



ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΙΩΑΝΝΙΝΩΝ

**ΣΧΟΛΗ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**  
**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**  
**ΠΜΣ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ ΔΙΚΤΥΩΝ**

**ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

**Ο ΜΙΚΡΟΕΛΕΓΚΤΗΣ ARDUINO ΩΣ ΕΡΓΑΛΕΙΟ ΜΑΘΗΣΗΣ**  
**ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΣΤΗΝ ΠΡΩΤΟΒΑΘΜΙΑ ΕΚΠΑΙΔΕΥΣΗ**

Νάσης Βασίλειος

Επιβλέπων καθηγητής: Επίκουρος Καθηγητής Λιαροκάπης Δημήτριος  
Άρτα, Ιανουάριος 2022

**THE ARDUINO MICROCONTROLLER AS A PROGRAMMING  
LEARNING TOOL IN ELEMENTARY EDUCATION**





## **Δήλωση μη λογοκλοπής**

Δηλώνω υπεύθυνα και γνωρίζοντας τις κυρώσεις του Ν. 2121/1993 περί Πνευματικής Ιδιοκτησίας, ότι η παρούσα μεταπτυχιακή εργασία είναι εκ ολοκλήρου αποτέλεσμα δικής μου ερευνητικής εργασίας, δεν αποτελεί προϊόν αντιγραφής ούτε προέρχεται από ανάθεση σε τρίτους. Όλες οι πηγές που χρησιμοποιήθηκαν (κάθε είδους, μορφής και προέλευσης) για τη συγγραφή της περιλαμβάνονται στη βιβλιογραφία.

Νάσης Βασίλειος

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Ευχαριστώ τον επιβλέποντα καθηγητή κύριο Λιαροκάπη Δημήτριο καθώς χωρίς την καθοδήγησή του και την υπομονή του κατά την πραγματοποίηση της διπλωματικής εργασίας αλλά και κατά την αναζήτηση θέματος θα ήταν αδύνατη η ολοκλήρωση της. Ακόμη, τον διευθυντή του 5<sup>ου</sup> Δημοτικού Σχολείου Νέας Φιλαδέλφειας κύριο Άγγελο Συμεωνίδα και την διευθύντρια του 4<sup>ου</sup> Δημοτικού Σχολείου Νέας Φιλαδέλφειας κυρία Πέπη Παπανικολάου για την παραχώρηση του σχολικού εξοπλισμού εκτός σχολικού ωραρίου.

## ΠΕΡΙΛΗΨΗ

Η παρούσα εργασία πραγματεύεται την χρήση του μικροελεγκτή Arduino σαν εργαλείο μάθησης στην πρωτοβάθμια εκπαίδευση, στο μάθημα της Πληροφορικής(Τεχνολογίες Πληροφορικής Επικοινωνιών). Πιο συγκεκριμένα επικεντρώνεται στην χρήση του στις τελευταίες τάξεις του Δημοτικού Σχολείου δηλαδή στις ηλικίες 10-12 ετών. Γίνεται βιβλιογραφική αναφορά στην ιστορία του μαθήματος της Πληροφορικής και στους στόχους που εκπληρώνονται μέσα σε αυτό ενώ παρουσιάζονται οι ελλείψεις και οι αστοχίες που παρουσιάζει το πρόγραμμα σπουδών. Στη συνέχεια παρουσιάζεται η προσπάθεια εισαγωγής των μαθητών στην υπολογιστική σκέψη και σε στοιχεία προγραμματισμού μέσω εφαρμογών και δραστηριοτήτων σε επίπεδο λογισμικού για να περάσουμε τελικά στον στόχο της εργασίας που είναι η χρήση του Arduino ως εργαλείο προγραμματισμού. Μέσα από μικρές προγραμματιστικές ασκήσεις οι μαθητές προσπαθούν να εξοικειωθούν με τον μικροελεγκτή αλλά και με την έννοια του προγραμματισμού ενώ επιχειρείται η σύνδεση με μια πραγματική γλώσσα προγραμματισμού όπως η javascript μέσα από απλά διαδικτυακά κουίζ στο Kahoot. Τέλος, παρουσιάζονται τα συμπεράσματα και προτείνονται θέματα για μελλοντική έρευνα.

**Λέξεις-κλειδιά:** Arduino, Δημοτικό σχολείο, προγραμματισμός, πληροφορική.

## **ABSTRACT**

This master thesis deals with using Arduino Microcontroller as a programming learning tool in Elementary Education as a part of the Informatics Class. More specifically, it focuses on the last two grades of Elementary Education. Chapter 1 includes a bibliographic reference about the history of the Informatics Course and discusses the objectives that are fulfilled in it, while the shortcomings and failures are presented too. Chapter 2 includes an introduction to Computational Thinking and Programming for Kids through block-based programming applications. In Chapter 3 STEM Methodology is presented as well as its implementation through Educational Robotics. Chapter 4 includes the main goal of this master thesis which is the use of Arduino Microcontroller as a programming learning tool. For this purpose, there have been created ten programming activities based on Arduino UNO. Moreover, an introduction to the programming language Javascript is made through online quizzes. Finally, the conclusions of the research are presented.

**Key-words:** Arduino, Elementary Education, Programming tool, Informatics Class



## Περιεχόμενα

ΕΥΧΑΡΙΣΤΙΕΣ.....	ii
ΠΕΡΙΛΗΨΗ.....	iii
ABSTRACT.....	iv
Κατάλογος Εικόνων.....	vii
Κεφάλαιο 1.....	1
1.1 Η πληροφορική στην εκπαίδευση.....	1
1.2 ΤΠΕ και Επιστήμη των Υπολογιστών.....	10
1.3 Θεωρίες μάθησης και διδακτικά μοντέλα.....	12
Κεφάλαιο 2.....	15
2.1 Η Υπολογιστική Σκέψη.....	15
2.2 Από την Υπολογιστική σκέψη στον προγραμματισμό.....	20
Κεφάλαιο 3.....	31
3.1 Τι είναι το STEM.....	31
3.2 Το STEM στην εκπαίδευση.....	33
3.3 STEM και Εκπαιδευτική Ρομποτική.....	35
Κεφάλαιο 4.....	45
4.1.1 Το Arduino.....	45
4.1.2 Τεχνικά χαρακτηριστικά του Arduino Uno.....	48
4.2 Το περιβάλλον προγραμματισμού Pictoblox.....	51
4.2.1 Τι είναι το Pictoblox.....	51
4.2.2 Το γραφικό περιβάλλον του Pictoblox.....	53

4.3 Φύλλα δραστηριοτήτων προγραμματισμού με Arduino.....	65
4.3.1 Εισαγωγή και προετοιμασία .....	65
4.3.2 1 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΑΝΑΒΟΣΒΗΝΩΝΤΑΣ ΤΟ LED».....	68
4.3.3 2 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΠΡΟΓΡΑΜΜΑΤΙΖΩ ΕΝΑΝ ΔΙΑΚΟΠΤΗ» .....	71
4.3.4 3 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΑΛΛΑΖΩ ΤΗΝ ΦΩΤΕΙΝΟΤΗΤΑ ΤΟΥ LED» .....	76
4.3.5 4 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΔΙΝΩ ΧΡΩΜΑΤΑ ΣΤΟ LED» .....	81
4.3.6 5 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΑΥΤΟΜΑΤΟΣ ΦΩΤΙΣΜΟΣ» .....	87
4.3.7 6 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΠΡΟΓΡΑΜΜΑΤΙΖΩ ΤΟΝ ΔΙΚΟ ΜΟΥ ΣΥΝΑΓΕΡΜΟ».....	93
4.3.8 7 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΕΛΕΓΧΩ ΕΝΑΝ ΣΕΡΒΟΚΙΝΗΤΗΡΑ» .....	97
4.3.9 8 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΜΕΤΡΑΩ ΤΗΝ ΘΕΡΜΟΚΡΑΣΙΑ» .....	103
4.3.10 9 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΒΡΙΣΚΩ ΤΗΝ ΑΠΟΣΤΑΣΗ ΑΠΟ ΕΜΠΟΔΙΑ» .....	108
4.3.11 10 <sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΠΡΟΓΡΑΜΜΑΤΙΖΩ ΕΝΑ ΜΟΤΕΡΑΚΙ».....	113
Βασικά Συμπεράσματα και προτάσεις για μελλοντική έρευνα.....	119
ΠΑΡΑΡΤΗΜΑ .....	123
Κώδικες σε JavaScript.....	123
Κώδικας 1 <sup>ου</sup> Φύλλου Εργασίας .....	123
Κώδικας 2 <sup>ου</sup> Φύλλου Εργασίας.....	123
Κώδικας 3 <sup>ου</sup> Φύλλου εργασίας .....	124
Κώδικας 4 <sup>ου</sup> Φύλλου εργασίας .....	125
Κώδικας 5 <sup>ου</sup> Φύλλου εργασίας .....	127
Κώδικας 6 <sup>ου</sup> Φύλλου Εργασίας .....	128
Κώδικας 7 <sup>ου</sup> Φύλλου εργασίας.....	128
Κώδικας 8 <sup>ου</sup> Φύλλου Εργασίας .....	130
Κώδικας 9 <sup>ου</sup> Φύλλου Εργασίας .....	131
Κώδικας 10 <sup>ου</sup> Φύλλου Εργασίας.....	131
Βιβλιογραφία .....	133

## Κατάλογος Εικόνων

ΕΙΚΟΝΑ 1. Η ΥΠΟΛΟΓΙΣΤΗ ΣΚΕΨΗ .....	19
ΕΙΚΟΝΑ 2. ΗΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ LOGO .....	22
ΕΙΚΟΝΑ 3. ΤΟ ARDUINO UNO .....	46
ΕΙΚΟΝΑ 4. ΤΟ ARDUINO MEGA .....	47
ΕΙΚΟΝΑ 5. ΤΟ ARDUINO LEONARDO .....	47
ΕΙΚΟΝΑ 6. ΤΟ ARDUINO NANO .....	47
ΕΙΚΟΝΑ 7. ΤΕΧΝΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ARDUINO UNO .....	48
ΕΙΚΟΝΑ 8. ΤΑ ΣΤΟΙΧΕΙΑ ΤΗΣ ΠΛΑΤΦΟΡΜΑΣ ARDUINO UNO .....	51
ΕΙΚΟΝΑ 9. ΤΟ ΠΕΡΙΒΑΛΛΟΝ ΤΟΥ ΡΙΣΤΟΒΛΟΧ.....	54
ΕΙΚΟΝΑ 10. ΤΟ ΠΡΟΓΡΑΜΜΑ ΤΟΥ 1ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	70
ΕΙΚΟΝΑ 11. ΚΟΥΙΖ 1ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	71
ΕΙΚΟΝΑ 12. ΤΟ ΠΡΟΓΡΑΜΜΑ ΤΟΥ 2ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	74
ΕΙΚΟΝΑ 13. ΚΟΥΙΖ 2ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	75
ΕΙΚΟΝΑ 14. ΤΟ ΠΡΟΓΡΑΜΜΑ ΤΟΥ 3ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	78
ΕΙΚΟΝΑ 15. ΤΟ ΚΟΥΙΖ ΤΟΥ 3ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	80
ΕΙΚΟΝΑ 16. ΤΟ ΠΡΟΓΡΑΜΜΑ ΤΟΥ 4ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	85
ΕΙΚΟΝΑ 17. ΤΟ ΚΟΥΙΖ ΤΟΥ 4ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	87
ΕΙΚΟΝΑ 18. ΤΟ ΠΡΟΓΡΑΜΜΑ ΤΟΥ 5ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	90
ΕΙΚΟΝΑ 19. ΤΟ ΚΟΥΙΖ ΤΟΥ 5ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	93
ΕΙΚΟΝΑ 20. ΤΟ ΠΡΟΓΡΑΜΜΑ ΤΟΥ 6ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	96
ΕΙΚΟΝΑ 21. ΤΟ ΚΟΥΙΖ ΤΟΥ 6ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	97
ΕΙΚΟΝΑ 22. ΤΟ ΠΡΟΓΡΑΜΜΑ ΤΟΥ 7ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	101
ΕΙΚΟΝΑ 23. ΤΟ ΚΟΥΙΖ ΤΟΥ 7ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	102
ΕΙΚΟΝΑ 24. ΤΟ 1Ο ΠΡΟΓΡΑΜΜΑ ΤΟΥ 8ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	106
ΕΙΚΟΝΑ 25. ΤΟ 2Ο ΠΡΟΓΡΑΜΜΑ ΤΟΥ 8ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	106
ΕΙΚΟΝΑ 26. ΕΝΑΛΛΑΚΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΑΙΣΘΗΤΗΡΑ ΘΕΡΜΟΚΡΑΣΙΑΣ .....	107
ΕΙΚΟΝΑ 27. ΚΟΥΙΖ 8ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	108
ΕΙΚΟΝΑ 28. ΤΟ ΠΡΟΓΡΑΜΜΑ ΤΟΥ 9ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ. ....	111
ΕΙΚΟΝΑ 29. ΤΡΟΠΟΠΟΙΗΣΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ ΤΟΥ 9ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	112
ΕΙΚΟΝΑ 30. ΚΟΥΙΖ 9ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	113
ΕΙΚΟΝΑ 31. ΠΡΟΓΡΑΜΜΑ 10ΟΥ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ .....	116

ΕΙΚΟΝΑ 32. ΠΛΑΚΙΔΙΟ ΔΙΑΧΕΙΡΙΣΗ ΠΟΛΛΑΠΛΩΝ ΚΙΝΗΤΗΡΩΝ

ΕΙΚΟΝΑ 33. SHIELDΓΙΑ

ΔΙΑΣΥΝΔΕΣΗ ΚΙΝΗΤΗΡΩΝ .....	117
ΕΙΚΟΝΑ 34. ΚΟΥΙΖ 100Υ ΦΥΛΛΟΥ ΕΡΓΑΣΙΑΣ.....	118

# Κεφάλαιο 1

## 1.1 Η πληροφορική στην εκπαίδευση

Ζώντας στην εποχή της πληροφορίας με τις Τεχνολογίες της Πληροφορικής και των Επικοινωνιών(ΤΠΕ) να είναι αναπόσπαστο κομμάτι σε κάθε πτυχή της καθημερινής μας ζωής δεν θα μπορούσε να ισχύει κάτι διαφορετικό για την εκπαίδευση. Το σύγχρονο εκπαιδευτικό περιβάλλον πρέπει να έχει ως προτεραιότητα την «αξιοποίηση των ΤΠΕ στην εκπαιδευτική διαδικασία».

Στα εκπαιδευτικά συστήματα των «ανεπτυγμένων» χωρών οι ΤΠΕ έχουν ενταχθεί από πολύ νωρίς είτε ως εργαλείο διδασκαλίας και μάθησης των διάφορων γνωστικών αντικειμένων είτε ως αντικείμενο γνώσης ακόμα και ως μέσο για την διαχείριση του σχολικού περιβάλλοντος και της σχολικής καθημερινότητας. (Εισαγωγή στις εκπαιδευτικές εφαρμογές των τεχνολογιών της πληροφορίας και των επικοινωνιών, 2004)

Ο Β. Κομής(2005) διαχωρίζει την διαδικασία ένταξης της πληροφορικής στα εκπαιδευτικά συστήματα των «ανεπτυγμένων» χωρών σε τέσσερις φάσεις:

- Μέχρι το 1970 η περίοδος μπορεί να χαρακτηριστεί ως περίοδος από την εισαγωγή κάποιων συσκευών στη εκπαίδευση. Παρατηρείται για πρώτη φορά η δημιουργία ειδικών συσκευών με στόχο την βοήθεια των εκπαιδευτικών στην διδασκαλία των μαθημάτων του εκάστοτε προγράμματος σπουδών.
- Την δεκαετία 1970-1980 επιχειρείται για πρώτη φορά σε πιλοτικό στάδιο η εισαγωγή της επιστήμης της Πληροφορικής ως αυτόνομο γνωστικό αντικείμενο στα προγράμματα σπουδών των «ανεπτυγμένων» χωρών. Η εισαγωγή αυτή ωστόσο γίνεται μόνο στο επίπεδο των τελευταίων τάξεων της δευτεροβάθμιας εκπαίδευσης.
- Στην δεκαετία που ακολουθεί (1980-1990) η Πληροφορική αντιμετωπίζεται και ως αυτόνομο γνωστικό αντικείμενο αλλά και ως μέσο για την διδασκαλία των υπόλοιπων γνωστικών αντικειμένων. Επίσης στην διάρκεια αυτής της δεκαετίας επιχειρείται για πρώτη φορά η γενικευμένη εισαγωγή της Πληροφορικής σε όλες τις

βαθμίδες της εκπαίδευσης. Γίνεται πλέον πανθομολογούμενο ότι η Πληροφορική ως επιστήμη αποτελεί αναγκαιότητα και παράλληλα αναγνωρίζεται η συμβολή της ως εκπαιδευτικό μέσο διδασκαλίας. Σε αυτό συνέβαλλε ασφαλώς η ραγδαία και συνεχής ανάπτυξη της τεχνολογίας των ηλεκτρονικών υπολογιστών με την ταυτόχρονη μείωση του κόστους τους.

- Από το 1990 και έπειτα οι ΤΠΕ έχουν πλέον εδραιωθεί στην εκπαίδευση. Ωστόσο, σε αυτή την περίοδο παρατηρείται μια σημαντική αναθεώρηση στον ρόλο της Πληροφορικής στα αναλυτικά προγράμματα σπουδών. Πιο συγκεκριμένα παρατηρείται, ειδικά στην πρωτοβάθμια και την δευτεροβάθμια εκπαίδευση, μια υποβάθμιση της Πληροφορικής από αυτόνομο γνωστικό αντικείμενο σε βοηθητικό εργαλείο για το σύνολο της εκπαιδευτικής διαδικασίας και για την καθημερινή λειτουργία του σχολείου. Σημαντικό ρόλο σε αυτό έπαιξε η παράλληλη ραγδαία ανάπτυξη του διαδικτύου αλλά και η συνολική ανάπτυξη της τεχνολογίας των ηλεκτρονικών υπολογιστών που επέτρεψε την ανάπτυξη και διάδοση εκπαιδευτικών εφαρμογών βασισμένων σε πολυμέσα.

Η διαδικασία της ένταξης των ΤΠΕ στην εκπαίδευση στηρίχθηκε σε τρεις βασικές προσεγγίσεις ή αλλιώς πρότυπα (Κομής, Εισαγωγή στην διδακτική της πληροφορικής, 2005):

- Το τεχνοκεντρικό πρότυπο το οποίο υπαγορεύει την εισαγωγή της επιστήμης της Πληροφορικής ως αυτόνομο γνωστικό αντικείμενο και έχει ως στόχο της απόκτηση γνώσεων τόσο σε επίπεδο πληροφορικού αλφαριθμητισμού αλλά και σε υψηλότερο επίπεδο γνώσεων πάνω στην Πληροφορική.
- Το ολιστικό πρότυπο σύμφωνα με το οποίο οι ΤΠΕ εντάσσονται σε όλα τα αντικείμενα του αναλυτικού προγράμματος σπουδών και σε όλες τις βαθμίδες εκπαίδευσης ως εργαλείο διδασκαλίας, έρευνας και μάθησης. Σύμφωνα με το Παιδαγωγικό Ινστιτούτο(1998) «το πρότυπο αυτό προϋποθέτει διαφορετικές παιδαγωγικές αντιλήψεις, τόσο στην επιλογή της γνώσης τόσο στην επιλογή της γνώσης και της διδακτικής πρακτικής, όσο και στην εκπαίδευση των εκπαιδευτικών και την υλικοτεχνική υποδομή». Η συγκεκριμένη προσέγγιση, ωστόσο, προκάλεσε

μεγάλες ανατροπές στα προγράμματα σπουδών και έγινε μη εφαρμόσιμη και επιπλέον δεν έχει επιτευχθεί σε κανένα εκπαιδευτικό σύστημα (Κομής, Εισαγωγή στις εκπαιδευτικές εφαρμογές των τεχνολογιών της πληροφορίας και των επικοινωνιών, 2004).

- Το πραγματολογικό πρότυπο της Πληροφορικής και των ΤΠΕ το οποίο μπορεί να περιγραφεί ως συνδυασμός των δύο παραπάνω προτύπων. Έτσι σύμφωνα με αυτή την προσέγγιση συνυπάρχουν τα αμιγώς μαθήματα Πληροφορικής με την χρήση υπολογιστικών τεχνολογικών εργαλείων σαν μέσο υποστήριξης της μαθησιακής διαδικασίας σε όλα τα γνωστικά αντικείμενα των προγραμμάτων σπουδών.

Στην χώρα μας όλοι σήμερα συμφωνούν ότι η Πληροφορική και εν γένει οι ψηφιακές τεχνολογίες πρέπει να παίζουν κυρίαρχο ρόλο στην εκπαίδευση από την προσχολική κιόλας ηλικία. Ήδη από τις αρχές του 2000 έχει επισημανθεί ότι το παραδοσιακό σχολείο που βασιζόταν στον εκπαιδευτικό που κατείχε την πληροφορία και τη γνώση και τη μετέδιδε στο μαθητή, πρέπει να μετατραπεί σε ένα νέο τύπο σχολείου, όπου ο ρόλος του εκπαιδευτικού θα είναι καθοδηγητικός και συμβουλευτικός και έτσι ο μαθητής θα είναι ικανός να αποκτά την πληροφορία και τη γνώση μέσω του υπολογιστή και των νέων τεχνολογιών, λειτουργώντας ως ερευνητής, καθοδηγούμενος από τον εκπαιδευτικό και καλλιεργώντας έτσι τις δεξιότητες και τα ιδιαίτερα χαρακτηριστικά του (Ένωση Πληροφορικών Ελλάδας, 2006).

Βασικό πλεονέκτημα των ψηφιακών τεχνολογιών είναι η αλληλεπιδραστικότητα που προσφέρει στο μαθητή τη δυνατότητα να συμμετέχει μαζί με τον δάσκαλό του στο σχεδιασμό των μαθησιακών δραστηριοτήτων και να εκφράζει ελεύθερα τις αντιλήψεις και τα συναισθήματά του. Επίσης, διαμορφώνεται η κατάλληλη ψυχοπαιδευτική σχολική ατμόσφαιρα και επικοινωνία μεταξύ των μελών της τάξης, στα πλαίσια μιας τάσης για ισότιμη σχέση, αλληλεπίδραση και ανατροφοδότησης (Ζωγόπουλος, 2001).

