληνισμένο.
- Να υπάρχει κοινότητα μάθησης εστιασμένης στους σκοπούς του μαθήματος.

Από τα παραπάνω κριτήρια, θεωρούμε ότι το AI τα ικανοποιεί όλα εκτός από τα δυο τελευταία, τουλάχιστον προς το παρόν. Λόγω του ελεύθερου και ανοικτού χαρακτήρα του AI, πιστεύουμε ότι είναι θέμα χρόνου η μετάφραση του περιβάλλοντος στα ελληνικά, όπως επίσης και η δημιουργία σχετικής κοινότητας μάθησης.

Επίσης, όλο το νέο Πρόγραμμα Σπουδών της υποχρεωτικής εκπαίδευσης βασίζεται στη χρήση διαδικασιών και το AI χειρίζεται διαδικασίες τόσο χωρίς, όσο και με παραμέτρους, ενώ παρέχει πολλαπλούς τρόπους εισόδου δεδομένων στις εφαρμογές, όπως και αλληλεπίδρασης του χρήστη με τη συσκευή (οθόνη αφής, GPS, SMS, πληθώρα αισθητήρων, κ.α.). Επίσης, υποστηρίζει δομές δεδομένων, με την χρήση λιστών.

6.9.2 AI και ομοιότητες με το Scratch

Με βάση τις πολυάριθμες έρευνες που συνηγορούν υπέρ της χρήσης του Scratch στη σχολική εκπαίδευση [54][77][81][83][86][87][91][92], μπορούμε να παρουσιάσουμε ορισμένα κοινά χαρακτηριστικά του AI με το Scratch, αλλά και να επιχειρήσουμε να εξετάσουμε το ενδεχόμενο να παρέχει βελτιωμένη εμπειρία χρήσης σε σχέση με το Scratch.

Το AI μπορεί να θεωρηθεί κάτι σαν το Scratch για έξυπνες φορητές συσκευές, καθώς ισχύει πως και οι δυο αυτές γλώσσες βασίστηκαν σε κοινή βιβλιοθήκη για τα πλακίδια (blocks) τους [56]. Οι παλέτες των πλακιδίων, ομαδοποιημένες σε κατηγορίες, με τα πλακίδια να «κλειδώνουν» μεταξύ τους, όταν επιτρέπεται, παρέχουν οπτικά ερεθίσματα για το πώς συναρμολογούνται οι εντολές και αποτρέπουν συντακτικά λάθη [62]. Επιπλέον, όμως, το AI ενσωματώνει πλακίδια που απλοποιούν τη χρήση τεχνολογιών, όπως αποστολής SMS ή αισθητήρων GPS, οπότε μπορούν κάλλιστα να αξιοποιηθούν για τη δημιουργία αντίστοιχων κατανοητών παραδειγμάτων, που δεν περιορίζονται σε εικονικούς κόσμους, αλλά έχουν νόημα και στον πραγματικό. Κατ' αυτόν τον τρόπο, ακόμα και τα εισαγωγικά παραδείγματα δίνουν κίνητρα και αφορμές στους μαθητές για περαιτέρω ενασχόληση. Ο Wolber [19][67] αναφέρει ενδεικτικά παραδείγματα, που φαίνεται ότι η έμπνευση των μαθητών του, μετουσιώθηκε, χωρίς ιδιαίτερο κόπο, σε ενδιαφέρουσες και πρωτότυπες εφαρμογές. Σε ορισμένες περιπτώσεις, μάλιστα, οι ιδέες των μαθητών προκάλεσαν το ενδιαφέρον επαγγελματιών του χώρου, που επιχείρησαν να έρθουν σε επαφή με τους μαθητές.

Ένα ιδιαίτερο χαρακτηριστικό του AI, κατά παρόμοιο τρόπο με το Scratch, είναι και η δυνατότητα παράλληλης εκτέλεσης τμημάτων κώδικα. Η γλώσσα Scratch υπάγεται σε μια νέα προγραμματιστική κατηγορία που ονομάζεται ταυτόχρονος προγραμματισμός (concurrent programming). *“Ταυτόχρονο πρόγραμμα ονομάζεται ένα σύνολο από ακολουθιακά τμήματα προγραμμάτων που εκτελούνται παράλληλα”* [83]. Αντιστοίχως στο AI, λόγω των πολλαπλών αντικειμένων που αλληλεπιδρούν ταυτόχρονα κατά την εκτέλεση της εφαρμογής, και ιδιαίτερος όταν χρησιμοποιούνται πολλαπλά ρολόγια (clock/timer component) είναι δυνατός ο ταυτόχρονος προγραμματισμός. Μέσα από τη διδασκαλία του ταυτόχρονου προγραμματισμού εξυπηρετούνται οι ευρύτεροι στόχοι της εκπαίδευσης αναφορικά με την επιστήμη των υπολογιστών [04].

Παρομοίως με το Scratch, η λογική του AI στηρίζεται στη δημιουργία «σεναρίων», με την τεχνική «σύρε και άσε», για καθένα από τα συστατικά που προστίθενται από το χρήστη, με χρήση των βασικών δομών που απαντώνται σε όλες τις γλώσσες προγραμματισμού (Δομή Ακολουθίας, Δομή Επιλογής και Δομή Επανάληψης), αλλά και με στοιχεία αντικειμενοστραφούς προγραμματισμού, προγραμματισμού οδηγούμενου από γεγονότα, ακόμα και ταυτόχρονου προγραμματισμού [91].

Κάποια από τα προβλήματα στην οικοδόμηση της έννοιας της μεταβλητής από τους μαθητές, όπως ο τύπος και οι κατηγορίες μεταβλητών που χρησιμοποιούνται από τους προγραμματιστές

[74], δεν υφίστανται στη γλώσσα προγραμματισμού Scratch, παρομοίως και στο AI, όπου δεν δηλώνεται τύπος για τις μεταβλητές.

Ένα από τα εμπόδια για την οικοδόμηση της έννοιας της δομής ελέγχου σε παραδοσιακά περιβάλλοντα μάθησης είναι η αναπαράσταση που έχουν οι μαθητές για τη σειριακή εκτέλεση του προγράμματος [74]. Στο Scratch και στο AI, ο χώρος αναφοράς του «εάν» και του «αλλιώς» μιας δομής επιλογής είναι οπτικά εμφανής, συμβάλλοντας έτσι σε μια πιο διαισθητική κατανόηση της δομής.

Η ανάπτυξη εφαρμογών για κινητά εμπεριέχει την ανάγκη οι μαθητές να καταλάβουν βασικές έννοιες του καθοδηγούμενου από γεγονότα (event-driven) προγραμματισμού, κάτι το οποίο απλοποιείται από το AI που ουσιαστικά αντικαθιστά πολύπλοκες συντακτικές και αντικειμενοστραφείς διαδικασίες από blocks [08]. Ταυτόχρονα, παρέχει συστατικά μέρη (components) που χειρίζονται πολύπλοκες αλληλεπιδράσεις με το λειτουργικό σύστημα του Android, ενώ και η επιτόπου εφαρμογή των αλλαγών του κώδικα στον εξομοιωτή ή στο συνδεδεμένο τηλέφωνο διευκολύνουν και επιταχύνουν την ανατροφοδότηση των μαθητών.

Το AI αποτελεί μια δυναμικά διερμηνευόμενη γλώσσα, καθώς επιτρέπει τις αλλαγές στον κώδικα, ακόμα και κατά τη διάρκεια της δοκιμαστικής εκτέλεσης των προγραμμάτων. Σε αντίθεση με τις γλώσσες γενικού σκοπού, που η εκτέλεση των βασικών λειτουργιών δεν είναι οπτικοποιημένη, στο AI ο χρήστης μπορεί να συνθέσει τον κώδικα και να δει επιτόπου τα αποτελέσματα και τη συμπεριφορά του προγράμματος. Επίσης, μπορεί κάνει αλλαγές στον κώδικα του ακόμη και αν το πρόγραμμα εκτελείται και να προκαλέσει ένα νέο παραγόμενο αποτέλεσμα. Έτσι, εξοικονομεί χρόνο και κυρίως είναι σε θέση να λαμβάνει άμεση ανατροφοδότηση, που με τη σειρά της θα βοηθήσει τον μαθητή να αξιολογήσει καλύτερα τον εαυτό του και το έργο του, να αναστοχαστεί πάνω στους στόχους του και να ενισχύσει την αυτορρύθμισή του. Λαμβάνοντας γρήγορα αποτελέσματα, οι μαθητές αποκτούν κίνητρο να προσπαθήσουν περαιτέρω και οι δείκτες εμπλοκής αυξάνονται [85].

Από τα σημαντικότερα πλεονεκτήματα του App Inventor, παρομοίως με το Scratch, είναι ότι δεν απαιτεί από τους μαθητές να έχουν απομνημονεύσει και να πληκτρολογούν τις εντολές, διότι τις προσφέρει έτοιμες. Επιπλέον, οι αρχάριοι προγραμματιστές νιώθουν ασφαλείς και αποκτούν μεγαλύτερη αυτοπεποίθηση, καθώς το ίδιο το πρόγραμμα τους εμποδίζει να υποπέσουν σε

βασικά συντακτικά λάθη, κατά τη συγγραφή του κώδικα. Τα πλακίδια που αναπαριστούν τις εντολές «κουμπώνουν» μαζί μόνο με συντακτικά ορθό τρόπο και η κάθε κατηγορία έχει το δικό της χρώμα για να είναι πιο αναγνωρίσιμη και να μην υπάρχει σύγχυση της μιας κατηγορίας εντολών με την άλλη. Έτσι, οι μαθητές έχουν την ευκαιρία να πειραματιστούν με το εργαλείο και να μάθουν κάνοντας (learn by doing), χωρίς τον φόβο ότι κάτι θα χαλάσουν ή θα κάνουν λάθος. Εξάλλου, με αυτό τον τρόπο μειώνεται το ενδογενές νοητικό φορτίο που απαιτείται για τη συγγραφή κώδικα και ταυτόχρονα επιτρέπεται στους χρήστες να επικεντρωθούν στη δομή και επίλυση ενός προβλήματος [39][54].

6.9.3 Σύνοψη των συμπερασμάτων από τη χρήση του AI στο εξωτερικό

Προς το παρόν δεν υπάρχουν επίσημα καταγεγραμμένες αναφορές από τη χρήση του AI στην ελληνική εκπαίδευση. Στο εξωτερικό, όμως, πρόσφατες έρευνες [20][26][28][40] [57][85] επισημαίνουν ότι υπάρχουν αρκετά πλεονεκτήματα στη χρήση του AI για τη διδασκαλία του προγραμματισμού, όπως:

- Μάθηση μέσω της επίλυσης προβλημάτων.
- Περιβάλλον εύκολο στη χρήση με πολλές δυνατότητες (low floor - high ceiling).
- Αντικειμενοστραφές μοντέλο οπτικού προγραμματισμού με δομές ελέγχου καθοδηγούμενες από γεγονότα (event-driven).
- Παροχή επιπλέον κινήτρων στους μαθητές σε σχέση με το Scratch εξαιτίας της φορητότητας και της πρακτικής χρήσης των εφαρμογών που δημιουργούνται.
- Ύπαρξη emulator (προσομοιωτή), οπότε ουσιαστικά δεν χρειάζονται πολλές ή και καθόλου συσκευές για την εισαγωγή σε μια σχολική τάξη.
- Υποστήριξη από τη Google και το MIT (Massachusetts Institute of Technology).

Ο Morelli [40] διερεύνησε διάφορους τρόπους χρήσης του AI για την ανάπτυξη υπολογιστικής σκέψης στην K-12 βαθμίδα εκπαίδευσης και τα συμπεράσματά του ήταν ενθαρρυντικά ως προς τη χρήση του AI για την ελκυστική εισαγωγή των αρχαρίων σε προγραμματισμό στις βασικές προγραμματιστικές έννοιες και στην ανάπτυξη ικανοτήτων επίλυσης προβλημάτων. Παρατηρεί επίσης ότι το AI, σε σύγκριση με το Scratch και το Alice, υπερτερεί ως προς την αμεσότητα επίδειξης λειτουργικών εφαρμογών και της διάδοσής τους στον κοινωνικό περίγυρο των μαθητών (viral effect), αναφέροντας χαρακτηριστικά ότι «είναι τόσο εύκολη, όσο το να βγάλουν

το κινητό τηλέφωνο από την τσέπη τους». Αυτή η φορητότητα συνεισφέρει και στη διάδοση του ίδιου του «μικροβίου» του προγραμματισμού μεταξύ των μαθητών.

Παρομοίως, ο Grover [20] μελέτησε την επίδραση του AI σε μαθητές Γυμνασίου και διαπίστωσε ότι το AI αποτέλεσε ένα άριστο, με προοπτικές και συνάμα διασκεδαστικό μέσο για την εισαγωγή τους στον προγραμματισμό. Επισήμανε επίσης πως τα video games και η ρομποτική δεν είναι τόσο «δημοκρατικά», καθώς συνήθως ανταποκρίνονται περισσότερα στα ενδιαφέροντα των αγοριών.

Ο Roy [57] μάλιστα, καταλήγει πως το AI μπορεί να αποτελέσει γέφυρα για τη μετάβαση σε προγραμματισμό Java, εισάγοντας τους μαθητές σε σχετικές προγραμματιστικές έννοιες μέσω της απλότητας που παρέχει το AI και μεταβαίνοντας σταδιακά και ομαλότερα στο Android SDK, χτίζοντας πάνω στις γνώσεις που αποκτήθηκαν.

Συνυπολογίζοντας, λοιπόν, το βαθμό εισχώρησης των έξυπνων ψηφιακών φορητών συσκευών στην καθημερινότητα των νέων γενιών, από πολύ μικρή ηλικία, σε συνδυασμό με την παιγνιώδη φύση του AI, θεωρούμε ότι το AI παρέχει ένα αυθεντικό περιβάλλον μάθησης, ελκυστικό προς τους μαθητές της δευτεροβάθμιας εκπαίδευσης για την εισαγωγή τους σε βασικές έννοιες προγραμματισμού, αλλά και στην καλλιέργεια υπολογιστικής σκέψης. Επιπλέον, οι έξυπνες φορητές συσκευές παύουν πλέον να είναι μαύρα κουτιά για τους μαθητές, που μπορούν να τις βλέπουν από άλλη οπτική. Το AI δεν συμβάλει μόνο στη διδασκαλία και μάθηση προγραμματισμού, αλλά δίνει την ευκαιρία στον οποιοδήποτε να είναι δημιουργός και όχι απλά καταναλωτής της τεχνολογίας, ενδυναμώνοντας το ρόλο του στην κοινωνία της πληροφορίας.

Κεφάλαιο 7

Μεθοδολογία

Στο κεφάλαιο αυτό παρουσιάζεται ο σχεδιασμός και τα βασικά χαρακτηριστικά της ερευνητικής διαδικασίας για την επαλήθευση ή την απόρριψη της ερευνητικής υπόθεσης της διευκόλυνσης της διδασκαλίας και μάθησης βασικών εννοιών προγραμματισμού στη δευτεροβάθμια εκπαίδευση με την αξιοποίηση του App Inventor.

7.1 Η μέθοδος

Η ερευνητική διαδικασία που ακολουθήθηκε ήταν συνδυασμός ποιοτικής και ποσοτικής μεθόδου. Η συλλογή στοιχείων βασίστηκε στη συμμετοχική παρατήρηση του ερευνητή, σε ηλεκτρονικά ερωτηματολόγια που συμπλήρωσαν τα υποκείμενα της μελέτης και σε ημιδομημένες συνεντεύξεις από τους εκπαιδευτικούς – επιμορφωτές. Έτσι, με την τεχνική της τριγωνοποίησης, τα δεδομένα που συλλέχθηκαν, αλληλοσυμπληρώνονται και αλληλοεπαληθεύονται. Οι εκπαιδευτικοί – επιμορφωτές λειτούργησαν και ως παρατηρητές, εξασφαλίζοντας την αξιοπιστία και την εγκυρότητα των δεδομένων.

Μετά την ολοκλήρωση της βιβλιογραφικής επισκόπησης για τα προβλήματα που συναντώνται στη διδασκαλία και μάθηση βασικών εννοιών προγραμματισμού στη δευτεροβάθμια εκπαίδευση και στις θετικές αναφορές ερευνητών άλλων χωρών για την αξιοποίηση των ψηφιακών φορητών συσκευών στο παραπάνω πλαίσιο, επιλέξαμε να σχεδιάσουμε την έρευνά μας με βάση στόχους που τίθενται στα Προγράμματα Σπουδών, όπως αυτοί παρουσιάστηκαν στο Κεφάλαιο 4. Ο πληθυσμός που επιλέχθηκε για το πείραμα περιλάμβανε μαθητές από Γ' τάξη Γυμνασίου μέχρι και Β' τάξη Λυκείου. Στόχος του πειράματος ήταν η διδασκαλία βασικών προγραμματιστικών εννοιών με τη χρήση του ΑΙ, ανεξάρτητα από την τάξη των μαθητών, επιλέγοντας τη μελέτη περίπτωσης δυο ομάδων που συμμετείχαν σε δυο παρόμοια εκπαιδευτικά εργαστήρια. Και στα δυο εργαστήρια ακολουθήθηκε η ίδια μέθοδος διδασκαλίας και το ίδιο υλικό, ενώ όλοι οι συμμετέχοντες συμπλήρωσαν τα ίδια ερωτηματολόγια.

7.2 Δείγμα έρευνας

Το δείγμα της έρευνάς μας ήταν 35 μαθητές Γυμνασίου και Λυκείου. Οι μαθητές του δείγματος συμμετείχαν σε δυο εκπαιδευτικά εργαστήρια που διοργανώθηκαν υπό την αιγίδα και σε συνεργασία με το Σύλλογο Εκπαιδευτικών Πληροφορικής Χίου, απευθύνονταν σε μαθητές Δ/θμιας εκπαίδευσης και υλοποιήθηκαν απογευματινές ώρες. Επίσης, και στα δυο εργαστήρια, συμμετείχαν ως παρατηρητές, εκτός του ερευνητή, συνολικά έξι (6) συνάδελφοι εκπαιδευτικοί Πληροφορικής.

Οι συμμετέχοντες στα δυο εκπαιδευτικά εργαστήρια ήταν μαθητές της Γ' τάξης Γυμνασίου, καθώς και των Α' και Β' τάξεων Λυκείου (ΓΕΛ και ΕΠΑΛ). Δεν επιλέχθηκαν μαθητές μικρότερης τάξης από τη Γ' Γυμνασίου, επειδή, με βάση τα τρέχοντα ΠΣ για την Πληροφορική, οι μαθητές έρχονται για πρώτη φορά σε επαφή με τον προγραμματισμό υπολογιστών στη Γ' τάξη Γυμνασίου. Δεν επιλέχθηκαν επίσης μαθητές της Γ' τάξης Λυκείου, κυρίως λόγω του φόρτου για την προετοιμασία τους για τις Πανελλαδικές Εξετάσεις. Εξάλλου, το εύρος περιορίστηκε σκόπιμα, ώστε μην υπάρχει έντονη διαφορά υπόβαθρου και γνώσεων ενός μαθητή Γυμνασίου σε σχέση π.χ. με έναν μαθητή της Τεχνολογικής Κατεύθυνσης της Γ' τάξης Γενικού Λυκείου.

Το **πρώτο** τμήμα του δείγματος αποτελούνταν από 16 μαθητές της Μαθητικής Στέγης Καρδαμύλων Χίου «Το Σπίτι της Μαρίας» του Ιδρύματος «Μαρία Τσάκος». Οι συνολικά 19

άρρηνες φιλοξενούμενοι μαθητές, τάξεων από Γ' Γυμνασίου μέχρι και Β' Λυκείου, προέρχονται από οικογένειες με χαμηλό οικονομικό εισόδημα, αλλά με γενική βαθμολογία άνω του δεκαπέντε. Το πείραμα της έρευνας, διεξήχθη στο πλαίσιο ευρύτερου εκπαιδευτικού σεμιναρίου (για αντικείμενα πληροφορικής που δεν διδάσκονται στο σχολείο). Για το αντικείμενο του App Inventor έγιναν τέσσερις (4) δίωρες απογευματινές συναντήσεις στο εργαστήριο πληροφορικής της Μαθητικής Στέγης, το Φεβρουάριο του 2014.

Το **δεύτερο** τμήμα του δείγματος αποτελούνταν από 19 μαθητές και επιλέχθηκε τυχαία, κατόπιν πρόσκλησης στα Γυμνάσια και Λύκεια της Χίου (με την έγκριση της Διεύθυνσης Δευτεροβάθμιας Εκπαίδευσης Χίου). Από τους 33 μαθητές που εκδήλωσαν ενδιαφέρον, τελικά στο πείραμα συμμετείχαν 21 μαθητές, σε έξι (6) δίωρες απογευματινές συναντήσεις, σε εργαστήριο πληροφορικής σχολικής μονάδας στην πόλη της Χίου. Το εκπαιδευτικό πρόγραμμα αυτό διεξήχθη το Μάρτιο και Απρίλιο του 2014.

7.3 Η δομή του πειράματος

Μέσα από τη διεξαγωγή του πειράματος, επιχειρούμε να εξετάσουμε κατά πόσο είναι εφικτή η αξιοποίηση των κινητών συσκευών για τη βελτίωση και διευκόλυνση της διδασκαλίας και μάθησης βασικών προγραμματιστικών εννοιών στη δευτεροβάθμια εκπαίδευση. Το υλικό που προσφέρεται στον επίσημο δικτυακό τόπο του ΑΙ δεν διατίθεται στην ελληνική γλώσσα, ενώ δεν βρέθηκε διαθέσιμο, εκπαιδευτικό υλικό που να καλύπτει τους απαιτούμενους στόχους της έρευνας. Έτσι, αναπτύχθηκε εκπαιδευτικό υλικό, το οποίο στο μεγαλύτερο μέρος του αποτελείται από πρωτότυπες δραστηριότητες.

Για τη συλλογή των στοιχείων δημιουργήθηκαν επίσης δυο ερωτηματολόγια, εκ των οποίων το πρώτο συμπληρώνεται από τους συμμετέχοντες μαθητές κατά την πρώτη συνάντηση και το άλλο με το τέλος του εργαστηρίου. Με τον τρόπο αυτό ερευνήθηκε η αλλαγή των στάσεων τους ως προς τον προγραμματισμό υπολογιστών, μετά τη χρήση του ΑΙ.

Επιπλέον, έγιναν ημιδομημένες συνεντεύξεις στους εκπαιδευτικούς πληροφορικής που ήταν παρόντες στα εργαστήρια, όπου καταγράφηκαν οι απόψεις τους περί πιθανής αξιοποίησης του ΑΙ προς βελτίωση του διδακτικού τους έργου στο αντικείμενο του προγραμματισμού.

7.3.1 Περιγραφή του πειράματος

Κατά την πρώτη συνάντηση, οι συμμετέχοντες συμπληρώνουν σύντομο ερωτηματολόγιο, όπου καταγράφονται τα βασικά χαρακτηριστικά του δείγματος, η άποψή τους για τον προγραμματισμό κ.α.. Το ερωτηματολόγιο παρατίθεται στο Παράρτημα Β.2.

Σε κάθε συνάντηση γίνεται μια σύντομη εισαγωγή (5-10 λεπτά) από τον εκπαιδευτικό με τη χρήση βιντεοπροβολέα και πίνακα μαρκαδόρου, για τις βασικές έννοιες του μαθήματος μέσα από συζήτηση για τις δραστηριότητες που καλούνται να υλοποιήσουν. Στη συνέχεια, οι μαθητές εργάζονται στους υπολογιστές του εργαστηρίου σε δυάδες, ακολουθώντας τις οδηγίες και τα βήματα που περιγράφονται στα φύλλα εργασίας κάθε μαθήματος. Ο εκπαιδευτικός παρέχει βοήθεια μόνο όταν μια ομάδα μαθητών δεν μπορεί να συνεχίσει, προτρέποντάς την πρώτα να ξαναμελετήσει τα παρεχόμενα φύλλα εργασίας και έπειτα να ρωτήσει κάποια άλλη ομάδα. Σε κάθε περίπτωση, γίνεται προσπάθεια να χρησιμοποιείται η τεχνική του scaffolding, ώστε να γεφυρωθεί το χάσμα ανάμεσα σε αυτό που μπορεί να κάνει ένας μαθητής από μόνος του και σε αυτό που μπορεί να κατορθώσει όταν καθοδηγείται από άλλους.

Στα πρώτα μαθήματα, οι δραστηριότητες και τα βήματα περιγράφονται αναλυτικά, ώστε οι μαθητές να γνωρίσουν και να κατανοήσουν τη διαδικασία σχεδίασης και ανάπτυξης εφαρμογών στο AI, αλλά και για να ανατρέχουν στα συγκεκριμένα σημεία των Φύλλων Εργασίας, όποτε και αν τους χρειαστεί.

Σταδιακά, οι λεπτομέρειες ελαττώνονται και τα βήματα και οι εντολές επεξηγούνται αναλυτικά, μόνο όταν πρόκειται για κάποια καινούρια έννοια. Ο βαθμός δυσκολίας είναι λοιπόν σταδιακά αυξανόμενος, ενώ στις περισσότερες δραστηριότητες, κάποιες ελλείψεις, επεκτάσεις ή βελτιώσεις δίνονται ως άσκηση για τους μαθητές.

Σύντομα, όταν οι μαθητές έχουν εξοικειωθεί με το περιβάλλον και το τμήμα σχεδίασης του περιβάλλοντος των εφαρμογών, πολύ συχνά παρέχεται έτοιμο το κομμάτι αυτό στους μαθητές (πηγαίος κώδικας με τα αρχεία πολυμέσων και τα βασικά αντικείμενα με τις αρχικές ιδιότητές τους), ώστε να εστιάζουν στην ανάπτυξη, επέκταση ή διόρθωση του κώδικα και όχι στη σχεδίαση των εφαρμογών.

Είναι επιθυμητό οι μαθητές να κάνουν δοκιμές και πειραματισμούς στη σχεδίαση του περιβάλλοντος και στην ανάπτυξη του σεναρίου/κώδικα των εφαρμογών που περιγράφονται στα φύλλα εργασίας, τροποποιώντας και παραλλάσσοντας τα χαρακτηριστικά τους, με βάση τις προτιμήσεις και επιθυμίες τους. Οι μαθητές προτρέπονται, μόλις αισθάνονται άνετα με το AI, να βάζουν προσωπικές πινακίδες στις εφαρμογές που αναπτύσσουν ή να επεκτείνουν/παραλλάσσουν τον τρόπο που αυτές λειτουργούν.

Στο τέλος κάθε συνάντησης, οι μαθητές προτρέπονται να υλοποιήσουν όποιες από τις δραστηριότητες για το σπίτι επιθυμούν, ώστε να ταιριάζουν στα ενδιαφέροντά τους.

Στο τέλος του προγράμματος, οι μαθητές συμπληρώνουν το τελικό ερωτηματολόγιο, όπου καταγράφονται ξανά οι απαντήσεις τους σε ερωτήματα που τέθηκαν στο πρώτο ερωτηματολόγιο, αλλά και οι απόψεις τους για το υλικό, για τις βασικές έννοιες προγραμματισμού και τα σημεία που τους άρεσαν/δυσκόλευσαν, κ.α.. Το ερωτηματολόγιο παρατίθεται στο Παράρτημα Β.3.

7.3.2 Ο ρόλος των εκπαιδευτικών

Από τη φάση του σχεδιασμού του πειράματος, προτού ακόμα ξεκινήσει το πείραμα, αποφασίσαμε τα εκπαιδευτικά προγράμματα να υλοποιηθούν συνεργατικά, τόσο ως προς την παραγωγή του υλικού, όσο και ως προς την διεξαγωγή τους. Σε κάθε συνάντηση με τους μαθητές, επιδιώχθηκε να παρευρίσκονται ταυτόχρονα, τουλάχιστον δυο εκπαιδευτικοί πληροφορικής.

Οι συνεργαζόμενοι εκπαιδευτικοί παρουσιάζουν σύντομα τα βασικά σημεία του κάθε μαθήματος και επιτρέπουν σε κάθε ομάδα μαθητών να συνεχίσει με το δικό της ρυθμό τη μελέτη των φύλλων εργασίας και να ολοκληρώσει τις εκπαιδευτικές δραστηριότητες. Βοήθεια από τον εκπαιδευτικό παρέχεται μόνο όταν μια ομάδα μαθητών δεν μπορεί να συνεχίσει, προτρέποντάς την, να ξαναμελετήσει τα παρεχόμενα φύλλα εργασίας ή εναλλακτικά να ρωτήσει κάποια άλλη ομάδα.

Παράλληλα, οι εκπαιδευτικοί, παρακολουθούν διακριτικά τους μαθητές, καθ' όλη τη διάρκεια των συναντήσεων, καταγράφοντας σε φύλλα παρατήρησης (ενδεικτικά δείγματα παρατίθενται

στο Παράρτημα Β.4) την πορεία των μαθητών, τις συνηθέστερες απορίες, παρανοήσεις, λάθη, ιδέες και πρωτοβουλίες τους.

Επομένως, ο ρόλος του εκπαιδευτικού είναι διπλός: συντονίζει και υποστηρίζει τη μαθησιακή διαδικασία, παρακινώντας και οδηγώντας έμμεσα και διακριτικά τη σκέψη των μαθητών στο σωστό δρόμο, προετοιμάζει το έδαφος για τις δραστηριότητες που πρόκειται να πραγματοποιηθούν, ενώ ταυτόχρονα, καλείται να παίξει το ρόλο του παρατηρητή και ερευνητή, συλλέγοντας, επεξεργάζοντας και αξιολογώντας δεδομένα και ευρήματα.

7.3.3 Ο ρόλος των μαθητών

Οι μαθητές παροτρύνθηκαν από τον εκπαιδευτικό να δημιουργήσουν διμελείς ομάδες, τόσο για να επιτευχθεί καλύτερη συνεργασία και επικοινωνία μεταξύ των μελών, όταν θα μελετούν το εκπαιδευτικό υλικό και θα υλοποιούν τις δραστηριότητες, όσο και λόγω περιορισμένου αριθμού διαθέσιμων υπολογιστών. Η στελέχωση της κάθε ομάδας έγινε με βάση τις προτιμήσεις των μαθητών, αλλά σε ελάχιστες περιπτώσεις, η σύσταση των ομάδων τροποποιήθηκε, για να επιτευχθεί μαθησιακή ανομοιογένεια μεταξύ των μελών τους. Κάθε ομάδα ακολούθησε τη δική της μαθησιακή πορεία με βάση τις εναλλακτικές επιλογές που παρείχαν οι εκπαιδευτικοί και προχώρησε στην υλοποίηση των δραστηριοτήτων με το δικό της ρυθμό.

Οι μαθητές δραστηριοποιήθηκαν σε ένα συνεργατικό περιβάλλον μάθησης, καθώς πέρα από τα ζευγάρια εργασίας ανά υπολογιστή (pair programming), κατά διαστήματα κλήθηκαν να σχηματίσουν μεγαλύτερες ομάδες και να συζητήσουν με τους συμμαθητές τους. Αναλυτικότερα, οι μαθητές καλούνται να μελετήσουν το υλικό που τους δίνεται, να επικοινωνήσουν και να συνεργαστούν, να βρουν τις πληροφορίες που χρειάζονται και να τις αξιοποιήσουν κατάλληλα. Λόγω της δημιουργικής μορφής των δραστηριοτήτων, οι εκπαιδευόμενοι προκαλούνται και συμμετέχουν ενεργά στη μαθησιακή διαδικασία. Στόχος δεν είναι να αντιγράψουν την ενδεικτική επίλυση μιας δραστηριότητας, αλλά να αναλάβουν πρωτοβουλίες. Είναι στη δική τους ευχέρεια να πειραματιστούν, να υποπέσουν σε σφάλματα, να κάνουν υποθέσεις και να τις αποδείξουν ή να τις απορρίψουν, κάτι που ευνοείται από το AI, καθώς επιτρέπει την άμεση δοκιμή των εφαρμογών και παρέχει ανατροφοδότηση προς τους μαθητές, σχεδόν σε πραγματικό χρόνο. Οι εκπαιδευόμενοι νιώθοντας ότι αυτοί ορίζουν το πως θα συντελεστεί η μάθησή τους, αποκτούν

αυτοπεποίθηση και εμπιστοσύνη στον εαυτό τους, χωρίς την ανάγκη συνεχούς καθοδήγησης και επιβεβαίωσης από τον καθηγητή.