Στην Ελλάδα, η πρώτη ολοκληρωμένη πρόταση του Παιδαγωγικού Ινστιτούτου(νυν ΙΕΠ) για την εφαρμογή των Τεχνολογιών της Πληροφορίας και Επικοινωνίας(ΤΠΕ) στην εκπαίδευση καταρτίστηκε το 1997 με τίτλο «Ένιαίο Πλαίσιο Προγράμματος Σπουδών Πληροφορικής» (ΕΠΠΣΠ) και αφορούσε την εισαγωγή και ένταξη των ΤΠΕ σε όλες τις

βαθμίδες της εκπαίδευσης. Για της πρωτοβάθμια εκπαίδευση το ΕΠΠΣΠ προβλέπει ότι οι μαθητές που τελειώνουν το δημοτικό σχολείο θα πρέπει να είναι σε θέση «να περιγράφουν τα βασικά στοιχεία της αρχιτεκτονικής των υπολογιστών(μνήμη, επεξεργασία, περιφερειακά), να αναγνωρίζουν την κεντρική μονάδα και τις βασικές περιφερειακές συσκευές, να μπορούν να τις θέτουν σε λειτουργία και να εξηγούν την χρησιμότητα της καθημιάς, να μπορούν να εργάζονται με σχετική αυτονομία σε ένα γραφικό περιβάλλον εργασίας, να χρησιμοποιούν λογισμικό γενικής χρήσης για να εκφράζουν τις ιδέες τους με πολλούς τρόπους και μέσα (χρησιμοποιώντας εικόνες, ήχους, κείμενα κτλ.), να χρησιμοποιούν εφαρμογές πολυμέσων εκπαιδευτικού περιεχομένου και να έχουν κατακτήσει τις έννοιες της πλοήγησης σε ένα δίκτυο πληροφοριών και της αλληλεπίδρασης με ένα πληροφορικό σύστημα, να αναζητούν πληροφορίες από απλές βάσεις δεδομένων, να επικοινωνούν και να αναζητούν πληροφορίες χρησιμοποιώντας τον παγκόσμιο ιστό πληροφοριών, να αναφέρουν εφαρμογές της πληροφορικής στο σύγχρονο κόσμο, να αντιλαμβάνονται τον υπολογιστή, τις περιφερειακές συσκευές και το χρησιμοποιούμενο λογισμικό ως ενιαίο σύστημα.»

Στην πρώτη προσπάθεια του Παιδαγωγικού Ινστιτούτου δεν υπήρχε καμία αναφορά στην εισαγωγή των μαθητών του δημοτικού στην υπολογιστική σκέψη και σε στοιχεία προγραμματισμού κάτι που ασφαλώς έχει να κάνει και με την γενικότερη μειωμένη εισχώρηση των ψηφιακών τεχνολογιών στην χώρα μας μέχρι την χρονική περίοδο σύνταξης του ΕΠΠΣΠ. Έτσι ο ηλεκτρονικός υπολογιστής εισάγεται στην εκπαιδευτική διαδικασία αποκλειστικά σαν ένα εργαλείο γραφείου που διευκολύνει την καθημερινή εργασία και σαν μια πηγή αναζήτησης και έκφρασης πληροφοριών.

Το δεύτερο, αναθεωρημένο και διευρυμένο, πλάνο για την εισαγωγή των ΤΠΕ στην εκπαίδευση δημοσιεύτηκε από το εκπαιδευτικό ινστιτούτο το Νοέμβριο του 2003 ως «Διαθεματικό Ενιαίο Πλαίσιο Σπουδών Πληροφορικής» (ΔΕΠΠΣΠ). Το ΔΕΠΠΣΠ προήλθε ως αποτέλεσμα συζήτησης με την εκπαιδευτική κοινότητα και εσωτερικής αξιολόγησης ενώ ακολούθησε το «ολιστικό πρότυπο» (Ενωση Πληροφορικών Ελλάδας, 2006). Σύμφωνα με αυτό «βασικός σκοπός της εισαγωγής της Πληροφορικής στην πρωτοβάθμια εκπαίδευση είναι να εξοικειωθούν οι μαθητές και οι μαθήτριες με τις βασικές λειτουργίες του υπολογιστή και να έλθουν σε μια πρώτη επαφή με διάφορες χρήσεις του ως εποπτικού μέσου



διδασκαλίας, ως γνωστικού – διερευνητικού εργαλείου και ως εργαλείου επικοινωνίας και αναζήτησης πληροφοριών στο πλαίσιο των καθημερινών σχολικών τους δραστηριοτήτων με τη χρήση κατάλληλου λογισμικού και ιδιαίτερα ανοικτού λογισμικού διερευνητικής μάθησης». Σύμφωνα με το ΔΕΠΠΣΠ η εισαγωγή των ΤΠΕ στην εκπαίδευση διευρύνεται και θα μπορούσε να χωριστεί σε τέσσερις κεντρικούς άξονες:

**Εργαλείο γνώσης και έρευνας:** Με την χρήση εκπαιδευτικού λογισμικού ανοιχτού κώδικα οι μαθητές αποκτούν την δυνατότητα να διευρύνουν τις δυνατότητες μάθησης αντίστοιχα με το επίπεδο τους. Το λογισμικό μπορεί να έχει την μορφή εκπαιδευτικών παιχνιδιών και αλληλεπιδραστικών πολυμέσων κάνοντας έτσι τον ηλεκτρονικό υπολογιστή μέσο ανάπτυξης δραστηριοτήτων, διεύρυνσης γνώσεων αλλά και εκπαιδευτικής ψυχαγωγίας παράλληλα.

**Εποπτικό μέσο διδασκαλίας:** Εντάσσοντας το λογισμικό ευρείας χρήσης(ζωγραφική, επεξεργαστές κειμένου, παρουσιάσεων και λογιστικών φύλλων) στην διδασκαλία των βασικών μαθημάτων της γλώσσας και των μαθηματικών από μικρή ηλικία, οι μαθητές αποκτούν ευχέρεια στην χρήση τους ενώ παράλληλα αποκτούν παραπάνω ερεθίσματα για την κατανόηση εννοιών της ύλης.

**Εργαλείο επικοινωνίας και αναζήτησης πληροφοριών:** Οι μαθητές θα μπορούν χρησιμοποιώντας τον υπολογιστή να αντλούν πληροφορίες από βάσεις δεδομένων και ηλεκτρονικές πηγές αλλά και να επικοινωνούν με άλλους μαθητές.

**Εισαγωγή στην επιστήμη της πληροφορικής:** Εισάγεται μια πρώτη προσέγγιση σε βασικά σημεία την επιστήμης των υπολογιστών τόσο σε επίπεδο υλικού όσο και λογισμικού. Στο επίπεδο του υλικού γίνεται εισαγωγή στις λειτουργίες και την αρχιτεκτονική των βασικών συστημάτων του υπολογιστή όπως η μνήμη και ο επεξεργαστής. Σε επίπεδο λογισμικού γίνεται η πρώτη προσπάθεια εισαγωγής στην υπολογιστική σκέψη και σε κάποια στοιχεία προγραμματισμού όπως οι ακολουθίες και οι συνθήκες. Με τον τρόπο αυτό δημιουργείται η προοπτική του τεχνολογικού αλφαριθμητισμού και της αναγνώρισης των δυνατοτήτων που παρέχει η επιστήμη των υπολογιστών.

Η Ένωση Πληροφορικών Ελλάδας(2006) τονίζει πως υιοθετώντας την ολιστική προσέγγιση το ΔΕΠΠΣΠ έθετε σαν προτεραιότητες την κατανόηση των βασικών αρχών της χρήσης των ηλεκτρονικών υπολογιστών σε σημαντικές ανθρώπινες δραστηριότητες πάντα στα πλαίσια των υπόλοιπων σχολικών δραστηριοτήτων. Σημαντικός παράγοντας θεωρήθηκε η εύκολη εξοικείωση των μαθητών αυτή της ηλικίας με τον υπολογιστή. Έτσι, έγινε η

προσπάθεια να ενταχθεί αυτός στα πλαίσια διδασκαλίας βασικών γνωστικών αντικειμένων και να μπορούν οι μαθητές να αναζητούν πληροφορίες στο Διαδίκτυο ή σε άλλες πηγές πληροφοριών και να συνεργαστούν για την εκτέλεση ομαδικών δραστηριοτήτων.

Ωστόσο το μάθημα της Πληροφορικής διδάσκεται μόνο στο μεταμεσημβρινό μάθημα του ολοήμερου δηλαδή το παρακολουθούν μόνο οι μαθητές που παραμένουν στο σχολείο μετά την ολοκλήρωση των πρωινών μαθημάτων.

Η ολιστική προσέγγιση διήρκησε εν μέρει μέχρι το 2010 οπότε με την υπουργική απόφαση Φ.3/609/60745/ Γ1: «Ορισμός 800 12/θέσιων Δημοτικών Σχολείων με Ενιαίο Αναμορφωμένο Εκπαιδευτικό Πρόγραμμα» ορίζεται η εισαγωγή της Πληροφορικής ως ξεχωριστό γνωστικό αντικείμενο στην Πρωτοβάθμια εκπαίδευση, στο πρωινό ωράριο. Το μάθημα είναι δίωρο σε κάθε τάξη και διδάσκεται από εκπαιδευτικούς Πληροφορικής. Ταυτόχρονα αλλάζει και ο τίτλος του μαθήματος και γίνεται «Τεχνολογίες της Πληροφορίας και των Επικοινωνιών (ΤΠΕ).(Φ.12/879/88413/Γ1-28-07-2010, ΦΕΚ 1139/2010, τ.Β). Ωστόσο και μέχρι το 2011 τα σχολεία με Ενιαίο Αναμορφωμένο Εκπαιδευτικό Πρόγραμμα που διδάσκεται το μάθημα ανερχόταν μόλις σε 961.

Ο διαχωρισμός αυτός διατηρήθηκε μέχρι την περίοδο 2016-2017 με την ίδρυση του Ενιαίου Τύπου Ολοήμερου Δημοτικού Σχολείου (Ε.Τ.Ο.Δ.Σ) (ΦΕΚ 1324/2016). Στον Ε.Τ.Ο.Δ.Σ υπάγονται με ενιαίο διδακτικό και ωρολόγιο πρόγραμμα **όλα** τα δημοτικά σχολεία της χώρας από 4/θέσια και άνω ενώ από το σχολικό έτος 2017-2018 το πρόγραμμα επεκτάθηκε και στα Ολιγοθέσια Δημοτικά Σχολεία. Με το νέο Ωρολόγιο πρόγραμμα οι ΤΠΕ διδάσκονται πλέον σε όλα τα δημοτικά σχολεία της χώρας από μια ώρα εβδομαδιαίως σε κάθε τάξη. Ωστόσο, στα σχολεία που λειτουργούσαν ως ΕΑΕΠ οι ώρες διδασκαλίας μειώθηκαν κατά μία στις τάξεις Δ, Ε και ΣΤ.

Έχοντας πλέον υιοθετηθεί το τεχνοκεντρικό πρότυπο, η διδασκαλία των ΤΠΕ έχει σαφή εργαστηριακό προσανατολισμό. Σύμφωνα με το ισχύουσες κατευθύνσεις του υπουργείου παιδείας (Ινστιτούτο Εκπαιδευτικής Πολιτικής, 2020) βασικός παράγοντας είναι η ενεργός συμμετοχή κάθε μαθητή η συνεχής αλληλεπίδραση και η συνεργασία με τον διδάσκοντα αλλά κυρίως με τους συμμαθητές του. Το Εργαστήριο Πληροφορικής αποτελεί για τους μαθητές χώρο μελέτης έρευνας ενεργητικής συμμετοχής και συνεργασίας. Έτσι,

ενθαρρύνεται και ευνοείται η διερευνητική προσέγγιση των νέων γνώσεων η αλληλεπιδραστική και συνεργατική μάθηση η αυτενέργεια και η δημιουργικότητα των μαθητών. Το πλαίσιο διδασκαλίας συνίσταται σε τέσσερις διαστάσεις:

**Τεχνολογική:** Περιλαμβάνει τεχνικές γνώσεις που αφορούν θεμελιώδεις έννοιες ΤΠΕ(υλικό, λογισμικό, αρχείο δίκτυο κλπ) και ικανότητες χρήσεις βασικών περιβάλλοντων των ΤΠΕ(εκπαιδευτικό λογισμικό, επεξεργασία κειμένου, εννοιολογική χαρτογράφηση, λογισμικό παρουσιάσεων, υπηρεσίες Διαδικτύου κλπ).

**Γνωστική:** Περιγράφει τις θεμελιώδεις δεξιότητες αξιοποίησης των ΤΠΕ ως εργαλείο έρευνας δημιουργίας επικοινωνίας και μάθησης στα πλαίσια όλων των μαθημάτων του προγράμματος σπουδών αλλά και της καθημερινής σχολικής ζωής των μαθητών.

**Επίλυση προβλημάτων:** Αφορά στην εφαρμογή και ολοκλήρωση των τεχνικών και γνωστικών δεξιοτήτων με στόχο την επίλυση προβλημάτων. Έτσι γίνεται εισαγωγή των μαθητών στην υπολογιστική σκέψη και τον προγραμματισμό.

**Κοινωνικές δεξιότητες:** Στόχος είναι οι μαθητές και οι μαθήτριες να αναπτύξουν εκείνεις τις κοινωνικές στάσεις που διαμορφώνουν την σύγχρονη ψηφιακή κουλτούρα. Η διάσταση αυτή αφορά ζητήματα ηλεκτρονικής ασφάλειας, προστασίας προσωπικών δεδομένων, πληροφορικής ηθικής και δεοντολογίας και σε κώδικες διαχείρισης και αξιοποίησης πληροφοριών από ηλεκτρονικές πηγές.

Συμφώνα με το Ινστιτούτο Εκπαιδευτικής Πολιτικής, κεντρικός στόχος του μαθήματος είναι όλοι οι μαθητές και οι μαθήτριες να αναπτύξουν τις γνώσεις και τις ικανότητες χρήσης των ΤΠΕ μέσα από δραστηριότητες που αφορούν στην αναζήτηση και διαχείριση πληροφοριών, στην επίλυση προβλημάτων στη δημιουργική έκφραση και στην επικοινωνία. Εξίσου σημαντικό είναι μέσα από κατάλληλες χρήσεις και δραστηριότητες βασισμένες σε ΤΠΕ, οι μαθητές να κατανοήσουν τα όρια και την επίδραση που έχουν οι σύγχρονες τεχνολογίες στα άτομα και στις ομάδες, στις κοινότητες και την κοινωνία ευρύτερα.

Οι δραστηριότητες που υλοποιούν οι μαθητές και οι μαθήτριες στον υπολογιστή θα πρέπει να είναι κλιμακούμενης δυσκολίας και να στοχεύουν στην συνδυασμένη ανάπτυξη τεχνικών, γνωστικών και κοινωνικών ικανοτήτων με στόχο την επίλυση προβλημάτων από το σχολικό πρόγραμμα σπουδών και την κοινωνική ζωή των μαθητών. Κατά συνέπεια ενώ

οι τεχνικές και οι γνωστικές δεξιότητες συνιστούν, κατ' αρχήν, διακριτές περιοχές γνώσεων και δεξιοτήτων, ο συνδυασμός και η ενσωμάτωσή τους σε ένα νοηματοδοτούμενο πλαίσιο δραστηριοτήτων επίλυσης προβλημάτων διαμορφώνουν προϋποθέσεις πολύπλευρης ανάπτυξης των μαθητών στις ΤΠΕ και στο πεδίο του Πληροφορικού Γραμματισμού.

Το Ινστιτούτο Εκπαιδευτικής Πολιτικής επισημαίνει ότι θεματικές ενότητες της διδασκαλίας του γνωστικού αντικείμενου έχουν σχεδιαστεί με τρόπο ώστε οι μαθησιακοί στόχοι να ανταποκρίνονται στον βαθμό ετοιμότητας και ανάπτυξης των μαθητών αλλά και στο επίπεδο της κάθε τάξης. Τονίζει επίσης το ΙΕΠ ότι δεν είναι απαραίτητο να διδαχθούν σειριακά και δίνει μια ενδεικτική προτεινόμενη κατανομή χρόνου. Ο εκπαιδευτικός θα πρέπει να κάνει τον χρονοπρογραμματισμό και τον σχεδιασμό της διάρθρωσης της ύλης με βάση τα μαθησιακά χαρακτηριστικά, τα ενδιαφέροντα και τις προϋπάρχουσες γνώσεις και εμπειρίες των μαθητών της εκάστοτε τάξης. Επίσης, σύμφωνα με το ΙΕΠ, θα πρέπει να χρησιμοποιεί τη σπειροειδή προσέγγιση και να επανέρχεται με κάθε ευκαιρία και ειδικά στα πλαίσια της υλοποίησης σχεδίων έρευνας/εργασίας, σε βασικές ενότητες που απαιτούν την χρήση ποικίλων εργαλείων λογισμικού.

Τέλος οι οδηγίες του ΙΕΠ καταλήγουν πως η διδακτική πορεία θα πρέπει να αξιοποιεί την έμφυτη περιέργεια και την αυτενέργεια των μαθητών και μαθητριών. Να συνδυάζει την θεωρία και την πράξη μέσα από μια ενιαία, συνεχή και δημιουργική διαδικασία η οποία θα ενθαρρύνει και θα βοηθά τους μαθητές να συμμετέχουν ενεργά να συνεργάζονται μεταξύ τους να αναπτύσσουν πρωτοβουλίες να ανακαλύπτουν τη νέα γνώση να εκφράζονται και να δημιουργούν. Ο εκπαιδευτικός είναι καθοδηγητής και συντονιστής των μαθησιακών δραστηριοτήτων των μαθητών, μεσολαβητής, συνεργάτης και σύμβουλος των μαθητών στην πορεία της ανακάλυψης της δημιουργίας της καλλιέργειας δεξιοτήτων της ανάπτυξης ικανοτήτων και τελικά της οικοδόμησης νέων γνώσεων.

Η ενδεικτική κατανομή των διδακτικών ωρών των αξόνων μαθησιακών στόχων ανά ενότητα και τάξη που ισχύει μέχρι και το διδακτικό έτος 2020-2021 φαίνονται στον παρακάτω πίνακα.

Άξονες μαθησιακών στόχων	Προτεινόμενες ώρες διδασκαλίας					
	Α'	Β'	Γ'	Δ'	Ε'	Στ'
<b>Γνωρίζω, δημιουργώ και εκφράζομαι με ΤΠΕ</b>	<b>12</b>	<b>12</b>	<b>10</b>	<b>10</b>	<b>6</b>	<b>6</b>
<ul style="list-style-type: none"> <li>Γνωρίζω και χειρίζομαι τον υπολογιστή</li> </ul>	4	4	2	2		
<ul style="list-style-type: none"> <li>Δημιουργώ και εκφράζομαι με τη ζωγραφική, τα πολυμέσα και τις παρουσιάσεις</li> </ul>	4	4	4	4	3	3
<ul style="list-style-type: none"> <li>Δημιουργώ με τον κειμενογράφο</li> </ul>	4	4	4	4	3	3
<b>Επικοινωνώ και συνεργάζομαι με ΤΠΕ</b>	<b>6</b>	<b>6</b>	<b>6</b>	<b>6</b>	<b>6</b>	<b>6</b>
<ul style="list-style-type: none"> <li>Γνωρίζω το Διαδίκτυο</li> </ul>	3	3	3	3	3	3
<ul style="list-style-type: none"> <li>Επικοινωνώ και συνεργάζομαι</li> </ul>	3	3	3	3	3	3
<b>Διερευνώ, ανακαλύπτω και λύνω προβλήματα με ΤΠΕ</b>	<b>10</b>	<b>10</b>	<b>12</b>	<b>12</b>	<b>16</b>	<b>16</b>
<ul style="list-style-type: none"> <li>Μοντελοποιώ με εννοιολογικούς χάρτες</li> </ul>	4	4	4	4		
<ul style="list-style-type: none"> <li>Λύνω προβλήματα με Υπολογιστικά Φύλλα</li> </ul>					4	4
<ul style="list-style-type: none"> <li>Προγραμματίζω τον υπολογιστή</li> </ul>					4	4
<ul style="list-style-type: none"> <li>Υλοποιώ σχέδια εργασίας/έρευνας (project)</li> </ul>	6	6	8	8	8	8
<b>Οι ΤΠΕ ως κοινωνικό φαινόμενο</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>2</b>
<ul style="list-style-type: none"> <li>Οικοδομώ ψηφιακή παιδεία και γραμματισμό</li> </ul>	2	2	2	2	2	2
<b>Σύνολο διδακτικών ωρών</b>	<b>30</b>	<b>30</b>	<b>30</b>	<b>30</b>	<b>30</b>	<b>30</b>

Πίνακας 1. Η δομή του μαθήματος των ΤΠΕ στο δημοτικό

Ωστόσο, και μετά την υιοθέτηση του τεχνοκεντρικού προτύπου και παρά τις κατευθύνσεις του ΙΕΠ, παραμένουν κάποια θεμελιώδη προβλήματα για την διείσδυση της πληροφορικής στο δημοτικό σχολείο. Τα σημαντικότερα από αυτά είναι: **Απουσία οργανικών θέσεων εκπαιδευτικών:** Στην πρωτοβάθμια εκπαίδευση έχουν ιδρυθεί ελάχιστες οργανικές θέσεις συγκρίνοντας με τις λειτουργικές ανάγκες των σχολείων. Έτσι, οι θέσεις καλύπτονται μόνο από αναπληρωτές εκπαιδευτικούς κάτι που έχει σαν αποτέλεσμα να χάνεται κάθε έννοια διδακτικής και παιδαγωγικής συνέχειας από το ένα σχολικό έτος στο

άλλο. Επιπροσθέτως, η μετακίνηση των εκπαιδευτικών πληροφορικής σε παραπάνω του ενός σχολείου ακόμα και μέσα στην ίδια μέρα δυσχεραίνει το έργο και την προετοιμασία των μαθημάτων και του εξοπλισμού.

**Απαρχαιωμένος εξοπλισμός:** Η πλειονότητα των εργαστηρίων πληροφορικής δεν έχει ανανεωθεί με συνέπεια να υπάρχουν εργαστήρια βασισμένα σε υλικό και λογισμικό της προηγούμενης δεκαετίας. Σε αρκετές περιπτώσεις δεν υπάρχουν καν σταθερά εργαστήρια πληροφορικής και ζητείται από τον εκπαιδευτικό η καθημερινή μεταφορά φορητών υπολογιστών σε αίθουσες άλλων χρήσεων. Αρκετές φορές ακόμα και μέσα στο ίδιο εργαστήριο πληροφορικής παρατηρείται ανομοιομορφία μεταξύ του υλικού και του λογισμικού των υπολογιστών με συνέπεια την δυσκολία παροχής σωστού εκπαιδευτικού έργου.

**Λίγες ώρες διδασκαλίας:** Όπως αναφέρθηκε παραπάνω το μάθημα των ΤΠΕ διδάσκεται μια από μια ώρα εβδομαδιαίως σε κάθε τάξη. Είναι χαρακτηριστικό ότι μαθήματα όπως τα Θρησκευτικά έχουν περισσότερες ώρες διδασκαλίας για σε Γ και Δ τάξη και ακριβώς τις ίδιες σε Ε και ΣΤ. Γίνεται λοιπόν εύκολα αντιληπτό ότι απαιτείται εκσυγχρονισμός του προγράμματος σπουδών ώστε αυτό να συμβαδίζει με τις απαιτήσεις της εποχής. Μια πρώτη προσπάθεια προς αυτή την κατεύθυνση γίνεται με την εισαγωγή της ρομποτικής ως ξεχωριστή ενότητα στο νέο μάθημα «Εργαστήρια Δεξιοτήτων» αφενός όμως εναπόκειται στον διδάσκοντα το αν θα διδαχτεί και αφετέρου το υπουργείο ανέθεσε την διδασκαλία της στους δασκάλους και όχι στους εκπαιδευτικούς πληροφορικής.

## **1.2 ΤΠΕ και Επιστήμη των Υπολογιστών**

Όπως αναφέρθηκε στην προηγούμενη ενότητα από το 2010 το μάθημα η Πληροφορική διδάσκεται στο δημοτικό σχολείο ως Τεχνολογίες της Πληροφορίας και των Επικοινωνιών(ΤΠΕ). Είναι σημαντικό, ωστόσο, να επισημανθούν οι σχέσεις που συνδέουν τις ΤΠΕ(Information and Communication Technology, ICT) με την Επιστήμη των Υπολογιστών(Computer Science, CS) και οι διαφορές μεταξύ τους. Στην διεθνή βιβλιογραφία διεξάγεται μεγάλη συζήτηση για την σχέση αυτή. Οι ΤΠΕ και η Επιστήμη των Υπολογιστών έχουν κοινά σημεία αλλά δεν υποκαθιστά η μία την άλλη. Η Ε.Υ ασχολείται με θέματα μερικά εκ των οποίων επικαλύπτονται με εκείνα των ΤΠΕ ενώ κάποια άλλα δεν έχουν καμία σχέση. Για παράδειγμα η πολυπλοκότητα των αλγορίθμων είναι θεμελιώδης έννοια της Ε.Υ αλλά είναι πολύ πιθανό να μην υπάρχει στο αναλυτικό πρόγραμμα των ΤΠΕ.

Οι ΤΠΕ είναι ένα εφαρμοσμένο πεδίο επιστήμης που εκμεταλλεύεται τα πρακτικά πλεονεκτήματα που προσφέρει η Ε.Υ. ωστόσο περιλαμβάνει ορισμένες μαθηματικές επιστημονικές και πρακτικές διαστάσεις (Ψυχάρης & Καλοβρέκτης, Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ, 2018). Στο πεδίο της εκπαίδευσης, αυτή η σχέση θα μπορούσε να περιγράψει ως ότι η Ε.Υ διδάσκει έναν μαθητή πως θα είναι αποτελεσματικός συγγραφέας δημιουργός υπολογιστικών εργαλείων ενώ οι ΤΠΕ «διδάσκουν» με ποιον τρόπο ο μαθητής θα είναι ικανός και σκεπτόμενος χρήστης αυτών των εργαλείων. Η διάκριση αυτή βέβαια εστιάζει στους ΗΥ αποκλειστικά ως «τεχνολογία».

Συμπερασματικά, σύμφωνα με τους Ψυχάρη και Καλοβρέκτη(2018) η Ε.Υ είναι μια γνωστική περιοχή που αναζητά την κατανόηση του κόσμου με χρήση υπολογιστικών όρων και υπολογιστικών διεργασιών. Συνδέεται με την μελέτη, την ανάλυση, τον σχεδιασμό την υλοποίηση των συστημάτων των υπολογιστών και των αρχών που τις διέπουν. Από την άλλη πλευρά, οι ΤΠΕ ασχολούνται με την «πρακτική» εφαρμογή των συστημάτων υπολογιστών την εγκατάσταση και των προσδιορισμό του υλικού και λογισμικού που χρειάζονται.

Ενδεχόμενη εισαγωγή της Ε.Υ στην εκπαίδευση θα πρέπει να περιλαμβάνει

- τους αλγόριθμους, δηλαδή επαναχρησιμοποιούμενες διαδικασίες με την μορφή αναλυτικών βημάτων
- τις δομές δεδομένων, δηλαδή τους τρόπους για να οργανωθούν τα δεδομένα ώστε να χρησιμοποιηθούν κατάλληλα σε μια εφαρμογή
- τα προγράμματα τα οποία καθοδηγούν τον ΗΥ για τις ενέργειες που πρέπει να κάνει. Κάθε πρόγραμμα είναι γραμμένο σε μια γλώσσα προγραμματισμού με διαφορετικές δυνατότητες και πρέπει να επιλέγεται κάθε φορά η κατάλληλη.
- Την αρχιτεκτονική η οποία περιγράφει την δομή των υπολογιστικών συστημάτων.
- Την επικοινωνία, δηλαδή τις συλλογές των υποσυστημάτων τα προγράμματα που αυτά τρέχουν για να επικοινωνούν με τα υπόλοιπα και τους τρόπους διαμοιρασμού της μνήμης.

### 1.3 Θεωρίες μάθησης και διδακτικά μοντέλα

Η πληροφορική και κατ' επέκταση το STEM μπορεί να αποτελούν σχετικά νέα και συνεχώς εξελισσόμενα αντικείμενα αλλά για τους εκπαιδευόμενους δεν παύουν να είναι ένα αντικείμενο μάθησης.

Για τον όρο μάθηση έχουν δοθεί πολλοί ορισμοί στην βιβλιογραφία. Σύμφωνα με τον H. Roth μάθηση είναι η απόκτηση νέων μορφών συμπεριφοράς και ικανοτήτων. Κατά τον R. Gagne μάθηση είναι η διαδικασία που υποβοηθάει τους οργανισμούς να τροποποιήσουν την συμπεριφορά τους σε έναν σχετικά σύντομο χρονικό διάστημα και με έναν μόνιμο τρόπο, ώστε η ίδια τροποποίηση ή αλλαγή να μην περιλαμβάνεται σε κάθε νέα περίπτωση. Σύμφωνα με την γνωστική ψυχολογία η μάθηση είναι μια διαδικασία η οποία πραγματοποιείται στον εσωτερικό κόσμο του μαθητή και έχει σαν αποτέλεσμα την τροποποίηση της συμπεριφοράς του. Η μάθηση εξαρτάται από τις προηγούμενες γνώσεις, το υπόβαθρο, την ιδιοσυγκρασία και τις εμπειρίες του κάθε μαθητή (Γκίνης, 2005).

Η μάθηση συνδέεται με την «γνώση» με βάση ένα γνωστικό αντικείμενο ή μια γνωστική περιοχή. Μια **θεωρία μάθησης** είναι ένα σύνολο από ισχυρισμούς και αξιώματα που επιχειρεί να περιγράψει και να ερμηνεύσει τους μηχανισμούς της μάθησης με ένα συνεπή και συνεκτικό τρόπο (Ψυχάρης & Καλοβρέκτης, 2018). Ασφαλώς δεν υπάρχει μια και μοναδική θεωρία μάθησης ικανή να απαντήσει σε όλα τα ερωτήματα αλλά περισσότερες οι οποίες επικαλύπτονται.

Οι θεωρίες μάθησης μπορούν να χωριστούν σε τέσσερις βασικές ομάδες (Σμυρναίου, 2014):

- τις συμπεριφοριστικές θεωρίες
- τις γνωστικές θεωρίες
- τις εποικοδομιστικές θεωρίες
- τις κοινωνικοπολιτισμικές θεωρίες μάθησης

Ο όρος **συμπεριφορισμός** βασίζεται στην θεώρηση ότι η γνώση είναι η κατανόηση της πραγματικότητας που υπάρχει ανεξάρτητα από κάθε υποκείμενο μάθησης (Στυλιάρης &



Δήμου, 2015). Σύμφωνα με την Σμυρναίου (2014) οι συμπεριφοριστικές θεωρίες μάθησης δίνουν έμφαση στην αναμετάδοση της πληροφορίας και στην τροποποίηση της συμπεριφοράς διερευνώντας και αναζητώντας σχέσεις μεταξύ εξωτερικών ερεθισμάτων και της παρατηρούμενης συμπεριφοράς. Πρωτεργάτες της συγκεκριμένης σχολής ήταν οι Watson (1913) και Pavlov (1927) οι οποίοι απέδειξαν στην πράξη την σχέση μεταξύ φυσικού ερεθίσματος και απόκρισης του υποκειμένου. Το συγκεκριμένο μοντέλο μάθησης συναντάται στην βιβλιογραφία ως κλασσική εξαρτημένη μάθηση.

Στην συνέχεια ο Skinner (1958) ανέπτυξε την θεωρία της συντελεστικής μάθησης η οποία έδωσε μεγάλη βαρύτητα στην ενίσχυση – επιβράβευση με βάση τα γεγονότα που προηγήθηκαν. Με βάση αυτή θεωρείται πως η συμπεριφορά που ακολουθείται από μια θετική ενίσχυση ή επιβράβευση έχει περισσότερες πιθανότητες να επαναληφθεί. Η συγκεκριμένη θεωρία για να εφαρμοστεί απαιτεί ενεργή συμμετοχή του μαθητή, δόμηση της διδακτέας ύλης σε σύντομες ενότητες, την προσαρμογή στους ρυθμούς του μαθητή και την άμεση επαλήθευση και ενίσχυση της ενδεχόμενης σωστής απάντησης (Σμυρναίου, 2014). Ένας ακόμα κύριος εκπρόσωπος της συμπεριφοριστικής θεωρίας, ο Thorndike (1992) σημειώνει πως το η μάθηση είναι διαρκής διαδικασία με δοκιμές και λάθη και υπάρχει σύνδεση ανάμεσα στο ερέθισμα και την αντίδραση.

Οι **γνωστικές** θεωρίες μάθησης εστιάζουν το ενδιαφέρον τους στο εσωτερικό του γνωστικού συστήματος στην δομή και την λειτουργία του (Σμυρναίου, 2014). Εστιάζουν στον ανθρώπινο νου και στον τρόπο με τον οποίο εκείνος αντιλαμβάνεται τις σχέσεις των πραγμάτων σκέφτεται και αντιδρά στις διάφορες καταστάσεις (Ψυχάρης & Καλοβρέκτης, 2018). Σύμφωνα με τους ίδιους, οι γνωστικές διεργασίες θεωρούνται οι ανώτερου επιπέδου νοητικές λειτουργίες που συμβαίνουν στον ανθρώπινο εγκέφαλο όπως αντίληψη, προσοχή, λογική, αιτιολόγηση και επίλυση προβλήματος.

Στις **εποικοδομιστικές** θεωρίες μάθησης η γνώση δεν μεταδίδεται αποκλειστικά αλλά είναι και προσωπική υπόθεση του εκπαιδευόμενου και βασίζεται σε προγενέστερες γνώσεις οι οποίες υφίστανται μετασχηματισμούς ώστε να συνδεθούν με τη νέα γνώση. Σημαντικότεροι εκπρόσωποι της συγκεκριμένης σχολής αποτελούν οι Piaget, Bruner, και Papert.

Ο Piaget, με τον **γνωστικό εποικοδομητισμό**, προσέγγισε την μάθηση όχι μόνο ως ψυχολογικό φαινόμενο αλλά κυρίως ως παιδαγωγικό και έδωσε βαρύτητα στην ανάπτυξη και εξέλιξη της λογικής σκέψης του παιδιού (Κομής, 2004). Αυτή, σύμφωνα με τον Piaget χωρίζεται σε τέσσερα στάδια: Το αισθησιοκινητικό στάδιο μέχρι 2 ετών, το στάδιο της προλογικής σκέψης από 2 έως 7 ετών, το στάδιο της συγκεκριμένης σκέψης από 7 έως 12 ετών και το στάδιο της τυπικής αφαιρετικής σκέψης από 12 ετών και άνω. Παρά το διαφορετικό επίπεδο γνωστικής ανάπτυξης, κοινό χαρακτηριστικό για κάθε ένα από τα παραπάνω στάδια είναι ότι το άτομο οικοδομεί την γνώση βάσει της αλληλεπίδρασής του με τον κόσμο.

Ο Bruner (1971) υποστήριξε την **μάθηση μέσω ανακάλυψης** όπου δίνεται έμφαση στην διευκόλυνση της μάθησης μέσω της κατανόησης των δομών και των επιστημονικών αρχών ενός αντικειμένου. Προτείνει την καθοδηγούμενη ανακάλυψη με την ανάπτυξη εσωτερικών κινήτρων μάθησης από τον μαθητή (Σμυρναίου, 2014). Σύμφωνα με τον Bruner ο μαθητής πρέπει να έρχεται αντιμέτωπος με προβληματικές καταστάσεις και ο εκπαιδευτικός να έχει ρόλο διευκολυντή, εμπνευστή και συντονιστή σε όλη την διαδικασία.

Συνδυαστικός κρίκος ανάμεσα στις παραδοσιακές θεωρίες μάθησης και τις σύγχρονες εκπαιδευτικές τάσεις όπως π.χ το STEM αποτέλεσε η **θεωρία κατασκευής της γνώσης** του Seymour Papert. Γνωστή και ως κονστρουκτιβισμός η θεωρία υποστηρίζει πως οι μαθητές κατασκευάζουν την γνώση στα πλαίσια μια αυθεντικής δραστηριότητας αποκτώντας έτσι νόημα για αυτούς (Ψυχάρης & Καλοβρέκτης, 2018). Για τον Papert ο κονστρουκτιβισμός δεν είναι μόνο θεωρία αλλά και εκπαιδευτική στρατηγική η οποία μπορεί να αξιοποιηθεί σε έναν πραγματικό εκπαιδευτικό σχεδιασμό μαζί με την υπολογιστική σκέψη. Σε τέτοια εκπαιδευτικά περιβάλλοντα οι μαθητές θα χειρίζονται, επεξεργάζονται ή θα παίζουν με αντικείμενα μαθαίνοντας έτσι πέρα από τον παραδοσιακό τρόπο μιας σχολικής τάξης (Papert, 1993). Η θεωρία του Papert δίνει μεγαλύτερη βαρύτητα στην κατασκευή – δημιουργία – πράξη και συνδέεται λειτουργικά με τις διαστάσεις της υπολογιστικής σκέψης είτε υλοποιούνται σε ηλεκτρονικό υπολογιστή είτε όχι (Ψυχάρης & Καλοβρέκτης, Διδακτική & Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM & ΤΠΕ, 2018, σ. 16)

## Κεφάλαιο 2

### 2.1 Η Υπολογιστική Σκέψη

Στην πρόταση της CSTA-Computer Science Standards για την Επιστήμη των Υπολογιστών στην Πρωτοβάθμια εκπαίδευση ([https://www.csteachers.org/page/CSTA\\_Standards](https://www.csteachers.org/page/CSTA_Standards) 2016, n.d.) γίνεται εμφατικά αναφορά στην Υπολογιστική Σκέψη ως μέθοδος για την επίλυση προβλημάτων, για την δημιουργία νέας γνώσης και την κατανόηση των λειτουργιών των περιορισμών των υπολογιστικών συστημάτων.

Ο αρχικός ορισμός για την Υπολογιστική Σκέψη σύμφωνα με την (Wing, Computational thinking. Communication of the ACM, 2006) αναφέρει πως «η Υπολογιστική Σκέψη αφορά στην επίλυση προβλημάτων, τον σχεδιασμό συστημάτων και την κατανόηση της ανθρώπινης συμπεριφοράς χρησιμοποιώντας έννοιες που είναι θεμελιώδους σημασίας για την επιστήμη των υπολογιστών. Αναφέρει επίσης πως είναι μια βασική ικανότητα που πρέπει να έχουν οι εκπαιδευόμενοι συμπληρωματικά με τις άλλες τρεις βασικές δεξιότητες την ανάγνωση, την γραφή και την αριθμητική. Με μία νεότερη δημοσίευση η Wing ορίζει την Υπολογιστική Σκέψη ως εκείνες τις διαδικασίες σκέψης που εμπλέκονται στην διαμόρφωση προβλημάτων και τις λύσεις τους με τέτοιο τρόπο ώστε οι λύσεις να μπορούν να παρουσιαστούν αποτελεσματικά από έναν πράκτορα επεξεργασίας πληροφοριών (Wing, 2010). Προσθέτει πως η Υπολογιστική Σκέψη περιγράφει την πνευματική δραστηριότητα στην διαμόρφωση προβλημάτων με υπολογιστική λύση που μπορεί να πραγματοποιηθεί από έναν άνθρωπο ή μηχανή ή συνδυασμό τους.

Πολλοί ακόμη ερευνητές έναντι προσπάθειες να δώσουν έναν πληρέστερο ορισμό για την Υπολογιστική Σκέψη περιλαμβάνοντας σ αυτή πολλά επίπεδα αφαιρετικότητας, την χρήση μαθηματικών για την ανάπτυξη αλγορίθμων και εξετάζει πόσο καλά μια λύση μπορεί να ταιριάζει σε διαφορετικά προβλήματα. Σύμφωνα με τον Denning(2009) η Υπολογιστική Σκέψη αναφέρεται σε ένα σύνολο δεξιοτήτων γενικά εφαρμόσιμο που όλοι θα ήταν πρόθυμοι να μάθουν και να χρησιμοποιούν. Στηρίζεται στην δύναμη αλλά και τους περιορισμούς των διαδικασιών υπολογισμού είτε αυτοί εκτελούνται από τον άνθρωπο είτε από τον υπολογιστή.

Οι Barr&Stephenson(2011) θεωρούν πως η Υπολογιστική Σκέψη είναι μια προσέγγιση για την επίλυση προβλημάτων με τέτοιο τρόπο ώστε να μπορεί να υλοποιηθεί από υπολογιστή. Τονίζουν πως οι μαθητές δεν γίνονται απλά χρήστες κάποιων εργαλείων αλλά και κατασκευαστές αυτών. Χρησιμοποιούν μια σειρά από έννοιες όπως είναι η αφαίρεση, η αναδρομή και η επανάληψη για την επεξεργασία και ανάλυση των δεδομένων. Είναι μια προσέγγιση επίλυσης προβλημάτων η οποία δύναται να αυτοματοποιηθεί, να μεταφερθεί και να εφαρμοστεί σε διάφορα θέματα.

Ο Gudzial(2008) υποστηρίζει πως από την δεκαετία του 1960 ο AlanPairlis έβλεπε την υπολογιστική σκέψη ως ένα βήμα για την καντανόηση της θεωρίας υπολογισμών που θα οδηγούσε τους φοιτητές να αναδιατυπώσουν τι καταλαβαίνουν σ ένα ευρύ φάσμα μαθημάτων και πρότεινε την διδασκαλία του προγραμματισμού. Υποστηρίζει επίσης (Gudzial, 2012) πως πρέπει να εστιάσουμε στην μετατόπιση από τον εννοιολογικό προσδιορισμό της Υπολογιστικής Σκέψης στον τρόπο που αυτή θα ενταχθεί στην διδακτική και μαθησιακή διαδικασία.

Το πανεπιστήμιο του Harvard στις πρώτες διαλέξεις του μαθήματος της Επιστήμης των Υπολογιστών(Computer Science) συνδέει την Υπολογιστική Σκέψη με την επίλυση προβλημάτων όταν υπάρχουν κάποια δεδομένα και απαιτείται να καταλήξεις σε κάποιο αποτέλεσμα (Malan, 2018). Ακόμα και η Googleέχει προσπαθήσει να ορίσει άτυπα την Υπολογιστική Σκέψη συνδέοντάς τη με την διαδικασία επίλυσης προβλήματος που περιλαμβάνει μια σειρά από χαρακτηριστικά όπως η λογική οργάνωση, η ανάλυση δεδομένων, η δημιουργία λύσεων που χρησιμοποιούν αλγόριθμους, η αναγνώριση προτύπων, η διάσπαση του προβλήματος και οι διαθέσεις των εκπαιδευομένων.