7.3.4 Προετοιμασία των χώρων διεξαγωγής

Πριν την έναρξη των προγραμμάτων έγινε έλεγχος των εργαστηρίων πληροφορικής, ώστε να πληρούν τις ελάχιστες απαιτούμενες προδιαγραφές για τη χρήση του AI, όπως αναφέρθηκαν στην Παράγραφο 6.2. Προληπτικά, έγινε εγκατάσταση του λογισμικού που παρέχει την εικονική μηχανή Android - προσομοιωτή (emulator), σε περίπτωση που δεν υπήρχε επαρκής αριθμός συμβατών συσκευών. Και τα δυο εργαστήρια διέθεταν βιντεοπροβολέα και ικανό αριθμό σταθερών υπολογιστών (τουλάχιστον 10) συνδεδεμένων σε τοπικό δίκτυο με πρόσβαση στο διαδίκτυο, ενώ πολύ βοηθητική ήταν η ύπαρξη ασύρματου δικτύου. Εάν δεν υπήρχε τέτοιο δίκτυο, θα τοποθετούσαμε προσωρινά δική μας συσκευή Ασύρματου Σημείου Πρόσβασης (Wireless Access Point) προς διευκόλυνση της διαδικασίας.

Πριν την έναρξη των μαθημάτων, είχαν δοθεί αναλυτικές οδηγίες στους επιλεγμένους μαθητές για τη δημιουργία του απαραίτητου λογαριασμού (Google Account) και την εγκατάσταση δυο βοηθητικών εφαρμογών στις Android συσκευές τους, του MIT AI2 Companion και ενός QR Code scanner (Παράρτημα A.5).

Με αυτό τον τρόπο, η δοκιμή και εγκατάσταση των παραγόμενων εφαρμογών έγινε εύκολα, μέσω σύνδεσης των συσκευών στο ασύρματο δίκτυο Wi-Fi των εργαστηρίων.

7.4 Το εκπαιδευτικό υλικό

Προτού παρουσιαστεί συνοπτικά το υλικό που χρησιμοποιήθηκε, παρουσιάζουμε το σκεπτικό ανάπτυξής του, τους στόχους του καθώς και τον τρόπο με τον οποίο χρησιμοποιήθηκε για τις ανάγκες του πειράματος.

7.4.1 Ανάπτυξη εκπαιδευτικού υλικού

Η ανάπτυξη διδακτικού υλικού που να είναι ελκυστικό για τους μαθητές, αποτελεί εκπαιδευτική προτεραιότητα στις μέρες μας. Με βάση τους στόχους των ΑΠΣ στα μαθήματα πληροφορικής, δίνεται η δυνατότητα στον εκπαιδευτικό να συνθέσει ο ίδιος ψηφιακό υλικό για χρήση την τάξη και διαμοιρασμό στην υπόλοιπη εκπαιδευτική κοινότητα, στηρίζοντας τη μάθηση των μαθητών με διαφορετικές εμπειρίες και χαρακτηριστικά, με βάση την υλικοτεχνική υποδομή που διατίθεται κάθε φορά.

Για τις ανάγκες του πειράματος αναπτύχθηκε εκπαιδευτικό υλικό, ώστε να αξιολογηθεί η επίδραση και η συνεισφορά του App Inventor στη διδασκαλία και μάθηση βασικών εννοιών προγραμματισμού από αρχάριους στον προγραμματισμό. Το ψηφιακό υλικό που αναπτύξαμε απευθύνεται σε μαθητές της δευτεροβάθμιας εκπαίδευσης (τελευταία τάξη Γυμνασίου, Γενικό και Επαγγελματικό Λύκειο) και έχει ως στόχο την εισαγωγή των εκπαιδευομένων στην διδασκαλία του προγραμματισμού και την καλύτερη κατανόηση εννοιών και δομών που αναφέρονται στα αντίστοιχα Αναλυτικά Προγράμματα Σπουδών.

Μάλιστα, προκειμένου τα εκπαιδευτικά εργαστήρια (workshops) που υλοποιήθηκαν, να αποτελούν μια κατά το δυνατόν πληρέστερη γνωριμία με το AI και τις δυνατότητές του και να δώσουν περαιτέρω κίνητρα στους συμμετέχοντες να συνεχίσουν να το χρησιμοποιούν, οι δραστηριότητες που αναπτύχθηκαν, επιχειρούν να αξιοποιήσουν πολλά ιδιαίτερα χαρακτηριστικά των ψηφιακών φορητών συσκευών, αλλά και του ίδιου του AI.

Ενδεικτικά, το εκπαιδευτικό υλικό αναφέρεται στη χρήση των αισθητήρων θέσης (GPS - Location Sensor), προσανατολισμού (Orientation Sensor) και επιτάχυνσης (Acceleration Sensor), της φωτογραφικής μηχανής (Camera), της πρόσβασης στις επαφές (Contacts Picker), την αποστολή μηνυμάτων SMS (Texting), τη μετατροπή κειμένου σε ομιλία (TTS - Text To Speech), τις λίστες (Lists), την αποθήκευση δεδομένων σε τοπική βάση δεδομένων (TinyDB) και το άνοιγμα άλλων εφαρμογών μέσα από το AI (Activity Starter). Όλα αυτά γίνονται σταδιακά, αξιοποιώντας βασικές έννοιες και προγραμματιστικές δομές, όπως μεταβλητές, διαδικασίες, τις δομές ακολουθίας, επιλογής και επανάληψης, αλλά και δομών δεδομένων.

7.4.2 Στόχοι διδασκαλίας

Στην προσπάθεια να διερευνηθεί σε ποιους τομείς συμβάλλει το AI στην αποτελεσματικότερη μάθηση βασικών προγραμματιστικών εννοιών από τους μαθητές και που όχι, έχουν τεθεί συγκεκριμένοι στόχοι. Έτσι, ως αποτέλεσμα της εκπαιδευτικής διαδικασίας, αναμένεται να

δημιουργηθούν οι κατάλληλες συνθήκες και ένα μαθησιακό περιβάλλον τέτοιο, ώστε μετά την ολοκλήρωση των εργαστηρίων να επιτευχθούν οι παρακάτω γνωστικοί και παιδαγωγικοί στόχοι για τους μαθητές:

Παιδαγωγικοί στόχοι

- Να μπορούν να αναλαμβάνουν πρωτοβουλίες, να πειραματίζονται, να αποτινάξουν το φόβο του σφάλματος και να μαθαίνουν στην πράξη, κάνοντας πράγματα.
- Να είναι σε θέση να διασπάσουν ένα πρόβλημα σε απλούστερα κομμάτια, να βρουν τις επιμέρους λύσεις του και να μπορούν να συνθέτουν αυτές τις λύσεις σε μία ολοκληρωμένη.

Γενικοί διδακτικοί - μαθησιακοί στόχοι

- Να αναπτύξουν βασικές δεξιότητες χρήσης ενός προγραμματιστικού περιβάλλοντος όπως το App Inventor.
- Να αναγνωρίζουν βασικές δομές και έννοιες του προγραμματισμού.
- Να είναι σε θέση να διαβάζουν και να επεξηγούν τον κώδικα ενός προγράμματος.
- Να ενισχυθεί το ενδιαφέρον τους για θέματα που αφορούν στον προγραμματισμό.

Ειδικοί διδακτικοί - μαθησιακοί στόχοι

Γενικός στόχος είναι η απόκτηση δεξιοτήτων ανάπτυξης εφαρμογών για Android συσκευές. Ειδικότερα, ως προς τον Προγραμματισμό, οι μαθητές προσδοκείται:

- Γεγονότα/συμβάντα:
 - Να ορίζουν ενέργειες και σενάρια που πρέπει να εκτελεστούν για την επίτευξη επιθυμητών γεγονότων.
 - Να συσχετίζουν γεγονότα με τις αντίστοιχες εντολές.
 - Να ορίζουν ιδιότητες και συμπεριφορά σε αντικείμενα.

- Μεταβλητές:
 - Να αναγνωρίζουν τη χρησιμότητα των μεταβλητών.
 - Να ορίζουν μεταβλητές, όπου απαιτείται.
 - Να αποδίδουν τιμές στις μεταβλητές κατά την εκτέλεση των εφαρμογών που αναπτύσσουν.

- Αριθμητικές και λογικές εκφράσεις:
 - Να χρησιμοποιούν τις λογικές τιμές TRUE/FALSE.
 - Να χρησιμοποιούν αριθμητικούς και λογικούς τελεστές για τη σύνταξη προτάσεων.

- Δομή επιλογής:
 - Να αναγνωρίζουν την αναγκαιότητα και τη χρησιμότητα των δομών επιλογής.
 - Να συντάσσουν τις κατάλληλες συνθήκες ελέγχου.
 - Να κάνουν χρήση των δομών απλής και πολλαπλής επιλογής.

- Διαδικασίες:
 - Να απαριθμούν τα πλεονεκτήματα από τη χρήση υποπρογραμμάτων.
 - Να διαχωρίζουν ένα πρόβλημα σε υποπροβλήματα.
 - Να δημιουργούν και να χρησιμοποιούν διαδικασίες (με ή χωρίς παραμέτρους).

Οι εκπαιδευτικοί/ερευνητές σε συνεργασία με τους μαθητές αναμένεται να αξιολογήσουν το βαθμό επίτευξης αυτών των στόχων, με βάση τα ερευνητικά ερωτήματα που τέθηκαν στο Κεφάλαιο 1. Βέβαια, λόγω της περιορισμένης χρονικής διάρκειας των εργαστηρίων, ενδέχεται ο

σχεδιασμός να εστιάσει στη διερεύνηση της επίτευξής του μεγαλύτερου δυνατού ποσοστού από τους αρχικούς στόχους.

7.4.3 Περιεχόμενο μαθημάτων

Το εκπαιδευτικό υλικό που αναπτύχθηκε, στο μεγαλύτερο μέρος του αποτελείται από πρωτότυπες δραστηριότητες και παρέχεται στην εκπαιδευτική κοινότητα με άδεια Creative Commons Αναφορά Δημιουργού - Παρόμοια Διανομή 4.0 ⁶³ από την ιστοσελίδα του Συλλόγου Εκπαιδευτικών Πληροφορικής Χίου (<http://www.sepchiou.gr>).

Οι στόχοι και οι δραστηριότητες των τριών πρώτων μαθημάτων, που καλύπτουν τους εξεταζόμενους στόχους της Μεταπτυχιακής διατριβής, περιγράφονται συνοπτικά στο Παράρτημα Α. Επισημαίνεται ότι ο τίτλος «Μάθημα» δεν υπονοεί ότι κάθε μάθημα προβλεπόταν να ολοκληρωθεί σε μια συνάντηση, αλλά ότι ο διαχωρισμός τους έγινε με βάση τους ευρύτερους στόχους τους και τις έννοιες που πραγματεύτηκαν.

Στο **1^ο μάθημα** γίνεται μια εισαγωγή στο περιβάλλον του App Inventor και διάκριση της σχεδίασης της οθόνης μιας εφαρμογής από το αντίστοιχο τμήμα εντολών της. Οι μαθητές αναγνωρίζουν την ύπαρξη απλών γεγονότων που «πυροδοτούν» την εκτέλεση κάποιων ενεργειών, οι οποίες σχετίζονται με τα διαθέσιμα αντικείμενα και τις ιδιότητές τους, μέσα από τη σχεδίαση και ανάπτυξη απλών εφαρμογών, για τις οποίες παρέχονται αναλυτικά βήματα.

Στο **2^ο μάθημα** οι μαθητές έρχονται σε επαφή με την έννοια της μεταβλητής και της χρησιμότητάς της, μέσα από τη δημιουργία απλών εφαρμογών όπου απαιτείται η αποθήκευση τιμών και η επεξεργασία τους, όπως και η καταμέτρηση συμβάντων. Στη συνέχεια, παίρνουν αποφάσεις για ενέργειες που επιθυμούν να εκτελεστούν, συντάσσοντας και ελέγχοντας την τιμή λογικών προτάσεων.

Στο **3^ο μάθημα** οι μαθητές αντιλαμβάνονται τα πλεονεκτήματα της δημιουργίας και χρήσης απλών διαδικασιών στις εφαρμογές που αναπτύσσουν. Σταδιακά, μαθαίνουν να δημιουργούν και να καλούν διαδικασίες με παραμέτρους, ενώ παράλληλα εφαρμόζουν έννοιες που έχουν γνωρίσει στα προηγούμενα μαθήματα για τη δημιουργία απλών παιχνιδιών.

⁶³ Άδεια χρήσης CC-BY-SA: <http://creativecommons.org/licenses/by-sa/4.0>

Στο **4^ο μάθημα**, το οποίο αποτέλεσε προαιρετικό αντικείμενο για όσους θα ολοκλήρωναν τις δραστηριότητες των προηγούμενων μαθημάτων, οι μαθητές έρχονται σε επαφή με τη δομή επανάληψης, με τη λίστα ως βασική δομή δεδομένων στο AI και με τη μόνιμη αποθήκευση δεδομένων σε τοπική βάση δεδομένων.

Με την ολοκλήρωση των τεσσάρων παραπάνω μαθημάτων, οι μαθητές αναμένουμε ότι θα είναι ικανοί να σχεδιάζουν και να υλοποιούν ένα μεγάλο εύρος εφαρμογών δικής τους επιλογής, χωρίς να απαιτούνται ιδιαίτερες επιπρόσθετες γνώσεις, παρά εφαρμογή και επέκταση των ήδη γνώριμων εννοιών.

Το πλήρες πακέτο του εκπαιδευτικού υλικού που διατίθεται στη διεύθυνση <http://www.sepchiou.gr/ai>, συνοδεύεται από τα αρχεία πολυμέσων που αναφέρονται σε αυτό, αλλά και από έργα έτοιμα για εισαγωγή στο AI, όταν η εστίαση είναι στην ανάπτυξη, διόρθωση ή επέκταση του ατελή κώδικα και όχι στη σχεδίαση της διεπαφής των εφαρμογών. Τα παραπάνω αρχεία (πολυμέσων και πηγαίου κώδικα) διατέθηκαν στο διαδίκτυο μέσω ενός δημόσιου φάκελου Dropbox, ενώ το υλικό των μαθημάτων εκτυπώθηκε σε μορφή βιβλίου (σε δυο τεύχη) και διανεμήθηκε στους μαθητές.

Ενδεικτικά, στο Παράρτημα A.4 παρατίθεται μια δραστηριότητα για την ανάπτυξη μιας εφαρμογής πυξίδας, αξιοποιώντας τη δομή επιλογής, όπως δόθηκε στους μαθητές στο 3^ο μάθημα.

7.5 Μέσα και μέθοδοι συλλογής δεδομένων

Στη συνέχεια, παρουσιάζονται αναλυτικότερα τα μέσα που χρησιμοποιήθηκαν για τη συλλογή των δεδομένων και η μέθοδος που χρησιμοποιήθηκε για την οργάνωση, επεξεργασία και παρουσίασή τους. Όπως έχει ήδη αναφερθεί, η διενέργεια των εκπαιδευτικών εργαστηρίων συνοδεύτηκε από τη συμπλήρωση ερωτηματολογίων από τους συμμετέχοντες μαθητές, από την καταγραφή παρατηρήσεων μέσω της φυσικής παρουσίας των εκπαιδευτικών και από τη διενέργεια προσωπικών συνεντεύξεων από εκπαιδευτικούς πληροφορικής που παρευρίσκονταν σε αυτά.

7.5.1 Ερωτηματολόγια

Με βάση τα ερευνητικά ερωτήματα και τους μαθησιακούς στόχους των μαθημάτων του εργαστηρίου, δημιουργήθηκαν τα κατάλληλα ερωτηματολόγια, εκ των οποίων το πρώτο συμπληρώνεται από τους συμμετέχοντες μαθητές κατά την πρώτη συνάντηση και το άλλο με το τέλος του εργαστηρίου. Έτσι, καταγράφονται οι απόψεις των μαθητών για το εκπαιδευτικό πρόγραμμα και τις δραστηριότητες, καθώς και για το ΑΙ, και μελετάται πιθανή αλλαγή στις στάσεις τους για τον προγραμματισμό υπολογιστών.

Η συμπλήρωση των ερωτηματολογίων έγινε στην αρχή και στο τέλος των εκπαιδευτικών εργαστηρίων κάθε ομάδας μαθητών, με κάθε μαθητή να χρησιμοποιεί συγκεκριμένο κωδικό κατά τη συμπλήρωσή τους, ώστε να είναι εφικτή η σύνδεση των ερωτηματολογίων μεταξύ τους. Οι μαθητές που έλειψαν σε περισσότερες από δυο συναντήσεις, με βάση τα τηρούμενα παρουσιολόγια, δεν συμπεριλήφθηκαν στο τελικό δείγμα. Τα ερωτηματολόγια αποφασίστηκε να διατεθούν σε ηλεκτρονική μορφή για εξοικονόμηση του κόστους εκτυπώσεων και του χρόνου ψηφιοποίησης των απαντήσεων. Συγκεκριμένα, επιλέχθηκε η χρήση της δωρεάν υπηρεσίας Google Forms, η οποία υποστηρίζει όλους τους απαιτούμενους τύπους ερωτήσεων και επιτρέπει την άμεση εξαγωγή των δεδομένων σε επεξεργάσιμη μορφή (.csv και .xlsx).

Το πρώτο ερωτηματολόγιο περιείχε δημογραφικές ερωτήσεις πολλαπλών επιλογών, αλλά και ερωτήσεις που χρησιμοποίησαν κλίμακα τύπου Likert με 4 ή 5 διαβαθμίσεις. Οι δημογραφικές ερωτήσεις κατέγραψαν τις πρότερες γνώσεις των μαθητών στον προγραμματισμό, τις απόψεις και τη διάθεσή τους απέναντι στο αντικείμενο, αλλά και την πληροφορική γενικότερα, καθώς και το βαθμό εξοικείωσής τους με τις ψηφιακές φορητές συσκευές (Παράρτημα Β.2).

Το δεύτερο ερωτηματολόγιο περιείχε αρκετές κοινές ερωτήσεις με το πρώτο, ώστε να μπορεί να διερευνηθεί πιθανή αλλαγή στη στάση των μαθητών πριν και μετά τη συμμετοχή τους στο πείραμα. Περιλάμβανε επίσης ερωτήσεις για την καταγραφή των προγραμματιστικών εννοιών που αξιοποίησαν οι συμμετέχοντες, της άποψής τους για το ΑΙ, των θετικών και αρνητικών του σημείων, των δραστηριοτήτων που υλοποίησαν και του βαθμού δυσκολίας τους (Παράρτημα Β.3).

Για την εξασφάλιση της εγκυρότητας των ερωτηματολογίων, αφού πρώτα έγινε προσπάθεια να εντοπιστούν οι διαστάσεις που συνθέτουν τις προς μελέτη μεταβλητές, έγινε ο μετασχηματισμός τους σε προτάσεις/ερωτήσεις και δόθηκαν δοκιμαστικά για συμπλήρωση από τέσσερις μαθητές ανάλογων χαρακτηριστικών με το δείγμα της έρευνας, οι οποίοι ήταν γνωστοί στον ερευνητή και δεν συμμετείχαν στο πείραμα. Από τις απαντήσεις των μαθητών και μέσα από συζήτηση μαζί τους, κρίναμε ότι οι ερωτήσεις μετράνε με σαφήνεια τις ζητούμενες έννοιες.

Η αξιοπιστία των κλιμάκων μέτρησης των ερωτηματολογίων αξιολογήθηκε ως προς την ομογένεια των ερωτήσεων, χρησιμοποιώντας τον συντελεστή Cronbach's alpha. Συγκεκριμένα, όσο μεγαλύτερη από 0.7 είναι η τιμή του συντελεστή αυτού, τόσο μεγαλύτερη είναι η αξιοπιστία εσωτερικής συνοχής. Η μέση τιμή του συντελεστή για τις ομάδες των ερωτήσεων είναι 0.713 (επιμέρους τιμές 0.764 - 0.788 - 0.588), εξαιτίας όμως του σχετικά μικρού μεγέθους του δείγματος (N=35), η μέτρηση δεν μπορεί να θεωρηθεί 100% αξιόπιστη.

Επισημαίνεται ότι οι μαθητές που κλήθηκαν να συμμετάσχουν στα εργαστήρια του πειράματος, για λόγους ηθικής και δεοντολογίας, εξ αρχής ενημερώθηκαν για το πλαίσιο διεξαγωγής των εκπαιδευτικών προγραμμάτων και προσκόμισαν ειδική Δήλωση Συγκατάθεσης συμπληρωμένη από τον κηδεμόνα τους (Παράρτημα Β.1).

7.5.2 Παρατήρηση

Η συλλογή των δεδομένων από παρατηρήσεις έγινε με φυσική παρουσία του ερευνητή στις συναντήσεις με τους συμμετέχοντες μαθητές, καθ' όλο το χρονικό διάστημα διεξαγωγής των δυο εργαστηρίων, δηλαδή από το Φεβρουάριο μέχρι τον Απρίλιο του 2014. Στο ρόλο του ερευνητή και παρατηρητή ήταν πάντοτε οι δυο εκπαιδευτικοί που είχαν παράλληλα το ρόλο του διδάσκοντα. Τηρήθηκε ημερολόγιο όπου καταγράφηκαν παρατηρήσεις και σχόλια για τη μαθησιακή διαδικασία, τις αντιδράσεις των μαθητών και τις ιδέες τους, ενώ επισημάνθηκαν και σημεία στα φύλλα εργασίας που έχρηζαν διορθώσεων. Παράλληλα, κατά την επιτόπια παρατήρηση και συζήτηση με τις ομάδες μαθητών, τόσο κατά τη διάρκεια, όσο και πριν την έναρξη και στο τέλος κάθε μαθήματος, συμπληρώθηκαν φύλλα παρατήρησης για το βαθμό επίτευξης των στόχων και τα δύσκολα σημεία κάθε μαθήματος (Παράρτημα Β.4).

7.5.3 Συνεντεύξεις

Οι εμπλεκόμενοι εκπαιδευτικοί πληροφορικής, μετά την ολοκλήρωση των εργαστηρίων, κλήθηκαν να εκθέσουν την άποψή τους για το ΑΙ και την προοπτική αξιοποίησής του για τη διευκόλυνση της διδασκαλίας και μάθησης βασικών προγραμματιστικών εννοιών, με τη διεξαγωγή ημιδομημένων συνεντεύξεων. Συγκεκριμένα, από τους έξι εκπαιδευτικούς πληροφορικής που συμμετείχαν στη διεξαγωγή των εργαστηρίων, τρεις ανέλαβαν υποστηρικτικούς ρόλους και άλλοι τρεις έλαβαν μέρος ως εκπαιδευόμενοι, ερχόμενοι για πρώτη φορά σε επαφή με το περιβάλλον του ΑΙ (Παράρτημα Β.5).

Τα δεδομένα που συλλέχθηκαν, χρησιμοποιώντας τα παραπάνω μέσα και μεθόδους, παρουσιάζονται αναλυτικά στο επόμενο κεφάλαιο.

Κεφάλαιο 8

Αποτελέσματα και συζήτηση

Στο κεφάλαιο αυτό παρουσιάζονται τα ευρήματα της συλλογής δεδομένων από τη διεξαγωγή του πειράματος, γίνεται συζήτηση των αποτελεσμάτων και μια αποτίμηση του εγχειρήματος.

8.1 Εισαγωγή

Πριν την παρουσίαση των ευρημάτων από τη διεξαγωγή των δυο εργαστηρίων, στο σημείο αυτό αξίζει να επισημανθεί μια σημαντική διαφορά μεταξύ των δυο τμημάτων. Οι μαθητές του πρώτου τμήματος, στο πλαίσιο των υποχρεώσεών τους στη Μαθητική Στέγη, υποχρεούνται να τηρούν το προβλεπόμενο ημερήσιο πρόγραμμα. Έτσι, τα απογεύματα του Σαββάτου, οπότε και πραγματοποιήθηκαν οι συναντήσεις, ουσιαστικά «θυσίασαν» την έξοδό τους για να συμμετάσχουν υποχρεωτικά στο πρόγραμμα. Αντιθέτως, παρόλο που οι συναντήσεις με το δεύτερο πρόγραμμα, πραγματοποιήθηκαν και αυτές απογεύματα Σαββάτου, οι μαθητές δήλωσαν εθελοντικά συμμετοχή στο πρόγραμμα και η ώρα και μέρα των συναντήσεων καθορίστηκε από κοινού μέσα από συζήτηση των συμμετεχόντων. Μάλιστα, η αίτηση συμμετοχής στο συγκεκριμένο πρόγραμμα, συμπληρώθηκε αρχικά από 33 μαθητές, κάτι που

φανερώνει το έντονο ενδιαφέρον τους ή την περιέργειά τους για το αντικείμενο. Μετά την πρώτη συνάντηση, βέβαια, όπου συζητήθηκε και καθορίστηκε η τελική ώρα και μέρα των συναντήσεων, υπήρξαν εξ αρχής αρκετές απώλειες, κυρίως λόγω υποχρεώσεων των μαθητών, με αποτέλεσμα να συμμετάσχουν τελικά 21 μαθητές.

Επίσης, σημειώνεται ότι σε κανένα από τα δυο τμήματα δεν χρειάστηκε να χρησιμοποιηθεί ο εγκατεστημένος Android προσομοιωτής στους υπολογιστές των εργαστηρίων, καθώς κάθε ομάδα μαθητών διέθετε τουλάχιστον μια συμβατή συσκευή (τηλέφωνο ή ταμπλέτα). Σε ότι αφορά τη διαδικασία προετοιμασίας των μαθητών, με βάση τις οδηγίες που τους είχαν δοθεί πριν την έναρξη των μαθημάτων, κανένας μαθητής δεν αντιμετώπισε πρόβλημα στη δημιουργία του Λογαριασμού Google. Ελάχιστοι μαθητές αντιμετώπισαν προβλήματα στην εγκατάσταση της εφαρμογής MIT AI2 Companion από το Google Play Store, οπότε αναγκάστηκαν να εγκαταστήσουν την εφαρμογή χειροκίνητα, κατεβάζοντας στη συσκευή τους το αντίστοιχο .apk αρχείο.

Τα δεδομένα που συλλέχθηκαν από τη συμπλήρωση των online ερωτηματολογίων στο Google Drive, ελέγχθηκαν και κωδικοποιήθηκαν εκ νέου, προκειμένου να γίνει η στατιστική επεξεργασία τους, χρησιμοποιώντας το λογισμικό IBM SPSS Statistics 22 σε συνδυασμό με το λογισμικό υπολογιστικών φύλλων Microsoft Excel 2013.

Η ανάλυση των δεδομένων περιλάμβανε αρχικά περιγραφική στατιστική ανάλυση και παρουσίαση των αποτελεσμάτων (συχνότητες κατανομής, μέσες τιμές και τυπική απόκλιση). Για τη μελέτη του τρόπου και του βαθμού επίδρασης του AI στις απόψεις των μαθητών, αξιοποιώντας τεχνικές ελέγχου υποθέσεων, μελετήθηκαν συσχετίσεις και ελέγχθηκε η ύπαρξη στατιστικά σημαντικών διαφορών (επίπεδο σημαντικότητας 0.05) πριν και μετά την παρέμβαση με τη διεξαγωγή του πειράματος στις ομάδες.

Σημειώνεται ότι για λόγους συντομίας, όποτε γίνεται σχετική αναφορά, χρησιμοποιούμε τον κωδικό K για την ομάδα των μαθητών που συμμετείχαν στο πρώτο εκπαιδευτικό εργαστήριο που διεξήχθη στη Μαθητική Στέγη των Καρδαμύλων της Χίου και τον κωδικό X για την ομάδα των μαθητών που συμμετείχαν στο δεύτερο εκπαιδευτικό εργαστήριο που διεξήχθη στην πόλη της Χίου. Επισημαίνεται πως, εξαιτίας χρονικών περιορισμών και υποχρεώσεων των μαθητών της ομάδας K, στο εν λόγω τμήμα το εκπαιδευτικό εργαστήριο διήρκεσε τέσσερις ώρες λιγότερο από εκείνο της ομάδας X. Επιπλέον, λόγω και του αργότερου ρυθμού με τον οποίο προχώρησαν οι συμμετέχοντες στην υλοποίηση των δραστηριοτήτων, στο πρώτο τμήμα (K), ολοκληρώθηκαν

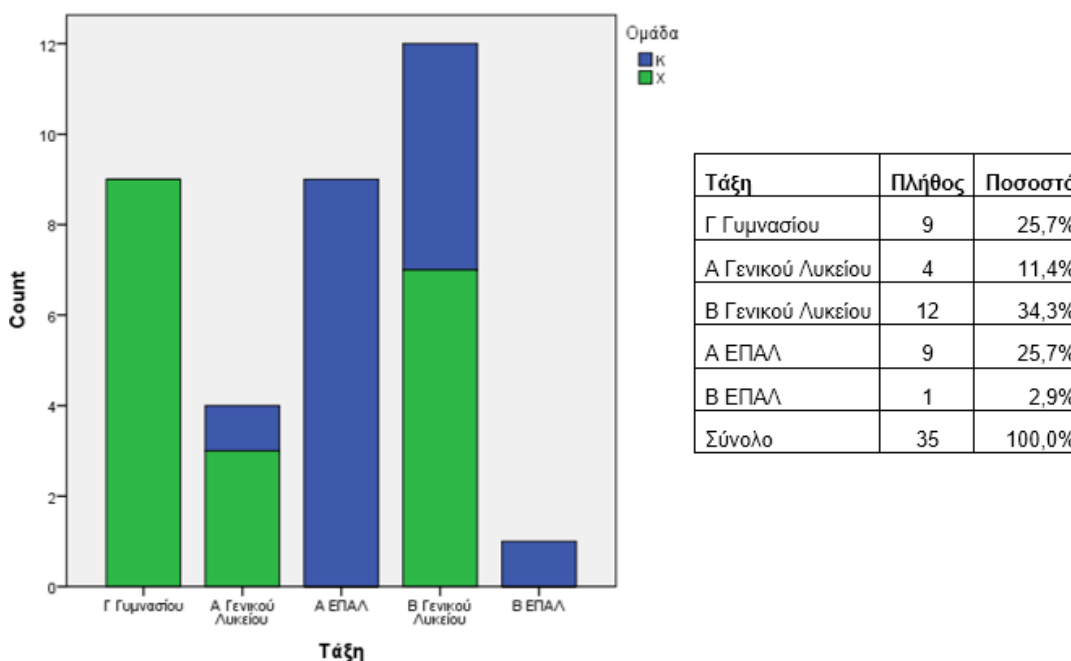
μόνο τα δυο πρώτα μαθήματα (σε 8 ώρες), σε αντίθεση με το δεύτερο τμήμα (X), όπου ολοκληρώθηκαν και τα τρία μαθήματα (σε 12 ώρες), ενώ κάποιοι μαθητές προχώρησαν από μόνοι τους σε ορισμένες δραστηριότητες του τέταρτου μαθήματος.

8.2 Δημογραφικά στοιχεία

Ως προς το προφίλ των υποκειμένων του πειράματος, είχαμε 35 μαθητές (32 αγόρια και 3 κορίτσια) από Γ' Γυμνασίου έως και Β' ΓΕΛ/ΕΠΑΛ. Επειδή το δείγμα μας περιλάμβανε μόνο τρία κορίτσια, δεν μελετήσαμε τελικά την επίδραση του ΑΙ ως προς το φύλο.

Τάξη

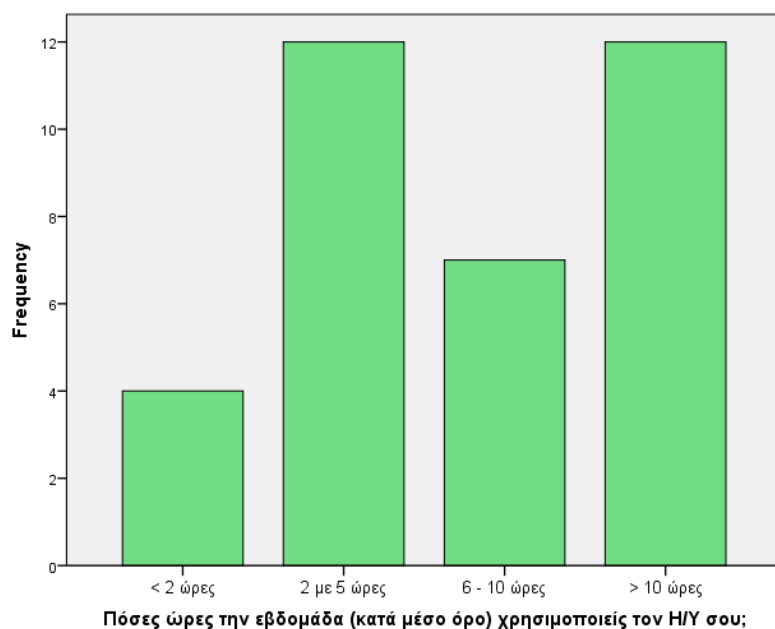
Το 25,7 % του δείγματος είναι μαθητές Γυμνασίου, ενώ το υπόλοιπο 74,3 % είναι μαθητές Λυκείου (Γενικού ή ΕΠΑΛ).



Διάγραμμα 8.1: Τάξη δείγματος.

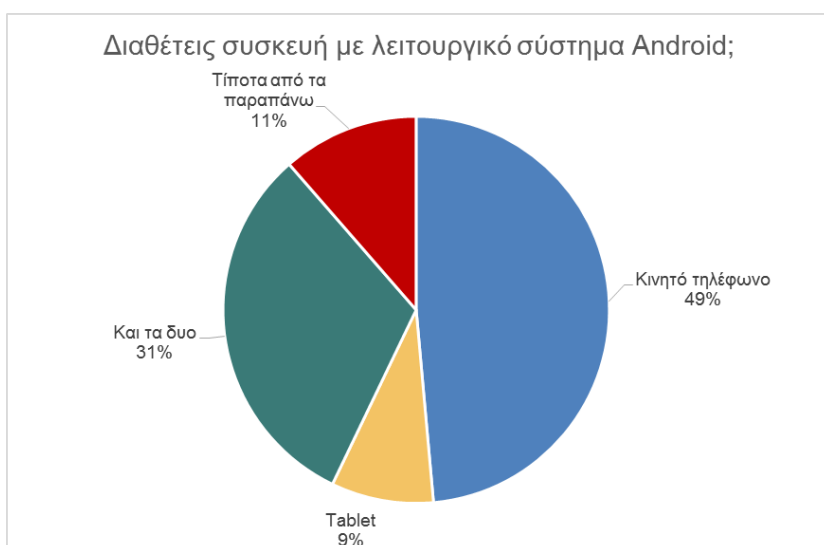
Κατοχή και χρήση Η/Υ και συσκευών Android

Το 88,7% του δείγματος διαθέτει Η/Υ και τον χρησιμοποιεί καθημερινά.



Διάγραμμα 8.2: Εβδομαδιαία χρήση Η/Υ.

Ως προς την κατοχή συσκευών με Android, το 49% διαθέτει τηλέφωνο, το 9% διαθέτει μόνο ταμπλέτα, ενώ το 11% (4 συμμετέχοντες) δεν διαθέτει συσκευή με Android. Το 31% του δείγματος διαθέτει τόσο τηλέφωνο, όσο και ταμπλέτα με Android, ανεβάζοντας το συνολικό ποσοστό κατόχων κινητών τηλεφώνων με Android στο 80%. Επίσης, το 88% των συμμετεχόντων έχει κατεβάσει και εγκαταστήσει εφαρμογή ή παιχνίδι στη συσκευή του από το Google Play.



Διάγραμμα 8.3: Κατοχή συσκευής με λειτουργικό σύστημα Android.

8.3 Πρότερες γνώσεις των μαθητών για τον προγραμματισμό

Οι μαθητές κλήθηκαν να επιλέξουν όσες βασικές προγραμματιστικές έννοιες θυμούνται, από εκείνες που αναμένουμε να έχουν διδαχθεί στο Δημοτικό ή στο Γυμνάσιο, καθώς και τα περιβάλλοντα προγραμματισμού που έχουν αξιοποιήσει.

Προγραμματιστικές έννοιες που οι μαθητές έχουν αξιοποιήσει στο παρελθόν

Στην ερώτηση σχετικά με το ποιες βασικές προγραμματιστικές έννοιες από τις αναφερόμενες έχουν αξιοποιήσει στο παρελθόν οι μαθητές (πριν το εκπαιδευτικό πρόγραμμα), οι μισοί συμμετέχοντες (50%) δεν ήταν απόλυτα σίγουροι για τις απαντήσεις τους, έντεκα (32%) ανέφεραν τη μεταβλητή σαν τη πιο γνώριμη έννοια, ενώ έξι μαθητές της Α΄ τάξης ΕΠΑΛ (26%) απάντησαν ότι δεν είχαν χρησιμοποιήσει καμία έννοια από όσες αναφέρονταν στην ερώτηση.



Διάγραμμα 8.4: Προγραμματιστικές έννοιες που χρησιμοποίησαν οι μαθητές στο παρελθόν.

Οπωσδήποτε, η έννοια της μεταβλητής αποτελεί πρωταρχικής σημασίας έννοια για τον προγραμματισμό και οι μαθητές είναι πολύ πιθανό να την έχουν χρησιμοποιήσει στο παρελθόν, αλλά, όπως έχουμε επισημάνει και στο Κεφάλαιο 4, όπου αναφέρθηκαν οι συνήθεις δυσκολίες και παρανοήσεις των μαθητών, δεν αποκλείεται η αναφορά τους στην έννοια της μεταβλητής να προήλθε σε μεγάλο βαθμό από τη σύγχυσή της με την αντίστοιχη μαθηματική έννοια.

Προγραμματιστικά περιβάλλοντα που οι μαθητές έχουν χρησιμοποιήσει στο παρελθόν

Το 65% των μαθητών (23 μαθητές) δήλωσε πως είχε χρησιμοποιήσει στο παρελθόν κάποια Logo-like γλώσσα προγραμματισμού, το 57% είχε χρησιμοποιήσει το Scratch (20 μαθητές), ενώ το 37% (13 μαθητές) είχε χρησιμοποιήσει κάποια text-based γλώσσα (Pascal/ΓΛΩΣΣΑ). Μόνο ένας μαθητής δήλωσε πως δεν έχει χρησιμοποιήσει καμία γλώσσα προγραμματισμού στο παρελθόν.

Σε ότι αφορά την κατανομή των απαντήσεων ως προς την τάξη των μαθητών, οι απαντήσεις ήταν σχετικά μοιρασμένες, με εξαίρεση την περίπτωση των μαθητών της Β' τάξης Λυκείου που έχουν χρησιμοποιήσει περισσότερο κάποιο Logo-like περιβάλλον, και τους μαθητές της Α' τάξης ΕΠΑΛ που έχουν έρθει σε επαφή κυρίως με το Scratch.

Χρήση γλωσσών προγραμματισμού στο παρελθόν ανά τάξη μαθητών		Τάξη					Total
		Γ Γυμνασίου	Α ΓΕΛ	Β ΓΕΛ	Α ΕΠΑΛ	Β ΕΠΑΛ	
Logo-like	Count	5	4	11	2	1	23
Scratch	Count	4	3	6	7	0	20
Διεργηνηυτής - ΓΛΩΣΣΑ - PASCAL	Count	4	3	6	0	0	13
Total	Count	8	4	12	9	1	34

Πίνακας 8.1: Χρήση γλωσσών προγραμματισμού στο παρελθόν ανά τάξη μαθητών.

Γνώση της ύπαρξης του App Inventor πριν την έναρξη των εργαστηρίων

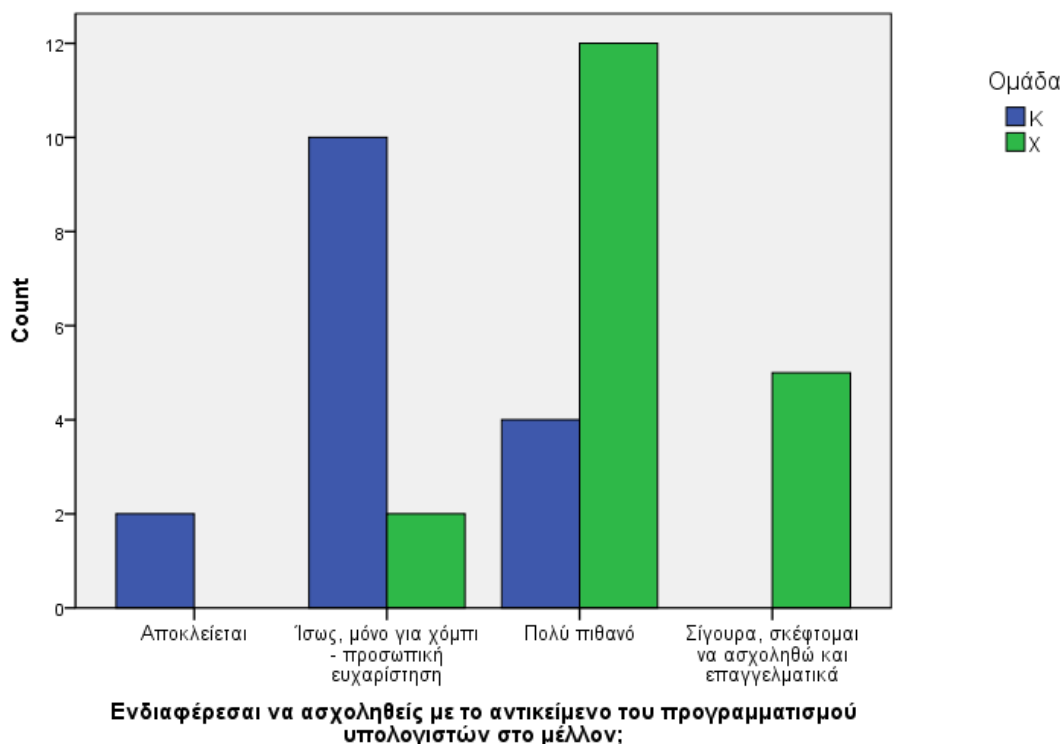
Το 91% των συμμετεχόντων δεν είχε ξαναχρησιμοποιήσει το App Inventor, εκτός από δυο μαθητές της Β' τάξης Γενικού Λυκείου.



Διάγραμμα 8.5: Γνώση της ύπαρξης του App Inventor πριν την έναρξη των εργαστηρίων.

Ενδιαφέρον ενασχόλησης με τον προγραμματισμό υπολογιστών στο μέλλον

Οι μαθητές που δήλωσαν από μόνοι τους, συμμετοχή στο εκπαιδευτικό πρόγραμμα (ομάδα X) ήταν εξ αρχής περισσότερο διατεθειμένοι να ασχοληθούν με το αντικείμενο του προγραμματισμού στο μέλλον, σε σχέση με τους μαθητές της ομάδας K, που παρακολούθησαν το πρόγραμμα τα απογεύματα του Σαββάτου, στα πλαίσια των υποχρεώσεών τους στη Μαθητική Στέγη. Σε γενικές γραμμές, η πλειοψηφία των μαθητών δήλωσε ότι είναι πολύ πιθανό να ασχοληθεί μελλοντικά με το αντικείμενο του προγραμματισμού, ανεξάρτητα με το αν αυτό θα συμβεί για προσωπικούς ή επαγγελματικούς λόγους.



Διάγραμμα 8.6: Ενδιαφέρον για μελλοντική ενασχόληση με τον προγραμματισμό υπολογιστών.

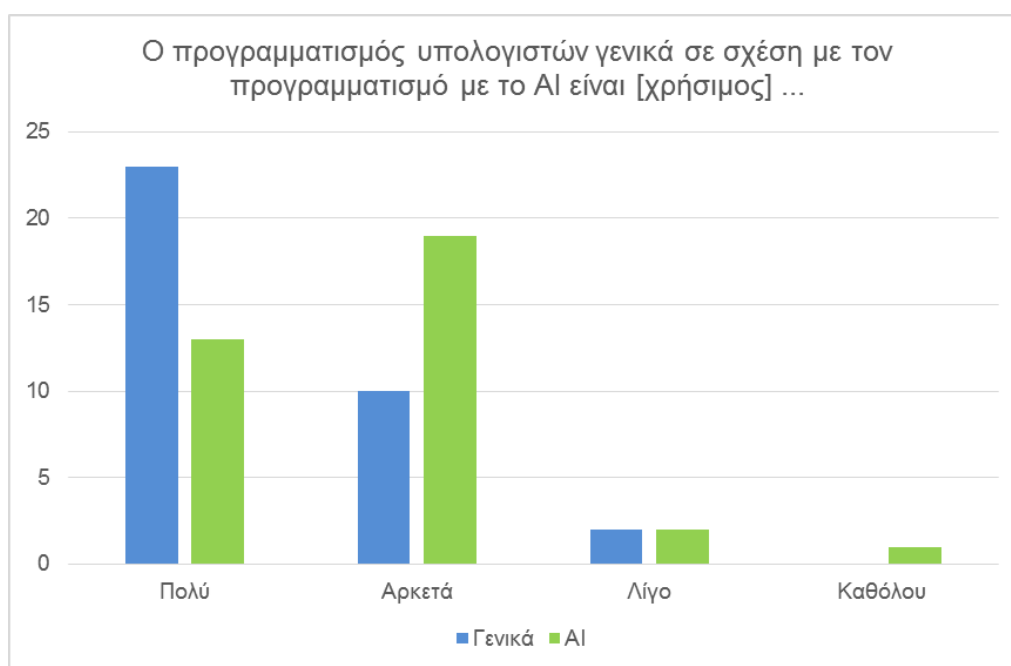
8.4 Ανίχνευση αλλαγών στις στάσεις των μαθητών για τον προγραμματισμό

Προκειμένου να μελετηθεί ο τρόπος και ο βαθμός επίδρασης της χρήσης του AI στις απόψεις των μαθητών, αξιοποιώντας τεχνικές ελέγχου υποθέσεων, μελετήθηκαν συσχετίσεις και ελέγχθηκε η ύπαρξη στατιστικά σημαντικών διαφορών πριν και μετά την παρέμβαση με τη διεξαγωγή του πειράματος στις ομάδες, με βάση τις απαντήσεις τους στα δυο ερωτηματολόγια (επίπεδο σημαντικότητας 0.05).

Με βάση τα αποτελέσματα των ελέγχων υποθέσεων για τα συσχετιζόμενα δείγματα (Paired Samples t-tests), από τις ερωτήσεις που επαναλήφθηκαν στο πρώτο και στο δεύτερο ερωτηματολόγιο, τα οποία παρατίθενται στο Παράρτημα Γ.1, διαπιστώθηκαν τα παρακάτω.

Παρατηρήθηκε στατιστικά σημαντική διαφορά στην άποψη των μαθητών, μετά τη χρήση του AI, σε σχέση με πριν, στο κατά πόσο θεωρούν ότι οι προγραμματιστικές έννοιες που πραγματεύτηκαν στο πρόγραμμα σχετίζονται με την καθημερινότητά τους ($p=0,017$). Το ίδιο ισχύει και για το κατά πόσο θεωρούν εύκολο ($p=0,002$) το αντικείμενο του προγραμματισμού με το AI, σε σχέση με τον προγραμματισμό υπολογιστών, γενικότερα.

Παράλληλα, όμως, διαπιστώθηκε μια στατιστικά σημαντική διαφορά στο κατά πόσο ο προγραμματισμός με το AI τους φάνηκε χρήσιμος, σε σχέση με τον προγραμματισμό υπολογιστών γενικότερα, όπου παρατηρήθηκε μια μικρή αρνητική τάση στις απαντήσεις ($p=0,002$ και Mean Differences = -0,343), όπως φαίνεται και στο παρακάτω διάγραμμα.

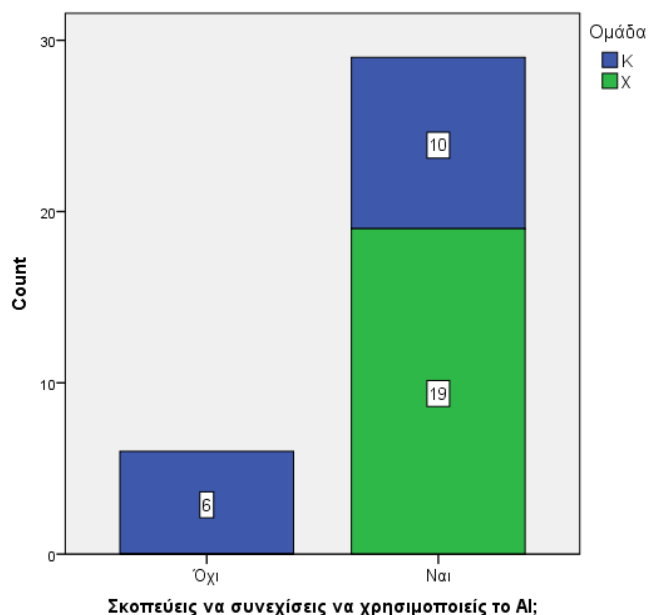


Διάγραμμα 8.7: Χρησιμότητα του προγραμματισμού σε AI σε σχέση με τον προγραμματισμό υπολογιστών.

Εξετάζοντας τις περιπτώσεις των μαθητών που εξέφρασαν αρνητική άποψη για τη χρησιμότητα του AI σε σχέση με τον προγραμματισμό υπολογιστών γενικότερα, δεν διαπιστώσαμε σημαντικές διαφορές μεταξύ των δυο ομάδων.

Και στις δυο ομάδες, παρατηρήθηκε αύξηση στη διάθεση των μαθητών να ξαναχρησιμοποιήσουν το AI, μετά το τέλος του προγράμματος, όπως επίσης και στο ενδιαφέρον τους να ασχοληθούν με τον προγραμματισμό υπολογιστών στο μέλλον, χωρίς όμως η διαφορά πριν και μετά να είναι στατιστικά σημαντική.

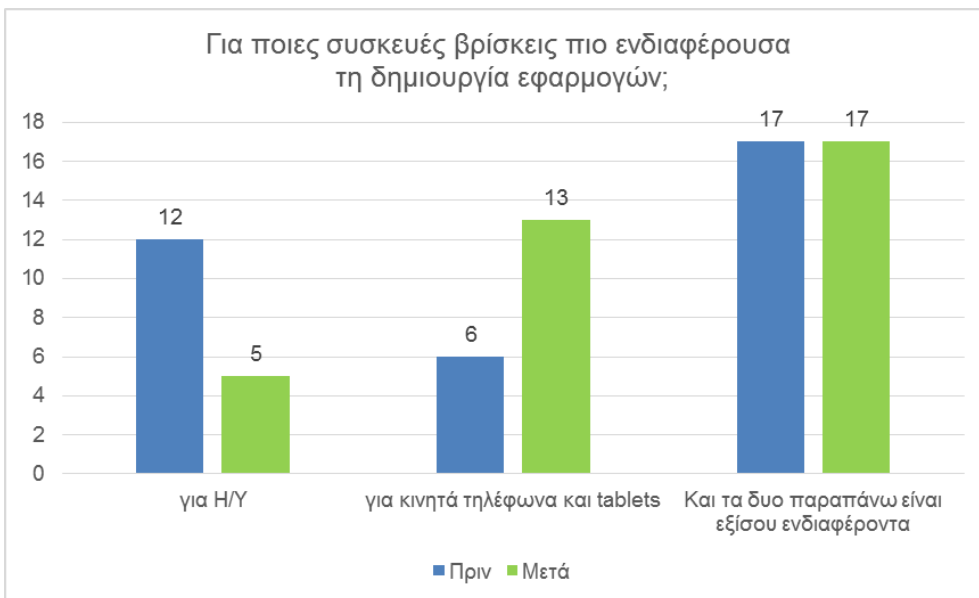
Ειδικά σε ότι αφορά τη διάθεση των μαθητών να συνεχίσουν να χρησιμοποιούν το AI μετά το τέλος των εκπαιδευτικών εργαστηρίων, παρατηρήθηκε πως το 83% των μαθητών απάντησε θετικά, με την ομάδα X να απαντάει καθολικά υπέρ, ενώ υπήρχαν έξι μαθητές (37,5%) της ομάδας K, οι οποίοι απάντησαν αρνητικά.



Διάγραμμα 8.8: Διάθεση για συνέχιση χρήσης του App Inventor μετά τη λήξη των εργαστηρίων.

Ενδιαφέρον ως προς τις συσκευές στόχο των αναπτυσσόμενων εφαρμογών

Παρατηρήθηκε μια σημαντική αλλαγή ως προς το είδος των συσκευών για τις οποίες οι μαθητές βρίσκουν πιο ενδιαφέρον τον προγραμματισμό. Πριν τη συμμετοχή τους στο εκπαιδευτικό πρόγραμμα, το 34% των μαθητών θεωρούσε πιο ενδιαφέρον τον προγραμματισμό υπολογιστών, έναντι ενός 17% υπέρ κινητών τηλεφώνων και ταμπλετών. Μετά τη συμμετοχή τους, στην ίδια ερώτηση, τα ποσοστά άλλαξαν σε 14% υπέρ του προγραμματισμού υπολογιστών, ανεβάζοντας το ποσοστό υπέρ των φορητών συσκευών σε 37%.



Διάγραμμα 8.9: Ενδιαφέρον ως προς τις συσκευές στόχο των αναπτυσσόμενων εφαρμογών.

8.5 Δεξιότητες που αναπτύχθηκαν

Μετά την ολοκλήρωση των εργαστηρίων, οι μαθητές κλήθηκαν να απαντήσουν σε ερωτήσεις σχετικές με δεξιότητες που θεωρούν ότι απέκτησαν με την αξιοποίηση του ΑΙ. Παράλληλα, οι εκπαιδευτικοί με τη φυσική τους παρουσία κατά τη διεξαγωγή του πειράματος και με την καταγραφή παρατηρήσεων, επισήμαναν, ανά μάθημα, τους μαθησιακούς στόχους που οι μαθητές φάνηκε να κατέκτησαν επαρκώς και εκείνους που τους δυσκόλευσαν.

8.5.1 Ικανότητα δημιουργίας προσωπικών εφαρμογών με το ΑΙ

Μετά το πέρας του εκπαιδευτικού προγράμματος, οι περισσότεροι μαθητές θεωρούς πως είναι σε θέση να φτιάξουν τη δική τους εφαρμογή στο ΑΙ, με μόλις τρεις μαθητές της ομάδας Κ να εκφράζουν αντίθετη άποψη.



Διάγραμμα 8.10: Η άποψη των μαθητών για το βαθμό ικανότητάς τους στην ανάπτυξη προσωπικών τους εφαρμογών στο AI.

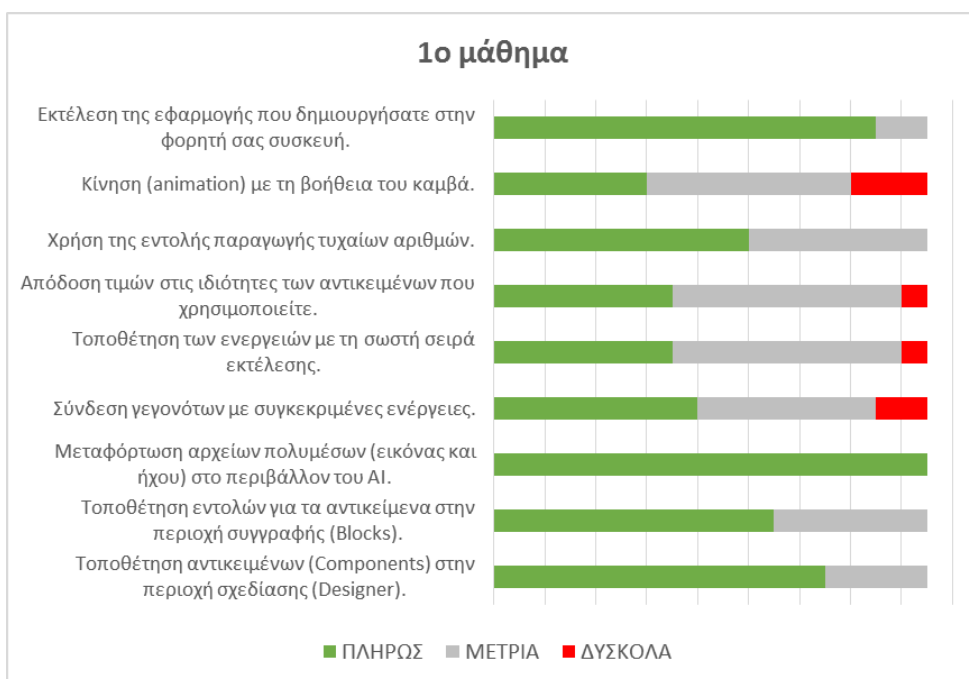
Μάλιστα, συγκρίνοντας τις δυο ακραίες ηλικιακά ομάδες μαθητών (Γ Γυμνασίου με Β Λυκείου), χρησιμοποιώντας έλεγχο ανεξάρτητων δειγμάτων (independent samples t-test) διαπιστώνουμε ότι δεν υπάρχουν στατιστικά σημαντικές διαφορές μεταξύ τους (Παράρτημα Γ.2). Βεβαίως είναι άξιο αναφοράς το γεγονός ότι οι μαθητές της Γ' τάξης Γυμνασίου διδάσκονται προγραμματισμό το τρέχον σχολικό έτος, σε αντίθεση με τους μαθητές της Β' τάξης Γενικού Λυκείου, που ενδέχεται να μην ξαναήρθαν σε επαφή με προγραμματιστικές έννοιες στο σχολικό περιβάλλον, από την φοίτησή τους στη Γ' τάξη Γυμνασίου.

8.5.2 Βαθμός κατάκτησης δεξιοτήτων

Για κάθε ένα από τα τρία μαθήματα, όπως αυτά παρουσιάστηκαν στο Κεφάλαιο 7, έγινε καταγραφή του βαθμού κατανόησης των βασικών εννοιών και επίτευξης των μαθησιακών αποτελεσμάτων, από τους εκπαιδευτικούς. Από τα στοιχεία που καταγράφηκαν στα σχετικά Φύλλα Παρατήρησης (Παράρτημα Β.4), προέκυψαν τα εξής:

Από το **1ο μάθημα**, κιάλας, οι μαθητές δεν αντιμετώπισαν καμία δυσκολία στην εκτέλεση και δοκιμή των εφαρμογών στη φορητή τους συσκευή, στην προσθήκη αρχείων πολυμέσων ή αντικειμένων σε αυτές.

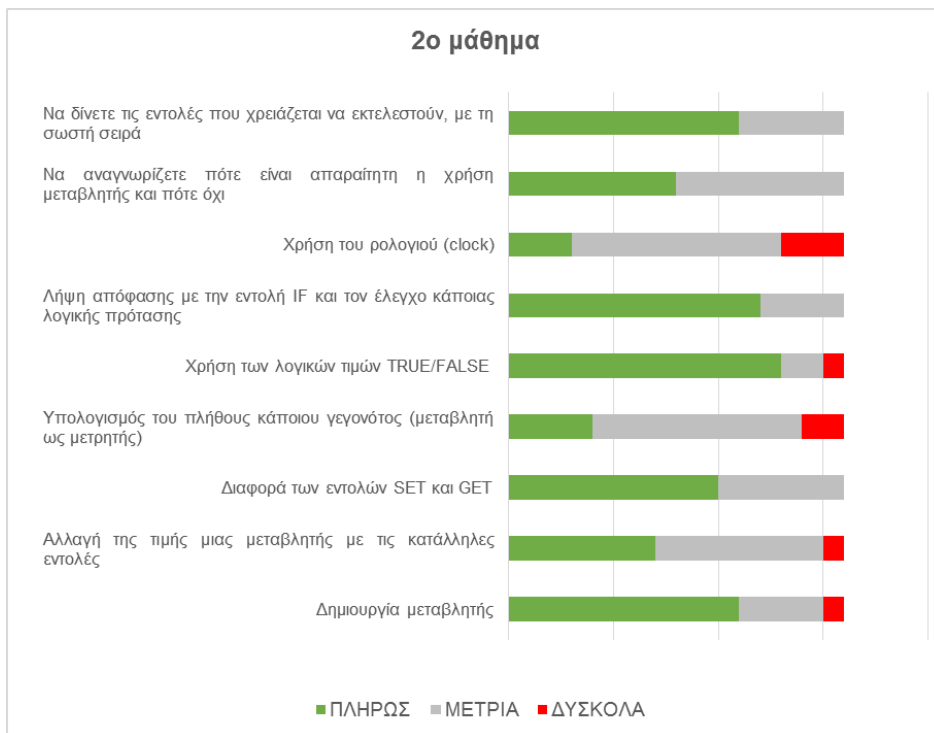
Η κύρια δυσκολία στο υλικό του 1ου μαθήματος εντοπίζεται στη δημιουργία κίνησης (animation) σε αντικείμενα με το άγγιγμα της οθόνης. Δυσκολίες εντοπίζονται και στις νοητικές διεργασίες που απαιτούνται για την απόδοση των κατάλληλων τιμών σε ιδιότητες των διαθέσιμων αντικειμένων, με τη σωστή σειρά και στη σύνδεση των παραπάνω ενεργειών με τα επιθυμητά γεγονότα.



Διάγραμμα 8.11: Επίτευξη στόχων 1^{ου} μαθήματος.

Στο **2ο μάθημα** υπήρξε μια θετική αλλαγή και εξοικείωση των μαθητών ως προς την δεξιότητα τοποθέτησης των απαιτούμενων εντολών στη σωστή σειρά. Δεν υπήρξαν ιδιαίτερες δυσκολίες, επίσης, ούτε στη δημιουργία μεταβλητών και στην αλλαγή της τιμής τους με τη σύνταξη κατάλληλων εντολών εκχώρησης, αλλά ούτε και στη σύνταξη λογικών προτάσεων και στη χρήση της δομής απλής επιλογής.