Συνοψίζοντας τους ορισμούς της βιβλιογραφίας για την Υπολογιστική Σκέψη μπορούμε να πούμε ότι τα κοινά αποδεκτά στοιχεία είναι τα εξής (Ψυχάρης & Καλοβρέκτης, Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ, 2018):

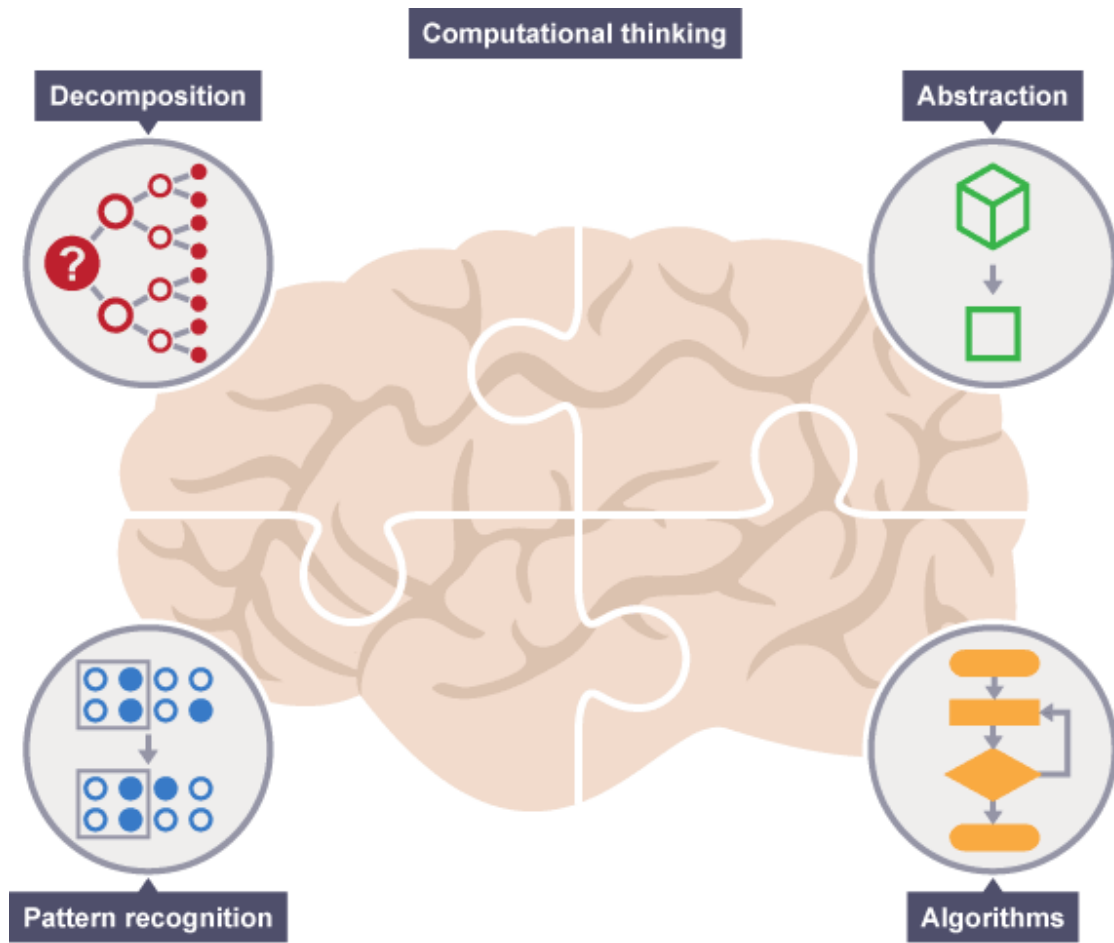
- **Η αφαιρετική σκέψη(Abstraction):** Είναι εκείνη η νοητική διαδικασία με την οποία κατανοούμε καλύτερα ένα τεχνολογικό κατασκεύασμα παραλείποντας τις μη αναγκαίες λεπτομέρειες. Η βασική δεξιότητα στην αφαιρετική σκέψη είναι η επιλογή στην λεπτομέρεια που πρέπει να παραλειφθεί ώστε το πρόβλημα να γίνει ευκολότερο

χωρίς όμως να χαθεί η ουσία του. Σημαντικό ρόλο στην αφαιρετική σκέψη παίζει η σωστή αναπαράσταση του προβλήματος καθώς διαφορετικές αναπαραστάσεις μπορεί να οδηγήσουν σε διαφορετικές μεθόδους επίλυσης με διαφορετική δυσκολία. Η αφαιρετική σκέψη και οι αφαιρετικές αναπαραστάσεις μπορούν να βοηθήσουν τους εκπαιδευόμενους να αναγνωρίζουν το μοντέλο ενός πραγματικού φαινομένου να ελέγχουν τις μεταβλητές και τις παραμέτρους του.

- **Η ολοκλήρωση της μαθηματικής σκέψης, της αλγοριθμικής σκέψης και τεχνικών από την επιστήμη των μηχανικών και την αριθμητική ανάλυση για παιδαγωγική αξιοποίηση:** Τα μαθηματικά παίζουν κυρίαρχο ρόλο σε όλες τις θετικές επιστήμες έτσι λοιπόν δεν γίνεται να μην χρησιμοποιούνται και στον υπολογιστικό τρόπο σκέψης ώστε να δημιουργούνται συστήματα που αντιστοιχούν σε εκείνα του πραγματικού κόσμου ή σε εικονικά συστήματα. Οι περιορισμοί των υπολογιστικών συσκευών ωστόσο οδηγούν σε διακριτικοποίηση των συστημάτων και σε χρήση στοχαστικών μοντέλων. Έτσι αρκετές φορές θα χρειαστούν αλγόριθμοι με προσεγγιστικές μεθόδους από την αριθμητική ανάλυση για την επίλυση προβλημάτων.
- **Οι έννοιες της Υπολογιστικής Σκέψης:** Ο υπολογιστικός τρόπος σκέψης δεν αφορά τόσο το λογισμικό και το υλικό αλλά κυρίως αφορά τις υπολογιστικές έννοιες που χρησιμοποιούμε για να λύσουμε προβλήματα και να επικοινωνήσουμε με άλλους ανθρώπους ενώ πολύ συχνά οι εκπαιδευόμενοι χρειάζεται να εμπλακούν στην υπολογιστική σκέψη χωρίς την χρήση τεχνολογικών συστημάτων για να λύσουν προβλήματα.
- **Η δομοστοιχείωση (Modularity):** Είναι η έννοια που περιγράφει την εκτέλεση συγκεκριμένων επαναληπτικών ενεργειών για μια συγκεκριμένη δράση χρησιμοποιώντας έτοιμες υπορουτίνες για διαφορετικά φαινόμενα.
- **Η τμηματοποίηση (Decomposition)** περιγράφει την διαδικασία διαχωρισμού του προβλήματος σε τμήματα που δεν μπορούν να τμηματοποιηθούν περαιτέρω τα οποία στην συνέχεια «ολοκληρώνονται»για να λυθεί το πρόβλημα.

- **Η αποσφαλμάτωση (Debugging)** αποτελεί μια βασική συνιστώσα της Υπολογιστικής Σκέψης και συνδέεται άμεσα με τον προγραμματισμό ως βασική διαδικασία της Υπολογιστικής Σκέψης. Η αποσφαλμάτωση στον προγραμματισμό σημαίνει τον εντοπισμό των λαθών σε έναν αλγόριθμο και εν συνεχεία την διόρθωση του. Κατά τον διδακτικό σχεδιασμό είναι πάντα χρήσιμο οι μαθητές να μπαίνουν στην διαδικασία της αποσφαλμάτωσης δίνοντας τους λάθος κώδικας ο οποίος θα πρέπει να συζητηθεί και θα διορθωθεί.
- **Ο εντοπισμός προτύπων-μοτίβων (Pattern recognition)** αφορά τον ορισμό προτύπων δηλαδή διαδικασιών ή φαινομένων που επαναλαμβάνονται, την αναγνώριση αυτών καθώς και το πώς αυτά θα χρησιμοποιηθούν κατάλληλα για την επίλυση κάποιου προβλήματος. Η συγκεκριμένη διαδικασία της Υπολογιστικής Σκέψης δεν έχει εφαρμογή μόνο στην Επιστήμη των Υπολογιστών και το STEM αλλά είναι χρήσιμη και για πτυχές της καθημερινής ζωής.
- **Η γενίκευση** περιγράφει τα μοτίβα και τις ομοιότητές αλλά και το πώς αυτά διασυνδέονται μεταξύ τους. Η γνώση ότι κάποιος αλγόριθμος μπορεί να λύσει παραπάνω από ένα πρόβλημα αποτελεί έννοια της γενίκευσης. Αποτελεί μια μέθοδο επίλυσης καινούριων προβλημάτων που βασίζεται στην εμπειρία από προηγούμενες μεθόδους και της αξιοποίηση τους.

Οπτικοποιώντας τις παραπάνω αναφορές η Υπολογιστική Σκέψη συνοψίζεται στον παρακάτω συνοπτικό οδηγό (Bitesize, n.d.):



Εικόνα 1. Η Υπολογιστή Σκέψη

Υιοθετώντας τις δεξιότητες της Υπολογιστικής Σκέψης οι μαθητές θα πρέπει να είναι σε θέση να εκπληρώνουν κάποιους στόχους όπως φαίνονται στον παρακάτω πίνακα (Atmatzidou & Demetriadis, 2016).

<u>Δεξιότητα</u>	<u>Οι μαθητές πρέπει να είναι σε θέση να...</u>
<b>Αφαίρεση</b>	<ul style="list-style-type: none"> <li>• Διαχωρίζουν την σημαντική από την περιττή πληροφορία</li> </ul>

	<ul style="list-style-type: none"> <li>• Εντοπίζουν κοινές συμπεριφορές ή προγραμματιστικές δομές μεταξύ διαφορετικών σεναρίων</li> </ul>
<b>Γενίκευση</b>	<ul style="list-style-type: none"> <li>• Επεκτείνουν μια υφιστάμενη λύση ενός ήδη δοθέντος προβλήματος για την κάλυψη περισσότερων περιπτώσεων</li> </ul>
<b>Αλγοριθμική Σκέψη</b>	<ul style="list-style-type: none"> <li>• Διατυπώνουν σαφώς τα βήματα ενός αλγορίθμου</li> <li>• Αναγνωρίζουν διαφορετικούς αποδοτικούς αλγόριθμους για ένα συγκεκριμένο πρόβλημα</li> <li>• Βρίσκουν τον πιο αποδοτικό αλγόριθμο</li> </ul>
<b>Επεκτασιμότητα</b>	<ul style="list-style-type: none"> <li>• Αναπτύσσουν αυτόνομα κομμάτια κώδικα για χρήση στο ίδιο ή σε διαφορετικό πρόβλημα</li> </ul>
<b>Ανάλυση</b>	<ul style="list-style-type: none"> <li>• Να τμηματοποιούν ένα μεγαλύτερο πρόβλημα σε μικρότερα τα οποία μπορούν να διαχειριστούν ευκολότερα</li> </ul>

## 2.2 Από την Υπολογιστική σκέψη στον προγραμματισμό

Από τις αναφορές του κεφαλαίου 1 προκύπτει πως η Υπολογιστική Σκέψη είναι ένα σύνολο δεξιοτήτων ή μια μέθοδος επίλυσης προβλημάτων. Το τελευταίο και σύμφωνα με τα

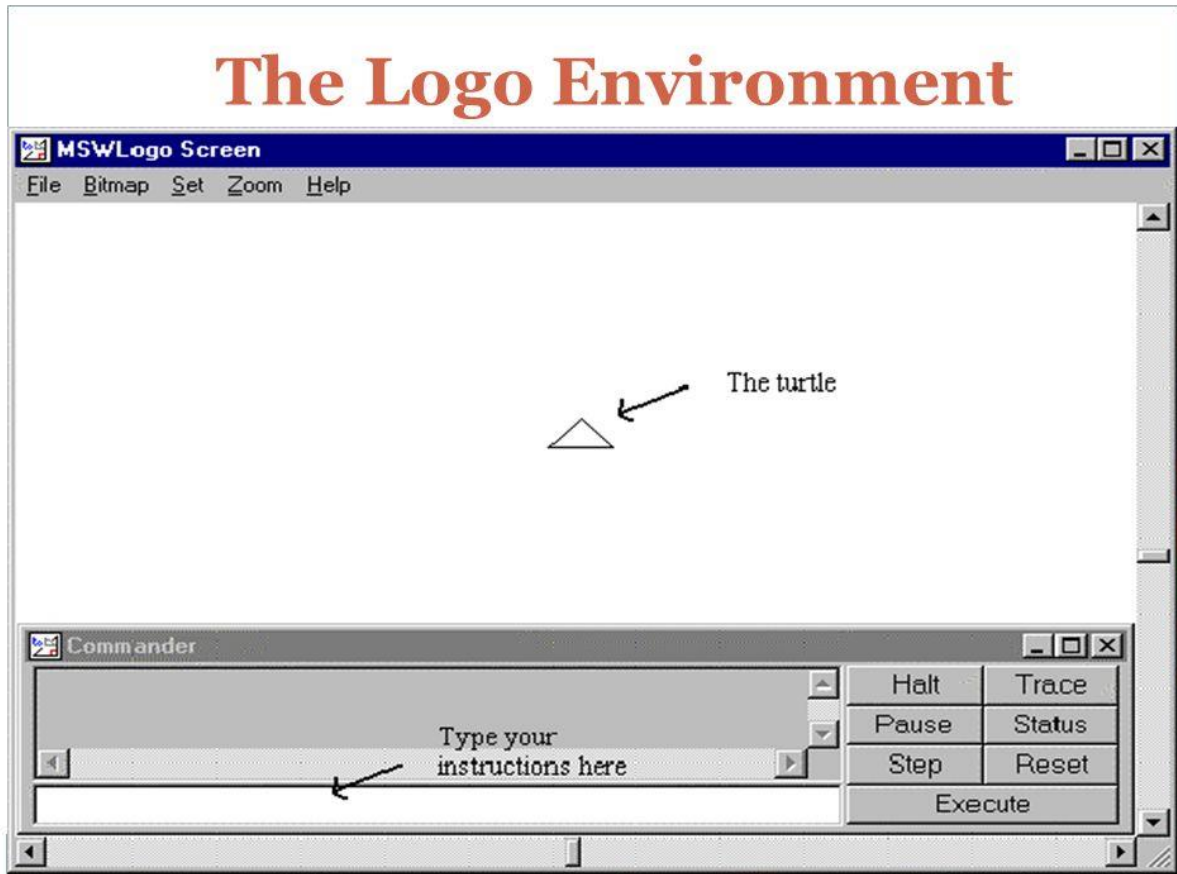


παραδείγματα ερευνών (Google, 2020) σχετίζεται άμεσα με τον προγραμματισμό αφού αυτός αποτελεί την καλύτερη μέθοδο επίλυσης προβλημάτων και υλοποίηση στην πράξη της Υπολογιστικής Σκέψης. Μέσω του προγραμματισμού οι εκπαιδευόμενοι(μαθητές ή φοιτητές) θα χρησιμοποιήσουν τους υπολογιστές ευνοώντας την δημιουργικότητα ενώ ταυτόχρονα γίνονται επιστημολογικοί μετασχηματισμοί ώστε ο εκπαιδευόμενος να αισθάνεται ότι δημιουργεί ο ίδιος την γνώση (Ψυχάρης & Καλοβρέκτης, Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ, 2018).

Η εισαγωγή του προγραμματισμού στην εκπαίδευση ως υλοποίηση στην πράξη της υπολογιστικής σκέψης δεν είναι μια νέα ιδέα. Πρώτος ο Seymour Papert(1991) υποστήριξε την διδασκαλία του προγραμματισμού σε παιδιά μέσα από την χρήση της γλώσσας προγραμματισμού LOGO η οποία είχε αναπτυχθεί από τον ίδιο και τον Wally Feurzeig ήδη από την δεκαετία του 1960. Ο Papert υποστηρίζει πως η εμπλοκή των παιδιών σε μια γλώσσα προγραμματισμού θα επηρεάσει γενικά τον τρόπο με τον οποίο τα παιδιά μαθαίνουν και να έχει θετική επίδραση στην δόμηση της γνώσης.

Το όνομα της γλώσσας προγραμματισμού LOGO προέρχεται από την ελληνική λέξη λόγος παραπέμποντας στις έννοιες λογική και υπολογισμός και ένας από τους κύριους σκοπούς δημιουργίας της ήταν να κάνει τον προγραμματισμό ηλεκτρονικού υπολογιστή προσιτό στους φοιτητές. Η LOGO αποτελείται από ένα απλό σύνολο εντολών με την μορφή λέξεων(π.χ μπροστά, πίσω) η οποίες μπορούν εύκολα να ταυτιστούν με την διαισθητική γνώση από την κίνηση του ανθρώπου. Έτσι οι μαθητές χρησιμοποιώντας αυτές τις απλές λέξεις «διδάσκουν» τον υπολογιστή και δημιουργούν προγράμματα. Στην εξέλιξη της η LOGO συνδέθηκε με το «γραφικό της χελώνας» (Papert, Νοητικές θύελλες: Παιδιά, ηλεκτρικοί υπολογιστές και δυναμικές ιδέες Μετάφραση Αίγλη Σταματίου, 1991) και χρησιμοποιήθηκε ως εργαλείο αναπαράστασης και έκφρασης μαθηματικών εννοιών. Ο όρος «γραφικό της χελώνας» χρησιμοποιείται για να περιγράψει μια οντότητα η οποία κινείται σε δύο διαστάσεις σύμφωνα με τις εντολές που δέχεται από τους εκπαιδευόμενους και χαρακτηρίζεται κάθε στιγμή από την θέση και την κατεύθυνση που έχει. Με την LOGO ο Papert επεδίωξε να συνδέσει την διδακτική των μαθηματικών με τον προγραμματισμό και την μαθηματική με την υπολογιστική και αλγοριθμική σκέψη.

# The Logo Environment



Εικόνα 2. Ηγλώσσα προγραμματισμού Logo

Στις μέρες μας ο προγραμματισμός έχει μπει στα αναλυτικά προγράμματα των περισσότερων «ανεπτυγμένων» χωρών. Ο προγραμματισμός θεωρείται πλέον μια παιδαγωγική τεχνική που αυξάνει τα κίνητρα για εμπλοκή στον σχεδιασμό τεχνολογικών έργων και φυσικών διαδικασιών και έχουν δημιουργηθεί πάρα πολλά εργαλεία και πρωτοβουλίες για να υλοποιήσουν αυτή την αρχή (Ψυχάρης & Καλοβρέκτης, Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ, 2018). Στις ΗΠΑ υπάρχει η «ώρα του κώδικα» και στην Μεγάλη Βρετανία το computing curricula όπου οι μαθητές αναγνωρίζουν και δημιουργούν αλγόριθμους και στην συνέχεια τους υλοποιούν στους Η/Υ. Τα τελευταία χρόνια η «ώρα του κώδικα» υλοποιείται και σε αρκετά ελληνικά σχολεία χωρίς ωστόσο κεντρικό σχεδιασμό αλλά με πρωτοβουλία των εκπαιδευτικών πληροφορικής.

Για τον σκοπό της εισαγωγής στις αρχές επίλυσης προβλημάτων και της διδακτικής του προγραμματισμού σε παιδιά έχουν αναπτυχθεί τα τελευταία χρόνια πολλές εφαρμογές. Ο σκοπός των εφαρμογών αυτών δεν είναι η εκμάθηση κώδικα με την μορφή κάποιας

συγκεκριμένης γλώσσας προγραμματισμού αλλά η εκμάθηση των εννοιών και του τρόπου σκέψης που πρέπει να έχουν στην διαδικασία του προγραμματισμού. (Hutchison, Nadolny, & Estara, 2015). Τελικός στόχος τέτοιων εφαρμογών είναι να μετατρέψει τους μαθητές από «καταναλωτές» της τεχνολογίας σε «παραγωγούς» της εισάγοντάς τους από μικρή ηλικία στην επιστήμη της πληροφορικής στις έννοιες του προγραμματισμού και στην ανάπτυξη της υπολογιστικής σκέψης. (Gedik, Cetin, & Koca, 2017).

Κάποιες από τις πιο δημοφιλείς εφαρμογές που χρησιμοποιούνται για την εισαγωγή των μαθητών στην έννοια του προγραμματισμού είναι οι παρακάτω (Τρακοςας, 2019):

- Scratch(<https://scratch.mit.edu>)

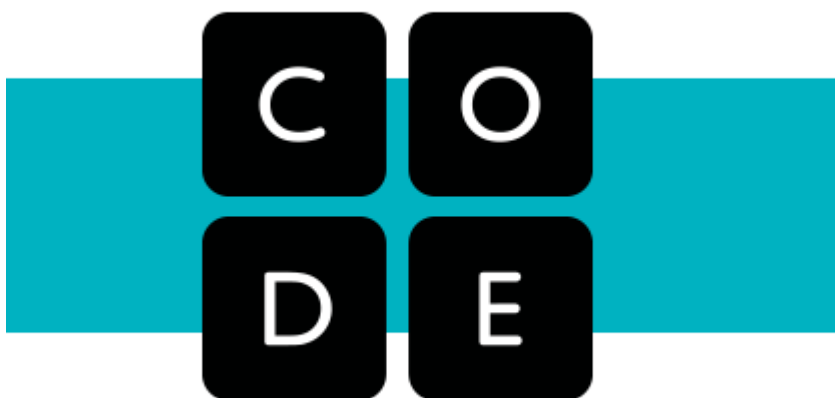


Το Scratch είναι ένα προγραμματιστικό περιβάλλον ανοιχτού κώδικα για αρχάριους που κυκλοφόρησε για πρώτη φορά το 2007 και βρίσκεται πλέον στην τρίτη του έκδοση. Χρησιμοποιεί έναν συνδυασμό πολυμέσων όπως έτοιμα γραφικά, ήχο και βίντεο για να κάνει ελκυστική προς τους χρήστες την δημιουργία διαδραστικών ιστοριών και παιχνιδιών. Επίσης εκτός των έτοιμων γραφικών ο χρήστης μπορεί να εισάγει τα δικά του μέσω της ζωγραφικής ή της εισαγωγής από εξωτερικές πηγές.

Στο Scratch για κάθε ένα από τα αντικείμενα που εισάγει ή δημιουργεί ο χρήστης υπάρχει ένα ή περισσότερα σενάρια. Μέσω αυτών ο χρήστης δίνει στα αντικείμενα τον τρόπο με τον οποίο επιθυμεί να αλληλεπιδρούν είτε με εκείνον, είτε με τα υπόλοιπα αντικείμενα είτε με τον εαυτό τους. Τα σενάρια δημιουργούνται συνδυάζοντας προγραμματιστικά block (block programming) τα οποία είναι χωρισμένα σε κατηγορίες ανάλογα με την λειτουργία τους. Τα πλακίδια(blocks) είναι με τέτοιο τρόπο σχεδιασμένα ώστε η σύνδεση τους να είναι δυνατή μόνο όταν υπάρχει λογική σύνταξη απλουστεύοντας με αυτό τον τρόπο την ανάπτυξη εφαρμογών για αρχάριους χρήστες, όπως οι μαθητές.

Το Scratch προσφέρει στον εκπαιδευτικό μια σειρά από σημαντικά πλεονεκτήματα για την μετάδοση βασικών προγραμματιστικών εννοιών στους μαθητές όπως είναι οι δομές ελέγχου (if, elseif, else) τις μεταβλητές και τους βρόγχους. Το Scratch ενσωματώνει στην πράξη πολύ καλά δύο μεθόδους διδασκαλίας: α) την γνωστική όπου διδάσκει τους μαθητές τις έννοιες του προγραμματισμού και της υπολογιστικής σκέψης και β) την συναισθηματική όπου η γνώση έρχεται στον μαθητή μέσα από την ψυχαγωγία (Kalelioglu & Gulbahar, The Effect of Instructional Techniques on Critical Thinking and Critical Thinking Dispositions in Online Discussion, 2013).