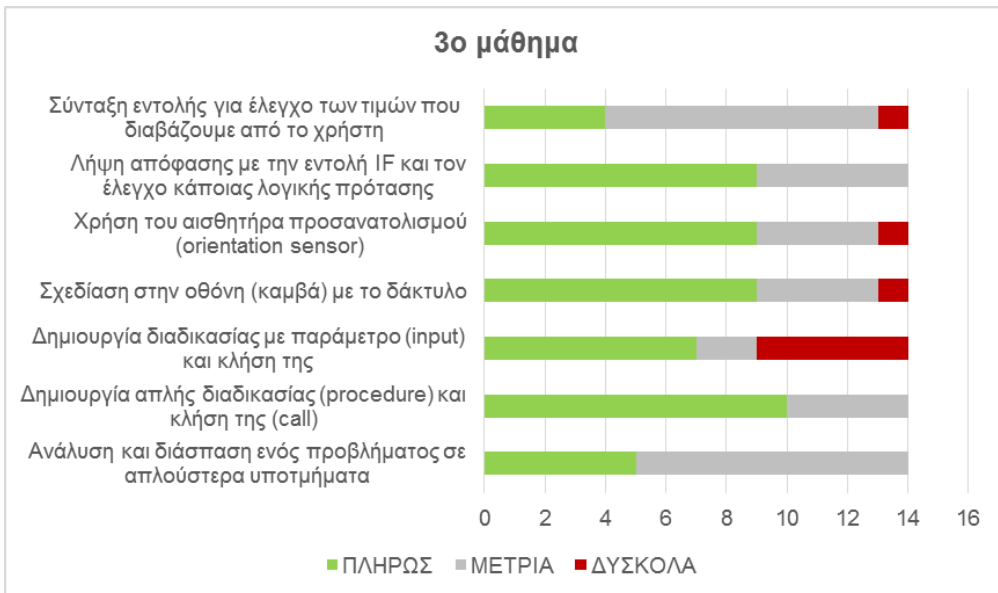
Οι μαθητές δυσκολεύτηκαν κυρίως στη χρήση του συστατικού του ρολογιού (clock), γεγονός που αναλύεται περισσότερο στη συνέχεια, και δευτερευόντως στη χρήση μεταβλητής ως μετρητή.



Διάγραμμα 8.12: Επίτευξη στόχων 2ου μαθήματος.

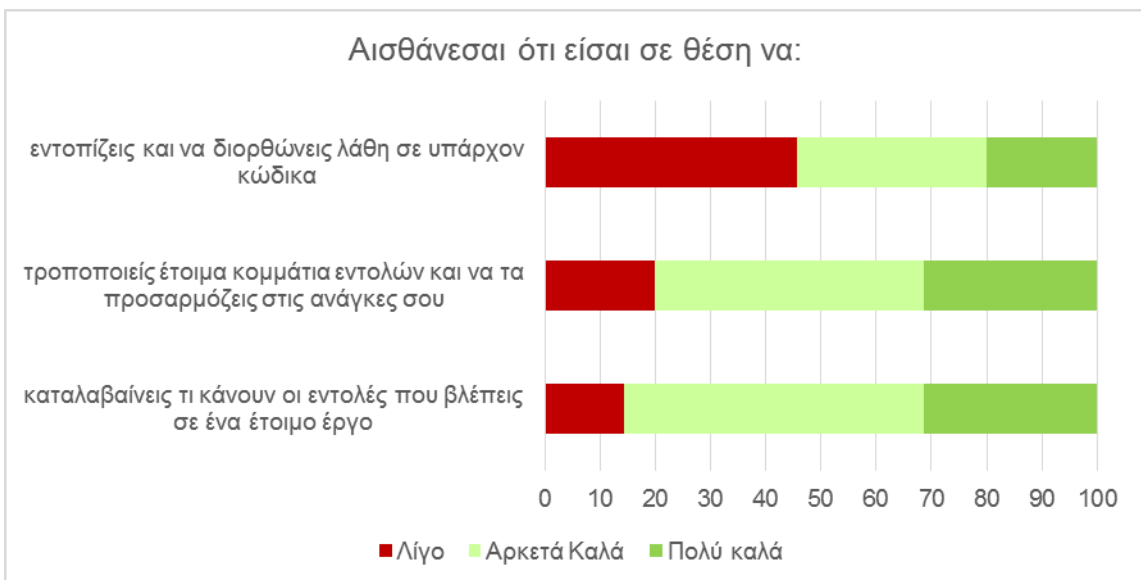
Στο 3ο και τελευταίο μάθημα, οι μαθητές της ομάδας X, που ήταν και οι μόνοι που έφτασαν μέχρι και το σημείο αυτό, εξακολούθησαν να μην αντιμετωπίζουν δυσκολίες στη χρήση της εντολής IF, αξιοποιώντας μάλιστα τιμές που προέρχονταν από αισθητήρα της συσκευής. Επίσης, δεν δυσκολεύτηκαν στην κατανόηση του ρόλου και των πλεονεκτημάτων μιας διαδικασίας σε μια εφαρμογή και στη δημιουργία και κλήση απλών διαδικασιών σε αυτή.

Αντιθέτως, ίσως και λόγω πίεσης χρόνου, οι μαθητές δεν πρόλαβαν να αντιληφθούν τον τρόπο δημιουργίας και χρήσης παραμετρικών διαδικασιών.



Διάγραμμα 8.13: Επίτευξη στόχων 3ου μαθήματος.

Οι ίδιοι οι συμμετέχοντες, μέσω του ερωτηματολογίου που συμπλήρωσαν, θεωρούν πως οι δυσκολίες τους στο ΑΙ εστιάζονται στην τροποποίηση έτοιμων κομματιών εντολών και στην προσαρμογή τους στις ανάγκες τους (20%), και σε ακόμα μεγαλύτερο βαθμό (46%) στον εντοπισμό και στη διόρθωση λαθών σε υπάρχον κώδικα.



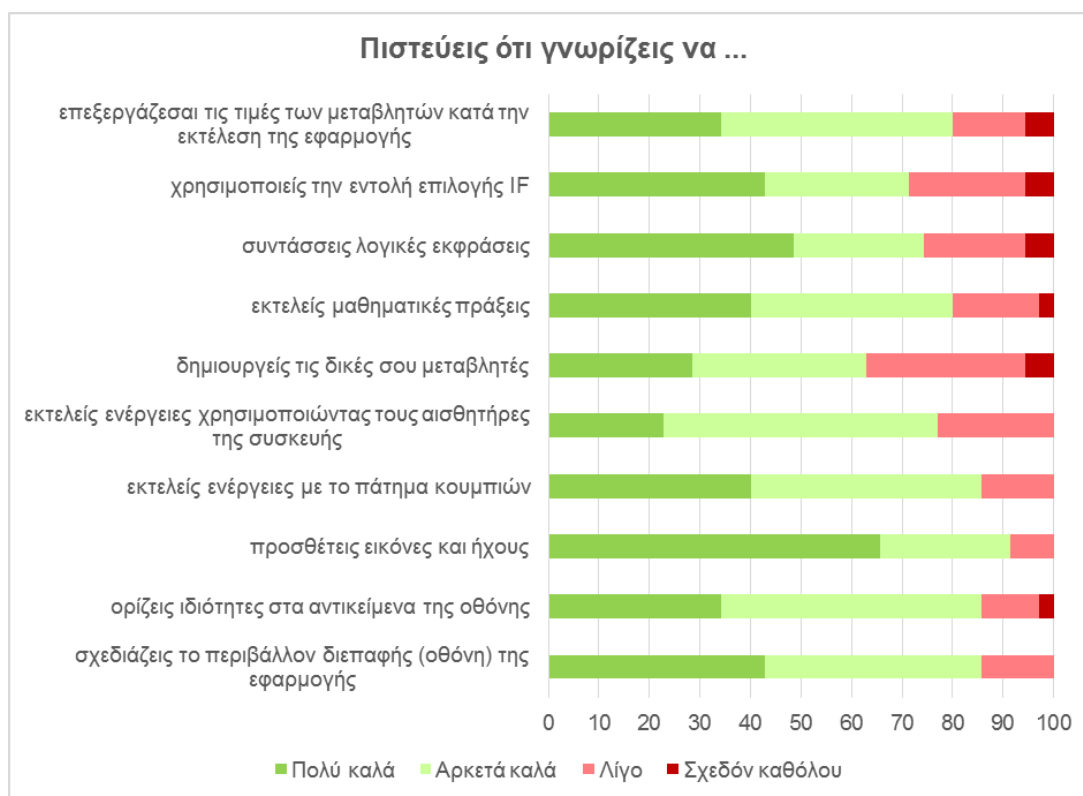
Διάγραμμα 8.14: Δυσκολίες που αναγνωρίζουν οι ίδιοι οι μαθητές.

Σε παρόμοια ερώτηση αυτοαξιολόγησης των μαθητών για το βαθμό επίτευξης των μαθησιακών στόχων, μετά το τέλος των εργαστηρίων (σε κλίμακα 1: Πολύ καλά έως 4: Σχεδόν Καθόλου), οι απαντήσεις τους (Παράρτημα Γ.4) δείχνουν πως η πολυμεσικότητα του ΑΙ και η σχεδίαση και

προσαρμογή του περιβάλλοντος διεπαφής των εφαρμογών δεν τους δημιούργησε δυσκολίες (θετικές απαντήσεις στο 91,4% και 85,7% και μέση τιμή 1,43 και 1,77 αντίστοιχα). Επίσης, η εκτέλεση ενεργειών με το πάτημα κουμπιών ή αξιοποιώντας τους αισθητήρες της συσκευής φαίνεται να έγιναν κατανοητές στην πλειοψηφία του δείγματος (θετικές απαντήσεις στο 85,7% και μέση τιμή 1,74). Αντιθέτως, η εκτέλεση ενεργειών αξιοποιώντας τους αισθητήρες της συσκευής έγινε μεν κατανοητή (θετικές απαντήσεις στο 77,1%), αλλά με υψηλότερη μέση τιμή (2,00).

Η εκτέλεση μαθηματικών πράξεων και η επεξεργασία των τιμών των μεταβλητών κατά τη διάρκεια εκτέλεσης των εφαρμογών φαίνεται να έγιναν κατανοητές από ικανοποιητικό ποσοστό μαθητών (θετικές απαντήσεις στο 80% και μέση τιμή 1,87). Η κύρια δυσκολία εντοπίζεται στη δημιουργία των απαραίτητων μεταβλητών, καθώς το 37,1% του δείγματος δήλωσε πως γνωρίζει λίγο έως σχεδόν καθόλου να δημιουργεί τις δικές του μεταβλητές με τις περισσότερες απαντήσεις να συγκεντρώνονται στην κλίμακα «Αρκετά καλά» και μέση τιμή 2,14.

Τέλος, η σύνταξη λογικών εκφράσεων και η χρήση της εντολής επιλογής IF έγιναν κατανοητές από τους μαθητές, με βάση τις απαντήσεις τους, σε ποσοστό 74,3% και 71,4% και μέση τιμή 1,83 και 1,91 αντίστοιχα.



Διάγραμμα 8.15: Αυτοαξιολόγηση των μαθητών για τις δεξιότητες που απέκτησαν.

Σε γενικές γραμμές δεν διαπιστώθηκαν ιδιαίτερες δυσκολίες στις δραστηριότητες που υλοποίησαν οι μαθητές, πέραν εκείνης όπου αποτελούσε επέκταση της προσομοίωσης ρίψης ενός νομίσματος για τη δημιουργία παιχνιδιού τύπου «Κορώνα – Γράμματα» με ποντάρισμα και animation της ρίψης. Οι λόγοι αναλύονται στην παράγραφο 8.7.1.

8.6 Οι απόψεις των μαθητών για το App Inventor

Ως προς τις απόψεις των μαθητών για το AI, χρησιμοποιήθηκε η πενταβάθμια κλίμακα Likert, με την τιμή 1 να αντιστοιχεί στην απόλυτη συμφωνία και την τιμή 5 να αντιστοιχεί στην απόλυτη διαφωνία.

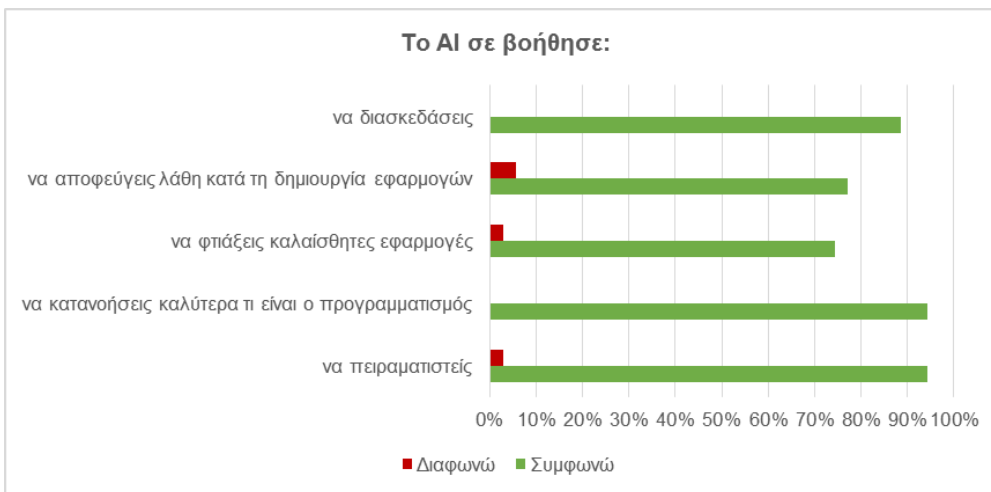
8.6.1 Θετικά σημεία του AI

Οι μέσες τιμές και τυπικές αποκλίσεις για τις απαντήσεις των μαθητών ως προς τα θετικά σημεία του AI φαίνονται στον παρακάτω πίνακα.

To App Inventor σε βοήθησε:	Πλήθος	Μέση τιμή	Τυπική Απόκλιση
Να πειραματιστείς	35	1,34	,684
Να κατανοήσεις καλύτερα τι είναι ο προγραμματισμός	35	1,31	,583
Να φτιάξεις καλαίσθητες εφαρμογές	35	1,91	,853
Να αποφεύγεις λάθη κατά τη δημιουργία εφαρμογών	35	1,89	,900
Να διασκεδάσεις	35	1,46	,701

Πίνακας 8.2: Που βοήθησε το AI τους μαθητές (και τι τους άρεσε).

Οι συμμετέχοντες, σε ποσοστό 94% ανέφεραν ότι το AI τους βοήθησε να πειραματιστούν, και να κατανοήσουν καλύτερα τι είναι ο προγραμματισμός και σε ποσοστό 89% να διασκεδάσουν. Σημαντικά θετική ήταν η στάση των μαθητών και ως προς τη συμβολή του AI στο να δημιουργήσουν καλαίσθητες εφαρμογές, αποφεύγοντας λάθη κατά τη δημιουργία τους. Επίσης, σε καμία ερώτηση δεν παρατηρήθηκαν στατιστικά σημαντικές διαφορές μεταξύ των απαντήσεων των δυο ομάδων (crosstab chi-square tests για $\alpha=0.05$).



Διάγραμμα 8.16: Που βοήθησε το AI τους μαθητές.

Μάλιστα, στην προαιρετική ερώτηση ελεύθερου κειμένου για το ίδιο θέμα, ένας μαθητής σημείωσε ότι «το AI μου άρεσε γιατί μου επέτρεψε να φτιάξω εύκολα δικές μου εφαρμογές για το κινητό μου τηλέφωνο (απλές μεν, αλλά δικές μου), ενώ δεν είχα ιδέα πως γίνεται κάτι τέτοιο».

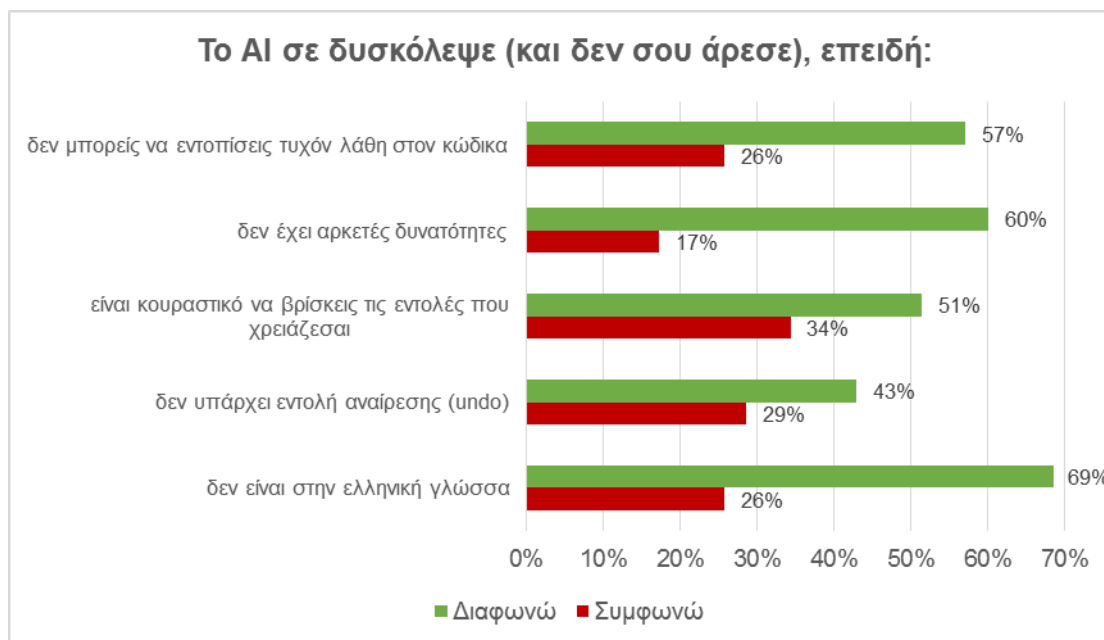
8.6.2 Αρνητικά σημεία του AI

Ως προς τα πιθανά χαρακτηριστικά ή ελλείψεις του AI που ενδεχομένως να δυσκόλευσαν ή να μην άρεσαν στους μαθητές, οι μέσες τιμές και τυπικές αποκλίσεις για τις απαντήσεις των μαθητών στις σχετικές ερωτήσεις, φαίνονται στον παρακάτω πίνακα (στην πενταβάθμια κλίμακα Likert, η τιμή 1 υποδηλώνει την απόλυτη συμφωνία και η τιμή 5 την απόλυτη διαφωνία).

Το App Inventor σε δυσκόλεψε (και δεν σου άρεσε), επειδή:	Πλήθος	Μέση τιμή	Τυπική Απόκλιση
Δεν είναι στην ελληνική γλώσσα	35	3,74	1,502
Δεν υπάρχει εντολή αναίρεσης (undo)	35	3,17	1,294
Είναι κουραστικό να βρίσκεις τις εντολές που χρειάζεσαι	35	3,34	1,349
Δεν έχει αρκετές δυνατότητες	35	3,63	1,087
Δεν μπορείς να εντοπίσεις τυχόν λάθη στον κώδικα	35	3,63	1,330

Πίνακας 8.2: Τι δυσκόλεψε τους μαθητές στο AI (και δεν τους άρεσε).

Οι μαθητές φαίνονται να συμφωνούν πως δεν αποτέλεσε πρόβλημα για αυτούς το γεγονός πως το περιβάλλον (διεπαφή και εντολές) του AI είναι στην αγγλική γλώσσα. Επίσης, μάλλον διαφωνούν ότι το AI δεν είχε αρκετές δυνατότητες ή δεν μπόρεσαν να εντοπίσουν τυχόν λάθη στον κώδικα. Οι κύριες δυσκολίες τους εντοπίζονται στον κόπο που απαιτήθηκε για να εντοπίζουν τις απαιτούμενες εντολές και στην απουσία της εντολής αναίρεσης. Όπως και πριν, σε καμία ερώτηση δεν παρατηρήθηκαν στατιστικά σημαντικές διαφορές μεταξύ των απαντήσεων των δυο ομάδων (crosstab chi-square tests για $\alpha=0.05$).



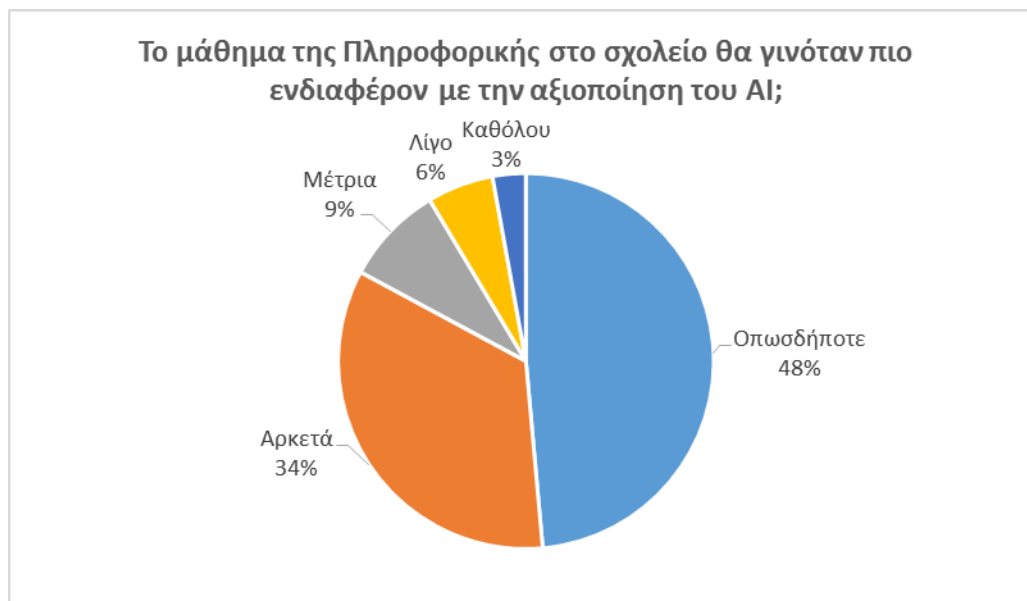
Διάγραμμα 8.17: Που δυσκόλεψε το AI τους μαθητές.

Στην προαιρετική ερώτηση ελεύθερου κειμένου σχετικά με το τι δυσκόλεψε ή δεν άρεσε στους μαθητές στο AI, δυο μαθητές αναφέρθηκαν στη δυσκολία να βρουν τις εντολές που χρειάζονταν, επισημαίνοντας πως αυτή ξεπεράστηκε στην πορεία.

8.6.3 Η συμβολή του AI στην ανάπτυξη του ενδιαφέροντος των μαθητών

Βασικός μας στόχος μας ήταν να εξετάσουμε κατά πόσο το AI συμβάλλει στη διευκόλυνση της διδασκαλίας και μάθησης του προγραμματισμού στη δευτεροβάθμια εκπαίδευση και όχι, εν γένει, αν η χρήση του κρίνεται ενδιαφέρουσα από τους μαθητές. Βεβαίως, στην ερώτηση σχετικά με το κατά πόσο πιστεύουν οι μαθητές ότι το μάθημα της πληροφορικής θα γινόταν πιο

ενδιαφέρον με την αξιοποίηση του ΑΙ, οι απαντήσεις φανερώνουν ότι το ΑΙ ενθουσίασε τους μαθητές του δείγματος, κάτι που είχε φανεί εξ αρχής. Συγκεκριμένα, το 82% του δείγματος εκφράστηκε θετικά (απαντήσεις «οπωσδήποτε» ή «αρκετά»). Μόνο τρεις μαθητές της ομάδας Κ (9% επί του συνόλου του δείγματος) εκφράστηκαν αρνητικά.



Διάγραμμα 8.18: Κατά πόσο πιστεύουν οι μαθητές ότι το μάθημα της πληροφορικής θα γινόταν πιο ενδιαφέρον με την αξιοποίηση του ΑΙ.

Στο σημείο αυτό αξίζει να αναφερθεί ένα χαρακτηριστικό περιστατικό που δείχνει τον ενθουσιασμό των μαθητών για το ΑΙ, καθώς και το ενδιαφέρον τους και την προσδοκία τους να υλοποιήσουν τις ιδέες τους με αυτό. Στην τέταρτη προγραμματισμένη συνάντηση της ομάδας Χ, παρουσιάστηκε ένα απρόσμενο τεχνικό πρόβλημα με τη σύνδεση του σχολείου στο διαδίκτυο και την πρόσβαση στο περιβάλλον του ΑΙ, οπότε προτάθηκε από τους εκπαιδευτές στους μαθητές, είτε να αποχωρήσουν, είτε να οργανωθούν σε ομάδες και να συζητήσουν για ιδέες πιθανών εφαρμογών που θα ήθελαν να αναπτύξουν, να καταλήξουν σε μια ιδέα που να τους βρίσκει όσο το δυνατό πιο σύμφωνους και να σχεδιάσουν σε χαρτί τη διεπαφή της εφαρμογής. Αξιοσημείωτο είναι πως κανένας μαθητής δεν εκδήλωσε ενδιαφέρον να αποχωρήσει. Αντιθέτως, αμέσως και με ενθουσιασμό, οι μαθητές ξεκίνησαν την εργασία σε ομάδες. Βέβαια, ορισμένες ιδέες των μαθητών, δεν ήταν δυνατό να υλοποιηθούν άμεσα, με τις γνώσεις τους στο ΑΙ τη δεδομένη στιγμή, αλλά θεωρούμε πως, όσοι μαθητές ολοκλήρωναν τις προτεινόμενες δραστηριότητες που περιλαμβάνονται στο πλήρες πακέτο εκπαιδευτικού υλικού των τεσσάρων μαθημάτων που τους δόθηκε, θα ήταν ικανοί να τις υλοποιήσουν σε ικανοποιητικό βαθμό.

8.7 Απόψεις των εκπαιδευτικών

Σε συνεργασία με συναδέλφους εκπαιδευτικούς πληροφορικής, που συμμετείχαν στη διεξαγωγή των εκπαιδευτικών προγραμμάτων, είτε αναλαμβάνοντας υποστηρικτικούς ρόλους ή απλά λαμβάνοντας μέρος ως εκπαιδευόμενοι, ερχόμενοι για πρώτη φορά σε επαφή με το περιβάλλον του AI, καταγράψαμε τις απόψεις τους και τις συνδύασαμε με τα ευρήματα της επιτόπιας συμμετοχικής μας παρατήρησης.

8.7.1 Ευρήματα από τη συμμετοχική παρατήρηση του ερευνητή

Σε συζήτηση με τους μαθητές κατά την πρώτη συνάντηση, διαπιστώθηκε πως από τους τρεις μαθητές που είχαν ξαναχρησιμοποιήσει το AI, οι δύο ήταν μαθητές της Β' τάξης Γενικού Λυκείου, οι οποίοι στο πλαίσιο της καμπάνιας της Ώρας του Κώδικα (Hour of Code) είχαν γνωρίσει το AI σε διδακτική ώρα του μαθήματος «Εφαρμογές Υπολογιστών», ενώ και μια μαθήτρια της Γ' τάξης Γυμνασίου είχε ασχοληθεί από μόνη της, έχοντας παρακολουθήσει δυο tutorials από την επίσημη ιστοσελίδα του AI.

Σε όλες τις συναντήσεις, οι μαθητές έδειξαν ικανοποιητικό ενδιαφέρον για το αντικείμενο του προγραμματισμού με το AI, χωρίς να εμφανίσουν σημάδια ανίας. Η άμεση και εύκολη δοκιμή των εφαρμογών, ακόμα και πριν αυτές ολοκληρωθούν, σε οποιαδήποτε συσκευή, απλά σαρώνοντας με την κάμερα τον κωδικό (QR code) που εμφανιζόταν στην οθόνη του Η/Υ τους, ενθουσίασε τους μαθητές. Πολλοί παρήγαγαν και τα αντίστοιχα .apk αρχεία για μόνιμη αποθήκευση των εφαρμογών στις συσκευές τους και αναρωτιόντουσαν πως μπορούν να τις μοιραστούν με φίλους τους ή και μεταξύ τους.

Οι μαθητές, από την πρώτη συνάντηση, μόλις απέκτησαν σχετική άνεση με το χειρισμό των αντικειμένων στην περιοχή σχεδίασης, προσωποποίησαν το γραφικό περιβάλλον των εφαρμογών που ανέπτυσαν, αλλάζοντας λ.χ. ετικέτες και θέσεις στα κουμπιά ή ορίζοντας διαφορετικό φόντο και χρώματα σε αντικείμενα. Δεν περιορίστηκαν όμως μόνο σε σχεδιαστικές αλλαγές και παρεμβάσεις, αλλά πειραματίστηκαν και στο τμήμα του κώδικα των εφαρμογών. Ορισμένα χαρακτηριστικά παραδείγματα που δείχνουν τη διάθεση των μαθητών να δώσουν την προσωπική τους πινελιά είναι τα παρακάτω:

- Ένας μαθητής τροποποίησε μια εικόνα ζαριού και δημιούργησε ένα νέο κενό ζάρι, παραλλάσσοντας παράλληλα τον κώδικα, ώστε να εμφανίζεται και το κενό ζάρι στην οθόνη.
- Ένας άλλος μαθητής, που υλοποίησε τους δυο διαφορετικούς τρόπους ριξίματος των ζαριών (με κουμπί και ταρακούνημα) «πείραξε» τις εντολές που ενεργοποιούνται με το πάτημα του κουμπιού, ώστε να η ζαριά να φέρνει μόνο 5 και 6 στα ζάρια.
- Κάποιοι άλλοι προχώρησαν ακόμα παραπέρα, προσθέτοντας ένα κρυφό πλαίσιο ελέγχου (checkbox), ώστε, όποτε το επιθυμούν, να το επιλέγουν και η επόμενη ζαριά να είναι εξάρες.
- Μια ομάδα σκέφτηκε και υλοποίησε την προσωρινή απενεργοποίηση της δυνατότητας ρίψης των ζαριών για ένα χρονικό διάστημα της επιλογής τους, αφού είχε πατηθεί το κουμπί ρίψης ή είχε ταρακουνηθεί η συσκευή.
- Μια ομάδα, αφού υλοποίησε τη δραστηριότητα της εφαρμογής αντίστροφου χρονομέτρου, επέκτεινε τις δυνατότητές τις, προσθέτοντας λεπτά, πέραν των δευτερολέπτων. Ο χρήστης μπορούσε μάλιστα να αυξάνει κατά ένα λεπτό ή δευτερόλεπτο το διαθέσιμο χρόνο, αγγίζοντας τα αντίστοιχα κουμπιά.
- Στην δραστηριότητα της εφαρμογής πυξίδας, μετά από μια σύντομη συζήτηση με τους μαθητές για τον αισθητήρα προσανατολισμού της συσκευής και τις τιμές που παρέχει, προτρέψαμε τους μαθητές να δημιουργήσουν μια σύνθετη λογική συνθήκη με βάση την τιμή του αισθητήρα αυτού, για ηχητική ειδοποίηση όταν κοιτάζουν προς το βορρά. Οι περισσότερες ομάδες δημιούργησαν παραλλαγές της συνθήκης, οι οποίες λειτουργούσαν εξίσου σωστά, κάτι το οποίο δείχνει ότι κατανόησαν ικανοποιητικά τον τρόπο σύνταξης σύνθετων λογικών προτάσεων.

Βεβαίως, παρουσιάστηκαν ορισμένες δυσκολίες και προβλήματα στους μαθητές, που άλλοτε ξεπεράστηκαν, άλλοτε δημιούργησαν εμπόδια στην επίτευξη των μαθησιακών στόχων.

Αρκετοί μαθητές επιχείρησαν να υλοποιήσουν τις δραστηριότητες των φύλλων εργασίας, χωρίς να δίνουν την απαραίτητη σημασία στα επεξηγηματικά κείμενα και συμβουλές, παρά μόνο προσπαθώντας να «συναρμολογήσουν» τις εικόνες με τα πλακίδια που συναντούσαν στο φύλλο

εργασίας που είχαν μπροστά τους. Αυτό είχε ως αποτέλεσμα να ρωτάνε συνέχεια τους συμμαθητές τους ή τους εκπαιδευτικούς, που βρίσκεται το πλακίδιο που αναζητούν. Ευτυχώς, γρήγορα αντιλήφθηκαν ότι στο κείμενο υπήρχαν σημαντικές και χρήσιμες πληροφορίες για τη δική τους διευκόλυνση και μελετούσαν με μεγαλύτερη προσοχή τα φύλλα εργασίας και απευθύνονταν σπανιότερα σε τρίτους για βοήθεια.