- **Code.org**(<https://code.org>)



Το Code.org είναι μια μη κερδοσκοπική online πλατφόρμα κώδικα που κυκλοφόρησε τον Ιανουάριο του 2013 και είναι διαθέσιμη σε 53 γλώσσες μεταξύ των οποίων και τα ελληνικά. Χρησιμοποιεί και αυτό μια οπτική γλώσσα προγραμματισμού με πλακίδια(blocks), την Blockly. Με την τεχνική του drag-and-drop ο χρήστης μετακινεί τα πλακίδια στην περιοχή του editor και δημιουργεί τον τελικό του κώδικα. Η ιστοσελίδα του Code.org είναι διαμορφωμένη με τρόπο που να θυμίζει παιχνίδι και αποτελείται από κατευθυνόμενα μαθήματα μέσα από μικρά βίντεο με διαλέξεις επιστημόνων της πληροφορικής αλλά και γενικές πληροφορίες γύρω από την επιστήμη της πληροφορικής από ανθρώπους γνωστούς στο ευρύ κοινό.

Το Code.org λειτουργεί ουσιαστικά σαν μια διαδικτυακή τάξη που προσφέρει στον εκπαιδευτικό την δυνατότητα διαχείρισης μαθημάτων προγραμματισμού αφού εγγράφει τους μαθητές του σε αυτή και παρακολουθεί τις δραστηριότητές τους το επίπεδο που φτάνουν ακόμη και πόση ώρα ξόδεψαν στην κάθε δραστηριότητα. Οι προγραμματιστικές δραστηριότητες που περιλαμβάνονται στο Code.org χωρίζονται σε τρία ηλικιακά επίπεδα: 4-8, 9-12 και 13+. Οι προγραμματιστικές δραστηριότητες που παρέχονται από το Code.org έχουν τη μορφή παιχνιδιών τύπου παζλ ή λαβύρινθου όπου για την λύση τους απαιτείται η δημιουργία κώδικα με την μορφή drag-and-drop πλακιδίων ενώ προσφέρει ακόμα και δραστηριότητες που δεν απαιτούν την χρήση ηλεκτρονικού υπολογιστή. Οι δραστηριότητες-ασκήσεις του Code.org βοηθούν τους μαθητές να μπουνε στην λογική των αλγορίθμων, των μεταβλητών και των βρόγχων με έναν τρόπο που συνδυάζει την μάθηση με την ψυχαγωγία(Kalelioglu, 2015). Τέλος το Code.org δίνει την δυνατότητα να εμφανίζεται ο κώδικα των πλακιδίων σε JavaScript.

- **Kodable** (<https://www.kodable.com>)



Το Kodable κυκλοφόρησε το Νοέμβριο του 2012 και είναι ένα πλήρες onlin επρόγραμμα σπουδών προγραμματισμού το επίπεδο του οποίου κυμαίνεται από απλές ασκήσεις για εισαγωγή στην υπολογιστική σκέψη μέχρι συγγραφή κώδικα σε γλώσσα Javascript. Το Kodable υπάρχει σαν δωρεάν έκδοση στην οποία παρέχει πρόσβαση στο διδακτικό υλικό και το starter kit των δραστηριοτήτων. Η premium έκδοση για σχολεία κοστίζει \$1250 και παρέχει όλα τα επιπλέον μαθήματα και δραστηριότητες εργαλεία διαχείρισης τάξης και αναλυτικές πληροφορίες για την πρόοδο κάθε χρήστη-μαθητή ενώ

με μια επιπλέον χρέωση \$750 αποκτά πρόσβαση σε 80 ακόμα σχέδια μαθήματος για ηλικίες K-5, προχωρημένες ενότητες με χρήση JavaScript και Swift και επιπλέον μαθήματα που καλύπτουν ρομποτική και ELA. Επίσης υπάρχει η ατομική έκδοση του Kodable που προορίζεται για γονείς με \$59.99/έτος ή \$6,99/μήνα. Σε αυτή περιλαμβάνονται δυνατότητα δημιουργίας έως και 4 προφίλ χρηστών, απεριόριστο δημιουργικό περιεχόμενο, δυνατότητα δημιουργίας παιχνιδιού από τον μαθητή και εισαγωγή στην γλώσσα Javascript.

Το Kodable θεωρείται ένα πλήρες πρόγραμμα σπουδών προγραμματισμού όπου οι μαθητές εισάγονται στην αλγοριθμική σκέψη και μπορεί να φτάσει μέχρι το επίπεδο όπου μαθαίνουν να γράφουν γραμμές κώδικα σε Javascript. Είναι ιδανικό για μαθητές της πρωτοβάθμιας εκπαίδευσης διότι χρησιμοποιεί στοιχεία παιχνιδιού και είναι απλό στην χρήση του ενώ παράλληλα είναι σχεδιασμένο ώστε να βοηθάει στην μετάβαση από τα σύμβολα στον γραπτό κώδικα (Fokides, 2017). Συνδυάζει το παιχνίδι και τα πολύχρωμα γραφικά για να κερδίσει την προσοχή του μαθητή ενώ χρησιμοποιεί drag-and-droπεντολές οι οποίες έχουν την μορφή βέλη κατεύθυνσης. Σε κάθε επίπεδο του Kodable εμφανίζεται ένας χνουδωτός χαρακτήρας ο οποίος πρέπει να οδηγηθεί στην έξοδο ενός λαβυρίνθου χρησιμοποιώντας και τοποθετώντας κατάλληλα τα βέλη-εντολές. Με αυτό τον τρόπο στοχεύει στο να μάθουν οι εκπαιδευόμενοι στο τέλος κάθε λαβυρίνθου για την αλληλουχία τους βρόγχους και τις συνθήκες.

Το Kodable παρέχει στον εκπαιδευτικό πλήρη και έτοιμα σχέδια μαθημάτων και εκπαιδευτικού υποστηρικτικού υλικού με πολλά επίπεδα για την εξάσκηση των μαθητών στις έννοιες του προγραμματισμού (Fokides, 2017). Επιπλέον, η εκπαιδευτική του βιβλιοθήκη συνεχώς μεγαλώνει παρέχοντας συνεχώς καινούριες πηγές στον εκπαιδευτικό. Τα χαρακτηριστικά του Kodable όπως τα πολύχρωμα γραφικά, η άμεση εφαρμογή των εντολών και η μεταφορά των παραδειγμάτων προγραμματισμού στην πραγματικότητα το καθιστούν ιδανικό για την ανάπτυξη των δεξιοτήτων των μαθητών (Gedik, Cetin, & Koca, 2017). Παράλληλα τα κλιμακούμενης δυσκολίας επίπεδα και η συλλογή εικονικών κερμάτων ως επιβράβευση για το ξεκλείδωμα χαρακτήρων αυξάνουν ακόμα περισσότερο το κίνητρο των μαθητών και καθιστούν την εφαρμογή ακόμα πιο θελκτική. Παρά το γεγονός ότι αποτελείται από παιχνίδια το Kodable μαθαίνει στους

μαθητές τις αρχές του προγραμματισμού με πολύ αποτελεσματικό τρόπο. Σύμφωνα με έρευνα οι μαθητές που χρησιμοποίησαν τους το Kodable ανέβασαν το επίπεδο της αλγοριθμικής σκέψης τους. (Pila, Alade, Sheehan, Lauricella, & Wartella, 2019).

- **MIT App Inventor** (<https://appinventor.mit.edu/>)



Το MIT AppInventor είναι ένα δωρεάν περιβάλλον οπτικού προγραμματισμού που κυκλοφόρησε τον Μάρτιο του 2012 και είναι διαθέσιμο σε 12 γλώσσες. Στο App Inventor ο χρήστης αναπτύσσει εφαρμογές για smartphone με λειτουργικό Android. Βασίζεται και αυτό στον προγραμματισμό με πλακίδια(blocks) και στοχεύει σε μια πιο ανεπτυγμένη εκπαίδευση πάνω στον προγραμματισμό. Για να το πετύχει χρησιμοποιεί οπτικά διαισθητικά μπλοκ απλοποιώντας τις προγραμματιστικές έννοιες αφού με αυτό τον τρόπο καθιστούν προσιτή την ροή της λογικής και του προγραμματισμού για τις μικρές ηλικίες (Ruan, Patton, & Tissenbaum, 2017). Η γλώσσα προγραμματισμού που χρησιμοποιεί το App Inventor είναι η blocky και τα μπλοκ προγραμματισμού της εφαρμογής χωρίζονται σε δύο κατηγορίες. Η πρώτη κατηγορία περιλαμβάνει τα built-in blocks τα οποία είναι πάντα διαθέσιμα προς χρήση και χωρίζονται σε 8 κατηγορίες ανάλογα με την λειτουργία τους: ελέγχου(control), λογικής(logic). Μαθηματικών (math), κειμένου(text), λίστες (lists), χρωμάτων (colors), μεταβλητών (variables), διαδικασιών(procedures). Η δεύτερη κατηγορία περιλαμβάνει τα μπλοκ τα οποία αποτελούν συστατικά άλλων μπλοκ ανάλογα με το είδος της εφαρμογής που προγραμματίζουμε. (Ruan, Patton, & Tissenbaum, 2017).

Στην αρχική έκδοση του AppInventor απαιτούνταν μια εγκατάσταση Java και ένας εξομοιωτής (emulator) ενώ στην δεύτερη έκδοση η οποία είναι online αρκεί μόνο ένας λογαριασμός Google.

Το AppInventor έχει όλα τα χαρακτηριστικά μια πραγματικής γλώσσας προγραμματισμού και δίνει την δυνατότητα να δημιουργηθούν εφαρμογές που θα χρησιμοποιηθούν και στην πραγματικότητα. Η χρήση οπτικών πλακιδίων προγραμματισμού (visual programming) αντί γραπτού κώδικα κάνει πιο εύκολη για τους μαθητές την αποφυγή συντακτικών λαθών κάτι που έχει σαν αποτέλεσμα να κρατάει περισσότερο το ενδιαφέρον τους (Morelli, και συν., 2018). Επιτρέπει επίσης την δημιουργία ισχυρών εφαρμογών αφού δίνει την δυνατότητα χρήσης από τις εφαρμογές βάσεων δεδομένων, εισερχόμενων SMS αλληλεπίδρασης μια αισθητήρες και GPS και επικοινωνίας με web APIs.

Το AppInventor είναι πολύ εύκολα προσβάσιμο από τους μαθητές και με λίγη εξάσκηση μπορούν να μάθουν να αναπτύσσουν τις δικές τους εφαρμογές (Chatzinikolakis & Papadakis, 2015). Η δυνατότητα που δίνει επίσης να μοιράζονται οι εφαρμογές που δημιουργούν οι μαθητές μέσα από την διαδικτυακή του κοινότητα και η μορφοποίηση κώδικα άλλων χρηστών δίνει έναν χαρακτήρα ανοιχτής κοινότητας γνώριμο στους μαθητές από την ενασχόληση τους με το Scratch. Οι εφαρμογές που θα δημιουργήσουν οι μαθητές μπορούν να ανέβουν ακόμα και στο επίσημο κατάστημα εφαρμογών της Google, το Google play.

Η εφαρμογή είναι πολύ ενδιαφέρουσα για τους μαθητές και ταυτόχρονα αποτελεσματική στην εκπαιδευτική διαδικασία γιατί είναι εύκολα κατανοητή, το περιβάλλον της είναι πολύ εύχρηστο ενώ ενσωματώνει στοιχεία που είναι γνώριμα στους μαθητές από την πρότερη ενασχόληση τους με άλλες εφαρμογές. Ένα ακόμη χαρακτηριστικό που προσθέτει μεγαλύτερη δημοφιλία στο AppInventor είναι το γεγονός πως οι εφαρμογές που δημιουργούν οι μαθητές παίζουν στα smartphones που είναι πλέον αναπόσπαστο κομμάτι της καθημερινότητας τους αλλά και καθιστά τις δημιουργίες τους κάτι το απτό και άμεσα χρησιμοποιήσιμο. Δηλαδή η μαθησιακή εμπειρία μεταφέρεται από κάτι θεωρητικό σε ένα καθημερινό εργαλείο χρήσης τους (Perdikuri, 2014). Στην Ελλάδα το AppInventor χρησιμοποιείται κυρίως στην δευτεροβάθμια εκπαίδευση και ιδιαίτερα στο γυμνάσιο. Ωστόσο, μπορεί να αποτελέσει ένα πολύ καλό εργαλείο μάθησης προγραμματισμού και για



τις τελευταίες τάξεις του δημοτικού σχολείου αφού είναι εύκολα προσβάσιμο, ενσωματώνει τον αντικειμενοστραφή προγραμματισμό όπως και το Scratch, βοηθάει τους μαθητές να βελτιώσουν την ικανότητα τους στην επίλυση προβλημάτων χωρίς να σπαταλούν χρόνο με συντακτικά λάθη και τέλος με το εύχρηστο περιβάλλον εργασίας που προσφέρει και την δυνατότητα χρήσης των εφαρμογών σε smartphones δημιουργεί την αίσθηση του κινήτρου στους μαθητές. (Morelli, και συν., 2018).

- CodeHS (<https://codehs.com>)



Το CodeHS είναι μια πλατφόρμα διδασκαλίας προγραμματισμού που κυκλοφόρησε τον Μάιο του 2012. Υπάρχει η δωρεάν έκδοση με πλήρη πρόσβαση στον κατάλογο μαθημάτων για τις ηλικίες 6-12 στην οποία παρέχει λειτουργίες διαχείρισης των εγγραφών της τάξης, πρόσβαση στον κώδικα των μαθητών, υλικό για τον εκπαιδευτικό και δυνατότητες δημιουργίας κουίζ, λίστας παιχνιδιών και ανάθεσης εργασιών. Επιπρόσθετα υπάρχει η Pro έκδοση που απευθύνεται σε σχολεία με κόστος περίπου \$2500. Στην συγκεκριμένη έκδοση περιλαμβάνονται πλήρη εργαλεία παρακολούθησης της προόδου των μαθητών, οδηγοί λύσεων για κάθε άσκηση, εκπαιδευτικά μαθήματα και δραστηριότητες χωρίς την χρήση υπολογιστή, βιβλίο βαθμολογίας, βραβεία για τους μαθητές, παραμετροποίηση ανά τάξη και ανά μάθημα και δυνατότητες εξαγωγής δεδομένων μαθητών. Τέλος, υπάρχει και η premium έκδοση της πλατφόρμας η οποία απευθύνεται σε πολλαπλά σχολεία ή ακόμα και σε ολόκληρες περιφέρειες σχολείων με το κόστος να φτάνει έως τα \$25.000.

Όπως αναφέρθηκε παραπάνω το CodeHS είναι μια πλατφόρμα διδασκαλίας η οποία βοηθάει τα σχολεία στην διδασκαλία του προγραμματισμού και της επιστήμης της πληροφορικής εν γένει. Ακολουθεί τα εκπαιδευτικά πρότυπα που έχουν θεσπίσει για την πρωτοβάθμια εκπαίδευση η CSTA (Computer Science Teachers Association) και η ISTE (International Society for Technology in Education) (Padlipsky, 2018).

Αν και σε μια πρώτη ματιά μπορεί να μοιάζει απλοϊκό είναι πολύ αποτελεσματικό στην εισαγωγή στην λογική σκέψη και την ικανότητα επίλυσης προβλημάτων μέσα από τον προγραμματισμό (Smith, Hartley, & Mehdi, 2013). Σε αντίθεση με τις εφαρμογές εκμάθησης προγραμματισμού παρουσιάστηκαν παραπάνω, το CodeHS δεν έχει λειτουργία drag-and-drop και πλακίδια προγραμματισμού αλλά κανονικό κώδικα ο οποίος γράφεται μέσω της λειτουργίας sandbox και είναι βασισμένος στις γλώσσες Javascript, Python, Java και HTML5. Παρέχει έτοιμα κομμάτια κώδικα προς επεξεργασία ώστε να επιτευχθεί το τελικό αποτέλεσμα σύμφωνα με τις οδηγίες της εκάστοτε άσκησης. Στο CodeHS δεν υπάρχει η δομή επανάληψης for οπότε κάθε εντολή που χρειάζεται να επαναληφθεί γράφεται πολλές φορές.

Όπως αναφέρθηκε, το CodeHS παρέχει στον εκπαιδευτικό ένα πλήρες πρόγραμμα σπουδών με πολλά εκπαιδευτικά εργαλεία και επιπλέον υποστηρικτικό υλικό. Η πλατφόρμα ακολουθώντας τα εκπαιδευτικά πρότυπα των CSTA και ISTE επιτυγχάνει α) να καταρτίζει στόχους μάθησης και να συμμετέχουν ενεργά οι ίδιοι οι μαθητές στην εκπαίδευσή τους, β) να κατανοούν τα δικαιώματα και τις ευθύνες που έχουν με την χρήση των νέων τεχνολογιών, γ) να αναγνωρίζουν τις αξιόπιστες πηγές online πληροφοριών, δ) να κατανοούν τις βασικές αρχές επίλυσης προβλημάτων, ε) να είναι σε θέση να δημιουργούν στρατηγικές για την επίλυση των προβλημάτων, στ) να μπορούν να εκφραστούν μέσω των ψηφιακών μέσων με συνοπτικό και σαφή τρόπο και ζ) να συνεργάζονται με τους υπόλοιπους μαθητές για την επίτευξη κοινών στόχων (Padlipsky, 2018).

## Κεφάλαιο 3

### 3.1 Τι είναι το STEM

Ο όρος STEM προέρχεται από τα αρχικά της φράσης «Science Technology Engineering and Mathematics». Πρωτοεμφανίστηκε το 2001 από τη βιολόγο Judith A. Ramaley, η οποία ως Διευθύντρια του Ιδρύματος Φυσικών Επιστημών των ΗΠΑ, ήταν υπεύθυνη για την ανάπτυξη νέων προγραμμάτων σπουδών. Σύμφωνα με τον Οργανισμό Εκπαιδευτικής Ρομποτικής, Επιστήμης, Τεχνολογίας και Μαθηματικών «το STEM είναι μια προσέγγιση στην Εκπαίδευση που σχεδιάζεται ώστε στη διδασκαλία των Μαθηματικών και των Φυσικών Επιστημών, που είναι ζωτικής σημασίας για μια βασική κατανόηση του σύμπαντος, να εισαχθούν οι Τεχνολογίες και η Επιστήμη των Μηχανικών, που αποτελούν για τον άνθρωπο τα μέσα αλληλεπίδρασης με το σύμπαν». Σύμφωνα με τον Morrison (2006), θεωρείται ένας «μετά-επιστημονικός» κλάδος, δηλαδή αφορά την δημιουργία ενός γνωστικού κλάδου που βασίζεται στην ολοκλήρωση άλλων γνωστικών κλάδων σε μία νέα «ολότητα» ενώ ως «μετά-επιστημονικός» κλάδος περιέχει έννοιες που συχνά συγκρούονται μεταξύ τους. Οι προσεγγίσεις STEM περιστρέφονται γύρω από την επίλυση πραγματικών προβλημάτων και υποστηρίζουν τη μάθηση μέσω ερευνητικής εργασίας. Οι μαθητές, δηλαδή, συμμετέχουν ενεργά σε δραστηριότητες επίλυσης προβλημάτων που υποστηρίζουν την εγκάρσια διεπιστημονικότητα και χρησιμοποιούνται μέθοδοι από ένα ή περισσότερα από τα υπόλοιπα αντικείμενα του STEM (Nicolescu, 2010).

Η εγκάρσια διεπιστημονικότητα (transdisciplinary) είναι ο πυρήνας της μεθοδολογίας STEM και εκφράζει τον προσανατολισμό του STEM προς την επίλυση πολύπλοκων προβλημάτων πραγματικών καταστάσεων χρησιμοποιώντας εργαλεία από διάφορα επιστημονικά πεδία (Ψυχάρης & Καλοβρέκτης, Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ, 2018). Σύμφωνα με τους ίδιους, ο όρος εγκάρσια διεπιστημονικότητα πρέπει να διαχωρίζεται από τους όρους διεπιστημονικότητα (interdisciplinary) και πολυδιάστατη επιστημονικότητα (multidisciplinary) που συναντάμε συχνά στην βιβλιογραφία και που όλοι αναφέρονται στην επίλυση σύνθετων προβλημάτων «διασχίζοντας» διαφορετικές γνωστικές περιοχές.

Η πολυδιάστατη προσέγγιση (multidisciplinary) χαρακτηρίζεται από την βιβλιογραφία ως η λιγότερο «ολοκληρώσιμη» προσέγγιση καθώς τα χαρακτηριστικά της

εστιάζονται κυρίως στην αξιοποίηση ορισμένων γνωστικών περιοχών που εφαρμόζονται μαζί για την διερεύνηση μιας κοινής περιοχής. Η διεπιστημονικότητα (interdisciplinary) θεωρείται ως ένα «βήμα παραπάνω» από την πολυδιάστατη προσέγγιση καθώς εστιάζει στην αντιμετώπιση πραγματικών προβλημάτων κάτι που έχει σαν αποτέλεσμα οι συμμετέχοντες (εκπαιδευόμενοι και εκπαιδευτές) να αναγκάζονται να διασχίσουν τα σύνορα μεταξύ των γνωστικών περιοχών ώστε να δημιουργηθεί νέα γνώση. Σε αυτό το σημείο έγκειται η διαφορά της με την πολυδιάστατη προσέγγιση αφού φτάνει σε μεγαλύτερο επίπεδο «ολοκλήρωσης» και συνεργασίας γεφυρώνοντας επιστημονικές απόψεις από διαφορετικές γνωστικές περιοχές και παρέχοντας δυνατότητα αξιολόγησης της γνώσης που αποκτήθηκε με την χρήση γειτονικών γνωστικών αντικειμένων (Mobjork, 2010). Η δια-επιστημονική προσέγγιση (transdisciplinary), στην οποία βασίζεται το STEM, θεωρεί ως πραγματικά προβλήματα αυτά στα οποία οι εκπαιδευόμενοι εφαρμόζουν γνώσεις και δεξιότητες από δύο ή περισσότερες διαφορετικές περιοχές. Οι μαθητές μέσα από συνεργατική δουλειά κατακτούν νέες γνώσεις και νέες συνδέσεις που ξεπερνούν τις εμπλεκόμενες γνωστικές περιοχές αφού θέτουν ερωτήματα και απαντούν συλλογικά σε αυτά και εξερευνούν έναν νέο κόσμο μέσω της διερευνητικής μάθησης (Vasquez, Comer, & Sneider, 2013). Η δια-επιστημονική προσέγγιση συνδέει έννοιες, ικανότητες και δεξιότητες κατά την επίλυση πραγματικών προβλημάτων εστιάζοντας σε επιστημονικές διαδικασίες. Συνήθως επιλέγεται κάποιο φαινόμενο που πρέπει να ερμηνευτεί και οι εκπαιδευόμενοι καλούνται να επιλέξουν ποιες έννοιες, θεωρίες και εργαλεία θα συνδυάσουν από διαφορετικές γνωστικές περιοχές και ταυτόχρονα να αξιοποιήσουν προϋπάρχουσες γνώσεις αλλά και ικανότητες σχεδιασμού και κατασκευής (Ψυχάρης & Καλοβρέκτης, Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ, 2018).