Οι μαθητές, για τη σύνταξη εντολών που τροποποιούν ορισμένες ιδιότητες των συστατικών στις εφαρμογές τους, και συγκεκριμένα για τον ορισμό των αρχείων εικόνας ή ήχου που θα χρησιμοποιηθούν, αναγκάζονταν να πληκτρολογήσουν οι ίδιοι, ολόκληρο ή ένα μέρος του ονόματος του σχετικού αρχείου, κάθε φορά. Συχνά, κατά τη δοκιμή της εφαρμογής τους, αντιμετώπισαν το πρόβλημα να μην εμφανίζεται το επιθυμητό αποτέλεσμα, λόγω συντακτικού λάθους, είτε στο όνομα του αρχείου, είτε στην επέκταση.

Κάτι αντίστοιχο συναντήσαμε και κατά την απόδοση τιμών σε μεταβλητές, όπου ορισμένοι μαθητές έθεταν σε αυτές, ολόκληρα τα ονόματα αρχείων, μαζί με την επέκταση, παρόλο που με τη χρήση της εντολής “join”, αρκούσε να χρησιμοποιείται μόνο το όνομα και όχι η επέκταση. Το πρόβλημα της λάθος πληκτρολόγησης θα μπορούσε να επιλυθεί, αν στις περιπτώσεις επιλογής αρχείων πολυμέσων στον κώδικα, το AI παρείχε τη δυνατότητα, αυτά να επιλέγονται από λίστα. Τέλος, παρατηρήθηκε συχνά, παρότι οι μαθητές ενημέρωναν τις τιμές των μεταβλητών, χρησιμοποιώντας τις σχετικές εντολές εκχώρησης τιμής, αμελούσαν να τροποποιήσουν και τις αντίστοιχες ετικέτες κειμένου, οπότε νόμιζαν ότι κάτι έκαναν λάθος.

Πολύ συχνά, κατά τη διαδικασία σύνταξης των εντολών για τις διάφορες δραστηριότητες, ζητούσαν βοήθεια, αντιμετωπίζοντας ως επί το πλείστο το ίδιο πρόβλημα: ακολουθώντας τις οδηγίες των φύλλων εργασίας ή επιχειρώντας να προσθέσουν πλακίδια εντολών για κάποιο γεγονός που σχετίζεται με ένα αντικείμενο, προσέθεταν ξανά στο χώρο σύνταξης του προγράμματος, το ίδιο πλακίδιο χειρισμού του γεγονότος για το αντικείμενο αυτό, οπότε το AI τους προειδοποιούσε πως υπάρχει κάποιο πρόβλημα, αλλά δεν μπορούσαν να το αντιληφθούν, ίσως και λόγω των αγγλικών όρων.



Εικόνα 8.19: Το σφάλμα της συνύπαρξης δυο πανομοιότυπων πλακιδίων διαχείρισης ενός συμβάντος.

Κατά τη σύνταξη εντολών σύνθετης επιλογής (if ... else), ορισμένοι μαθητές επαναλάμβαναν τις ίδιες εντολές που ακολουθούσαν τον έλεγχο και στις δυο περιπτώσεις, παρότι θα μπορούσαν να τις τοποθετήσουν μια φορά μόνο, μετά τη δομή επιλογής.

Κατά τη σύνταξη σύνθετων λογικών εκφράσεων, κάποιοι προτίμησαν να χρησιμοποιήσουν εμφωλευμένες εντολές IF αντί να χρησιμοποιήσουν το λογικό τελεστή AND ή τοποθέτησαν απλές εντολές ελέγχου ακολουθιακά, αντί να χρησιμοποιήσουν το λογικό τελεστή OR.

Γενικά, η δομή επιλογής, λόγω και της οπτικής αναπαράστασής της, αλλά και με τη χρήση παραδειγμάτων που αξιοποίησαν τους αισθητήρες της συσκευής, φάνηκε να έγινε κατανοητή από τους μαθητές, καθώς στις επεκτάσεις που τους ζητήθηκε να την εφαρμόσουν, το αποτέλεσμα ήταν ιδιαίτερος ικανοποιητικό.

Σε ότι αφορά τις βασικές έννοιες του δομημένου προγραμματισμού, η δομή επανάληψης δεν εξετάστηκε σε μεγάλο βαθμό, καθώς στον καθοδηγούμενο από γεγονότα προγραμματισμό που υποστηρίζει το AI, η επαναληπτικότητα επιτυγχάνεται με άλλους τρόπους, όπως π.χ. με τη χρήση του συστατικού του ρολογιού.

Όπως αναφέρθηκε και στην παράγραφο 8.5.2, το συστατικό του ρολογιού, ως χαρακτηριστικό παράδειγμα του ταυτόχρονου προγραμματισμού που υποστηρίζει το AI, φάνηκε να δυσκόλεψε τους μαθητές και να προκάλεσε τις περισσότερες παρανοήσεις εκ μέρους τους. Η σχετική δραστηριότητα αποτελούσε επέκταση της προσομοίωσης ρίψης ενός νομίσματος για τη δημιουργία παιχνιδιού τύπου «Κορώνα – Γράμματα» με ποντάρισμα και animation της ρίψης. Ο βασικός λόγος ήταν η δυσκολία κατανόησης του τρόπου λειτουργίας και χρήσης του συστατικού του AI με την ονομασία clock.

Το συγκεκριμένο συστατικό μπορεί να λειτουργήσει σαν χρονόμετρο ή σαν πυροδότης εντολών ανά χρονικά διαστήματα της επιλογής των χρηστών και είναι στη διάθεσή τους για ενεργοποίηση, απενεργοποίηση και επεξεργασία του, ανά πάσα στιγμή κατά την εκτέλεση μιας εφαρμογής. Οι περισσότεροι μαθητές φάνηκε να συγχέουν το γεγονός της πυροδότησης του ρολογιού με την ενεργοποίησή του. Αντιλαμβάνονται το γεγονός της πυροδότησης του ρολογιού σαν την κλήση ενός υποπρογράμματος που διακόπτει την κανονική ροή εκτέλεσης των εντολών και μετά την ολοκλήρωσή των εντολών που περιλαμβάνει, επιστρέφει στο κυρίως πρόγραμμα. Η πραγματικότητα είναι ότι από τη στιγμή της ενεργοποίησης του ρολογιού και μέχρι την απενεργοποίησή του, οι εντολές που περιλαμβάνει το γεγονός της πυροδότησής του,

εκτελούνται περιοδικά και παράλληλα με τις υπόλοιπες εντολές της εφαρμογής. Πιο συγκεκριμένα, με βάση το χρονικό όριο που θέτει ο προγραμματιστής στο κάθε ρολόι, εκτελούνται επαναλαμβανόμενα οι εντολές που περικλείονται σε αυτό.

Η συγκεκριμένη δυσκολία, εκτιμούμε ότι θα περιοριζόταν μέσα από απλούστερα παραδείγματα που να κάνουν εμφανή τα αποτελέσματα εκτέλεσης των εντολών με την περιοδική πυροδότηση του ρολογιού, παράλληλα με την κανονική ροή εκτέλεσης της εφαρμογής. Η συγκεκριμένη δραστηριότητα του παιχνιδιού «Κορώνα - Γράμματα», αποδείχθηκε δυσνόητη, λόγω της ύπαρξης δυο ρολογιών: ενός για τη δημιουργία κίνησης στο κέρμα και ενός για το σταμάτημά της μετά από προκαθορισμένο χρονικό διάστημα. Εξάλλου, όμως, αν εξαρέσουμε τις δυνατότητες που παρέχει το συστατικό του ρολογιού και δικαιολογούν τη χρήση του, η συγκεκριμένη έννοια δεν αποτελεί βασικό μαθησιακό στόχο των μαθημάτων προγραμματισμού στη Δευτεροβάθμια εκπαίδευση, οπότε μπορεί να παραλειφθεί.

8.7.2 Ευρήματα συνεντεύξεων

Με βάση τις συνεντεύξεις που δόθηκαν από τους έξι εκπαιδευτικούς που συμμετείχαν στα εκπαιδευτικά εργαστήρια (Παράρτημα Δ), ασχέτως του ρόλου που είχαν σε αυτά, προέκυψαν πολλά κοινά στοιχεία, χωρίς την εμφάνιση σημείων διαφωνίας.

Η γενική σύνθεση απόψεων ως προς το AI είναι ότι είναι ελκυστικό προς τους μαθητές, ενδιαφέρον και συγχρόνως διασκεδαστικό, εύκολο στη χρήση και μοιάζει σε αρκετά σημεία με το Scratch. Επιπλέον, το γεγονός της άμεσης εμφάνισης του αποτελέσματος στις συσκευές τους, αλλά και γενικότερα της δυνατότητας που τους παρείχε να φτιάξουν πραγματικές εφαρμογές για αυτές, γοήτευσε τα παιδιά.

Ως προς την καμπύλη εκμάθησης (learning curve), όμως, το AI υστερεί σε σχέση με το Scratch. Απαιτείται περισσότερος χρόνος και εξάσκηση, προτού να μπορέσει ο αρχάριος χρήστης να αναπτύξει κάτι λειτουργικό και χρήσιμο. Κάποιοι απέδωσαν το φόρτο αυτό και στο γεγονός ότι, τουλάχιστον προς το παρόν, το περιβάλλον του εργαλείου διατίθεται αποκλειστικά στην αγγλική γλώσσα, αποτελώντας εμπόδιο για αρκετούς μαθητές. Σε κάθε περίπτωση, βέβαια η καμπύλη εκμάθησης δεν είναι τόσο απότομη, όσο εκείνη των text-based γλωσσών, χωρίς το AI να στερείται δυνατοτήτων.

Οι ερωτώμενοι εξέφρασαν την άποψη πως απαιτούνται αρκετές διδακτικές ώρες, μέχρις ότου οι μαθητές να φτάσουν σ' έναν ικανοποιητικό βαθμό κατανόησης και χειρισμού του περιβάλλοντος. Έτσι, θεωρούν ότι το σχολικό μάθημα στο οποίο θα ενταχθεί, πρέπει να προβλέπει αρκετές ώρες στο αντικείμενο του προγραμματισμού ή τουλάχιστον να μην είναι η πρώτη επαφή των μαθητών με παρόμοια εργαλεία και προγραμματιστικές έννοιες.

Το οπτικό περιβάλλον διευκολύνει τη σύνταξη του προγράμματος, χωρίς συντακτικά λάθη, ωστόσο επειδή μερικές εντολές απαιτούν την σύνθεση τους από πολλαπλά σημεία, οι μαθητές ορισμένες φορές μπερδεύονται.

Όταν αυξηθούν αρκετά οι χρησιμοποιούμενες εντολές, είναι δύσκολο να τις δει κανείς όλες μαζί στην οθόνη και δεν είναι αισθητικά ωραίο και ευανάγνωστο το τελικό αποτέλεσμα της σύνθεσης, δυσχεραίνοντας τον εντοπισμό προβλημάτων ή την πραγματοποίηση αλλαγών. Ο κοινός χώρος για την ανάπτυξη του κώδικα, κρύβοντας τον αντικειμενοστρεφή χαρακτήρα του εργαλείου, οδηγεί σε δύσκολα διαχειρίσιμα προγράμματα. Ο χρήστης αναγκάζεται να διαχειριστεί διαφορετικές εντολές για κάθε αντικείμενο, οι οποίες εν τέλει «συνωστίζονται» στον ίδιο χώρο.

Ως προς τις υποστηριζόμενες λειτουργίες, το AI έχει ορισμένους περιορισμούς. Οι εκπαιδευτικοί που συγκεκριμενοποίησαν και αιτιολόγησαν τη θέση τους, ανέφεραν την αδυναμία να δημιουργεί ο χρήστης νέα προγραμματιστικά αντικείμενα, αντιπαραθέτοντας τη δυνατότητα που προστέθηκε πλέον στο Scratch, αυτό να γίνεται ακόμα και δυναμικά, κατά την εκτέλεση του προγράμματος.

Κάποιες προγραμματιστικές έννοιες, όπως η δομή της επανάληψης, είναι δύσκολο να ενταχθούν στη διδασκαλία του προγραμματισμού μέσω AI, και να έχουν πραγματικό νόημα, ξεφεύγοντας από τις κλασικές ασκήσεις που διδάσκονται σε ένα text-based εργαλείο προγραμματισμού. Στις περισσότερες περιπτώσεις, στο AI η επαναληπτική εκτέλεση των εντολών αντικαθίσταται από το μηχανισμό του καθοδηγούμενου από γεγονότα προγραμματισμού.

Τέλος, εκφράστηκε προβληματισμός από τους εκπαιδευτικούς για τη χρήση κινητών τηλεφώνων την ώρα του μαθήματος, καθώς τη θεωρούν απαραίτητη, παρά το γεγονός ότι το AI παρέχει και προσομοιωτή (emulator). Αναφέρθηκαν, τέλος, στην εγκύκλιο του Υπουργείου Παιδείας που απαγορεύει τη χρήση τέτοιων συσκευών στο χώρο του σχολείου.

Συμπερασματικά, όλοι συμφωνούν πως αξίζει να έρθουν οι μαθητές σε επαφή με το AI, αλλά χρειάζεται ορισμένες βελτιώσεις για να αποτελέσει την πρώτη επιλογή ως εργαλείο εκμάθησης προγραμματισμού σε αρχάριους. Ένας μαθητής μικρής ηλικίας, θα δυσκολευτεί αρκετά, τουλάχιστον στα αρχικά στάδια χρήσης του, οπότε καλύτερα, προτού χρησιμοποιηθεί το AI, να έχει προηγηθεί η ενασχόληση των μαθητών με κάποιο απλούστερο και περισσότερο δοκιμασμένο περιβάλλον οπτικού προγραμματισμού, όπως είναι το Scratch.

8.8 Αποτίμηση του πειράματος

Το πείραμα έδειξε ότι όλοι οι μαθητές, ανεξαρτήτως ηλικίας και υποβάθρου, κατάφεραν να ανταποκριθούν σε παρόμοιο βαθμό στις απαιτήσεις του εργαστηρίου και στην υλοποίηση των δραστηριοτήτων, χωρίς ιδιαίτερη βοήθεια από τον εκπαιδευτικό. Στο πλαίσιο της εποικοδομιστικής προσέγγισης, ο ρόλος των εκπαιδευτικών ήταν περισσότερο υποστηρικτικός και καθοδηγητικός, όταν αυτό κρινόταν απαραίτητο, σε μια προσπάθεια οι μαθητές να αντιμετωπίζουν οι ίδιοι τα προβλήματα που συναντούσαν, κάτι που έδωσε περιθώρια στους μαθητές να νιώσουν ελεύθεροι να εξερευνήσουν το εργαλείο και να κάνουν δοκιμές.

Η ελευθερία που δόθηκε στους μαθητές να εργαστούν και να συνεργαστούν με το δικό τους ρυθμό, οδήγησε σε διαφορετικές μαθησιακές πορείες ανά ομάδα μαθητών, καθώς, συχνά, επέλεξαν την υλοποίηση δραστηριοτήτων με βάση τα προσωπικά τους ενδιαφέροντα, από τις εναλλακτικές που παρέχονταν στα φύλλα εργασίας. Αυτός ο σχεδιασμός συνέβαλε στην διατήρηση του ενδιαφέροντος των μαθητών σε υψηλό επίπεδο μέχρι το τέλος των συναντήσεων.

Η ανατροφοδότηση των μαθητών κατά τη διεξαγωγή των συναντήσεων, οδήγησε συχνά σε μικροαλλαγές στο εκπαιδευτικό υλικό, από το ένα μάθημα στο άλλο, για την κάλυψη των αναγκών των συμμετεχόντων. Οι δυσκολίες που παρατηρήθηκαν σε ορισμένα σημεία των φύλλων εργασίας, ενδεχομένως να αντιμετωπίζονταν αποτελεσματικότερα, εάν κάποιες δραστηριότητες επανασχεδιάζονταν, με την προσθήκη απλούστερων παραδειγμάτων και ασκήσεων πριν από την υλοποίηση των κυρίως εφαρμογών. Όμως, για τις ανάγκες της έρευνας, τα δυο εκπαιδευτικά εργαστήρια όφειλαν να περιλαμβάνουν κοινές δραστηριότητες και για τις δυο ομάδες μαθητών.

Κεφάλαιο 9

Συμπεράσματα

Μετά την επεξεργασία και ανάλυση των ευρημάτων του πειράματος, που ακολούθησε τη βιβλιογραφική ανασκόπηση, στο κεφάλαιο αυτό γίνεται παρουσίαση των συμπερασμάτων από την όλη ερευνητική διαδικασία. Παρουσιάζονται οι περιορισμοί που προέκυψαν, σκόπια ή μη, και προτείνονται ιδέες για μελλοντικές έρευνες.

9.1 Συμπεράσματα της έρευνας

Βασικός σκοπός της μεταπτυχιακής διατριβής ήταν η διερεύνηση της δυνατότητας αξιοποίησης του App Inventor για τη διευκόλυνση της διδασκαλίας και μάθησης βασικών εννοιών προγραμματισμού στη δευτεροβάθμια εκπαίδευση. Διαπιστώθηκε πως η αξιοποίηση των φορητών συσκευών των μαθητών σε συνδυασμό με τη χρήση του εργαλείου AI συνέβαλε θετικά στην εκμάθηση βασικών προγραμματιστικών εννοιών στους μαθητές του δείγματος. Τα συμπεράσματα που απαντάνε και στα ερευνητικά ερωτήματα, που σχετίζονται με τον παραπάνω σκοπό, παρουσιάζονται στη συνέχεια.

Μέσα από τη βιβλιογραφική επισκόπηση, παρατηρήσαμε ότι ο προγραμματισμός, παρά τη σημασία του, αποτελεί ένα δύσκολο αντικείμενο για τους μαθητές. Ένας από τους τρόπους που προτείνονται για την αντιμετώπιση των σχετικών δυσκολιών και παρανοήσεων, είναι η χρήση ειδικών εκπαιδευτικών περιβαλλόντων προγραμματισμού. Τα αποτελέσματα από την επιλογή του AI ως τέτοιου εργαλείου, σε συνδυασμό με την προτεινόμενη διδακτική προσέγγιση, αποτιμήθηκαν μέσα από την καταγραφή των δυσκολιών που αντιμετώπισαν οι συμμετέχοντες μαθητές.

Από τα ευρήματα της έρευνάς μας, από αυτή τη μελέτη περίπτωσης, συμπεραίνουμε ότι τόσο οι μαθητές, όσο και οι εκπαιδευτικοί συμφωνούν πως το App Inventor αποτελεί ένα περιβάλλον προγραμματισμού με ελκυστικά χαρακτηριστικά, είναι εύκολο και ευχάριστο στη χρήση. Διαπιστώθηκε ότι αποτελεί κίνητρο για τους μαθητές, η ευκαιρία αξιοποίησης των ιδιαίτερων χαρακτηριστικών και λειτουργιών των αγαπημένων τους προσωπικών φορητών συσκευών για την ανάπτυξη πλήρως λειτουργικών και αυθεντικών εφαρμογών. Μάλιστα, το γεγονός της άμεσης εκτέλεσης και δοκιμής των αναπτυσσόμενων εφαρμογών στις συσκευές τους, ενθουσίασε από την πρώτη στιγμή τους μαθητές. Η διεπαφή οπτικού προγραμματισμού που διαθέτει το AI, του επιτρέπει να διατηρήσει τα πλεονεκτήματα των εργαλείων αυτού του τύπου, αποτρέποντας τους μαθητές από συντακτικά λάθη, παρέχοντας οπτικά ερεθίσματα και συμβάλλοντας σε μια πιο διαισθητική κατανόηση των βασικών δομών. Τα παραπάνω γεγονότα, συνδυαζόμενα, εξοικονομούν χρόνο, αλλά κυρίως μειώνουν το απαιτούμενο νοητικό φορτίο και δίνουν την ευκαιρία στους μαθητές να πειραματιστούν και να μάθουν από τα λάθη τους, επικεντρώνοντας στη δομή και επίλυση του προβλήματος. Παρατηρήθηκε μάλιστα, εξ αρχής, έντονη τάση των μαθητών για απόδοση προσωπικών «πινελιών» στις εφαρμογές, τόσο στο σχεδιαστικό τμήμα, όσο και σε αυτό των λειτουργιών των εφαρμογών.

Οι μαθητές ολοκλήρωσαν επιτυχώς τις δραστηριότητες των φύλλων εργασίας, καθώς και τις επεκτάσεις τους, ερχόμενοι σε επαφή με τις περισσότερες προγραμματιστικές έννοιες και δομές στις οποίες εστίασε η έρευνα, χωρίς να αντιμετωπίσουν ιδιαίτερες δυσκολίες στην αξιοποίησή τους. Διαπιστώσαμε πως, για τις έννοιες του δομημένου προγραμματισμού, στις οποίες έχουν καταγραφεί βιβλιογραφικά αρκετά προβλήματα και παρανοήσεις, οι μαθητές του δείγματος δεν αντιμετώπισαν ιδιαίτερα προβλήματα.

Πιο συγκεκριμένα, το AI συνέβαλε θετικά στην κατανόηση και αξιοποίηση από τους μαθητές των λογικών εκφράσεων και της δομής επιλογής, κυρίως λόγω της οπτικής αναπαράστασης που παρείχαν τα σχετικά πλακίδια. Επίσης, η ανάγκη για διάσπαση ενός προβλήματος σε

υποπροβλήματα και η δημιουργία διαδικασιών χωρίς παραμέτρους, έγιναν άμεσα αντιληπτές από τους μαθητές, οι οποίοι προχώρησαν πολύ αβίαστα στην αξιοποίησή τους. Δεν συνέβη το ίδιο στη δημιουργία διαδικασιών με παραμέτρους, στις οποίες δεν προλάβσαμε να αφιερώσουμε αρκετό χρόνο.

Ένα σημείο, όμως, όπου παρατηρούνται δυσκολίες από τους μαθητές με βάση τη βιβλιογραφία, και στο οποίο το AI δεν συνέβαλε θετικά, αλλά ούτε και αρνητικά, ήταν η ικανότητα των μαθητών να αποφασίζουν και να ορίζουν οι ίδιοι τις απαραίτητες μεταβλητές στις εφαρμογές τους. Η επεξεργασία των τιμών τους, εφόσον αυτές είχαν ήδη οριστεί, δεν αποτέλεσε πρόβλημα για τους μαθητές. Η επιλογή όμως και ο ορισμός εκείνων των μεταβλητών που απαιτούνται, ήταν ένα πρόβλημα που εξακολούθησε να υφίσταται στους μαθητές, παρά την παρέμβασή μας με τη χρήση του AI.

Τα υπόλοιπα προβλήματα που εντοπίστηκαν, έχουν να κάνουν με τον κόπο των μαθητών να εντοπίζουν τις απαιτούμενες εντολές, με παραλείψεις τους στην ενημέρωση ετικετών για εμφάνιση τιμών στην οθόνη και κυρίως, με τη χρήση λειτουργιών του ταυτόχρονου προγραμματισμού.

Συμπερασματικά, το AI μπορεί να χρησιμοποιηθεί για τη διδασκαλία και μάθηση όλων των βασικών προγραμματιστικών εννοιών και δομών που προβλέπονται στα Προγράμματα Σπουδών της Δευτεροβάθμιας εκπαίδευσης στην Ελλάδα, καθώς ικανοποιεί τους στόχους τους και μάλιστα, εφόσον το επιθυμεί ο εκπαιδευτικός, με αντισυμβατικό τρόπο, κάνοντας δυνατή τη χρήση νέων παραδειγμάτων, πέρα από εκείνα που χρησιμοποιούνται παραδοσιακά στα συμβατικά περιβάλλοντα προγραμματισμού. Ιδιαίτερα σημαντικό όμως πλεονέκτημά του είναι ότι επιτρέπει στους μαθητές να δημιουργούν εφαρμογές που έχουν πρακτικό νόημα για τους ίδιους και μπορούν να τις μεταφέρουν πάντα μαζί τους.

Το AI μοιάζει σε αρκετά σημεία με το Scratch, προσφέροντας επιπλέον λειτουργίες λόγω της φύσης των συσκευών στις οποίες απευθύνεται. Όμως, το Scratch, λόγω και της μεγάλης και ενεργής κοινότητας χρηστών που διαθέτει, αναπτύσσεται και εξελίσσεται με γρήγορους ρυθμούς, με νέες δυνατότητες να προστίθενται συχνά σε αυτό. Το AI, ως σχετικά νέο, ελεύθερο και ανοικτό λογισμικό, έχει προοπτικές ως προς την ανάπτυξη σχετικής κοινότητας μάθησης, καθώς και του ίδιου του λογισμικού. Το AI και οι χρήστες του θα μπορούσαν να επωφεληθούν επίσης από την ανάπτυξη μιας πλατφόρμας διαμοιρασμού των παραγόμενων έργων, παρόμοιας

με εκείνης του Scratch, το οποίο διαθέτει μια ιδιαιτέρως ενεργή online κοινότητα διαμοιρασμού των έργων και αλληλεπίδρασης των χρηστών της.

Εφόσον η κοινότητα και οι χρήστες του AI αυξηθούν, η μετάφρασή του στα ελληνικά θα είναι θέμα χρόνου. Το τελευταίο, βέβαια, δεν φάνηκε να επηρέασε ιδιαίτερα τους μαθητές του δείγματος. Αντιθέτως, εντοπίστηκαν ορισμένες ελλείψεις σε χρήσιμες λειτουργίες, όπως η αδυναμία κλωνοποίησης αντικειμένων, δημιουργίας αντικειμένων από το χρήστη ή δυναμικής δημιουργίας τους κατά την εκτέλεση του προγράμματος, αλλά και η έλλειψη της εντολής αναίρεσης. Εάν αυτοί οι περιορισμοί επιλυθούν, τότε το App Inventor θα αποτελέσει ένα ακόμα πληρέστερο και πιο εύχρηστο περιβάλλον για την ανάπτυξη εφαρμογών από μαθητές, ανεξαρτήτως υποβάθρου.

Η χρήση του AI θα έχει ακόμα μεγαλύτερα οφέλη για τους μαθητές και τους εκπαιδευτικούς αν δεν αποτελέσει το πρώτο εργαλείο οπτικού προγραμματισμού με το οποίο έρχονται σε επαφή οι μαθητές. Για μικρότερης ηλικίας και αρχάριους στον προγραμματισμό μαθητές, απαιτείται κάποιος χρόνος για την εξοικείωσή τους με αυτό, ο οποίος συνήθως δεν διατίθεται λόγω πίεσης που αισθάνονται οι εκπαιδευτικοί στην κάλυψη της διδακτέας ύλης. Προτείνεται λοιπόν, προτού χρησιμοποιηθεί το AI, να έχει προηγηθεί η ενασχόληση των μαθητών με κάποιο απλούστερο περιβάλλον οπτικού προγραμματισμού, όπως είναι το Scratch. Σύμφωνα με τα πιλοτικά Προγράμματα Σπουδών για την Πληροφορική στο Δημοτικό σχολείο και στο Γυμνάσιο, ο μέσος μαθητής του δημοτικού έχει ήδη γνωρίσει το Scratch ή κάποιο παρόμοιο εργαλείο στις Ε' και ΣΤ' τάξεις, ενώ στο Γυμνάσιο, το αντικείμενο του προγραμματισμού διδάσκεται και στις τρεις τάξεις. Τα παραπάνω μας οδηγούν στην πρόταση το AI να χρησιμοποιηθεί στην Γ' τάξη Γυμνασίου ή/και στις πρώτες τάξεις του Λυκείου.

Τέλος, προτείνεται να επιτρέπεται η περιορισμένη χρήση φορητών συσκευών από τους μαθητές σε σχολικό περιβάλλον για εκπαιδευτικούς σκοπούς, έτσι ώστε να γίνει δυνατή η πλήρης εκμετάλλευση των ιδιαίτερων δυνατοτήτων που μπορεί να προσφέρει το AI. Η χρήση των κινητών τηλεφώνων από τους μαθητές του δείγματος έδειξε ότι οι μαθητές δεν αποσπάστηκαν από τη χρήση των συσκευών τους, αντιθέτως η ελευθερία φάνηκε να τους ενθουσίασε και να τους έδωσε ισχυρότερα κίνητρα να εργαστούν, όπως άλλωστε έχουν δείξει και σχετικές διεθνείς έρευνες για την πρακτική «Φέρε τη δική σου συσκευή» (BYOD). Εναλλακτικά, θα μπορούσαν να διατίθενται συσκευές στους μαθητές από το σχολείο, αποκλειστικά για χρήση στο εργαστήριο πληροφορικής, ώστε να αποφευχθούν τα προβλήματα που οδήγησαν το Υπουργείο Παιδείας στην απαγόρευση χρήσης τέτοιων συσκευών στα σχολεία. Άλλωστε, το κόστος των παραπάνω

συσκευών δεν είναι πλέον απαγορευτικό, αντιστοιχίζοντας μια συσκευή με λειτουργικό σύστημα Android ανά Η/Υ.

9.2 Περιορισμοί της έρευνας

Ο αρχικός σχεδιασμός για το πλαίσιο στο οποίο θα διεξαγόταν το πείραμα, ήταν αυτό να πραγματοποιηθεί σε πραγματικές συνθήκες σχολικής τάξης. Κάτι τέτοιο δεν κατέστη εφικτό να πραγματοποιηθεί από τον ίδιο τον ερευνητή - εκπαιδευτικό πληροφορικής, καθώς λόγω υπηρεσιακών αναγκών που προέκυψαν κατά τη διάρκεια εκπόνησης της Μεταπτυχιακής Διατριβής, αυτός διατέθηκε για το τρέχον σχολικό έτος σε σχολική μονάδα της Πρωτοβάθμιας εκπαίδευσης. Η δυσκολία να βρεθούν εκπαιδευτικοί για τη διεξαγωγή του πειράματος με τους ακριβείς όρους του σχεδιασμού του, σε συνδυασμό με τους περιορισμένους πόρους και τη χρονική πίεση για την έγκαιρη ολοκλήρωση του πειράματος, μας ανάγκασαν να περιορίσουμε την έκταση του πειράματος σε δυο εκπαιδευτικά εργαστήρια με συγκεκριμένα χαρακτηριστικά.

Συγκεκριμένα, η ομάδα μαθητών της Μαθητικής Στέγης Καρδαμύλων Χίου είχε την ιδιαιτερότητα να «υποχρεωθεί» να παρακολουθήσει τα δώρα μαθήματα για τέσσερα απογεύματα Σαββάτου, ως προγραμματισμένη δραστηριότητα του οικοτροφείου. Από την άλλη, στη δεύτερη ομάδα μαθητών, παρά το γεγονός ότι τα παιδιά συμμετείχαν και αυτά σε έξι δίωρες συναντήσεις, απογεύματα Σαββάτου, ήταν τα ίδια τα οποία ζήτησαν να συμμετάσχουν στο εκπαιδευτικό πρόγραμμα, γνωρίζοντας εξ αρχής τις τυπικές υποχρεώσεις τους.

Επίσης, το τελικό δείγμα ήταν σχετικά μικρό, καθώς αποτελούνταν από 35 μαθητές. Δυστυχώς, στο σύνολο αυτό, περιλαμβάνονταν μόνο τρία κορίτσια, οπότε δεν μπορέσαμε να βγάλουμε συμπεράσματα για την επίδραση που είχε το ΑΙ ως προς το φύλο. Σημαντική επίτευξη, πάντως, αποτελεί το γεγονός ότι παρά τους περιορισμούς, το παραπάνω δείγμα ήταν αντιπροσωπευτικό ως προς τις ηλικίες των μαθητών, καθώς υπήρχαν συμμετέχοντες από όλες τις τάξεις που μας ενδιέφερε να εξετάσουμε. Παρόλα αυτά, λόγω του μεγέθους του δείγματος, της γεωγραφικής του κατανομής και των ιδιαίτερων χαρακτηριστικών των δυο ομάδων, δεν μπορούμε να γενικεύσουμε τα συμπεράσματά μας στο σύνολο του μαθητικού πληθυσμού.