Ανακεφαλαιώνοντας, σύμφωνα με τους Ψυχάρη και Καλοβρέκτη (2018), το στοιχείο που κάνει την διεπιστημονική (interdisciplinarity) και την δια-επιστημονική προσέγγιση (transdisciplinarity) να διαφέρουν από την πολυδιάστατη προσέγγιση (multidisciplinarity) είναι ότι ενσωματώνουν την επαναληπτική διαδικασία, ξεπερνάνε τα σύνορα των εκάστοτε γνωστικών περιοχών και, τέλος, εμπλέκουν άτομα που δεν προέρχονται από την εκπαίδευση ή την έρευνα. Οι δύο πρώτες προσεγγίσεις διαφέρουν ελάχιστα μόνο στο γεγονός ότι η δια-επιστημονική προσέγγιση σκοπεύει να δημιουργήσει νέες γνωστικές περιοχές και θεωρίες.

### 3.2 Το STEM στην εκπαίδευση

Στα προηγούμενα κεφάλαια αναφέρθηκαν οι έννοιες Υπολογιστική Σκέψη, διερευνητική μάθηση, δια-επιστημονική προσέγγιση. Γίνεται λοιπόν κατανοητό ότι η ένταξη του STEM στην εκπαίδευση θα πρέπει να είναι μια σύνθεση όλων αυτών των εννοιών.

Όλες οι προαναφερθείσες έννοιες έχουν σαν κοινή συνισταμένη την επίλυση προβλημάτων. Έτσι η εισαγωγή του STEM στην εκπαίδευση θα πρέπει να περιλαμβάνει την μεθοδολογία του υπολογιστικού πειράματος, την δια-επιστημονική προσέγγιση και να υλοποιείται ως μεθοδολογία επίλυσης προβλήματος με την διερευνητική/ ανακαλυπτική μάθηση (Ψυχάρης & Καλοβρέκτης, Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ, 2018). Σκοπός της μεθοδολογίας STEM στην εκπαίδευση θα πρέπει να είναι οι μαθητές να αποκτήσουν ένα επαρκές και «ολοκληρωμένο» σώμα γνώσεων από όλες τις γνωστικές περιοχές που το αποτελούν αλλά και τις ήδη υπάρχουσες δεξιότητες. Να αναπτύξουν, δηλαδή, ικανότητες ανάλυσης, σχεδιασμού και επίλυσης προβλημάτων με αναζήτηση διάφορων εναλλακτικών λύσεων και παράλληλα να εξοικειωθούν με τον τρόπο μετατροπή της θεωρητικής σκέψης σε πράξη (Stohlman, Moore, & Roehrig, 2012). Η εκπαίδευση στο STEM θα πρέπει να περιλαμβάνει επίσης διδακτικές ακολουθίες σε όλο το φάσμα της εκπαίδευσης από την προσχολική ηλικία μέχρι το μετά-διδακτορικό επίπεδο τόσο κατά τη διάρκεια του παραδοσιακού ωρολογίου προγράμματος του σχολείου όσο και με επιπλέον δραστηριότητες μετά την λήξη τους (Gonzalez & Kuenzi, 2102).

Αναμφίβολα στα εφόδια που καλούνται να έχουν οι σημερινοί μαθητές είναι οι δεξιότητές τους να συνδυάζονται με τις νέες τεχνολογίες. Σύμφωνα με τον Morrison (2006), η ικανότητα επίλυσης προβλημάτων, η συνεργατική μάθηση, η δημιουργική και κριτική σκέψη είναι απαραίτητες δεξιότητες για τους μαθητές και προς αυτή την κατεύθυνση συνεισφέρει το STEM τόσο με τα επιστημονικά πεδία που το αποτελούν όσο και με τα εργαλεία και τις δραστηριότητες που προσφέρει. Προσθέτει πως οι μαθητές που ολοκληρώνουν μια εκπαίδευση βασισμένη σε στοιχεία του STEM θα είναι ικανοί να επιλύουν προβλήματα, να θέτουν ερωτήσεις να ερευνούν και να ανακαλύπτουν συνεχώς.

Οι δραστηριότητες του STEM δεν είναι απαραίτητο να εμπλέκουν κάθε φορά όλα τα γνωστικά πεδία του. Το βασικό είναι να έχουν κατανοήσει οι εκπαιδευτικοί την δια-

επιστημονική φύση του STEM και να συμμετέχουν σε δραστηριότητες όπου ένα πρόβλημα θα αντιμετωπίζεται από διαφορετικές γνωστικές περιοχές που θα ανατροφοδοτούνται μεταξύ τους (Kelley & Knowles, 2016). Οι Vasquez, Comer, & Sneider (2013) προτείνουν πέντε βασικές αρχές που πρέπει να λαμβάνονται υπόψιν για την σχεδίαση μαθημάτων βασισμένων στο STEM:

- **Εστίαση στην ολοκλήρωση:** Η συγκεκριμένη αρχή αναφέρεται στην δια-επιστημονική προσέγγιση μεταξύ δύο ή περισσότερων γνωστικών πεδίων η οποία βοηθά τους μαθητές να κατανοούν την σύνδεση θεμάτων που αρχικά φαίνεται ασύνδετη. Συνδυάζοντας π.χ την επιστήμη με την μηχανική ή τις κοινωνικές επιστήμες με τα μαθηματικά οι μαθητές θα είναι σε θέση να συνδέσουν πληροφορίες και γνώσεις από διαφορετικά γνωστικά πεδία ώστε να προτείνουν λύσεις σε πραγματικά προβλήματα.
- **Δημιουργία συνδέσεων:** Κάποιες φορές οι μαθητές δεν αντιλαμβάνονται το πώς οι γνώσεις που αποκτούν μπορεί να είναι εφαρμόσιμες στην πράξη. Σε αυτό το σημείο η μεθοδολογία STEM πρέπει να βοηθάει ώστε να συνδέονται οι νέες γνώσεις με προβλήματα και καταστάσεις του πραγματικού κόσμου που απαιτούν τεχνολογικές λύσεις. Τα ερωτήματα για το που είναι απαραίτητες οι γνώσεις που απέκτησαν και αν θα τους «είναι χρήσιμες» στο μέλλον είναι τα πλέον συνηθισμένα ανάμεσα στους μαθητές.
- **Έμφαση στις σύγχρονες δεξιότητες:** Ένα σημαντικό θέμα για το σύγχρονο εργατικό δυναμικό δεν είναι τόσο η ποσότητα των γνώσεων που μπορεί να επεξεργαστεί αλλά κυρίως το πώς μπορεί να έχει πρόσβαση σε πληροφορίες όταν τις χρειάζεται και πώς θα τις χρησιμοποιήσει κατάλληλα για να επιλύσει προβλήματα.
- **Πρόκληση του ενδιαφέροντος των μαθητών:** Χρησιμοποιώντας κατάλληλες δραστηριότητες στο πλαίσιο της ερευνητικής μάθησης θα πρέπει να προκαλείται το ενδιαφέρον των μαθητών. Είναι σημαντικό οι δραστηριότητες αυτές να μην έχουν αυξημένο βαθμό δυσκολίας ώστε να μην τις εγκαταλείπουν οι μαθητές αλλά και να μην είναι πολύ εύκολες με συνέπεια να τις αποφεύγουν.

- **Ποικιλία δραστηριοτήτων και προσεγγίσεων:** Οι δραστηριότητες STEM που σχεδιάζονται θα πρέπει να παρέχουν μια ποικιλία μαθησιακών στόχων έτσι ώστε οι μαθητές να εκφράζουν τις γνώσεις με πολλούς τρόπους, να μοιράζονται τις εμπειρίες τους και να διευρύνουν το σύνολο των δεξιοτήτων τους. Οι προσεγγίσεις θα πρέπει να είναι βασισμένες στην επίλυση προβλημάτων στα οποία οι μαθητές θα προσπαθούν να δώσουν δημιουργικές λύσεις. Παράλληλα θα πρέπει να χρησιμοποιούνται προσεγγίσεις βασισμένες στην μάθηση μέσω πρότζεκτ (project-based learning) στα οποία οι μαθητές θα εμπλέκονται σε μια ενεργό αναζήτηση που αφορά τον πραγματικό κόσμο μέσω της σύνδεσης με πραγματικά γεγονότα και αληθινά προβλήματα.

### 3.3 STEM και Εκπαιδευτική Ρομποτική

Όπως αναφέρθηκε στο κεφάλαιο 3.2, ο πυρήνας της μεθοδολογίας του STEM είναι η «ολοκλήρωση» δύο ή περισσότερων γνωστικών πεδίων με σκοπό την βελτίωση της ικανότητας των μαθητών στο να επιλύουν προβλήματα μέσα από συνεργατική και ανακαλυπτική project-based μάθηση. Μία από τις πιο αποτελεσματικές και πιο σύγχρονες υλοποιήσεις της μεθοδολογίας STEM στην εκπαίδευση είναι η Ρομποτική.

Η Εκπαιδευτική Ρομποτική περιλαμβάνει κατασκευές και χρήση του ηλεκτρονικού υπολογιστή οπότε είναι φυσικό να κεντρίζει εύκολα το ενδιαφέρον του εκπαιδευόμενου. Σκοπός, ωστόσο, της Εκπαιδευτικής Ρομποτικής ως δραστηριότητα STEM δεν είναι να μάθουν οι εκπαιδευόμενοι να κατασκευάζουν ρομποτικούς μηχανισμούς και να τους ελέγχουν αλλά να αποτελέσει το μέσο για την επίτευξη διδακτικών στόχων στην μαθησιακή διαδικασία (Ψυχάρης & Καλοβρέκτης, Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ, 2018). Με την κατάλληλη παιδαγωγική υποστήριξη οι δραστηριότητες εκπαιδευτικής ρομποτικής μπορούν να βοηθήσουν σημαντικά τους μαθητές να βελτιώσουν την ικανότητα τους στην επίλυση προβλημάτων (Kraetzschmar, 2009).

Μια ακόμη σημαντική ιδιότητα της ρομποτικής είναι πως βοηθά τους μαθητές στο να κατανοήσουν με πιο λεπτομερή τρόπο την τεχνολογία κάτι που δεν είναι αυτονόητο παρά την εξοικείωση που έχουν οι σημερινοί μαθητές με πολλά προϊόντα της. Η εκπαιδευτική ρομποτική φέρνει τους μαθητές πιο κοντά με το πώς λειτουργεί η τεχνολογία σε τομείς της καθημερινότητας (Eguchi, 2014).

Η ενσωμάτωσή της στην εκπαιδευτική διαδικασία βοηθά τους μαθητές να γίνουν πιο συνειδητοποιημένοι όσον αφορά τον τεχνολογικό κόσμο που τους περιβάλλει και τις αρχές που τον διέπουν. Μέσω των δραστηριοτήτων της εκπαιδευτικής ρομποτικής αναπτύσσουν σταδιακά έναν επιστημονικό τρόπο σκέψης και προσεγγίζουν την συμπεριφορά που έχει ένας ερευνητής (Μπάρας & Βασιλόπουλος, 2014). Η ενασχόληση με δραστηριότητες της Εκπαιδευτικής Ρομποτικής αφορούν δύο τομείς: την κατασκευή και τον προγραμματισμό. Έτσι στην ίδια δραστηριότητα οι μαθητές συνδυάζουν και εφαρμόζουν έννοιες από πολλά γνωστικά πεδία(μαθηματικά, φυσική, προγραμματισμό, μηχανική) και καθώς εμπλέκονται συνεργατικά και μέσα από παιχνίδι σε αυτές αυξάνεται το κίνητρό τους για κάθε μαθησιακό αντικείμενο (Γριζιώτη, Ξένος, & Κυνηγός, 2016).

Σημαντικός παράγοντας για την αποτελεσματική εφαρμογή της εκπαιδευτικής ρομποτικής αποτελεί η επιλογή του ρομποτικού μηχανισμού πάνω στον οποίο θα στηθούν τα διδακτικά σενάρια. Ο πρώτος παράγοντας ο οποίος πρέπει να ληφθεί υπόψιν για την επιλογή είναι ασφαλώς η ηλικία των μαθητών αλλά υπάρχει και ένα σύνολο από επιπλέον κριτήρια. Σύμφωνα με τους Ψυχάρη & Καλοβρέκτη (2018) αυτά είναι:

- **Τα διαθέσιμα σενάρια:** Είναι σημαντικό τα εκπαιδευτικά σενάρια που μπορεί να υλοποιήσει ο ρομποτικός μηχανισμός να συμβαδίζουν με τους διδακτικούς στόχους που θέτει το εκάστοτε Αναλυτικό Πρόγραμμα Σπουδών.Επιπλέον θα πρέπει να είναι εφικτή η επίτευξη των στόχων με βάση το σύνολο διδακτικών ωρών που έχουμε στην διάθεσή μας. Η αποτύπωση των στόχων μπορεί να γίνει απαντώντας σε κάποια συγκεκριμένα ερωτήματα όπως το τι θέλουμε να μάθουν οι μαθητές βάσει των οδηγιών του Α.Π.Σ, τι θα είναι ικανοί να κάνουν μετά το πέρας του διδακτικού σεναρίου και ποιες στάσεις ως προς κάποιες προγενέστερες στάσεις θα αναπτυχθούν ή θα αλλάξουν.
- **Η διαθεματικότητα των διαθέσιμων σεναρίων:** Ο ρομποτικός μηχανισμός που θα επιλεγεί πρέπει να καλύπτει τις ανάγκες του Α.Π.Σ για την τάξη για την οποία θα επιλεγεί.



- **Το μέγεθος των στοιχείων:** Το μέγεθος των συνοδευτικών στοιχείων του ρομποτικού μηχανισμού αποτελεί βασικό κριτήριο καταλληλότητας ειδικά όταν πρόκειται να χρησιμοποιηθεί σε μαθητές μικρότερων τάξεων.
- **Ο αριθμός των συναρμολογούμενων μερών του:** Ο μεγάλος αριθμός συναρμολογούμενων στοιχείων σε έναν ρομποτικό μηχανισμό μπορεί να αποτελέσει μειονέκτημα καθώς μπορεί να μειώσει σταδιακά το ενδιαφέρον των μαθητών. Παράλληλα είναι πιθανόν να αυξηθεί ο «διδακτικός θόρυβος» αφού ένας αριθμός μαθητών θα αδυνατεί να ανταποκριθεί στην δραστηριότητα.
- **Η μέθοδος προγραμματισμού:** Η μέθοδος προγραμματισμού του ρομποτικού μηχανισμού θα πρέπει και αυτή να συμβαδίζει με το εκάστοτε Α.Π.Σ. Για παράδειγμα στην πρωτοβάθμια εκπαίδευση θα πρέπει να μπορεί να προγραμματιστεί σε κάποιο Scratch-based ή Block-based περιβάλλον.
- **Οι διαθέσιμοι αισθητήρες:** Ο τύπος και ο αριθμός των αισθητήρων που συνοδεύουν έναν ρομποτικό μηχανισμό έχουν άμεση εξάρτηση με το πλήθος των διδακτικών σεναρίων που μπορούμε να πραγματοποιήσουμε.
- **Οι διαθέσιμοι ενεργοποιητές:** Ο τύπος και ο αριθμός των ενεργοποιητών που συνοδεύουν έχουν και αυτοί άμεση εξάρτηση με το πλήθος των διδακτικών σεναρίων που μπορούμε να πραγματοποιήσουμε.

Με βάση τα παραπάνω κριτήρια οι καταλληλότερες και πιο δημοφιλείς εκπαιδευτικής ρομποτικής πέραν την πλατφόρμας Arduinoπου αναλύεται διεξοδικά στο επόμενο κεφάλαιο είναι οι εξής:

- **Η έξυπνη μέλισσα (Bee-bot) –<https://bee-bot.us>**



Το προγραμματιζόμενο ρομπότ Beebot απευθύνεται κυρίως σε μαθητές προσχολικής ηλικίας και των πρώτων τάξεων του δημοτικού ως μια πρώτη εισαγωγή στην εκπαιδευτική ρομποτική. Στην ιστοσελίδα του υπάρχουν διάφορες εκπαιδευτικές δραστηριότητες προγραμματισμού. Προγραμματίζεται μέσω πλήκτρων που υπάρχουν πάνω στο ρομπότ χωρίς να χρειάζεται χρήση εξωτερικής εφαρμογής κάτι που συμβάλλει στο να είναι φιλικό προς τα παιδιά αυτής της ηλικίας. Ωστόσο για να αξιοποιηθεί πλήρως η πλατφόρμα απαιτείται να κατασκευαστεί η «πίστα» πάνω στην οποία θα κινείται το ρομπότ ή να αγοραστεί κάποιο έτοιμο «χαλί-πίστα» από το site της εταιρίας.

- **Ο ρομποτικός μηχανισμός Edison**



Το ρομπότ Edison απευθύνεται σε μαθητές των πρώτων τάξεων του δημοτικού σχολείου έως τις πρώτες τάξεις της δευτεροβάθμιας εκπαίδευσης και έχει το χαρακτηριστικό ότι μπορεί να χρησιμοποιηθεί και για δραστηριότητες τύπου LEGO αφού φέρει βάση σύνδεσης για τουβλάκια. Για τον προγραμματισμό του το Edison χωρίζεται σε τέσσερα

επίπεδα ανάλογα με την ηλικία στην οποία απευθύνεται. Στο πρώτο επίπεδο για μαθητές προσχολικής ηλικίας και πρώτων τάξεων του δημοτικού το Edison δίνει την δυνατότητα προγραμματισμού μέσω τηλεχειρισμού και barcodes κάνοντας έτσι διασκεδαστική την πρώτη επαφή των μαθητών με τον προγραμματισμό (<https://meetiedison.com/barcodes/>). Στο δεύτερο επίπεδο και στο τρίτο επίπεδο για τις ηλικίες 7-12 το Edison παρέχει δύο επιλογές block-based προγραμματισμού. Στο δεύτερο επίπεδο είναι το περιβάλλον Edblocks στην οποία τα block προσομειώνουν κινήσεις και αντιδράσεις του ρομπότ και ο χρήστης ρυθμίζει την διάρκεια που θα έχει το καθένα ενώ δίδεται και η δυνατότητα δημιουργία «προκλήσεων» για τους μαθητές (<https://meetiedison.com/robot-programming-software/>). Στο τρίτο επίπεδο είναι η επιλογή του περιβάλλοντος EdScratch που ενσωματώνει πλήρως το περιβάλλον του Scratch και ενδίκνυται για πιο πολύπλοκες προγραμματιστικές δομές (<https://meetiedison.com/robot-programming-software/>). Στο τέταρτο επίπεδο για τις μεγαλύτερες ηλικίες το Edison παρέχει το περιβάλλον προγραμματισμού EdPy το οποίο δεν είναι block-based αλλά βασίζεται σε κώδικα της γλώσσας Python (<https://meetiedison.com/robot-programming-software/edpy/>).

- **Ο ρομποτικός μηχανισμός WeDo της LEGO**



Ο ρομποτικός μηχανισμός LEGO WeDo απευθύνεται σε μαθητές δημοτικού και αποτελεί μια εισαγωγή στην εκπαιδευτική ρομποτική μέσα από ένα σύνολο 12 θεματικών δραστηριοτήτων. Οι μαθητές λειτουργώντας συνεργατικά μαθαίνουν να χειρίζονται δομικά

στοιχεία και κάνουν μια πρώτη εισαγωγή στην αλγοριθμική σκέψη. Επίσης, τα πολλά αποσπόμενα στοιχεία του WeDO βοηθούν στην κατανόηση φυσικών μεγεθών όπως η απόσταση, η κλίση και άλλα (Ψυχάρης & Καλοβρέκτης, Εκπαιδευτική Ρομποτική, 2018).

Το WeDo προγραμματίζεται μέσω του Scratch και αποτελείται από κομμάτια τύπου LEGO, σύνδεση USB για την επικοινωνία και τον προγραμματισμό μέσω ηλεκτρονικού υπολογιστή, κινητήρα, αισθητήρα κλίσης και αισθητήρα κίνησης ο οποίος μπορεί να ανιχνεύσει αντικείμενα σε απόσταση 15 εκατοστών. Σύμφωνα με τους Ψυχάρη & Καλοβρέκτη (2018) τα πλεονεκτήματα που προσφέρει η ρομποτικός μηχανισμός WeDostους μαθητές είναι:

- (1) Η ανάπτυξη των επικοινωνιακών και και συνεργατικών δεξιοτήτων τους στα πλαίσια της ομάδας.*
- (2) Η κατανόηση της μεθόδου δοκιμής-επαλήθευσης.*
- (3) Η ανάπτυξη δεξιοτήτων χειρισμών αντικειμένων.*
- (4) Εισαγωγή στον κώδικα.*
- (5) Η εισαγωγή διαθεματικής προσέγγισης στην επίλυση προβλημάτων.*
- (6) Εισαγωγή σε βασικές προγραμματιστικές έννοιες όπως δομή, επανάληψη, διακοπή.*
- (7) Συμβατότητα με το Scratch.*
- (8) Κατανόηση μηχανικών λειτουργιών*

- **Η πλατφόρμαMindstorms EV3 της LEGO**



Το LEGO Mindstorms EV3 είναι ίσως η πιο διαδεδομένη πλατφόρμα εκπαιδευτικής ρομποτικής αυτή την στιγμή και απευθύνεται σε μαθητές των δύο τελευταίων τάξεων του δημοτικού αλλά και μεγαλύτερους. Το LEGO Mindstorms αποτελείται από περίπου 1000 στοιχεία όπως τουβλάκια LEGO και ειδικά εξαρτήματα για την κατασκευή των εκπαιδευτικών ρομπότ. Για τον έλεγχο και την οδήγηση των ρομποτικών κατασκευών χρησιμοποιείται ο πολύ ισχυρός ελεγκτής EV3. Ο EV3 είναι εξοπλισμένος με επεξεργαστή ARM 9 και μια LCD οθόνη στην οποία μπορούμε να βλέπουμε τις εργασίες που εκτελούνται. Ακόμη διαθέτει 64 MB RAM και 16 MB FlashMemory, 4 εισόδους για την σύνδεση αισθητήρων και 4 εξόδους για την σύνδεση των κινητήρων οδήγησης των μηχανικών στοιχείων. Τέλος, διαθέτει θύρα USB και Wi-Fi για την επικοινωνία με τον υπολογιστή αλλά και υψηλής ποιότητας ηχείο (<https://www.lego.com/en-gr/product/ev3-intelligent-brick-45500>).

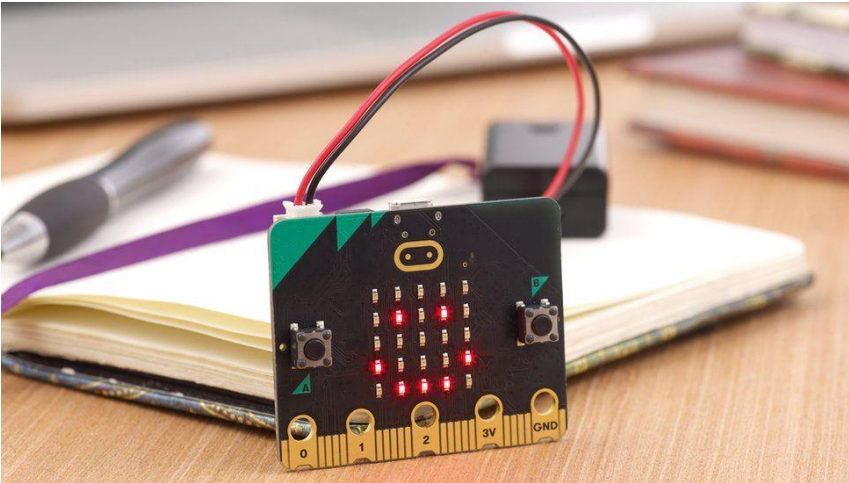
Για τον προγραμματισμό του το LEGO Mindstorms EV3 παρέχει εφαρμογή block-programming η οποία δίνει την δυνατότητα να μεταφερθεί το πρόγραμμα είτε απευθείας μέσω της θύρας USB, είτε μέσω κάρτας SD είτε με Bluetooth. Κάθε μπλοκ αναπαριστά και ένα στοιχείο του ρομπότ και μεταβλητές για τον έλεγχο του κίνησης έτσι διασκεδαστική την διαδικασία του προγραμματισμού και κατά συνέπεια πιο ελκυστική την εκπαιδευτική διαδικασία.

Η μεγάλη δύναμη του LEGO Mindstorms EV3 είναι τα περιφερειακά στοιχεία που διαθέτει. Τα σημαντικότερα από αυτά είναι τα εξής σύμφωνα με τους Ψυχάρη και Καλοβρέκτη (2018):

- *Κινητήρες:* Υπάρχουν δύο είδη διαθέσιμων κινητήρων από την πλατφόρμα. Ένας μεγάλος κινητήρας ο οποίος χρησιμοποιείται ως ο κύριος κινητήρας οδήγησης των ρομποτικών κατασκευών. Έχει ενσωματωμένο αισθητήρα στροφών με ευαισθησία 1 μοίρας και ο ρυθμός περιστροφής του είναι 160-170 ανά λεπτό και η ροπή στρέψης του αγγίζει τα 20Ncm. Ο δεύτερος κινητήρας ο οποίος είναι μικρότερος και ελαφρύτερος χρησιμοποιείται για να δώσει κίνηση στα μικρότερα μηχανικά μέρη του ρομπότ. Έχει ρυθμό περιστροφής 240-250 ανά λεπτό ενώ η ροπή στρέψης του αγγίζει τα 8Ncm. Αμφότεροι οι κινητήρες διαθέτουν ειδική υποδοχή για σύνδεση με τον κεντρικό ελεγκτή EV3.
- *Αισθητήρες:* Αποτελούν την δίοδο επικοινωνίας του ρομπότ με το εξωτερικό περιβάλλον, μέσω του ελεγκτή, και δίνουν τις πληροφορίες που θα καθορίσουν την συμπεριφορά του ρομποτικού μηχανισμού. Αυτοί είναι: α) ο αισθητήρας προσέγγισης ο οποίος είναι αισθητήρας υπέρυθρης ακτινοβολίας. Εκπέμπει και λαμβάνει σήματα υπέρυθρων και με αυτόν τον τρόπο μετράει την απόσταση από κάποιο άλλο αντικείμενο. Ακόμα υπάρχει β) ο αισθητήρας αφής ο οποίος είναι ένας αναλογικός αισθητήρας. Ανιχνεύει πότε έχει πατηθεί ή απελευθερωθεί το κόκκινο πλήκτρο που βρίσκεται πάνω του και με αυτό τον τρόπο ανιχνεύει αν η ρομποτική κατασκευή έχει έρθει σε επαφή με άλλο αντικείμενο. Τέλος υπάρχει γ) ο αισθητήρας ανίχνευσης χρωμάτων ο οποίος έχει την δυνατότητα να ανιχνεύει το χρώμα και την ένταση της ακτινοβολίας ενός αντικειμένου που βρίσκεται μπροστά του. Λειτουργεί είτε ως ανιχνευτής χρωμάτων είτε ως ανιχνευτής έντασης ακτινοβολίας ανάλογα με την απαιτήσεις της εκάστοτε εφαρμογής.
- *Συσκευή Remote Infrared Beacon:* Χρησιμοποιείται ως τηλεχειριστήριο υπέρυθρων για τον έλεγχο του ρομπότ, σε συνδυασμό με τον αισθητήρα προσέγγισης.

- *Διασύνδεση των στοιχείων:* Για την διασύνδεση των αισθητήρων και των κινητήρων με τον ελεγκτή χρησιμοποιείται ειδικό καλώδιο με το κατάλληλο βύσμα. Κάθε είσοδο αντιστοιχεί σε έναν προεπιλεγμένο αισθητήρα.

➤ **To BBC micro:bit**



Το micro:bit είναι μια προγραμματιζόμενη πλακέτα σε μέγεθος πιστωτικής κάρτας η οποία δημιουργήθηκε το 2015 από τον αγγλικό ραδιοτηλεοπτικό φορέα BBC με σκοπό να βοηθήσει τα σχολεία στην διδασκαλία της Πληροφορικής. Το micro:bit παρέχει 25 εξωτερικούς ακροδέκτες για τη σύνδεση αισθητήρων, LED, κινητήρων και άλλων περιφερειακών. Ακόμη ενσωματώνει δύο κουμπιά στο μπροστινό του μέρος ενώ διαθέτει κεραία Bluetooth Low Energy για ασύρματη επικοινωνία με υπολογιστές ή άλλες συσκευές. Ακόμη δίνει την δυνατότητα ασύρματης επικοινωνίας μέσω ραδιοκυμάτων και ενσωματώνει μαγνητόμετρο για τον προσδιορισμό της κατεύθυνσης. (<https://microbit.org>)

Για τον προγραμματισμό του το micro:bit παρέχει τον scratch-based editor Makecode και επιπλέον δίνει την δυνατότητα χρήσης του editor microPython, του Scratch αλλά και του Pictoblox που αναλύεται στα πλαίσια της παρούσας εργασίας.

Συνοψίζοντας, είναι εμφανές πως η εκπαιδευτική ρομποτική προσφέρει σημαντικά στοιχεία στην εκπαιδευτική διαδικασία. Ως εργαλείο μάθησης η εκπαιδευτική ρομποτική:

- ✓ Ελκύει το ενδιαφέρον των μαθητών λόγω της ψυχαγωγικής φύσης των δραστηριοτήτων της.

- ✓ Εμπλέκει με ενεργό τρόπο τους μαθητές στην εκπαιδευτική διαδικασία, βοηθώντας στην αποτελεσματικότερη απόκτηση γνώσεων.
- ✓ Δημιουργεί πραγματικά περιβάλλοντα μάθησης εισάγοντας τους μαθητές στην επίλυση πραγματικών προβλημάτων.
- ✓ Βοηθάει στην ανάπτυξη πνευματικών δραστηριοτήτων όπως η κριτική σκέψη, η δημιουργικότητα, η συνεργατικότητα και ο επιστημονικός τρόπος σκέψης.
- ✓ Επηρεάζει θετικά τις στάσεις απέναντι στην μάθηση εν γένει.
- ✓ Ενσωματώνει πλήρως τις έννοιες της μεθοδολογίας STEM.
- ✓ Έχει διαθεματικό χαρακτήρα καθώς ενσωματώνει πολλά γνωστικά αντικείμενα.
- ✓ Βοηθά τους μαθητές να αποκτήσουν μια πρώτη γεύση σχετικά με το πώς λειτουργεί η τεχνολογία.
- ✓ Είναι κατάλληλη για κάθε ηλικία και δεν απαιτεί πρότερες γνώσεις ή επίπεδο εξοικείωσης με την τεχνολογία.



## Κεφάλαιο 4

### 4.1.1 Το Arduino

Το Arduino έκανε την εμφάνιση του το 2005 όταν ο καθηγητής Massimo Banzi έψαχνε τρόπο να δώσει στους μαθητές την ευκαιρία να ανακαλύπτουν πράγματα μόνοι τους αντί απλά να τα ακούν θεωρητικά μειώνοντας παράλληλα το κόστος σε σχέση με άλλες αντίστοιχες συσκευές. Για τον σκοπό αυτό ζήτησε την βοήθεια του καθηγητή του Πανεπιστημίου του Malmö David Cuatrecasas και μαζί με την βοήθεια δύο φοιτητών ξεκίνησαν την δημιουργία ενός μικροελεγκτή πιο προσιτού στην χρήση του (Ψυχάρης & Καλοβρέκτης, 2018).

Σύμφωνα με τους δημιουργούς του το Arduino είναι μια *ανοιχτού κώδικα πλατφόρμα «πρωτοτυποποίησης» ηλεκτρονικών κυκλωμάτων* βασισμένη σε ευέλικτο και εύκολο στην χρήση hardware και software και προορίζεται για οποιονδήποτε έχει λίγη ή και καθόλου προγραμματιστική εμπειρία, στοιχειώδεις γνώσεις ηλεκτρονικών και ενδιαφέρεται να δημιουργήσει διαδραστικά projects. Διευκολύνει επίσης την διασύνδεση των περιφερειακών στοιχείων που απαιτούνται κάθε φορά αφού παρέχει έτοιμες μονάδες που διασυνδέονται είτε πάνω στην πλατφόρμα με την μορφή κάρτας είτε με την χρήση καλωδίων πάνω σε ένα breadboard, μέθοδος η οποία χρησιμοποιείται στα πλαίσια της παρούσας εργασίας. Κάθε περιφερειακή μονάδα του Arduino παρέχει το δικό της datasheet για τον τρόπο διασύνδεσης.

Ανάλογα με την εφαρμογή που θέλουμε να δημιουργήσουμε, μπορούμε να επιλέξουμε από ένα σύνολο 4 μονάδων Arduino με διαφοροποιημένα τεχνικά χαρακτηριστικά όπως φαίνεται στον παρακάτω πίνακα:

Μονάδα	Μικροελεγκτής	Τάση Λειτουργίας	Τάση Εισόδου	Ψηφιακές εισοδοί/έξοδοι	Είσοδοι PWM	Χρονισμός κροεπέξεργαστή (MHz)	Χωρητικότητα μνήμης flash (KB)
Arduino UNO	ATMega 328p	5V	7-12V	14	6	16	32
Arduino Leonardo	ATMega 32u4	5V	7-12V	20	7	16	32
Arduino Nano	ATMega 168 ή ATMega328	5V	7-12V	14	6	16	16 ή 32
Arduino Mega	ATMega 28	5V	7-12V	54	15	16	128

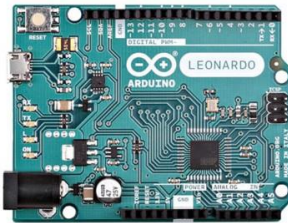
Πίνακας 2. Οι πλατφόρμες Arduino



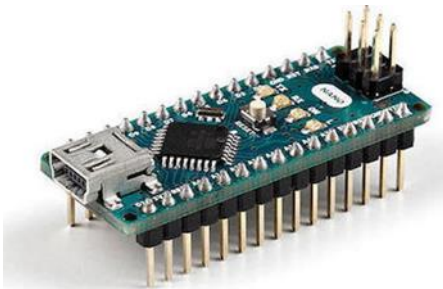
Εικόνα 3. Το Arduino Uno



Εικόνα 4. Το Arduino MEGA



Εικόνα 5. Το Arduino Leonardo



Εικόνα 6. Το Arduino NANO

Φυσικά εκτός από τις παραπάνω μονάδες Arduino που είναι κατάλληλες για εκπαιδευτική χρήση το Arduino παρέχει αρκετές ακόμα μονάδες κατάλληλες για πιο πολύπλοκες εφαρμογές, όπως τα Arduino Due και Arduino Mega 2560 και άλλα ενώ παρέχει και ειδικές μονάδες για εφαρμογές Internet of Things όπως το Arduino Nano 33 IoT και

άλλα. Στα πλαίσια της παρούσας εργασίας έχει επιλεγεί η πλατφόρμα Arduinoαφού εμπεριέχεται στο πακέτο Arduino starter kit. Το συγκεκριμένο kit μαζί με τους αισθητήρες που χρησιμοποιούνται για τις δραστηριότητες είναι προσιτό οικονομικά με κόστος κάτω από 40 ευρώ κάτι που αποτελεί το βασικό κριτήριο για να μπορεί να αποκτηθεί από τις σχολικές μονάδες.

#### 4.1.2 Τεχνικά χαρακτηριστικά του Arduino Uno

Όπως αναφέρθηκε παραπάνω για την παρούσα εργασία επιλέχθηκε το Arduino Uno λόγω του προσιτού του κόστους και της ευκολίας προγραμματισμού του από αρχάριους χρήστες και μαθητές μικρής ηλικίας.

Σύμφωνα με την επίσημη ιστοσελίδα του, το Arduino Uno είναι μια πλατφόρμα ανοιχτού κώδικα το οποίο είναι στημένο πάνω σε μια πλακέτα η οποία ενσωματώνει τον μικροεπεξεργαστή ATmega328p της Atmel. Έχει 14 ψηφιακούς ακροδέκτες οι οποίοι μπορούν να χρησιμοποιηθούν σαν είσοδοι/έξοδοι ενώ 6 από αυτές μπορούν να χρησιμοποιηθούν ως έξοδοι PWM. Οι μονάδες αυτές χωρίζονται σε ψηφιακές και αναλογικές. Επιπλέον υπάρχει μεγάλη πληθώρα περιφερειακών συσκευών συμβατών με την πλατφόρμα του Arduino. Κάποιες από αυτές είναι: αισθητήρες θερμοκρασίας, υγρασίας απόστασης και άλλοι.

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

Εικόνα 7. Τεχνικά χαρακτηριστικά Arduino UNO

Για την τροφοδοσία του, το Arduino Uno παρέχει δύο επιλογές. Η πρώτη μέσω της θύρας USB για διασύνδεση με υπολογιστή και η δεύτερη με εξωτερική τροφοδοσία μέσω της υποδοχής 2,1mm που βρίσκεται στην κάτω αριστερή πλευρά της πλακέτας. Η εξωτερική τροφοδοσία θα πρέπει να είναι τάσης 7 έως 12V. Οι ακροδέκτες τροφοδοσίες παρουσιάζονται στην πλακέτα με τις παρακάτω ενδείξεις:

- *Vin*: Από τον συγκεκριμένο ακροδέκτη γίνεται η τροφοδοσίας τάσης όταν χρησιμοποιείται εξωτερική πηγή ενέργειας.
- *5V*: Ο συγκεκριμένος ακροδέκτης παρέχει τάση ίση με 5V.
- *3.3V*: Ο ακροδέκτης με αυτή την ένδειξη δίνει τάση 3.3V η οποία παράγεται από το ολοκληρωμένο FTDI.
- *GND*: Ακροδέκτης γείωσης.

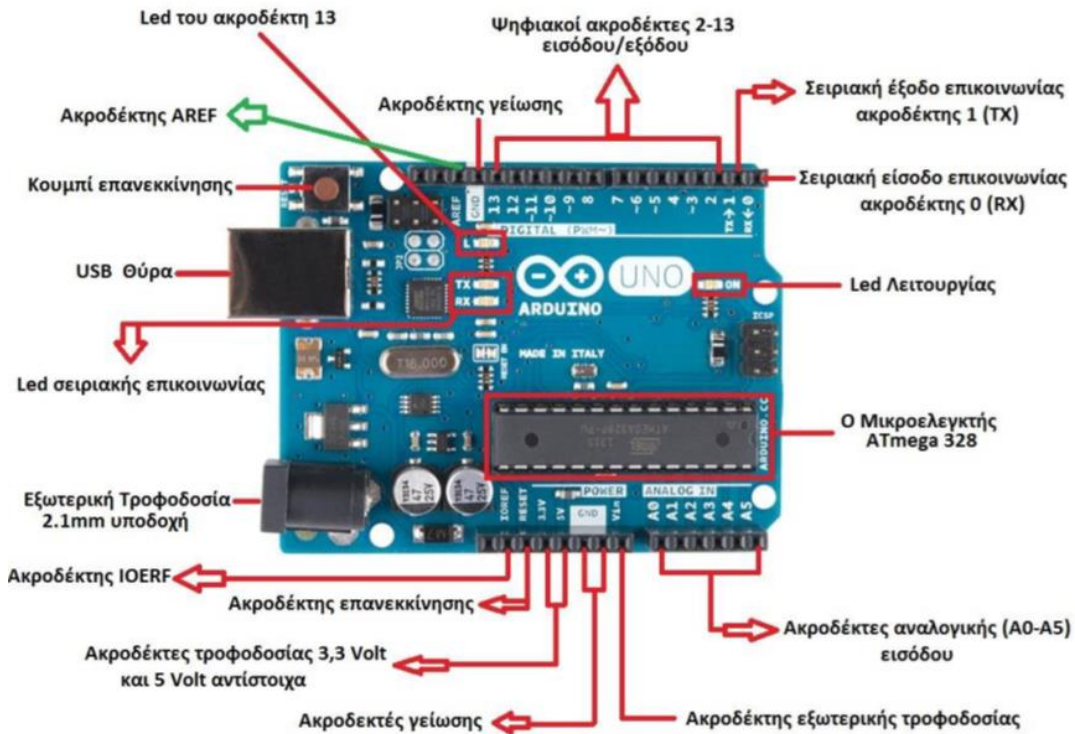
Περνώντας στο κομμάτι της μνήμης το Arduino Uno παρέχει τρία διαφορετικά είδη μνήμης:

- **Μνήμη τύπου flash χωρητικότητας 32KB**: Από τα 32KB τα 2KB χρησιμοποιούνται ήδη για το firmware του Arduino το οποίο έχει εγκαταστήσει ο κατασκευαστής. Στα υπόλοιπα 30KB αποθηκεύονται τα προγράμματα που γράφονται για το Arduino Uno (sketches) αφού πρώτα μεταγλωττιστούν. Η εγγραφή της μνήμης flash παραμένουν και αφού σταματήσει η τροφοδοσία της πλακέτας.
- **Μνήμη SRAM (static random access memory) χωρητικότητας 2KB**: Είναι η ωφέλιμη μνήμη η οποία χρησιμοποιείται από τα προγράμματα για την αποθήκευση πινάκων, μεταβλητών κλπ. Η συγκεκριμένη μνήμη χάνει τα δεδομένα της με την παύση τροφοδοσίας ρεύματος ή σε περίπτωση επανεκκίνησης.
- **Μνήμη EEPROM χωρητικότητας 1KB**: Μπορεί να χρησιμοποιηθεί για την εγγραφή δεδομένων από τα προγράμματα και σε αντίθεση με την SRAM δεν χάνει τα δεδομένα της με την παύση της τροφοδοσίας.

Όπως αναφέρθηκε παραπάνω το Arduino Uno έχει συνολικά 20 ακροδέκτες εισόδου/εξόδου από τους οποίους οι 14 είναι ψηφιακοί και οι 6 αναλογικοί. Κάποιοι από αυτούς έχουν επιπλέον ενδείξεις δίπλα τους αφού επιτελούν και κάποιες πιο συγκεκριμένες λειτουργίες για την πλατφόρμα.

- Ακροδέκτες 0 (RX) και 1 (TX): Χρησιμοποιούνται για την λήψη και αποστολή αντίστοιχα δεδομένων όταν χρησιμοποιείται από κάποιο πρόγραμμα η σειριακή θύρα. Συνεπώς όταν χρησιμοποιείται το σειριακό interface η πλατφόρμα έχει 2 λιγότερες ψηφιακές εισόδους/εξόδους.
- Ακροδέκτες 3, 5, 6, 9, 10 και 11: Οι ακροδέκτες αυτοί έχουν δίπλα τους την ένδειξη PWM κάτι που σημαίνει ότι μπορούν να λειτουργήσουν ως «ψευδοαναλογικοί» αφού ενσωματώνουν την λειτουργία Pulse Width Modulation.
- Οι ακροδέκτες 0 έως 5 που έχουν δίπλα τους την ένδειξη «ANALOG IN» είναι οι αναλογικοί ακροδέκτες του Arduino Uno. Μπορούν να δώσουν 1024 διαφορετικές τιμές και λαμβάνουν τιμές τάσης από 0 έως 5V

Όλοι οι 14 ψηφιακοί ακροδέκτες του Arduino μπορούν να χρησιμοποιηθούν είτε ως είσοδοι είτε ως έξοδοι ανάλογα με τον τρόπο που θα προγραμματιστούν από τον χρήστη. Όλοι λειτουργούν σε τάση 5V συνεπώς μπορούν να παρέχουν ή να λάβουν ρεύμα έντασης έως 40mA. Αναλυτικά όλα τα στοιχεία της πλακέτας Arduino φαίνονται στην παρακάτω εικόνα:



Εικόνα 8. Τα στοιχεία της πλατφόρμας Arduino UNO

## 4.2 Το περιβάλλον προγραμματισμού Pictoblox

### 4.2.1 Τι είναι το Pictoblox

Το λογισμικό Pictoblox της εταιρείας STEMedia είναι ένα γραφικό περιβάλλον προγραμματισμού, το οποίο είναι βασισμένο στην τρίτη έκδοση του Scratch και είναι διαθέσιμο για Windows, IOS, Android και Linux. Το Scratch είναι ένα εκπαιδευτικό περιβάλλον προγραμματισμού το οποίο αναπτύχθηκε από το Lifelong Kindergarten Group του MIT Media Lab με τη συμβολή των: National Science Foundation, Microsoft, Intel Foundation, Nokia και Media Lab. Θα μπορούσε να χαρακτηριστεί ως μια διερμηνεύσιμη δυναμική γλώσσα οπτικού προγραμματισμού (interpreted dynamic visual programming language) η οποία έχει υλοποιηθεί με την αντικειμενοστραφή γλώσσα Squeak. Το γραφικό περιβάλλον που διαθέτει το Scratch καθιστά την διαδικασία του προγραμματισμού πιο οικεία και ευχάριστη και έχει αναδειχθεί στο πιο δημοφιλές εργαλείο μάθησης προγραμματισμού για παιδιά από 8 ετών και άνω αλλά και για εφήβους και αρχάριους προγραμματιστές. Βρίσκεται πλέον στην τρίτη έκδοση του ενώ υπάρχουν και πολλές παραλλαγές του. Κάποιες

από αυτές είναι το Scratch junior για παιδιά κάτω των 7 ετών το S4A (Scratch for Arduino) αλλά και το Pictoblox που αναλύεται στην παρούσα εργασία.