9.3 Προτάσεις για μελλοντική έρευνα

Η αξιοποίηση ψηφιακών φορητών συσκευών στην ελληνική εκπαίδευση, ειδικά για την μάθηση και διδασκαλία προγραμματισμού, είναι ένας τομέας που δεν έχει διερευνηθεί σε βάθος.

Μέσα από τα ευρήματα της έρευνας διαφάνηκε ότι η χρήση του AI σε μαθητές Δ/θμιας εκπαίδευσης ενίσχυσε μεν τη διάθεσή τους για εκμάθηση και αξιοποίηση βασικών προγραμματιστικών εννοιών, δεν διερευνήθηκε ιδιαίτερα, όμως, εάν ενισχύθηκε η ικανότητα των μαθητών να επιλύουν προβλήματα, πέρα από εκείνα που τους δόθηκαν. Λόγω περιορισμού στο χρόνο, δεν προλάβσαμε να εξετάσουμε τη συμβολή του AI στη διδασκαλία και μάθηση δομών δεδομένων, ενώ ακόμα σημαντικότερη παρατήρηση είναι ότι δεν δόθηκε η ευκαιρία στους μαθητές να επιχειρήσουν την υλοποίηση εντελώς δικών τους εφαρμογών και ακολούθως στους εκπαιδευτικούς και στους μαθητές να τις αξιολογήσουν.

Ιδανικά, η χρήση του AI θα έπρεπε να αξιολογηθεί σε συνθήκες πραγματικής τάξης και μάλιστα, θα παρουσίαζε ενδιαφέρον να μελετηθεί η επίδρασή του ως προς τη δημιουργικότητα και δέσμευση των μαθητών σε σχέση με το φύλο τους.

Τέλος, θα μπορούσε να συγκριθεί η συμβολή του AI, χρησιμοποιώντας ομάδα ελέγχου, για τη σύγκριση της επίδρασής του στη διδασκαλία και μάθηση συγκεκριμένων εννοιών προγραμματισμού, σε σχέση με κάποιο άλλο παρόμοιο εργαλείο ανάπτυξης εφαρμογών, σε σχολικά τμήματα μαθητών με περισσότερα κοινά χαρακτηριστικά και σταθερό βαθμό επίδρασης των εξωτερικών παραγόντων που επηρεάζουν το πείραμα.

9.4 Επίλογος

Οι φορητές συσκευές, με τις οποίες είναι πλέον προσκολλημένη η πλειοψηφία των νέων, έχουν τη δυναμική να διευκολύνουν τη διδασκαλία και μάθηση προγραμματιστικών εννοιών, εφόσον αξιοποιηθούν σωστά. Το AI παρέχει ένα ελκυστικό και αυθεντικό περιβάλλον ανάπτυξης εφαρμογών, που παρέχει την ευκαιρία και κίνητρα στα νέα παιδιά να πειραματιστούν και να δουν τις συσκευές αυτές, από την οπτική του δημιουργού και όχι απλά του καταναλωτή της τεχνολογίας. Σημαντικοί παράγοντες για την επιτυχία του παραπάνω εγχειρήματος, είναι η ενίσχυση της κοινότητας ανάπτυξης του AI και η χρήση μαθησιακών δραστηριοτήτων που να αξιοποιούν τις ιδιαίτερες δυνατότητες των συσκευών και να έχουν νόημα στην καθημερινότητα των παιδιών.

Βιβλιογραφικές αναφορές

- [01] A. Aho. "Computation and computational thinking". *Computer Journal*, 55, 832-835, 2012.
- [02] R. Ballagas, M. Rohs, J. Sheridan, J. Borchers. "BYOD: Bring your own device". In *UbiComp 2004 Workshop on Ubiquitous Display Environments*, 2004.
- [03] V. Barr, C. Stephenson. "Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community?". *ACM Inroads*, 2(1), 48-54, 2011.
- [04] M. Ben-Ari, Y. Kolikant. "Thinking Parallel: The Process of Learning Concurrency". *ITiCSE*, pp. 13-16. Cracow, 1998.
- [05] A. Blackwell. "What is programming?". 14th workshop of the Psychology of Programming Interest Group, Brunel, Middlesex, UK. (pp.204-218), 2002.
- [06] P. Brusilovsky, A. Kouchnirenko, P. Miller, I. Tomek. "Teaching programming to novices: A review of approaches and tools". In *Proceedings of ED-MEDIA'94* (pp. 103-110), 1994.
- [07] S. Cooper, S. Cunningham. "Teaching computer science in context". *ACM Inroads*, 1, 5-8, 2010.
- [08] M. Dabney, B. Dean, T. Rogers. "No sensor left behind: enriching computing education with mobile devices". In *Proceeding of the 44th ACM technical symposium on Computer science education* (pp. 627-632). ACM, 2013.
- [09] V. Dagididelis, M. Satratzemi, G. Evangelidis. "Introducing Secondary Education Students to Algorithms and Programming". *Education and Information Technologies*, 9(2), 159-173, 2004.

- [10] C. Davidson. "Why We Need a 4th R: Reading, wRiting, aRithmetic, algoRithms". Digital Media and Learning Research Hub, 2012.
- [11] S. Dehnadi, R. Bornat, R. Adams. "Meta-analysis of the effect of consistency on success in early learning of programming". PPIG, 2009.
- [12] P. Denning, P. Rosenbloom. "The profession of IT Computing: the fourth great domain of science". Communications of the ACM, 52(9), 27-29, 2009.
- [13] B. Du Boulay. "Some difficulties of learning to program". E. Soloway & J. C. Spohrer (Eds.), Studying the Novice Programmer, 283-299, Hillsdale, NJ: Lawrence Erlbaum Associates, 1989.
- [14] G. Fletcher, J. Lu. "Education Human computing skills: rethinking the K-12 experience". Communications of the ACM, 52(2), 23-25, 2009.
- [15] M. Fox, N. Hackerman. "*Evaluating and improving undergraduate teaching in science, technology, engineering, and mathematics*". National Academies Press, 2002.
- [16] M. Goadrich, M. Rogers. "Smart smartphone development: iOS versus Android". In Proceedings of the 42nd ACM technical symposium on Computer science education (pp. 607-612), ACM, 2011.
- [17] A. Gomes, A. Mendes. "Learning to program-difficulties and solutions". In International Conference on Engineering Education-ICEE, 2007.
- [18] K. Gittman. "Bringing Your Own Mobile Device to the Classroom". Research Report. Virginia Tech, 2013.
- [19] J. Gray, A. Abelson, D. Wolber, M. Friend. "Teaching CS principles with app inventor". In Proceedings of the 50th Annual Southeast Regional Conference (ACM-SE '12). ACM, New York, NY, USA, 405-406, 2012.
- [20] S. Grover, R. Pea. "Using a discourse-intensive pedagogy and android's app inventor for introducing computational concepts to middle school students". In Proceeding of the 44th ACM technical symposium on Computer science education (pp. 723-728). ACM, 2013.

- [21] S. Grover, R. Pea. "Computational Thinking in K-12. A Review of the State of the Field". *Educational Researcher*, 42(1), 38-43, 2013.
- [22] M. Guzdial, E. Soloway. "Teaching the Nintendo generation to program". *Communications of the ACM*, 45(4), 17-21, 2002.
- [23] E. Hartnell-Young, N. Heym. "How mobile phones help learning in secondary schools. A report to Becta". *Isri Report. University of Nottingham, 2008*.
- [24] J. Holz, T. Leonhardt, U. Schroeder. "Using smartphones to motivate secondary school students for informatics". In *Proceedings of the 11th Koli Calling International Conference on Computing Education Research* (pp. 89-94). ACM, 2011.
- [25] E. Horowitz, *Fundamental of programming languages*, Science Press, 1984.
- [26] Y-C Hsu, K. Rice, L. Dawley. Empowering educators with Google's Android App Inventor: An online workshop in mobile app design. *British Journal of Educational Technology*, 43(1) E1-E5, 2012.
- [27] J. Huizenga, W. Admiraal, S. Akkerman, G. Dam. "Mobile game-based learning in secondary education: engagement, motivation and learning in a mobile city game". *Journal of Computer Assisted Learning*, 25(4), 332-344, 2009.
- [28] L. Johnson, R. Smith, H. Willis, A. Levine, K. Haywood. "The 2011 Horizon Report". Austin, Texas: The New Media Consortium, 2011.
- [29] L. Johnson, S. Adams, M. Cummins. "NMC Horizon Report: 2012 K-12 Edition". Austin, Texas: The New Media Consortium, 2012.
- [30] L. Johnson, S. Adams Becker, M. Cummins, V. Estrada, A. Freeman, H. Ludgate. "NMC Horizon Report: 2013 K-12 Edition". Austin, Texas: The New Media Consortium, 2013.
- [31] M. Karakus, S. Uludag, E. Guler, S. Turner, A. Ugur. "Teaching computing and programming fundamentals via App Inventor for Android". In *Information Technology Based Higher Education and Training (ITHET), 2012 International Conference on* (pp. 1-8). IEEE, 2012.

- [32] D. Knuth. "The art of computer programming", Volume I, Fundamental Algorithms, Addison-Wesley Inc, 1968.
- [33] A. Kornecki. "Computing curricula for the 21st Century". Distributed Systems Online, IEEE, 9(2), 2-2, 2008.
- [34] S. Kurkovsky. "Engaging students through mobile game development". In ACM SIGCSE Bulletin (Vol. 41, No. 1, pp. 44-48). ACM, 2009.
- [35] E. Lahtinen, K. Ala-Mutka, H. Järvinen. "A study of the difficulties of novice programmers". In ACM SIGCSE Bulletin (Vol. 37, No. 3, pp. 14-18). ACM, 2005.
- [36] I. Lee, F. Martin, J. Denner, B. Coulter, W. Allan, I. Lee, F. Martin, J. Denner, B. Coulter, W. Allan, J. Erickson, J. Malyn-Smith, L. Werner. "Computational thinking for youth in practice". ACM Inroads, 2(1), 32-37, 2011.
- [37] X. Liu, D. Murphy. "Using Mobile Apps to Entice General Education Students into Technology Fields". *Information Systems Education Journal*, 11(1), 25, 2013.
- [38] J. Margolis, A. Fisher. "Unlocking the clubhouse: Women in computing". Cambridge: MIT Press, 2003.
- [39] L. Margulieux, M. Guzdial, R. Catrambone. "Subgoal-labeled instructional material improves performance and transfer in learning to develop mobile applications". In *Proceedings of the ninth annual international conference on International computing education research* (pp. 71-78). ACM, 2012.
- [40] R. Morelli, T. Lanerolle, P. Lake, N. Limardo, E. Tamotsu, C. Uche. "Can Android App Inventor Bring Computational Thinking to K-12?". Proc. 42nd ACM technical symposium on Computer science education (SIGCSE'11), 2011.
- [41] S. Papert. "Mind-Storms, Children, Computers and Powerful Ideas". New York: Basic Books, 1980.
- [42] S. Papert. "Situating constructionism". In I. Harel & S. Papert (Eds.), *Constructionism*. (pp. 1-11). Norwood, NJ: Ablex, 1991.

- [43] S. Papert. "The Children's Machine: Rethinking School in the Age of the Computer". New York: Basic Books, 1993.
- [44] R. Pea. "Language-independent conceptual 'bugs' in the novice programming", *Journal of Educational Computing Research*, 2(1), 25-36, 1986.
- [45] N. Pennington, A. Lee, B. Rehder. "Cognitive activities and levels of abstraction in procedural and object-oriented design". *Human-Computer Interaction*, 10, 171-226, 1995.
- [46] A. Perlis. "The computer in the university". In M. Greenberger, Ed., *Computers and the World of the Future*, MIT Press, Cambridge, MA, 180-219, 1962.
- [47] S. Peyton Jones. "Computing at School International comparisons, Version 5", 2011.
- [48] M. Prensky. "Digital natives, digital immigrants part 1". *On the horizon*, 9(5), 1-6, 2001.
- [49] N. Ragonis. "Computing Pre-University: Secondary Level Computing Curricula". *Wiley Encyclopedia of Computer Science and Engineering*, 2008.
- [50] H. Ramadhan. "Programming by discovery". *Journal of Computer Assisted Learning*, 16(1), 83-93, 2000.
- [51] R. Reichert. "Theory of computation as a vehicle for teaching fundamental concepts of computer science". Dissertation No. 15035, ETH Zürich, 2003.
- [52] A. Repenning, D. Webb, A. Ioannidou. "Scalable game design and the development of a checklist for getting computational thinking into public schools". In *Proceedings of the 41st ACM technical symposium on Computer science education* (pp. 265-269). ACM, 2010.
- [53] M. Resnick. "Sowing the seeds for a more creative society". *Learning and Leading with Technology*, 35(4), 18, 2007.
- [54] M. Resnick, J. Maloney, A. Monroy-Hernandez, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman, Y. Kafai. "Scratch: Programming for All". *Communications of the ACM*, vol. 52, no. 11, pp. 60-67, 2009.

- [55] J. Rogalski, C. Samurcay. "Acquisition of programming knowledge & skills". In J.M. Hoc, T.R.G. Green, R. Samurcay & D.J. Gillmore (Eds.), *Psychology of Programming* (pp. 157–174). London: Academic Press, 1990.
- [56] R. Roque. "OpenBlocks: an extendable framework for graphical block programming systems". Dept. of Electrical Engineering and Computer Science, MIT, 2007.
- [57] K. Roy. "App inventor for android: report from a summer camp". In *Proceedings of the 43rd ACM technical symposium on Computer Science Education (SIGCSE '12)*. ACM, New York, NY, USA, 283-288, 2012.
- [58] M. Sime, A. Arblaster, T. Green. "Reducing programming errors in nested conditionals by prescribing a writing procedure". *International Journal of Man-Machine Studies*, 9, 119-126, 1977
- [59] I. Singh, M. Palmieri. "Comparison of cross-platform mobile development tools". In *16th International Conference on Intelligence in Next Generation Networks, 2012*.
- [60] J. Spohrer, E. Soloway. "Alternatives to construct-based programming misconceptions". In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '86)*, Marilyn Mantei and Peter Orbeton (Eds.). ACM, New York, NY, USA, 183-191, 1986.
- [61] S. Tigrek, M. Obadat. "Teaching smartphones programming using (Android Java): Pedagogy and innovation". In *Information Technology Based Higher Education and Training (ITHET), 2012 International Conference on* (pp. 1-7), IEEE, 2012.
- [62] F. Turbak, S. Sandu, O. Kotsopoulos, E. Erdman, E. Davis, K. Chadha. "Blocks languages for creating tangible artifacts". In *Visual Languages and Human-Centric Computing (VL/HCC), 2012 IEEE Symposium on* (pp. 137-144), IEEE, 2012.
- [63] C. Wilson, L. Sudol, C. Stephenson, M. Stehlik. "Running on empty: The failure to teach K-12 computer science in the digital age". ACM and CSTA, 2010.
- [64] J. Wing. "Computational thinking". *Communications of the ACM*, 49(3), 33-35, 2006.

- [65] J. Wing. "Computational thinking and thinking about computing". *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717-3725, 2008.
- [66] J. Wing. "Computational Thinking: What and Why". *The Link*. Spring 2011. Carnegie-Melon University, 2011.
- [67] D. Wolber. "App inventor and real-world motivation". 42nd ACM technical symposium on Computer science education (SIGCSE '11). ACM, New York, NY, USA, 601-606, 2011.
- [68] Κ. Γλέζου, Ε. Σταμούλη, Μ. Γρηγοριάδου. «Εναλλακτική Προσέγγιση Διδασκαλίας της Δομής Επιλογής για Αρχάριους Προγραμματιστές με Αξιοποίηση του MicroWorlds Pro». 3^ο Πανελλήνιο Συνέδριο «Διδακτική της Πληροφορικής», Πανεπιστήμιο Πελοποννήσου, Κόρινθος, 2005.
- [69] Μ. Γρηγοριάδου, Α. Γόγουλου, Ε. Γουλή. «Εναλλακτικές Διδακτικές Προσεγγίσεις σε Εισαγωγικά Μαθήματα Προγραμματισμού: Προτάσεις Διδασκαλίας». 3^ο Συνέδριο ΕΤΠΕ, Τόμος Α', σ. 239-248, Ρόδος, 2002.
- [70] Β. Δαγδιλέλης. «Η διδασκαλία του προγραμματισμού: αντιλήψεις των σπουδαστών για την κατασκευή και επικύρωση προγραμμάτων και διδακτικές καταστάσεις για τη διαμόρφωσή τους». Διδακτορική διατριβή. Τμήμα Εφ. Πληροφορικής, Πανεπιστήμιο Μακεδονίας, 1996.
- [71] Β. Εφόπουλος, Γ. Ευαγγελίδης, Β. Δαγδιδέλης, Α. Κλεφτοδήμος. «Οι δυσκολίες των Αρχάριων Προγραμματιστών». 3^ο Πανελλήνιο Συνέδριο «Διδακτική της Πληροφορικής», Κόρινθος, 2005.
- [72] Β. Κόμης. «Μελέτη Βασικών Εννοιών του Προγραμματισμού στο Πλαίσιο μιας Οικοδομιστικής Διδακτικής Προσέγγισης», *Θέματα στην Εκπαίδευση*, 2(2-3), σ. 243-270, 2001.
- [73] Β. Κόμης. «Πληροφορική στην εκπαίδευση». Τόμος Β', Πάτρα, Ομάδα εκτέλεσης έργου ΕΑΠ/1997-2001.

- [74] Β. Κόμης. «Εισαγωγή στη Διδακτική της Πληροφορικής», Εκδόσεις Κλειδάριθμος, Αθήνα, 2005.
- [75] Β. Κόμης, Α. Τζιμογιάννης. «Ο προγραμματισμός ως μαθησιακή δραστηριότητα: από τις εμπειρικές προσεγγίσεις στη γνώση παιδαγωγικού περιεχομένου». Θέματα στην Εκπαίδευση, 7(3), σ. 229-255, 2006.
- [76] Μ. Κορδάκη, Π. Ψώμος. «Μια πρόταση διδασκαλίας του Προγραμματισμού μέσω δημιουργίας Εκπαιδευτικών Ψηφιακών Αφηγήσεων στο περιβάλλον Storytelling Alice». 6ο Πανελλήνιο Συνέδριο «Διδακτική της Πληροφορικής», Φλώρινα, 2012.
- [77] Ι. Κοσμοπούλου, Χ. Φλώρου, Α. Μπαγιάτη, Η. Χούστης. «Ανάπτυξη Διαδραστικής Εφαρμογής για τη Διδασκαλία του Προγραμματισμού στο Δημοτικό με Χρήση του Προγράμματος Scratch, βασισμένη σε Rubrics Αξιολόγησης και Αυτο-αξιολόγησης». 5ο Συνέδριο «Διδακτική της Πληροφορικής», Αθήνα, 2010.
- [78] Ι. Κοτίνη, Σ. Τζελέπη. «Η Συμβολή της Υπολογιστικής Σκέψης στην Προετοιμασία του Αυριανού Πολίτη». 4ο συνέδριο «Η Πληροφορική στην Εκπαίδευση». Πειραιάς, 2012.
- [79] Τ. Λαδιάς. «Ο προγραμματισμός Η/Υ στο νέο Πρόγραμμα Σπουδών της υποχρεωτικής εκπαίδευσης στο πλαίσιο του μαθήματος για τον Πληροφορικό Γραμματισμό», 2012.
- [80] Α. Μάργαρης, Μ. Παπαστεργίου. «Εισάγοντας αρχάριους στον προγραμματισμό με τα περιβάλλοντα Kara: Μια προσέγγιση βασισμένη στη θεωρία υπολογισμού». 4ο Πανελλήνιο Συνέδριο με Διεθνή Συμμετοχή «Διδακτική της Πληροφορικής», σ. 81-90, Πάτρα, 2008.
- [81] Γ. Μαυροχαλυβίδης, Γ. Μακρής, Ν. Μπέκος. «Διδακτική προσέγγιση του Αντικειμενοστραφούς Προγραμματισμού με το Scratch». 6ο Πανελλήνιο Συνέδριο «Διδακτική της Πληροφορικής», Φλώρινα, 2012.
- [82] Γ. Μπουκέας. «Βασικές Έννοιες Πληροφορικής για Ένα Μάθημα Γενικής Παιδείας». 7ο Πανελλήνιο Συνέδριο Καθηγητών Πληροφορικής, Θεσσαλονίκη, 2013.
- [83] Δ. Νικολός, Β. Κόμης. «Μια διδακτική πρόταση για τη γλώσσα προγραμματισμού Scratch». 5ο Συνέδριο «Διδακτική της Πληροφορικής», Αθήνα, 2010.

- [84] Σ. Ξυνόγαλος, Μ. Σατρατζέμη, Β. Δαγδιδέλης. «Η εισαγωγή στον προγραμματισμό: Διδακτικές Προσεγγίσεις και Εκπαιδευτικά Εργαλεία». 2ο Συνέδριο ΕΤΠΕ, Πάτρα, 2000.
- [85] Σ. Παπαδάκης, Μ. Καλογιαννάκης, Ν. Ζαράνης. «Δημιουργώντας εφαρμογές για έξυπνες φορητές συσκευές με το AppInventor». 7ο Πανελλήνιο Συνέδριο Καθηγητών Πληροφορικής «Η Πληροφορική στην Πρωτοβάθμια και Δευτεροβάθμια Εκπαίδευση. Προκλήσεις & Προοπτικές», Πανεπιστήμιο Μακεδονίας, Θεσσαλονίκη, 2013.
- [86] Γ. Παπαευθυμίου. «Διδακτική του Προγραμματισμού για μαθητές Γυμνασίου, με χρήση του εργαλείου γραφικού προγραμματισμού Scratch». Διπλωματική. Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Πειραιά, 2008.
- [87] Ι. Σαρημπαλίδης. «Μάθηση Προγραμματισμού Η/Υ από μαθητές Α' Λυκείου με το Scratch». 6ο Πανελλήνιο Συνέδριο «Διδακτική της Πληροφορικής», Φλώρινα, 2012.
- [88] Α. Τζιμογιάννης, Β. Κόμης. «Επίλυση προβλημάτων σε προγραμματιστικό περιβάλλον: η οικοδόμηση της δομής ελέγχου από τους μαθητές του Ενιαίου Λυκείου». Πρακτικά 4ου Πανελληνίου Συνεδρίου Διδακτικής της Πληροφορικής, σ. 221-230, Πάτρα, 2009.
- [89] Α. Τζιμογιάννης. «Η διδασκαλία του Προγραμματισμού Η/Υ στη Δευτεροβάθμια Εκπαίδευση. Δυσκολίες και αντιλήψεις των μαθητών για την έννοια της μεταβλητής». Η Βάση, 2, σ. 35-42, 2000.
- [90] Γ. Φεσάκης, Δ. Δημητρακοπούλου. «Γνωστικές δυσκολίες μαθητών δευτεροβάθμιας εκπαίδευσης σχετικά με την έννοια της προγραμματιστικής μεταβλητής και προτεινόμενες παρεμβάσεις». 3ο πανελλήνιο συνέδριο «Διδακτική της Πληροφορικής», Κόρινθος, 2005.
- [91] Γ. Φεσάκης, Κ. Σεραφείμ. «Μάθηση προγραμματισμού ΗΥ από εκκολαπτόμενους εκπαιδευτικούς με το SCRATCH». 1ο Εκπαιδευτικό Συνέδριο «Ένταξη και Χρήση των ΤΠΕ στην Εκπαιδευτική Διαδικασία», Βόλος, 2009.
- [92] Δ. Χασανίδης, Θ. Μπράτιτσης. «Μαθήματα αλγοριθμικής σκέψης στη Γ' Λυκείου, με χρήση του Scratch: Μια πρόταση για τη διδασκαλία της δομής επιλογής». 5ο Συνέδριο «Διδακτική της Πληροφορικής», Αθήνα, 2010.

Κατάσταση ακρωνυμίων

AI	App Inventor
ACM	Association for Computing Machinery
BYOD	Bring Your Own Device
CS	Computer Science
CSTA	Computer Science Teacher Association
GPS	Global Positioning System
IDE	Integrated Development Environment
IT	Information Technology
ICT	Information and Communication Technology
MIT	Massachusetts Institute of Technology
SDK	Software Development Kit
SMS	Short Message Service
VM	Virtual Machine
ΑΕΠΠ	Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον
ΑΠΣ	Αναλυτικό Πρόγραμμα Σπουδών
ΓΕΛ	Γενικό Λύκειο
Δ/θμια	Δευτεροβάθμια
ΕΑΕΠ	Ενιαίο Αναμορφωμένο Εκπαιδευτικό Πρόγραμμα
ΕΛ/ΛΑΚ	Ελεύθερο Λογισμικό / Λογισμικό Ανοικτού Κώδικα
ΕΠΑΛ	Επαγγελματικό Λύκειο
Π/θμια	Πρωτοβάθμια
ΤΕΕ	Τεχνολογικό Επαγγελματικό Εκπαιδευτήριο
ΤΠΕ	Τεχνολογία Πληροφοριών και Επικοινωνίας
ΥΠΕΠΘ	Υπουργείο Εθνικής Παιδείας και Θρησκευμάτων

Παράρτημα Α

Σύνοψη Δραστηριοτήτων

Το πλήρες υλικό διατίθεται μέσω της ιστοσελίδας του Συλλόγου Εκπαιδευτικών Πληροφορικής Χίου στη διεύθυνση <http://www.sepchiou.gr/ai> και αποτελείται από δυο μέρη, συνολικά περίπου 200 σελίδων. Στη συνέχεια γίνεται μια συνοπτική περιγραφή των δραστηριοτήτων που περιλαμβάνονται στα τρία πρώτα μαθήματα και που χρησιμοποιήθηκαν στις ομάδες μαθητών που συμμετείχαν στα εκπαιδευτικά προγράμματα.

Ένα μικρό δείγμα από τις εφαρμογές που αναπτύχθηκαν και περιλαμβάνονται στις δραστηριότητες των εργαστηρίων διατίθεται στο Google Play ⁶⁴.

⁶⁴ Δείγμα εφαρμογών στο Google Play:

<https://play.google.com/store/apps/developer?id=Σύλλογος+Εκπαιδευτικών+Πληροφορικής+Χίου>

A.1 Μάθημα 1

Στόχοι για τους μαθητές:

- Να διακρίνουν το τμήμα της σχεδίασης από το τμήμα εντολών και να μπορούν να τοποθετούν σε αυτά, αντικείμενα (Components) και εντολές (blocks), αντίστοιχα.
- Να μεταφορτώνουν αρχεία πολυμέσων (εικόνας και ήχου) στο περιβάλλον του AI.
- Να διακρίνουν διάφορα απλά γεγονότα που μπορούν να προκληθούν σε μια κινητή συσκευή, όπως το άγγιγμα της οθόνης ή ενός κουμπιού ή το ταρακούνημα της συσκευής.
- Να συνδέουν τα γεγονότα με συγκεκριμένες ενέργειες.
- Να τοποθετούν τις ενέργειες αυτές με τη σωστή σειρά εκτέλεσης.
- Να δίνουν τιμές στις ιδιότητες των αντικειμένων που χρησιμοποιούν.
- Να χρησιμοποιούν τις εντολές παραγωγής τυχαίων αριθμών.
- Να δημιουργούν κίνηση αντικειμένων σε καμβά.
- Να δοκιμάζουν την εφαρμογή που αναπτύσσουν στην φορητή τους συσκευή.

1^η Δραστηριότητα: Ζάρια (v.1)

Περιγραφή: Απλή εφαρμογή για ρίξιμο δύο ζαριών με το πάτημα ενός κουμπιού.

Έννοιες: Γεγονότα (events), τυχαίοι (random) αριθμοί.

Επέκταση: Εναλλακτικός τρόπος ρίψης ζαριών με ταρακούνημα, αξιοποιώντας το επιταχυνσιόμετρο (accelerometer) της συσκευής.

2^η Δραστηριότητα: Σκύλος - Φύλακας (v.1)

Περιγραφή: ο σκύλος της κεντρικής οθόνης της εφαρμογής ενοχλείται (αλλάζει όψη και γαβγίζει) όταν κάποιος αγγίζει την οθόνη της συσκευής, ενώ επιπλέον τρέχει προς το σημείο που αγγίζουμε την οθόνη.

Έννοιες: αλλαγή κατεύθυνσης και ταχύτητας, γεγονότα επαφής με οθόνη (touchdown και touchup), κίνηση αντικειμένου σε καμβά με βάση συντεταγμένες επαφής.

Επέκταση: Δόνηση συσκευής, όταν ο χρήστης αγγίζει το σκύλο.

Δραστηριότητες για το σπίτι:

- Κατηγοριοποίηση δοσμένων εντολών σε γεγονότα και ενέργειες.
- Κορώνα – γράμματα (v.1).
- Πιάνο.
- Κουμπί πανικού για αναπαραγωγή ήχου και αποστολή SMS σε προκαθορισμένο αριθμό, με κουμπί διακοπής.

Στην παρακάτω εικόνα φαίνονται ενδεικτικά στιγμιότυπα των οθονών των εφαρμογών.



Ζάρια (v.1)



Σκύλος - Φύλακας (v.1)



Κορώνα - Γράμματα (v.1)



Πιάνο



Κουμπί Πανικού

Εικόνα Α.1: Ενδεικτικά στιγμιότυπα εφαρμογών από τις δραστηριότητες του 1^{ου} μαθήματος.

A.2 Μάθημα 2

Στόχοι για τους μαθητές:

- Να αναφέρουν τι είναι και γιατί χρειάζεται μια μεταβλητή.
- Να μπορούν να δημιουργήσουν τη δική τους μεταβλητή και να της δίνουν αρχική τιμή.
- Να αλλάζουν την τιμή των μεταβλητών μέσα στον κώδικα της εφαρμογής.
- Να χρησιμοποιούν μεταβλητές για να υπολογίσουν το πλήθος κάποιων συμβάντων (μετρητές).
- Να χρησιμοποιούν τις λογικές τιμές True/False (Αληθής/Ψευδής).
- Να είναι σε θέση να συντάσσουν τις κατάλληλες συνθήκες ελέγχου.
- Να ελέγχουν την τιμή λογικών προτάσεων για να παίρνουν αποφάσεις με τις εντολές απλής ή σύνθετης επιλογής.
- Να χρησιμοποιούν το αντικείμενο Clock (ρολόι) για την παραγωγή animation - κινούμενης εικόνας ή/και για την εκτέλεση ενεργειών ανά τακτά χρονικά διαστήματα.

1^η Δραστηριότητα: Ζάρια (v.2)

Περιγραφή: Επέκταση της προηγούμενης εφαρμογής με εμφάνιση μηνύματος επιτυχίας σε περίπτωση εξάρων και προσθήκη σκορ.

Έννοιες: Αποθήκευση τυχαίων αριθμών σε μεταβλητές, Σύνταξη σύνθετης λογικής πρότασης, εντολή απλής επιλογής If, μεταβλητή - μετρητής (σκορ).

Επεκτάσεις: Διόρθωση bug με επαναφορά κενής ετικέτας σε κάθε ρίψη, μετατροπή ελέγχου για εξάρες σε διπλές, προσθήκη δυνατότητας για μηδενισμό σκορ (ενημέρωση μεταβλητής και αντίστοιχης ετικέτας).

2^η Δραστηριότητα: Σκύλος - Φύλακας (v.2)

Περιγραφή: Επέκταση της προηγούμενης έκδοσης με ανίχνευση του σηκώματος της συσκευής και αντίδραση με γάβγισμα και αλλαγή εικόνας σκύλου.

Έννοιες: Δημιουργία εντολής ελέγχου με βάση την τιμή επιτάχυνσης που ανιχνεύει ο αισθητήρας, αξιοποίηση ρολογιού/χρονομέτρου (clock) με ενεργοποίηση και απενεργοποίησή του για επαναφορά σκύλου σε ήρεμη κατάσταση μετά από ένα χρονικό διάστημα ακινησίας.

3^η Δραστηριότητα: Κορώνα - Γράμματα (v.2)

Περιγραφή: Επέκταση της πρώτης έκδοσης με επιλογή κορώνας ή γραμμάτων και ρίξιμο νομίσματος. Εμφάνιση μηνύματος «Κέρδισες» ή «Έχασες» και αναπαραγωγή αντίστοιχου ήχου.

Έννοιες: Εντολή ελέγχου If ... then ... else ..., animation με τη βοήθεια ρολογιών και μεταβλητής λογικού τύπου.

Δραστηριότητες για το σπίτι

- Απλές ασκήσεις αλλαγών τιμών σε μεταβλητές.
- Επέκταση ζαριών που μετράει παιχνίδια και εμφανίζει ποσοστό διπλών.
- Εφαρμογή χρονομέτρου αντίστροφης μέτρησης, όπου ο χρήστης ορίζει αριθμό δευτερολέπτων (με έλεγχο εισαγωγής τιμής) και πατώντας κάποιο κουμπί η μέτρηση ξεκινάει, δείχνοντας έναν έναν τους αριθμούς στην οθόνη. Όταν τα δευτερόλεπτα μηδενιστούν αναπαράγεται κάποιος ήχος.
- Εφαρμογή μετατροπής ενός μεγέθους σε άλλο (π.χ. θερμοκρασιών ή ταχύτητας).

Στην παρακάτω εικόνα φαίνονται ενδεικτικά στιγμιότυπα των οθονών των εφαρμογών.



Ζάρια (v.2)



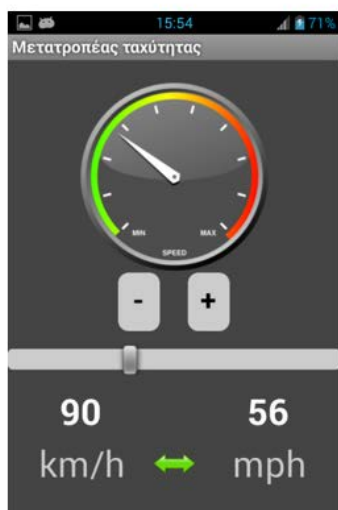
Κορώνα - Γράμματα (v.2)



Αντίστροφη μέτρηση



Μετατροπέας θερμοκρασίας



Μετατροπέας ταχύτητας

Εικόνα Α.2: Ενδεικτικά στιγμιότυπα εφαρμογών από τις δραστηριότητες του 2^{ου} μαθήματος.

A.3 Μάθημα 3

Στόχοι για τους μαθητές:

- Να αναλύουν ένα πρόβλημα σε απλούστερα τμήματα και να επιλύουν το κάθε τμήμα με ένα ξεχωριστό υποπρόγραμμα.
- Να δημιουργούν υποπρογράμματα (με ή χωρίς παραμέτρους) και να τα καλούν, όπου/όπως χρειάζεται.
- Να σχεδιάζουν στην οθόνη με τη βοήθεια του καμβά.
- Να χρησιμοποιούν τον αισθητήρα προσανατολισμού της συσκευής (orientation sensor) για κίνηση αντικειμένων εντός του γραφικού περιβάλλοντος.

1^η Δραστηριότητα: Γκαλερί φωτογραφιών

Περιγραφή: Παρουσίαση φωτογραφιών, με μουσική στο υπόβαθρο, με τρεις τρόπους (κουμπιά, σύρσιμο δακτύλου στην οθόνη, αυτόματα με χρόνο) και ανακύκλωση (από την τελευταία στην πρώτη και αντίστροφα).

Έννοιες: Μεταβλητή δείκτης για την τρέχουσα εικόνα, διαδικασία μετάβασης στην επόμενη ή προηγούμενη εικόνα (αρχικά χωρίς και έπειτα με παράμετρο), Εντολή επιλογής if για ανακύκλωση, Εντολή if else if για έλεγχο συρσίματος δακτύλου στην οθόνη και κλήση της κατάλληλης διαδικασίας με βάση τη φορά του.

Επέκταση: Επανεκκίνηση του χρονομέτρου στο σύρσιμο ή στο πάτημα των κουμπιών.

2^η Δραστηριότητα: Αριθμομηχανή

Περιγραφή: Εφαρμογή αριθμομηχανής για τις τέσσερις βασικές αριθμητικές πράξεις.

Έννοιες: Διαδικασίες με παραμέτρους, διάβασμα και αποθήκευση αριθμών σε μεταβλητές, εντολή επιλογής if ... else if ... (Αν ... Αλλιώς Αν ...) για υλοποίηση επιλεγμένης πράξης.

3^η Δραστηριότητα: Ζωγραφική με τα δάκτυλα

Περιγραφή: Εφαρμογή ζωγραφικής με τα δάκτυλα, η οποία επιτρέπει να σχεδιάζουμε σε κενό καμβά ή πάνω σε φωτογραφίες που τραβάμε με την κάμερα της συσκευής, αλλά και να αποθηκεύσουμε τις δημιουργίες μας.

Επεκτάσεις: Διαδικασία με παράμετρο την επιλογή χρώματος και ένδειξη τρέχοντος χρώματος.

4^η Δραστηριότητα: Πυξίδα

Περιγραφή: Εφαρμογή προσανατολισμού με ένδειξη των σημείων του ορίζοντα και της τρέχουσας κατεύθυνσης της συσκευής.

Έννοιες: Αξιοποίηση του κατάλληλου αισθητήρα (orientation sensor) για ανίχνευση του βορρά, με αλλαγή κατεύθυνσης εικόνας πυξίδας πάνω σε καμβά.

Επέκταση: Ηχητική/οπτική ειδοποίηση όταν κοιτάμε το βορρά.

5^η Δραστηριότητα: Παιχνίδι «Πιάσε τη σημαία»

Περιγραφή: Ο παίκτης χειρίζεται μια μπάλα γέροντας τη συσκευή και «κυνηγεί» τη σημαία-στόχο, ενώ προσπαθεί να αποφύγει τις τρύπες της πίστας. Η σημαία ανά διαστήματα μετακινείται σε τυχαίες θέσεις.

Έννοιες: Αξιοποίηση του Orientation Sensor ως χειριστήριο, σύγκρουση αντικειμένων, εντολή επιλογής (if ... else if ...), ρολόι για μετακίνηση στόχου σε τυχαία θέση στον καμβά.

Επεκτάσεις: Μετακίνηση σημαίας σε τυχαία θέση όταν την «πιάνει» ο παίκτης, δημιουργία σκορ, τυχαία μετακίνηση των τρυπών ανά διαστήματα, ζωές παίκτη, τέλος παιχνιδιού (game over) και δυνατότητα έναρξης νέου παιχνιδιού.

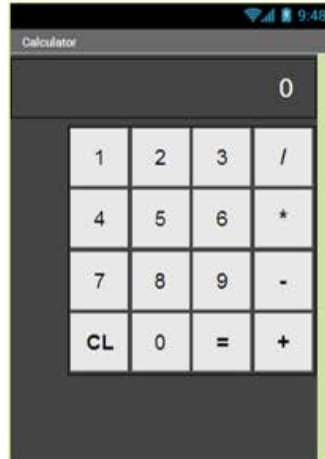
Δραστηριότητες για το σπίτι

- Παιχνίδι MoleMash: εμφανίζονται αντικείμενα σε τυχαίες θέσεις και ο παίκτης πρέπει να προλάβει να τα χτυπήσει πριν αυτά εξαφανιστούν.
- Παιχνίδι Pong: μια μπάλα κινείται συνεχώς και αναπηδά στα τοιχώματα. Ο παίκτης χειρίζεται μια σανίδα, την οποία μετακινεί αριστερά δεξιά με το δάκτυλο. Ο παίκτης μετακινώντας τη σανίδα προσπαθεί να χτυπάει την μπάλα και να μην της επιτρέπει να αγγίξει την κάτω πλευρά της οθόνης (στην περίπτωση αυτή το παιχνίδι τερματίζει και ο παίκτης χάνει).
- Παιχνίδι ζαριών με σκορ, όπου ο παίκτης έχει το δικαίωμα να ρίξει τα ζάρια μέχρι π.χ. 4 φορές και ανάλογα με τη ζαριά του, κερδίζει κάποιους πόντους και συνεχίζει ή χάνει και σταματάει.
- Κουίζ προσθέσεων μεταξύ τυχαίων αριθμών.

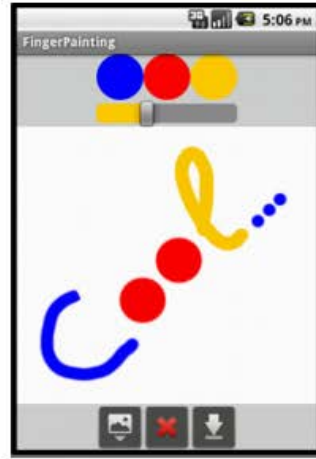
Στην παρακάτω εικόνα φαίνονται ενδεικτικά στιγμιότυπα των οθονών των εφαρμογών.



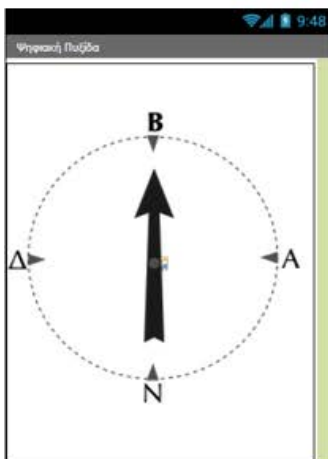
Γκαλερί φωτογραφιών



Αριθμομηχανή



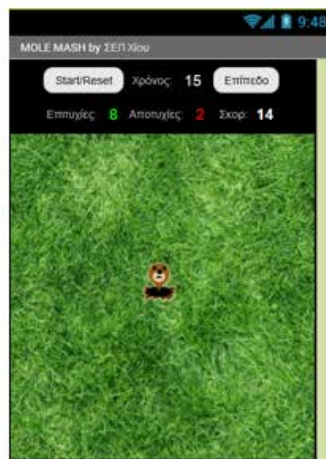
Ζωγραφική (v.1)



Πυξίδα



Πιάσε τη σημαία



Mole Mash



Pong



Παιχνίδι με ζάρια



Κουίζ μαθηματικών

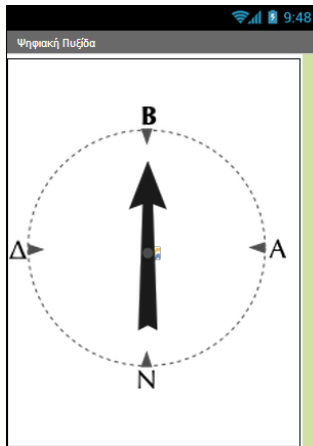
Εικόνα Α.3: Ενδεικτικά στιγμιότυπα εφαρμογών από τις δραστηριότητες του 3ου μαθήματος.

A.4 Ενδεικτικό παράδειγμα δραστηριότητας

Πρόκειται για την 4^η δραστηριότητα του 3^{ου} μαθήματος.

Δραστηριότητα: Πυξίδα

Θα κατασκευάσουμε μια εφαρμογή - πυξίδα που θα μας δείχνει προς τα πού είναι ο Βορράς.



Εικόνα 4 - Το περιβάλλον της εφαρμογής Πυξίδα

Βασικές έννοιες:

- Αισθητήρας ανίχνευσης προσανατολισμού της συσκευής (Orientation Sensor).
- Εντολή επιλογής if (Av).

Βήμα 1 Δημιουργία νέου project και ρυθμίσεις οθόνης

Ξεκινώντας, δημιουργούμε ένα καινούργιο project με όνομα Compass.

Designer Γραφικό Περιβάλλον Εφαρμογής

Αρχικά, το μοναδικό διαθέσιμο συστατικό της εφαρμογής είναι η οθόνη (Screen) και θα χρειαστεί να μεταβάλλουμε κάποιες από τις ιδιότητές της.



επιλέγουμε το
αντικείμενο

Screen1

μεταβάλλουμε τις
ιδιότητες

AlignHorizontal: Center

Screen Orientation: Portrait

Scrollable: No

Title: Ψηφιακή Πυξίδα

Βήμα 2 Προσθήκη αρχείων πολυμέσων

Ας προσθέσουμε στο project μας το αρχείο εικόνας που θα χρησιμοποιεί η εφαρμογή μας, κατεβάζοντάς το από τη διεύθυνση <http://tiny.cc/aifiles> και το φάκελο με όνομα Compass.



ανεβάζουμε το αρχείο με όνομα	σύντομη περιγραφή
compass-arrow.png	Εικόνα πυξίδας με βέλος

Βήμα 3

Προσθήκη αντικειμένων στο γραφικό περιβάλλον

Θα χρειαστούμε το αντικείμενο του καμβά, πάνω στο οποίο θα τοποθετηθεί το γραφικό της πυξίδας (ImageSprite), καθώς και τον αισθητήρα προσανατολισμού της συσκευής.



από την ομάδα	μεταφέρουμε το αντικείμενο	του δίνουμε το όνομα	μεταβάλλουμε τις ιδιότητες
Drawing and Animation	Canvas	CompassCanvas	Width: Fill Parent Height: Fill Parent
Drawing and Animation	ImageSprite	CompassSprite	X: 0 Y: 50 Width: 300 pixels Height: 300 pixels
Sensors	OrientationSensor	CompassSensor	

Βήμα 4

Ανιχνεύοντας την αλλαγή προσανατολισμού

Συγγραφή Προγράμματος

Blocks

Η εφαρμογή μας θα ανιχνεύει το γεγονός της αλλαγής του προσανατολισμού της κινητής συσκευής και θα προσαρμόζει κατάλληλα την εικόνα της πυξίδας.



Το αντικείμενο OrientationSensor μας δίνει τρεις σημαντικές πληροφορίες σχετικές με την κλίση και τον προσανατολισμό της κινητής μας συσκευής.

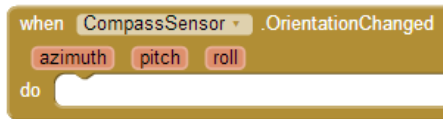
Roll: Έχει τιμή 0 όταν η συσκευή είναι επίπεδη, αυξάνεται στις 90° όταν η συσκευή είναι γυρισμένη στη δεξιά της πλευρά κ.ο.κ.

Pitch: Έχει τιμή 0 όταν η συσκευή είναι επίπεδη, αυξάνεται στις 90° όταν η πάνω πλευρά της συσκευής κοιτάει προς τα κάτω κ.ο.κ.

Azimuth: Έχει τιμή 0 όταν η συσκευή κοιτάζει στο Βορρά, 90° όταν κοιτάζει προς την Ανατολή, 180° στο Νότο και 270° στη Δύση.

Στην προκειμένη περίπτωση, θα χρειαστούμε την τιμή της ιδιότητας Azimuth (Αζιμούθιο) μιας και αυτή μας δείχνει τον προσανατολισμό της συσκευής στον άξονα Βορρά - Νότου.

Από το αντικείμενο CompassSensor χρησιμοποιούμε την εντολή:



Εικόνα 5 - Το γεγονός αλλαγής προσανατολισμού

Στη συνέχεια, θα πρέπει, κάθε φορά που αλλάζει ο προσανατολισμός της συσκευής, να προσαρμόζουμε την κατεύθυνση (heading) της εικόνας της πυξίδας.

Από το αντικείμενο CompassSprite επιλέγουμε την εντολή ...



Εικόνα 6 - Εντολή αλλαγής κατεύθυνσης μιας εικόνας

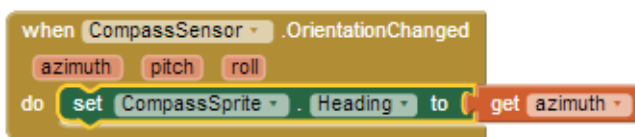
... και την τοποθετούμε μέσα στο γεγονός της αλλαγής προσανατολισμού.

Για να πάρουμε την τιμή του αζιμούθιου της συσκευής τοποθετούμε το δείκτη του ποντικιού πάνω στο πεδίο **azimuth** και τραβάμε την αντίστοιχη εντολή, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 7 - Παίρνοντας την τιμή του αζιμούθιου

Τέλος, κουμπώνουμε την παραπάνω εντολή με το αζιμούθιο της συσκευής.



Εικόνα 8 - Θέτοντας τον προσανατολισμό της πυξίδας στο Βορρά



Η εφαρμογή μας είναι έτοιμη!
Ωρα να την ελέγξουμε με την κινητή μας συσκευή.

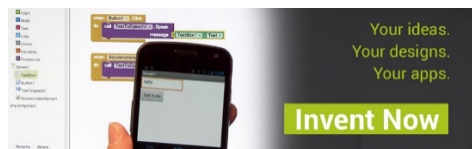
ΕΠΕΚΤΑΣΕΙΣ

1. Προσθέστε στην εφαρμογή μια ετικέτα κειμένου που να εμφανίζει την τιμή της ιδιότητας Azimuth και καταγράψτε τι τιμές εμφανίζονται όταν κοιτάτε προς το Βορρά.
2. Επεκτείνετε τη λειτουργία της εφαρμογής της πυξίδας, ώστε όταν η συσκευή δείχνει προς το Βορρά (με απόκλιση +/- 10 μοίρες) να αναπαράγει ένα ηχητικό σήμα της επιλογής σας ή/και να εμφανίζει κάποια ένδειξη στην οθόνη.

A.5 Οδηγίες προετοιμασίας πριν το 1^ο μάθημα

Το ακόλουθο φύλο στάλθηκε στους μαθητές που συμμετείχαν στα εργαστήρια για την προετοιμασία των συσκευών τους.

Οδηγίες προετοιμασίας



Η επίσημη ιστοσελίδα του App Inventor είναι η <http://appinventor.mit.edu> στην οποία μπορείτε να βρείτε οδηγούς χρήσης και παραδείγματα, αλλά και να δημιουργήσετε άμεσα εφαρμογές, χωρίς να εγκαταστήσετε τίποτα στον Η/Υ σας.

Βήμα 1: Εγκατάσταση εφαρμογής QR code scanner στη συσκευή

Σίγουρα θα έχετε ξαναδεί αυτά τα τετράγωνα barcodes. Με τη βοήθεια της φορητής συσκευής και της κάμεράς της, οποιοσδήποτε μπορεί να σαρώσει έναν τέτοιο κωδικό και να τον αποκωδικοποιήσει, ώστε πολύ εύκολα και γρήγορα, να μεταβεί στην ιστοσελίδα (URL) που αντιστοιχεί στο συγκεκριμένο QR code.



Προκειμένου να μπορείτε να σαρώνετε κι εσείς τέτοιους κωδικούς και να διευκολυνθείτε στη μετάβασή σας σε διάφορες χρήσιμες διευθύνσεις, συνιστάται η εγκατάσταση στη συσκευή σας κάποιας εφαρμογής σάρωσης QR κωδικών (QR Code Scanner), όπως π.χ. **οποιασδήποτε** από τις δύο παρακάτω δωρεάν εφαρμογές στο Play Store:

[BarCode Scanner](#)

από την ZXing Team



[QR Code Reader](#)

από την Scan, Inc.



Βήμα 2: Εγκατάσταση της εφαρμογής MIT AI2 companion στη συσκευή

Ο ευκολότερος τρόπος για επιτόπου έλεγχο των εφαρμογών που θα φτιάχνετε, απαιτεί σύνδεση της Android συσκευής σας με τον Η/Υ, μέσω ασύρματου δικτύου (Wi-Fi).



Απαραίτητη προϋπόθεση για τη σύνδεση μέσω Wi-Fi είναι η εγκατάσταση στην Android συσκευή σας της εφαρμογής **MIT AI2 Companion**:

είτε από το [Play Store](#)

QR code:



είτε από τη σελίδα του AI ([.apk](#))

QR code:



Προτείνεται η εγκατάσταση από το Play Store λόγω δυνατότητας αυτόματων ενημερώσεων.



Αυτά προς το παρόν... Είστε έτοιμοι να δημιουργήσετε την πρώτη σας εφαρμογή!
Τα υπόλοιπα από κοντά...

Παράρτημα Β

Μέσα Συλλογής Δεδομένων

Στο Παράρτημα Β παρατίθενται η δήλωση συγκατάθεσης για συμμετοχή των μαθητών στην έρευνα, ενδεικτικά φύλλα παρατήρησης που χρησιμοποιήθηκαν από τους εκπαιδευτικούς, καθώς και τα δύο online ερωτηματολόγια που συμπλήρωσαν οι συμμετέχοντες μαθητές κατά την πρώτη και την τελευταία συνάντηση.

B.1 Δήλωση συγκατάθεσης συμμετοχής στην έρευνα

Η παρακάτω δήλωση συμπληρώθηκε και υπογράφηκε από τους κηδεμόνες των συμμετεχόντων μαθητών, προς ενημέρωσή τους και έγκριση της συμμετοχής των παιδιών στην έρευνα.

Σημειώνεται ότι οι δηλώσεις για τους μαθητές της Μαθητικής Στέγης των Καρδαμύλων Χίου υπογράφηκαν από τον νόμιμο εκπρόσωπο των παιδιών και Διευθυντή της, κύριο Εμμανουήλ Κουκούλη.

Δήλωση συγκατάθεσης

Στο πλαίσιο μεταπτυχιακής εργασίας, διερευνάται η καταλληλότητα και η επίδραση του εργαλείου App Inventor, αλλά και γενικότερα της δημιουργίας εφαρμογών για ψηφιακές φορητές συσκευές, στη μάθηση βασικών εννοιών προγραμματισμού από μαθητές δευτεροβάθμιας εκπαίδευσης.

Οι μαθητές που συμμετέχουν στο σχετικό εκπαιδευτικό πρόγραμμα, που διοργανώνεται από το Σύλλογο Εκπαιδευτικών Πληροφορικής Χίου, θα κληθούν να συμπληρώσουν δυο σύντομα ερωτηματολόγια, το ένα κατά την πρώτη συνάντηση και το άλλο κατά το τέλος του προγράμματος.

Παράλληλα, οι εκπαιδευτές του προγράμματος, θα παρατηρούν την πορεία των μαθητών, τις προτιμήσεις τους, τις δυσκολίες που ενδεχομένως να αντιμετωπίσουν, καθώς και το ποσοστό επίτευξης των μαθησιακών στόχων και θα καταγράφουν μέρος των παραπάνω σε συγκεκριμένα φύλλα παρατήρησης.

Προαιρετικά, εφόσον κάποιος μαθητής το επιθυμεί, μπορεί να δώσει συνέντευξη στον ερευνητή για τα σημεία του προγράμματος που του άρεσαν περισσότερο ή λιγότερο και πιθανές προτάσεις για βελτίωση του μαθησιακού υλικού.

Όλα τα στοιχεία που θα συλλεχθούν θα είναι **ανώνυμα** και δεν θα χρησιμοποιηθούν για άλλους σκοπούς πέραν της συγκεκριμένης έρευνας.

**Διάβασα και κατανόησα το παρόν κείμενο και εγκρίνω τη συμμετοχή του/της
..... στο εκπαιδευτικό πρόγραμμα και
στην έρευνα αυτή με τους όρους που προαναφέρονται.**

....., / / 2014

Ερευνητής

Γονέας ή κηδεμόνας ή άλλος
νόμιμος εκπρόσωπος

Χατζηνικολάκης Γεώργιος
Εκπαιδευτικός Πληροφορικής

.....
.....

B.2 1^ο ερωτηματολόγιο

Το παρακάτω ερωτηματολόγιο συμπληρώθηκε από τους συμμετέχοντες μαθητές με την έναρξη των εκπαιδευτικών προγραμμάτων. Υλοποιήθηκε με τη χρήση της υπηρεσίας Google Forms και διατέθηκε online στη διεύθυνση: <http://tiny.cc/aiform1>

Δημιουργία εφαρμογών με το App Inventor

Στο πλαίσιο μεταπτυχιακής εργασίας για τη διδασκαλία και μάθηση βασικών εννοιών προγραμματισμού σε μαθητές δευτεροβάθμιας εκπαίδευσης, μέσω της δημιουργίας εφαρμογών για ψηφιακές φορητές συσκευές, συγκεντρώνουμε ορισμένα στατιστικά στοιχεία από τους συμμετέχοντες μαθητές. Το παρόν ερωτηματολόγιο είναι ανώνυμο και απαιτούνται περίπου 5' για τη συμπλήρωσή του.

Ευχαριστούμε προκαταβολικά για την συμμετοχή σας!

Σπουδαστής: Γεώργιος Χατζηνικολάκης

Ίδρυμα: Ανοικτό Πανεπιστήμιο Κύπρου

Μεταπτυχιακό Πρόγραμμα: Πληροφοριακά και Επικοινωνιακά Συστήματα

Επιβλέπων καθηγητής: Δρ. Σπυρίδων Παπαδάκης

** Απαιτείται*

1. Συμπλήρωσε τον κωδικό σου * _____

Κάθε μαθητής θα έχει τον δικό του ίδιο κωδικό κατά τη συμπλήρωση των 2 ερωτηματολογίων, ώστε να μπορεί να γίνει σύνδεσή του ενός με του άλλου.

2. Τάξη *

- Γ' Γυμνασίου
- Α' Γενικού Λυκείου
- Β' Γενικού Λυκείου
- Α' ΕΠΑΛ
- Β' ΕΠΑΛ

3. Φύλο *

- Αγόρι
- Κορίτσι

4. Διαθέτεις Η/Υ στο σπίτι σου; *

- Ναι
- Όχι

5. Εάν απάντησες Ναι στην προηγούμενη ερώτηση, πόσες ώρες την εβδομάδα (κατά μέσο όρο) χρησιμοποιείς τον Η/Υ σου;

- Λιγότερες από 2 ώρες
- 2 με 5 ώρες
- 6 - 10 ώρες
- Περισσότερες από 10 ώρες

6. Διαθέτεις συσκευή με λειτουργικό σύστημα Android; *

- Κινητό τηλέφωνο
- Tablet
- Και τα δυο
- Τίποτα από τα παραπάνω

7. Εάν απάντησες θετικά στην προηγούμενη ερώτηση, έχεις κατεβάσει ποτέ κάποια εφαρμογή ή παιχνίδι από το Google Play στη συσκευή σου;

- Ναι
- Όχι

8. Πόσο καλός/καλή (με βάση την επίδοσή σου) είσαι στα μαθήματα της πληροφορικής; *

- Καθόλου
- Λίγο
- Μέτρια
- Αρκετά
- Πολύ

9. Ποια/ποιες γλώσσες προγραμματισμού από τις παρακάτω έχεις διδαχθεί στο παρελθόν; *
Επιλέξτε όλα όσα ισχύουν.

- App Inventor
- EasyLogo
- K-Turtle
- MicroWorlds Pro
- Pascal
- Scratch
- Χελωνόκοσμος
- Διερμηνευτής της ΓΛΩΣΣΑΣ - Ψευδογλώσσα
- Καμία
- Άλλο:

10. Εάν έχεις χρησιμοποιήσει ξανά κάποια γλώσσα προγραμματισμού, ποια/ποιες από τις παρακάτω έννοιες έχεις αξιοποιήσει στα προγράμματα που έχεις κατασκευάσει;
Επιλέξτε όλα όσα ισχύουν.

- Μεταβλητή
- Δομή επιλογής
- Δομή επανάληψης
- Διαδικασία (υποπρόγραμμα)
- Δεν είμαι απόλυτα σίγουρος/η
- Καμία

11. Σε τι βαθμό θεωρείς ότι οι παραπάνω προγραμματιστικές έννοιες σχετίζονται με την καθημερινότητά σου; *

- Καθόλου
- Λίγο
- Μέτρια
- Αρκετά
- Πολύ

12. Το αντικείμενο του προγραμματισμού υπολογιστών είναι... *

	Πολύ	Αρκετά	Λίγο	Καθόλου
Ενδιαφέρον				

	Πολύ	Αρκετά	Λίγο	Καθόλου
Εύκολο				
Χρήσιμο				

13. Έχεις επιχειρήσει να δημιουργήσεις κάποιο πρόγραμμα (εφαρμογή, παιχνίδι), από προσωπικό σου ενδιαφέρον, εκτός σχολικού ωραρίου; *

- Ναι
- Όχι

14. Για ποιες συσκευές βρίσκεις πιο ενδιαφέρουσα τη δημιουργία εφαρμογών; *

- για Η/Υ
- για κινητά τηλέφωνα και tablets
- Και τα δυο παραπάνω είναι εξίσου ενδιαφέροντα/αδιάφορα
- Άλλο:

15. Ενδιαφέρεσαι να ασχοληθείς με το αντικείμενο του προγραμματισμού υπολογιστών στο μέλλον; *

- Αποκλείεται
- Ίσως, μόνο για χόμπι - προσωπική ευχαρίστηση
- Πολύ πιθανό
- Σίγουρα, σκέφτομαι να ασχοληθώ και επαγγελματικά

16. Το App Inventor είναι ένα δωρεάν εργαλείο που θα σε βοηθήσει να δημιουργείς εφαρμογές για Android συσκευές. Γνωρίζεις από πριν για την ύπαρξή του; *

- Όχι, δεν το έχω ξανακούσει
- Το έχω ακούσει/δει, αλλά δεν γνωρίζω πολλά γι' αυτό
- Το γνωρίζω, το βρίσκω ενδιαφέρον, αλλά δεν το έχω χρησιμοποιήσει
- Ναι, το έχω χρησιμοποιήσει κιόλας

17. Πιστεύεις ότι θα χρησιμοποιήσεις το App Inventor, για να πειραματιστείς ή να δημιουργήσεις κάποια εφαρμογή, πέρα από τις ώρες που γίνεται το σεμινάριο; *

- Όχι
- Μάλλον όχι
- Ίσως
- Ναι

18. Παρατηρήσεις / σχόλια

Μπορείς (προαιρετικά) να αναφέρεις οτιδήποτε σε προβληματίζει ή θες να σχολιάσεις για το εκπαιδευτικό πρόγραμμα και τη συμμετοχή σου σε αυτό.

B.3 2^ο ερωτηματολόγιο

Το παρακάτω ερωτηματολόγιο συμπληρώθηκε από τους συμμετέχοντες μαθητές με τη λήξη των εκπαιδευτικών προγραμμάτων. Υλοποιήθηκε με τη χρήση της υπηρεσίας Google Forms και διατέθηκε online στη διεύθυνση: <http://tiny.cc/aiform2>

Δημιουργία εφαρμογών με το App Inventor

Στο πλαίσιο μεταπτυχιακής εργασίας για τη διδασκαλία και μάθηση βασικών εννοιών προγραμματισμού σε μαθητές δευτεροβάθμιας εκπαίδευσης, μέσω της δημιουργίας εφαρμογών για ψηφιακές φορητές συσκευές, συγκεντρώνουμε ορισμένα στατιστικά στοιχεία από τους συμμετέχοντες μαθητές. Το παρόν ερωτηματολόγιο είναι ανώνυμο και απαιτούνται περίπου 10' για τη συμπλήρωσή του.

Ευχαριστούμε προκαταβολικά για την συμμετοχή σας!

Σπουδαστής: Γεώργιος Χατζηνικολάκης
Ίδρυμα: Ανοικτό Πανεπιστήμιο Κύπρου
Μεταπτυχιακό Πρόγραμμα: Πληροφοριακά και Επικοινωνιακά Συστήματα
Επιβλέπων καθηγητής: Δρ. Σπυρίδων Παπαδάκης

* Απαιτείται

1. Συμπλήρωσε τον κωδικό σου * _____

Το παρόν ερωτηματολόγιο αποτελεί τη συνέχεια του προηγούμενου ερωτηματολογίου. Συμπλήρωσε λοιπόν τον κωδικό που είχες δηλώσει κατά τη συμπλήρωση του 1ου ερωτηματολογίου.

2. Τάξη *

- Γ' Γυμνασίου
- Α' Γενικού Λυκείου
- Β' Γενικού Λυκείου
- Α' ΕΠΑΛ
- Β' ΕΠΑΛ

3. Φύλο *

- Αγόρι
- Κορίτσι

4. Ποια/ποιες από τις παρακάτω έννοιες αξιοποίησες στα προγράμματα που ανέπτυξες με το App Inventor;

Επίλεξε όλα όσα ισχύουν.

- Μεταβλητή
- Δομή επιλογής
- Δομή επανάληψης
- Διαδικασία (υποπρόγραμμα)
- Δεν είμαι απόλυτα σίγουρος/η
- Καμία

5. Σε τι βαθμό θεωρείς ότι οι προγραμματιστικές έννοιες (μεταβλητές, δομή επιλογή, δομή επανάληψης, διαδικασίες)) σχετίζονται με την καθημερινότητά σου; *

- Καθόλου
- Λίγο

- Μέτρια
- Αρκετά
- Πολύ

6. Για ποιες συσκευές βρίσκεις πιο ενδιαφέρουσα τη δημιουργία εφαρμογών; *

- για Η/Υ
- για κινητά τηλέφωνα και tablets
- Και τα δυο παραπάνω είναι εξίσου ενδιαφέροντα/αδιάφορα
- Άλλο:

7. Ο προγραμματισμός με το App Inventor είναι... *

	Πολύ	Αρκετά	Λίγο	Καθόλου
Ενδιαφέρον				
Εύκολος				
Χρήσιμος				

8. Σκοπεύεις να συνεχίσεις να χρησιμοποιείς το App Inventor; *

- Όχι
- Ναι

9. Ενδιαφέρεσαι να ασχοληθείς με το αντικείμενο του προγραμματισμού υπολογιστών στο μέλλον; *

- Αποκλείεται
- Ίσως, μόνο για χόμπι - προσωπική ευχαρίστηση
- Πολύ πιθανό
- Σίγουρα, σκέφτομαι να ασχοληθώ και επαγγελματικά

10. Το μάθημα της Πληροφορικής στο σχολείο θα γινόταν πιο ενδιαφέρον με αξιοποίηση του App Inventor; *

- Οπωσδήποτε
- Αρκετά
- Μέτρια
- Λίγο
- Καθόλου

Σχετικά με τον προγραμματισμό στο App Inventor

11. Σε τι βαθμό θεωρείς ότι μπορείς να φτιάξεις τη δική σου εφαρμογή με τη βοήθεια του App Inventor;

- Καθόλου
- Μικρό
- Μέτριο
- Μεγάλο

12. Πιστεύεις ότι γνωρίζεις ... *

(μετά τη συμμετοχή σου στο εκπαιδευτικό πρόγραμμα)

	Πολύ καλά	Αρκετά καλά	Λίγο	Σχεδόν Καθόλου
Να σχεδιάζεις το περιβάλλον διεπαφής (οθόνη) της εφαρμογής				
Να ορίζεις ιδιότητες στα αντικείμενα της οθόνης				
Να προσθέτεις εικόνες και ήχους				
Να εκτελείς ενέργειες με το πάτημα κουμπιών				
Να εκτελείς ενέργειες χρησιμοποιώντας τους αισθητήρες της συσκευής				
Να δημιουργείς τις δικές σου μεταβλητές				
Να εκτελείς μαθηματικές πράξεις				
Να συντάσσεις λογικές εκφράσεις				
Να χρησιμοποιείς την εντολή επιλογής IF				
Να επεξεργάζεσαι τις τιμές των μεταβλητών κατά την εκτέλεση της εφαρμογής				

13. Αισθάνεσαι ότι είσαι σε θέση ... *

(μετά τη συμμετοχή σου στο εκπαιδευτικό πρόγραμμα)

	Πολύ	Αρκετά	Λίγο	Σχεδόν Καθόλου
--	------	--------	------	----------------

	Πολύ	Αρκετά	Λίγο	Σχεδόν Καθόλου
Να καταλαβαίνεις τι κάνουν οι εντολές που βλέπεις σε ένα έτοιμο έργο				
Να τροποποιείς έτοιμα κομμάτια εντολών και να τα προσαρμόζεις στις ανάγκες σου				
Να εντοπίζεις και να διορθώνεις λάθη σε υπάρχον κώδικα				

14. Το App Inventor σε βοήθησε να ... *

	Συμφωνώ απόλυτα	Μάλλον συμφωνώ	Ούτε συμφωνώ, ούτε διαφωνώ	Μάλλον διαφωνώ	Διαφωνώ
Να πειραματιστείς					
Να κατανοήσεις καλύτερα τι είναι ο προγραμματισμός					
Να φτιάξεις καλαίσθητες εφαρμογές					
Να αποφεύγεις λάθη κατά τη δημιουργία εφαρμογών					
Να διασκεδάσεις					

15. Τι άλλο σε βοήθησε (και σου άρεσε) στο App Inventor;

[Προαιρετικό] Εκφράσου ελεύθερα και ειλικρινά. Η γνώμη σου είναι σημαντική!

16. Το App Inventor σε δυσκόλεψε (και δεν σου άρεσε), επειδή ... *

	Συμφωνώ απόλυτα	Μάλλον συμφωνώ	Ούτε συμφωνώ, ούτε διαφωνώ	Μάλλον διαφωνώ	Διαφωνώ
δεν είναι στην ελληνική γλώσσα					
δεν υπάρχει εντολή					

	Συμφωνώ απόλυτα	Μάλλον συμφωνώ	Ούτε συμφωνώ, ούτε διαφωνώ	Μάλλον διαφωνώ	Διαφωνώ
ανάιρεσης (undo)					
είναι κουραστικό να βρίσκεις τις εντολές που χρειάζεσαι					
δεν έχει αρκετές δυνατότητες					
δεν μπορείς να εντοπίσεις τυχόν λάθη στον κώδικα					

17. Τι άλλο σε δυσκόλεψε ή δεν σου άρεσε στο App Inventor;
[Προαιρετικό] Εκφράσου ελεύθερα και ειλικρινά. Η γνώμη σου είναι σημαντική!

Σχετικά με τις δραστηριότητες και τις εφαρμογές που δημιούργησες

18. Ποιες δραστηριότητες ολοκλήρωσες; *

(από το 1ο μάθημα)

	Πολύ εύκολα	Αρκετά εύκολα	Μέτρια	Δύσκολα	Καθόλου
τα ζάρια με ταρακούνημα της συσκευής					
την κορώνα- γράμματα					
το κουμπί πανικού					
το πιάνο					
το σκύλο φύλακα με δόνηση					

19. Ποιες δραστηριότητες ολοκλήρωσες; *

(από το 2ο μάθημα)

	Πολύ εύκολα	Αρκετά εύκολα	Μέτρια	Δύσκολα	Καθόλου
τα ζάρια με έλεγχο για εξάρεις και σκορ					
τα ζάρια με έλεγχο για διπλές και κουμπί μηδενισμού του σκορ					
το σκύλο φύλακα με έλεγχο σηκώματος της συσκευής και χρονόμετρο "ηρεμίας"					
την κορώνα-γράμματα με ποντάρισμα και animation της ρίψης					
τα ζάρια με εμφάνιση του ποσοστού διπλών					
το αντίστροφο χρονόμετρο με έλεγχο τιμών					
την εφαρμογή μετατροπής					

20. Ποιες δραστηριότητες ολοκλήρωσες; *

(από το 3ο μάθημα)

	Πολύ εύκολα	Αρκετά εύκολα	Μέτρια	Δύσκολα	Καθόλου
τη γκαλερί φωτογραφιών					
τη ζωγραφική					
την πυξίδα					
τη γκαλερί φωτογραφιών με διαδικασία					

	Πολύ εύκολα	Αρκετά εύκολα	Μέτρια	Δύσκολα	Καθόλου
τη ζωγραφική με διαδικασία για το χρώμα					
την πυξίδα με ηχητική ειδοποίηση για βορρά					

21. Παρατηρήσεις / σχόλια

Μπορείς (προαιρετικά) να αναφέρεις οτιδήποτε θες να σχολιάσεις για τους εκπαιδευτές σου, για το εκπαιδευτικό πρόγραμμα και τη συμμετοχή σου σε αυτό.

B.4 Φύλλα Παρατήρησης

Τα παρακάτω ενδεικτικά φύλλα παρατήρησης συμπληρώθηκαν από τους εκπαιδευτικούς κατά τη διεξαγωγή των συναντήσεων και κατόπιν της επιτόπιας παρακολούθησης και συζήτησης με τους μαθητές.

	ΜΑΘΗΜΑ 1 - ΒΑΘΜΟΣ ΚΑΤΑΝΟΗΣΗΣ ΚΑΙ ΕΠΙΤΕΥΞΗΣ	ΠΛΗΡΩΣ	ΜΕΤΡΙΑ	ΔΥΣΚΟΛΑ
1	Τοποθέτηση αντικειμένων (Components) στην περιοχή σχεδίασης (Designer).			
2	Τοποθέτηση εντολών για τα αντικείμενα στην περιοχή συγγραφής (Blocks).			
3	Μεταφόρτωση αρχείων πολυμέσων στο περιβάλλον του AI.			
4	Σύνδεση γεγονότων με συγκεκριμένες ενέργειες.			
5	Τοποθέτηση των ενεργειών με τη σωστή σειρά εκτέλεσης.			
6	Απόδοση τιμών στις ιδιότητες των αντικειμένων που χρησιμοποιούνται.			
7	Χρήση της εντολής παραγωγής τυχαίων αριθμών.			
8	Κίνηση (animation) με τη βοήθεια του καμβά.			
9	Εκτέλεση και δοκιμή της εφαρμογής στην φορητή συσκευή.			

	ΜΑΘΗΜΑ 2 - ΒΑΘΜΟΣ ΚΑΤΑΝΟΗΣΗΣ ΚΑΙ ΕΠΙΤΕΥΞΗΣ	ΠΛΗΡΩΣ	ΜΕΤΡΙΑ	ΔΥΣΚΟΛΑ
1	Δημιουργία μεταβλητής			
2	Αλλαγή της τιμής μιας μεταβλητής με τις κατάλληλες εντολές			
3	Διαφορά των εντολών SET και GET			
4	Υπολογισμός του πλήθους κάποιου γεγονότος (μεταβλητή ως μετρητής)			
5	Χρήση των λογικών τιμών TRUE/FALSE			
6	Λήψη απόφασης με την εντολή IF και τον έλεγχο κάποιας λογικής πρότασης			
7	Χρήση του ρολογιού (clock)			
8	Να αναγνωρίζετε πότε είναι απαραίτητη η χρήση μεταβλητής και πότε όχι			
9	Να δίνετε τις εντολές που χρειάζεται να εκτελεστούν, με τη σωστή σειρά			

	ΜΑΘΗΜΑ 3 - ΒΑΘΜΟΣ ΚΑΤΑΝΟΗΣΗΣ ΚΑΙ ΕΠΙΤΕΥΞΗΣ	ΠΛΗΡΩΣ	ΜΕΤΡΙΑ	ΔΥΣΚΟΛΑ
1	Ανάλυση και διάσπαση ενός προβλήματος σε απλούστερα υπομνήματα			
2	Δημιουργία απλής διαδικασίας και κλήση της			
3	Δημιουργία διαδικασίας με παράμετρο και κλήση της			
4	Σχεδίαση στην οθόνη (καμβά) με το δάκτυλο			
5	Χρήση του αισθητήρα προσανατολισμού (orientation sensor)			
6	Λήψη απόφασης με την εντολή IF και τον έλεγχο κάποιας λογικής πρότασης			
7	Σύνταξη εντολής για έλεγχο των τιμών που διαβάζουμε από το χρήστη			

Παράρτημα Γ

Πίνακες Στατιστικής Ανάλυσης

Οι πίνακες που παρήχθησαν από το SPSS για τη στατιστική ανάλυση των δεδομένων και αναφέρονται στο κεφάλαιο 8, παρατίθενται στη συνέχεια.

Γ.1 Στάσεις μαθητών ως προς τον προγραμματισμό πριν και μετά τη διεξαγωγή των εργαστηρίων

Τα αποτελέσματα των ελέγχων υποθέσεων για τα συσχετιζόμενα δείγματα (Paired Samples t-tests), από τις ερωτήσεις που επαναλήφθηκαν στο πρώτο και στο δεύτερο ερωτηματολόγιο, συνοψίζονται στον ακόλουθο πίνακα:

Paired Samples Test									
		Paired Differences					t	df	Sig. (2-tailed)
		Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference				
					Lower	Upper			
Pair 1	Σε τι βαθμό θεωρείς ότι οι προγραμματιστικές έννοιες σχετίζονται με την καθημερινότητά σου;	-,429	1,008	,170	-,775	-,082	-2,514	34	,017
Pair 2	Το αντικείμενο του προγραμματισμού υπολογιστών είναι... [Ενδιαφέρον] - Ο προγραμματισμός με το AI είναι... [Ενδιαφέρον]	,057	,684	,116	-,178	,292	,495	34	,624
Pair 3	Το αντικείμενο του προγραμματισμού υπολογιστών είναι... [Εύκολο] - Ο προγραμματισμός με το AI είναι... [Εύκολος]	,514	,658	,111	,288	,740	4,620	34	,000
Pair 4	Το αντικείμενο του προγραμματισμού υπολογιστών είναι... [Χρήσιμο] - Ο προγραμματισμός με το AI είναι... [Χρήσιμος]	-,343	,684	,116	-,578	-,108	-2,967	34	,005
Pair 5	Ενδιαφέρεσαι να ασχοληθείς με το αντικείμενο του προγραμματισμού υπολογιστών στο μέλλον; - Ενδιαφέρεσαι να ασχοληθείς με το αντικείμενο του προγραμματισμού υπολογιστών στο μέλλον;	-,057	,482	,081	-,223	,108	-,702	34	,487
Pair 6	Διάθεση χρήσης του AI - Σκοπεύεις να συνεχίσεις να χρησιμοποιείς το AI;	-,057	,416	,070	-,200	,086	-,813	34	,422

Με σκίαση σημειώνονται τα κελιά όπου παρατηρήθηκε στατιστικά σημαντική διαφορά (Pair 1, 3 και 4) και επισημαίνονται επίσης οι μέσες τιμές που δείχνουν θετική στάση απέναντι στο App Inventor (pair 1, 2, 3, 5 και 6). Με κόκκινη και πλάγια γραφή επισημάνθηκε το ζεύγος στο οποίο φάνηκε να διαμορφώνεται αρνητική στάση για το App Inventor σε σχέση με τον προγραμματισμό υπολογιστών γενικότερα (Pair 4).

Γ.2 Σύγκριση του βαθμού ικανότητας ανάπτυξης εφαρμογής στο AI μεταξύ μαθητών Γ' Γυμνασίου και Β' Λυκείου

Τα αποτελέσματα του ελέγχου ανεξάρτητων δειγμάτων (independent samples t-test) για τις απαντήσεις των μαθητών στην ίδια ερώτηση, σε σχέση με τις δυο ακραίες ηλικιακά ομάδες του δείγματος, είναι:

Group Statistics					
	Τάξη	N	Mean	Std. Deviation	Std. Error Mean
Σε τι βαθμό θεωρείς ότι μπορείς να φτιάξεις τη δική σου εφαρμογή με τη βοήθεια του AI;	Γ Γυμνασίου	9	3,44	,527	,176
	Β Γενικού Λυκείου	12	3,50	,522	,151

Independent Samples Test										
		Levene's Test for Equality of Variances		t-test for Equality of Means						
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference	
									Lower	Upper
Σε τι βαθμό θεωρείς ότι μπορείς να φτιάξεις τη δική σου εφαρμογή με τη βοήθεια του AI;	Equal variances assumed	,136	,717	-,240	19	,813	-,056	,231	-,539	,428
	Equal variances not assumed			-,240	17,298	,813	-,056	,231	-,543	,432

Δεν παρατηρούμε στατιστικά σημαντική διαφορά.

Γ.3 Δυσκολίες που αντιλαμβάνονται οι μαθητές

Οι ίδιοι οι μαθητές απαντούν σε τετραβάθμια κλίμακα (1: Πολύ – 2: Αρκετά – 3: Λίγο – 4: Σχεδόν καθόλου) στο κατά πόσο θεωρούν ότι κατέκτησαν τις αναφερόμενες δεξιότητες.

Statistics				
Αισθάνεσαι ότι είσαι σε θέση ...		[Να καταλαβαίνεις τι κάνουν οι εντολές που βλέπεις σε ένα έτοιμο έργο]	[Να τροποποιείς έτοιμα κομμάτια εντολών και να τα προσαρμόζεις στις ανάγκες σου]	[Να εντοπίζεις και να διορθώνεις λάθη σε υπάρχον κώδικα]
N	Valid	35	35	35
	Missing	0	0	0
Mean		1,83	1,89	2,26

Αισθάνεσαι ότι είσαι σε θέση ... [Να καταλαβαίνεις τι κάνουν οι εντολές που βλέπεις σε ένα έτοιμο έργο]					
		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Πολύ καλά	11	31,4	31,4	31,4
	Αρκετά καλά	19	54,3	54,3	85,7
	Λίγο	5	14,3	14,3	100,0
	Total	35	100,0	100,0	

Αισθάνεσαι ότι είσαι σε θέση ... [Να τροποποιείς έτοιμα κομμάτια εντολών και να τα προσαρμόζεις στις ανάγκες σου]					
		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Πολύ καλά	11	31,4	31,4	31,4
	Αρκετά καλά	17	48,6	48,6	80,0
	Λίγο	7	20,0	20,0	100,0
	Total	35	100,0	100,0	

Αισθάνεσαι ότι είσαι σε θέση ... [Να εντοπίζεις και να διορθώνεις λάθη σε υπάρχον κώδικα]					
		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Πολύ καλά	7	20,0	20,0	20,0
	Αρκετά καλά	12	34,3	34,3	54,3
	Λίγο	16	45,7	45,7	100,0
	Total	35	100,0	100,0	

Γ.4 Αντίληψη των μαθητών για την ικανότητά τους στην κατάκτηση των δεξιοτήτων

Οι απαντήσεις των μαθητών σε τετραβάθμια κλίμακα (1: Πολύ καλά- 2: Αρκετά καλά – 3: Λίγο – 4: Σχεδόν καθόλου).

Statistics											
Πιστεύεις ότι γνωρίζεις να ...		σχεδιάζεις το περιβάλλον διεπαφής (οθόνη) της εφαρμογής	ορίζεις ιδιότητες στα αντικείμενα της οθόνης	προσθέτεις εικόνες και ήχους]	εκτελείς ενέργειες με το πάτημα κουμπιών	εκτελείς ενέργειες χρησιμοποιώντας τους αισθητήρες της συσκευής	δημιουργείς τις δικές σου μεταβλητές	εκτελείς μαθηματικές πράξεις	συντάσσεις λογικές εκφράσεις	χρησιμοποιείς την εντολή επιλογής IF	επεξεργάζεσαι τις τιμές των μεταβλητών κατά την εκτέλεση της εφαρμογής
N	Valid	35	35	35	35	35	35	35	35	35	35
	Missing	0	0	0	0	0	0	0	0	0	0
Mean		1,71	1,83	1,43	1,74	2,00	2,14	1,83	1,83	1,91	1,91
Std. Deviation		,710	,747	,655	,701	,686	,912	,822	,954	,951	,853

Πιστεύεις ότι γνωρίζεις να ...	Πολύ καλά	Αρκετά καλά	Λίγο	Σχεδόν καθόλου
σχεδιάζεις το περιβάλλον διεπαφής (οθόνη) της εφαρμογής	42,9	42,9	14,3	0
ορίζεις ιδιότητες στα αντικείμενα της οθόνης	34,3	51,4	11,4	2,9
προσθέτεις εικόνες και ήχους	65,7	25,7	8,6	0
εκτελείς ενέργειες με το πάτημα κουμπιών	40,0	45,7	14,3	0
εκτελείς ενέργειες χρησιμοποιώντας τους αισθητήρες της συσκευής	22,9	54,3	22,9	0
δημιουργείς τις δικές σου μεταβλητές	28,6	34,3	31,4	5,7
εκτελείς μαθηματικές πράξεις	40,0	40,0	17,1	2,9
συντάσσεις λογικές εκφράσεις	48,6	25,7	20,0	5,7
χρησιμοποιείς την εντολή επιλογής IF	42,9	28,6	22,9	5,7
επεξεργάζεσαι τις τιμές των μεταβλητών κατά την εκτέλεση της εφαρμογής	34,3	45,7	14,3	5,7

Παράρτημα Δ

Αποσπάσματα συνεντεύξεων

Στη συνέχεια παρατίθενται οι απόψεις έξι εκπαιδευτικών, όπως καταγράφηκαν στις ημιδομημένες συνεντεύξεις τους για την αποτύπωση των εντυπώσεών τους από τη χρήση του AI. Οι ερωτήσεις εστίασαν στην άποψή τους για το AI ως εργαλείο για τη διδασκαλία και μάθηση βασικών προγραμματιστικών εννοιών στη δευτεροβάθμια εκπαίδευση, στον εντοπισμό του δυνατού του σημείου ή κάποιου που τους προβληματίζει και σε τι πιστεύουν ότι θα βοηθήσει ή που υστερεί.

Εκπαιδευτικός Α:

Σε γενικές γραμμές θα έλεγα ότι είναι ένα εργαλείο που ενδείκνυται για εκμάθηση προγραμματισμού με μεγάλο πλεονέκτημα το γεγονός ότι είναι πολύ ελκυστικό για τους μαθητές. Για την ακρίβεια δεν νομίζω ότι υπάρχει εργαλείο που να είναι πιο ενδιαφέρον και συγχρόνως διασκεδαστικό για τους μαθητές της δευτεροβάθμιας εκπαίδευσης! Αυτό που με προβληματίζει είναι η απαραίτητη (κατά τη γνώμη μου) χρήση του κινητού τηλεφώνου τους την ώρα του μαθήματος. Επίσης, θεωρώ πως χρειάζεται να αφιερωθούν αρκετές διδακτικές ώρες μέχρις ότου οι μαθητές να φτάσουν σ' έναν ικανοποιητικό βαθμό κατανόησης και χρήσης του

περιβάλλοντος ή/και των εννοιών, οπότε δεν μπορώ να πω με σιγουριά σε ποιο μάθημα του ΓΕΛ θα μπορούσε να ενταχθεί (λόγω έλλειψης ωρών). Στο Γυμνάσιο είναι πιο ελαστικά τα πράγματα και εύκολα θα μπορούσε να διδαχθεί στη Γ' Γυμνασίου.

Εκπαιδευτικός Β:

Το App Inventor ήταν πολύ άνετο στην χρήση του, καθώς σε αρκετά σημεία μοιάζει με το Scratch. Βέβαια ένα σημαντικό μειονέκτημά του είναι ότι η γλώσσα που χρησιμοποιείται είναι η Αγγλική με την οποία αρκετοί είχαν πρόβλημα. Το γραφικό περιβάλλον για την δημιουργία της εφαρμογής είναι πολύ άνετο και έγινε άμεσα αποδεκτό από τους μαθητές. Ως προς το κομμάτι προγραμματισμού των blocks, το drag and drop των εντολών επιταχύνει και διευκολύνει την σύνταξη του προγράμματος. Ωστόσο μερικές εντολές απαιτούν την σύνθεση τους από πολλαπλά σημεία, με αποτέλεσμα οι μαθητές ορισμένες φορές να μπερδεύονται. Ένα επιπλέον πρόβλημα που παρατήρησα είναι πως, όταν το πρόγραμμα αποτελείται από πολλά blocks εντολών, δυσκολεύεται να τα δεις όλα μαζί ταυτόχρονα, ειδικά όταν δεν έχεις οθόνη με διάσταση από 20" και πάνω. Σε γενικές γραμμές άρεσε το γεγονός της άμεσης εμφάνισης του αποτελέσματος στα κινητά των παιδιών και πιστεύω ότι με λίγες βελτιώσεις, το App Inventor θα είναι ότι καλύτερο για διδασκαλία προγραμματισμού σε Γυμνάσιο και Α' Λυκείου.

Εκπαιδευτικός Γ:

Το δυνατό σημείο του App Inventor είναι η διασύνδεσή του με φορητές υπολογιστικές συσκευές (κινητά, ταμπλέτες). Οι μαθητές κρατάνε στα χέρια τους το υπολογιστικό σύστημα στο οποίο εκτελείται το πρόγραμμά τους και αλληλοεπιδρούν με αυτό με πολύ φυσικό τρόπο (εξαιτίας των διαθέσιμων αισθητήρων). Αυτό καθιστά το περιβάλλον στο οποίο μαθαίνουν να προγραμματίζουν ιδιαίτερα ευχάριστο και ελκυστικό, ενώ έχουν, ούτως ή άλλως, πολύ στενή σχέση με τις συσκευές αυτές. Ωστόσο, το ίδιο το περιβάλλον έχει αρκετούς περιορισμούς, οι οποίοι γίνονται άμεσα αντιληπτοί όταν οι μαθητές θελήσουν να υλοποιήσουν τις δικές τους εφαρμογές ή να τροποποιήσουν τα παραδείγματα που τους δίνονται με αυθαίρετο τρόπο. Ενδεικτικά αναφέρω την αδυναμία να δημιουργήσουν προγραμματιστικά νέα αντικείμενα. Επίσης, το γεγονός ότι υπάρχει ένας κοινός χώρος για την ανάπτυξη του κώδικα (και όχι

ξεχωριστοί χώροι ανά αντικείμενο, όπως στο Scratch) οδηγεί σε προγράμματα τα οποία δεν είναι εύκολα διαχειρίσιμα και δεν ευνοεί την ανάπτυξη κώδικα με μια αντικειμενοστρεφή θεώρηση. Προσωπικά έχω ανάμεικτα συναισθήματα για την χρήση του AppInventor στην εκπαίδευση. Σίγουρα αξίζει να έρθουν οι μαθητές σε επαφή με αυτό, αλλά δεν είμαι βέβαιος ότι είναι ένα επαρκώς ώριμο και δομημένο περιβάλλον ώστε να αποτελέσει την πρώτη επιλογή για την διδασκαλία προγραμματιστικών εννοιών.

Εκπαιδευτικός Δ:

Στα θετικά του AI είναι η "γοητεία" που ασκεί στους μαθητές, εξαιτίας της δυνατότητας που τους δίνει να φτιάξουν πραγματικές εφαρμογές για τις κινητές τους συσκευές. Είναι ένα περιβάλλον οπτικού προγραμματισμού που γλυτώνει τον αρχάριο χρήστη από τη δυσκολία των συντακτικών σφαλμάτων. Παρόλα αυτά δεν στερείται δυνατοτήτων.

Από την άλλη, αν το συγκρίνουμε με περιβάλλοντα όπως το Scratch ή το Snap! (BYOB), όσον αφορά στην καμπύλη εκμάθησης, χάνει στα σημεία. Καταρχήν, κρύβοντας ουσιαστικά ένα περιβάλλον αντικειμενοστραφούς προγραμματισμού, το AI αναγκάζει τον χρήστη του να έρθει αντιμέτωπος με μια σειρά διαφορετικών εντολών (μεθόδων) για κάθε αντικείμενο που θέλει να χρησιμοποιήσει. Επιπλέον, κάποιες προγραμματιστικές έννοιες, όπως η δομή της επανάληψης είναι δύσκολο να ενταχθούν στη διδασκαλία του προγραμματισμού μέσω AI, μιας και στις περισσότερες περιπτώσεις η επαναληπτική εκτέλεση των εντολών αντικαθίσταται από το μηχανισμό του καθοδηγούμενου από γεγονότα προγραμματισμού. Σημαντικό εμπόδιο αποτελεί και η έκδοση του AI μόνο στα Αγγλικά. Τέλος, προς το παρόν στερείται κάποιων δυνατοτήτων που θα του προσέδιδαν μεγαλύτερη ευελιξία, όπως τη δυναμική δημιουργία αντικειμένων (βλέπε κλώνοι του Scratch).

Εν κατακλείδι, το AI δεν είναι το καταλληλότερο περιβάλλον για την εισαγωγή στον προγραμματισμό. Ένας αρχάριος χρήστης και ιδιαίτερα μικρής ηλικίας θα δυσκολευτεί αρκετά, τουλάχιστον στα αρχικά στάδια χρήσης του. Καλύτερα η χρήση του να ακολουθήσει τη διδασκαλία κάποιου άλλου περιβάλλοντος παρόμοιου τύπου, όπως το Scratch.

Εκπαιδευτικός Ε:

Τα προφανή πλεονεκτήματα του AI, με βάση τη σχετική μικρή εμπειρία μου από τη χρήση του είναι:

1. Καλό εργαλείο για εισαγωγή στον προγραμματισμό, δομημένο (δομές ακολουθίας, επιλογής, επανάληψης) αλλά και αντικειμενοστρεφές και καθοδηγούμενο από τα γεγονότα.
2. Περιβάλλον φιλικό για νέους χρήστες και WYSIWYG, στα αγγλικά βέβαια, αλλά μόνο η Γλώσσα είναι στα ελληνικά και κάτι Logo. Λογικά δεν πρέπει να είναι πολύ δύσκολο να μεταφραστεί.
3. Άμεση σύνδεση για έλεγχο του αποτελέσματος με κινητά και τάμπλετ που αποτελούν πλέον αναπόσπαστα, όχι απλά εργαλεία, αλλά «μέλη των σωμάτων» των παιδιών.
4. Δυνατότητα κατασκευής αξιόλογων εφαρμογών, χρησιμοποιώντας δυνατότητες όπως αξελερόμετρα, sms, πολυμέσα κ.τ.λ. που αρέσουν στα παιδιά.

Από την άλλη, ενδεχόμενα μειονεκτήματα είναι:

1. Δεν ξέρω πόσο εύκολο είναι να κάνεις πραγματικές βαριές εφαρμογές με χρήση π.χ. πινάκων, χειριστηρίων ή πολλών αντικειμένων. Σίγουρα πάντως δεν είναι αυτός ο σκοπός μας στη Δ/θμια.
2. Δυνατότητες debugging έχει; Δεν το έχω ψάξει αρκετά για να ξέρω.
3. Έχει ωριμάσει αρκετά, ώστε ένας καθηγητής να "ρискάρει" να αφιερώσει χρόνο και ενέργεια να το μάθει, να φτιάξει υλικό κ.τ.λ. ή τυχόν μια επόμενη έκδοση θα ανατρέψει το σκηνικό; Δε μιλάω προσωπικά, ούτε και θυμάμαι πόσα πράγματα έμαθα με χαρά στο παρελθόν και τώρα πια δεν ισχύουν αλλά μου άφησαν τη μαγιά για να μάθω τα επόμενα. Συνάδελφοι όμως σίγουρα θα θέλουν να πεισθούν.
4. Και βέβαια, τα κινητά ΑΠΑΓΟΡΕΥΟΝΤΑΙ στο σχολείο!

Καταλήγω στο τελικό αποτέλεσμα: 4-4 :), οπότε στην παράταση το τελικό αποτέλεσμα κατέληξε υπέρ του App Inventor... Σοβαρά τώρα, η βαρύτητα των πλεονεκτημάτων δε συγκρίνεται με τα τυχόν μειονεκτήματα.

Εκπαιδευτικός ΣΤ:

Πιστεύω ότι το App Inventor έχει άμεση απήχηση στα παιδιά, από τη στιγμή που τους παρέχεται η δυνατότητα να δημιουργούν εύκολα εφαρμογές για τις φορητές τους συσκευές. Κατά τη γνώμη μου, όσον αφορά το Γυμνάσιο, θα μπορούσε να χρησιμοποιηθεί σε μαθητές Γ' τάξης, όχι όμως σε προχωρημένο επίπεδο.