Το Pictoblox, όπως και το Scratch, επικεντρώνεται στις θεμελιώδεις έννοιες προγραμματισμού και παρέχει ένα σύνολο βασικών τεχνικών προγραμματισμού χωρισμένων σε κατηγορίες. Έτσι έχουμε στην διάθεσή μας εκτός από τα μεμονωμένα μπλοκ εντολών δομές επιλογής, επανάληψης, ελέγχου και την δυνατότητα δημιουργίας μεταβλητών. Υποστηρίζει την επικοινωνία παράλληλων διεργασιών μέσω μετάδοσης μηνυμάτων ενώ από την άλλη πλευρά δεν υποστηρίζονται οι μαθηματικές συναρτήσεις και η διαχείριση αρχείων δεδομένων. Επιπλέον, το Pictoblox διαθέτει τις απαραίτητες επεκτάσεις για την δημιουργία έργων Μηχανικής Μάθησης και Τεχνητής νοημοσύνης. Το πιο βασικό του χαρακτηριστικό, ωστόσο, είναι η υποστήριξη για τα πιο γνωστά εργαλεία εκπαιδευτικής ρομποτικής όπως το Lego Minestorms, Lego Boost, Lego WeDo αλλά και τις πλατφόρμες Raspberry Pi και Arduino.

Η φιλοσοφία του Pictoblox και όλων των Scratch-based λογισμικών είναι να παρέχει στον μαθητή-χρήστη την ευκολία χρήσης με την διαδικασία του «σύρω και αφήνω» (drag and drop). Έτσι ο χρήστης δημιουργεί τον δικό του κώδικα χωρίς πρότερες γνώσεις κάποιας γλώσσας προγραμματισμού απλά σύροντας και αφήνοντας πλακίδια εντολών τα οποία ενώνονται με την μορφή παζλ για την δημιουργία του προγράμματος. Με τον τρόπο αυτό το παζλ-πρόγραμμα μπορεί ανά πάσα στιγμή να τροποποιηθεί και να δοκιμαστεί.

Το Pictoblox δίνει την δυνατότητα δημιουργίας αλληλεπιδραστικών ιστοριών, παιχνιδιών, έργων ρομποτικής και υποστηρίζει τον εμπλουτισμό τους με ήχους, γραφικά και κινούμενα σχέδια τα οποία είτε παρέχονται από το ίδιο το Pictoblox είτε μπορούν να εισαχθούν ως εξωτερικά αρχεία από την χρήστη. Οποιοδήποτε εξωτερικό αρχείο μπορεί να γίνει μέρος ενός έργου του Pictoblox.

Η παρούσα έκδοση του Pictoblox, σε αντίθεση με το Scratch, δεν υποστηρίζει τον Online διαμοιρασμό κάθε έργου και την αναζήτηση των έργων των άλλων χρηστών. Παρ' όλα αυτά με τον διαμοιρασμό του αρχείου .sb3 οκάθε χρήστης μπορεί να δει όχι μόνο το τελικό αποτέλεσμα αλλά και τον κώδικα του έργου και να τον τροποποιήσει όπως επιθυμεί.

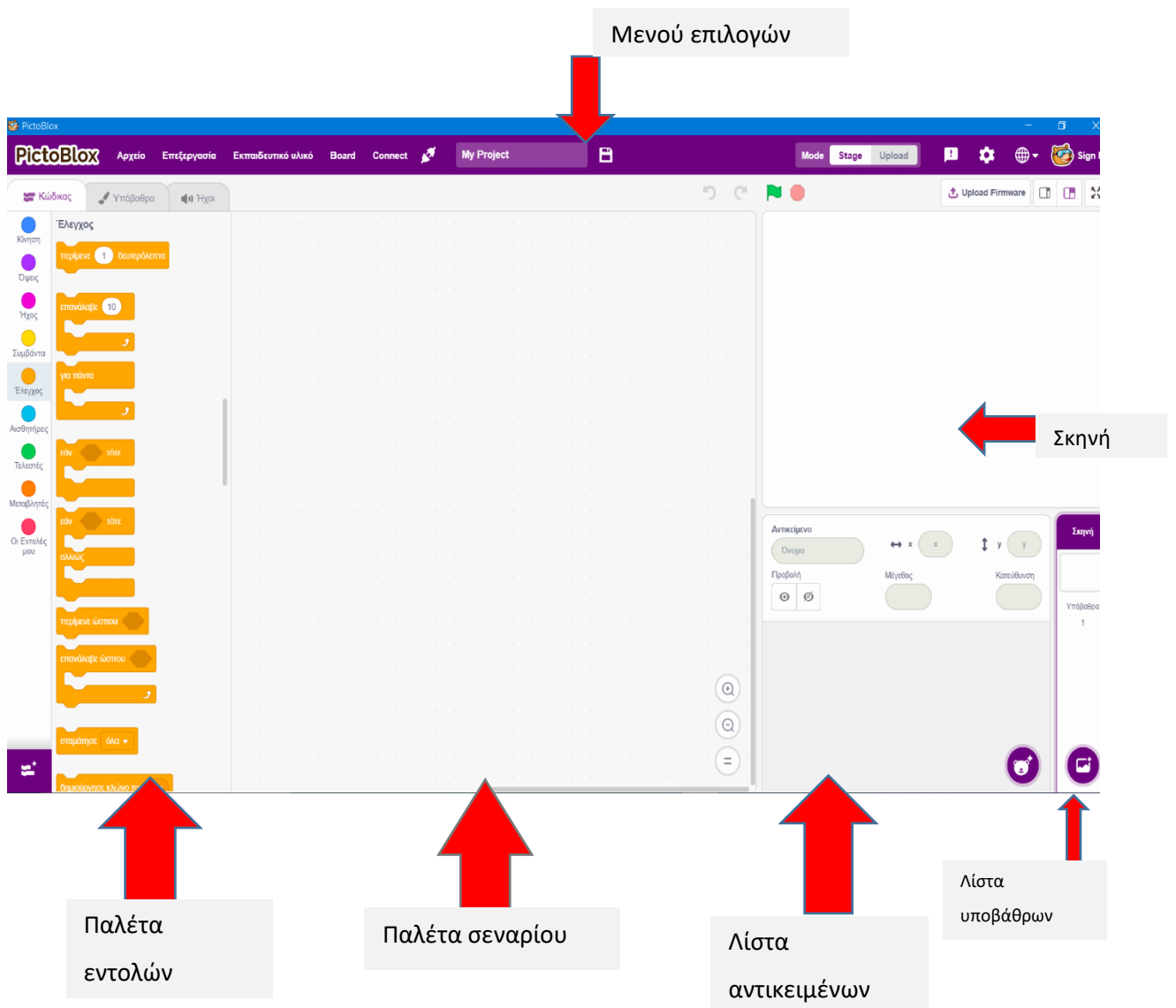


Η φιλοσοφία του Scratch ακολουθείται και από το Pictoblox σύμφωνα με την οποία ένα έργο που έχει αναπτυχθεί από κάποιον χρήστη θα πρέπει να μπορεί να χρησιμοποιηθεί και να τροποποιηθεί από κάποιον άλλον με σκοπό είτε την βελτίωση του είτε την δημιουργία κάποιου νέου έργου. Επιπλέον το Pictoblox προσφέρει την δυνατότητα καταγραφής της οθόνης ώστε να γίνεται επίδειξη και διαμοιρασμός του αποτελέσματος του προγραμματισμού. Μία ακόμη σημαντική δυνατότητα που προσφέρει είναι η παροχή μαζικών αναγνωριστικών χρήστη (user IDs) μετά από επικοινωνία με την εταιρεία. Η δυνατότητα αυτή είναι κομβική για την χρήση και διδασκαλία του Pictoblox σε μια σχολική τάξη καθώς αποφεύγεται η χρονοβόρα διαδικασία δημιουργίας μεμονωμένων λογαριασμών με διαφορετικά e-mails αλλά και τα ζητήματα προσωπικών δεδομένων των μαθητών που ανακύπτουν. Τέλος, το Pictoblox προσφέρει δωρεάν μεγάλη ποικιλία εκπαιδευτικού υλικού για κάθε τομέα που υποστηρίζει ενώ και το τμήμα υποστήριξης της STEMpedia προσφέρει πολύ γρήγορη απόκριση και είναι ιδιαίτερα βοηθητικό σε κάθε ζήτημα που προκύπτει.

Τα χαρακτηριστικά που αναφέρθηκαν παραπάνω σε συνδυασμό με την προυπάρχουσα διάδοση της τρίτης έκδοσης του Scratch μεταξύ των μαθητών οδήγησαν στην επιλογή του Pictoblox για την παρούσα εργασία, έναντι των υπολοίπων εκπαιδευτικών εργαλείων προγραμματισμού.

#### **4.2.2 Το γραφικό περιβάλλον του Pictoblox**


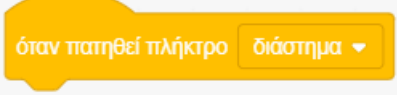
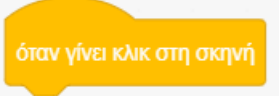
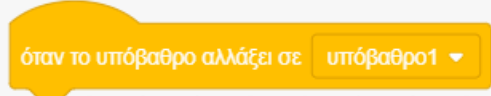

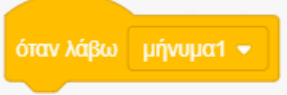

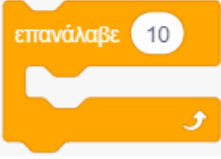
Αφού γίνει η εύκολη διαδικασία της δωρεάν λήψης από το site της STEMpedia και η εγκατάσταση στον υπολογιστή μας είμαστε έτοιμοι να μπούμε στην εφαρμογή και να περιηγηθούμε στο γραφικό της περιβάλλον και στις επιλογές που προσφέρει. Το περιβάλλον εργασίας του Pictoblox χωρίζεται σε έξι περιοχές όπως φαίνεται στην παρακάτω εικόνα:




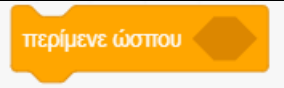






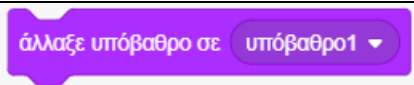
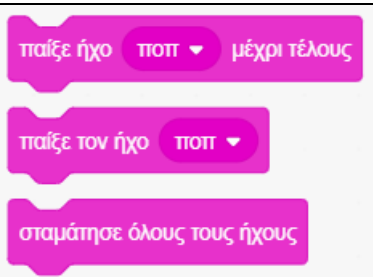
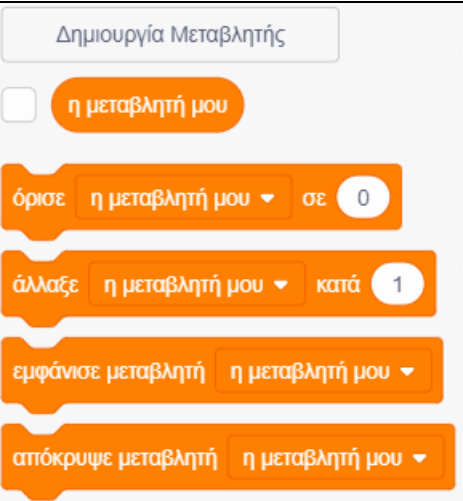



Εικόνα 9. Το περιβάλλον του PictoBlox



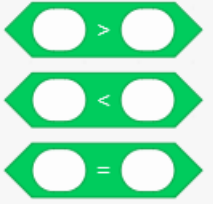




- Μενού επιλογών:** Μέσω του μενού επιλογών γίνεται η περιήγηση στις βασικές λειτουργίες της εφαρμογής όπως η αποθήκευση του έργου μας, το άνοιγμα κάποιου ήδη υπάρχοντος για επεξεργασία και επιλογές όπως αναίρεση της τελευταίας ενέργειας και η επαναφορά κάποιου αντικειμένου. Ακόμη μέσω της επιλογής *Board* γίνεται η επιλογή της πλατφόρμας ρομποτικής που θέλουμε να προγραμματίσουμε με το PictoBlox και στην συνέχεια μέσω της επιλογής *connect* επιλέγουμε την θύρα στην οποία θα την συνδέσουμε. Ακόμη μπορούμε να δώσουμε όνομα στο έργο μας, να κάνουμε είσοδο στον Pictoblox λογαριασμό μας και να επιλέξουμε την γλώσσα της εφαρμογής.

- **Παλέτα Σεναρίου:** Στην συγκεκριμένη περιοχή δημιουργείται κάθε σενάριο του Pictoblox με την μέθοδο του drag-and-drop των πλακιδίων της παλέτας εντολών. Κάθε αντικείμενο του εκάστοτε έργου έχει το δικό του σενάριο και πρέπει να επιλεγεί πρώτα από την λίστα αντικειμένων ή από την επιλογή «Board» αν πρόκειται για πλατφόρμα ρομποτικής.
- **Λίστα αντικειμένων:** Εδώ βλέπουμε τα αντικείμενα που συμμετέχουν στο έργο μας. Μπορούμε επίσης να προσθέσουμε καινούρια είτε από αυτά που παρέχει η εφαρμογή είτε κάποιο εξωτερικό αρχείο να το εισάγουμε ως αντικείμενο και να το προγραμματίσουμε. Ακόμη μπορούμε να αλλάξουμε τις βασικές παραμέτρους για κάθε αντικείμενο όπως η θέση, η αρχική τους κατεύθυνση το μέγεθος ή το όνομά τους,
- **Λίστα υποβάθρων(backgrounds):** Στο πλαίσιο αυτό υπάρχουν τα υπόβαθρα που συμμετέχουν στο έργο μας καθώς και η επιλογή να προσθέσουμε εξωτερικά αρχεία ως υπόβαθρα στο έργο μας. Η επεξεργασία γίνεται από την καρτέλα «Υπόβαθρα» στην παλέτα εντολών.
- **Σκηνή:** Εδώ εκτυλίσσεται κάθε πρόγραμμα ή διαδραστικό έργο που δημιουργούμε στο Pictoblox. Όταν προγραμματίζουμε με το PictoBlox μια εξωτερική πλατφόρμα όπως το Arduino, η σκηνή μπορεί να λειτουργήσει αλληλεπιδραστικά με αυτό παρόλο που δεν απεικονίζεται άμεσα κάτι σε αυτή.
- **Παλέτα εντολών:** Είναι η περιοχή στην οποία βρίσκονται τα πλακίδια (blocks) εντολών του Pictoblox χωρισμένα σε κατηγορίες που αντιπροσωπεύονται με διαφορετικό χρώμα η κάθε μια ώστε να βοηθούν στην απομνημόνευσή τους από τους μαθητές. Αρχικά υπάρχουν μόνο οι βασικές κατηγορίες εντολών ωστόσο με την σύνδεση κάποιας εξωτερικής πλατφόρμας εμφανίζονται και οι πιο εξειδικευμένες κατηγορίες εντολών όπως θα δούμε παρακάτω. Ακόμη στην παλέτα εντολών υπάρχουν οι επιλογές για επεξεργασία των υποβάθρων και των ήχων. Στον παρακάτω πίνακα επεξηγείται αναλυτικά η λειτουργία της κάθε εντολής.

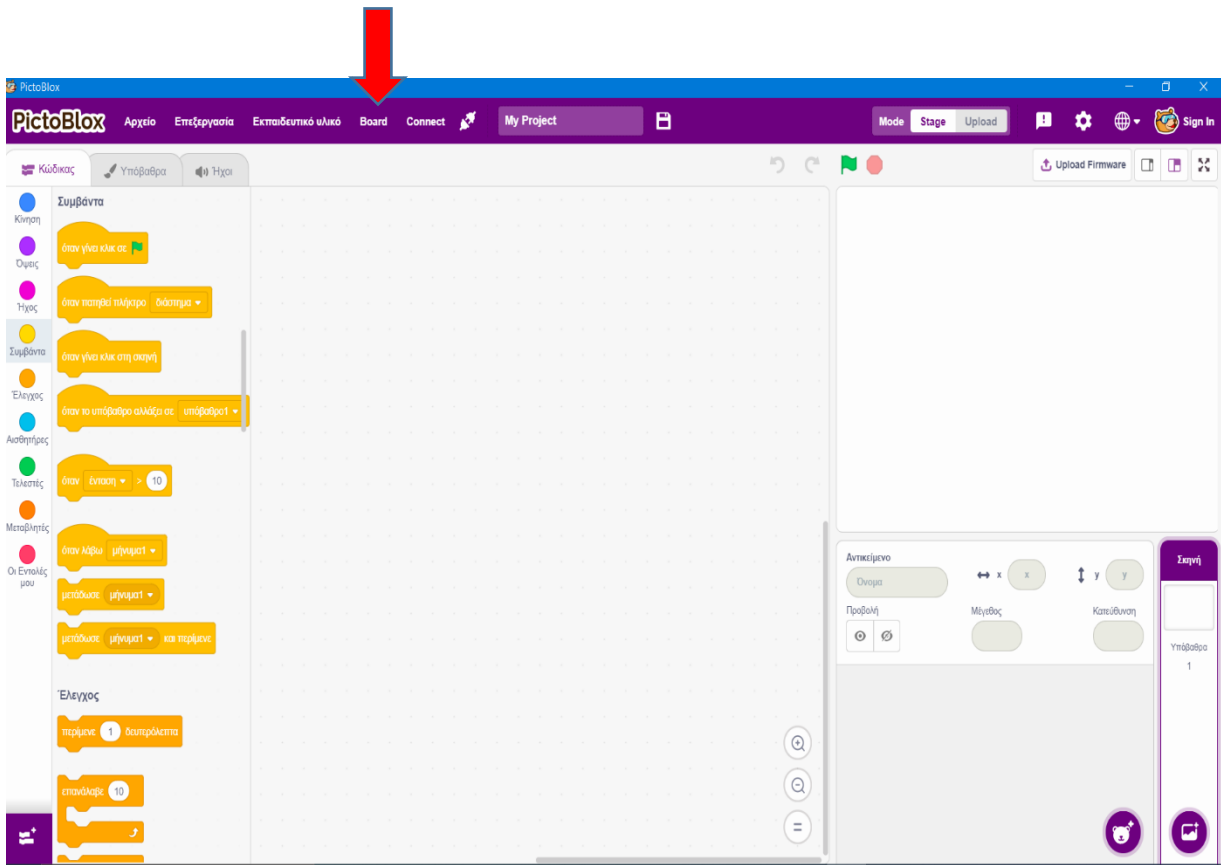
	<p>Ξεκινά ένα σενάριο με κλικ στο πράσινο σημαϊάκι.</p>
	<p>Ξεκινά ένα σενάριο με το πάτημα κάποιου πλήκτρου.</p>
	<p>Ξεκινά ένα σενάριο με κλικ πάνω στην σκηνή.</p>
	<p>Ξεκινά ένα σενάριο με την εμφάνιση συγκεκριμένου υποβάθρου.</p>
	<p>Ξεκινά ένα σενάριο όταν μια μεταβλητή ξεπεράσει ένα όριο.</p>
	<p>Ξεκινά ένα σενάριο όταν ληφθεί κάποιο μήνυμα.</p>
	<p>Δομή ελέγχου με την οποία ο κώδικας σταματάει για όσα δευτερόλεπτα ορίσουμε πριν προχωρήσει στην επόμενη εντολή.</p>
	<p>Δομή επανάληψης με την οποία ο κώδικας που τοποθετείται μέσα στο blockεκτελείται όσες φορές ορίσουμε στο πλαίσιο κειμένου.</p>

	<p>Δομή επανάληψης με την οποία ο κώδικας που τοποθετείται μέσα στο block εκτελείται συνεχώς.</p>
	<p>Δομή επιλογής η οποία ελέγχει αν είναι αληθής η συνθήκη που τοποθετείται στο πλαίσιο και αν ναι εκτελείται το κομμάτι κώδικα που τοποθετείται μέσα στο block.</p>
	<p>Δομή πολλαπλής επιλογής η οποία ελέγχει αν είναι αληθής η συνθήκη που τοποθετείται στο πλαίσιο και αν ναι εκτελείται ο κώδικας του πρώτου block αλλιώς εκτελείται το δεύτερο.</p>
	<p>Δομή ελέγχου με την οποία ο κώδικας συνεχίζεται μόνο όταν πραγματοποιηθεί το γεγονός που τοποθετείται στο πλαίσιο.</p>
	<p>Δομή επανάληψης με την οποία επαναλαμβάνεται ο κώδικας που τοποθετείται μέσα στο μπλοκ μέχρι να γίνει αληθής η συνθήκη που τοποθετείται στο πλαίσιο.</p>
	<p>Εντολή η οποία σταματά είτε ολόκληρο το έργο είτε το συγκεκριμένο σενάριο είτε όλα τα σενάρια του συγκεκριμένου αντικειμένου.</p>

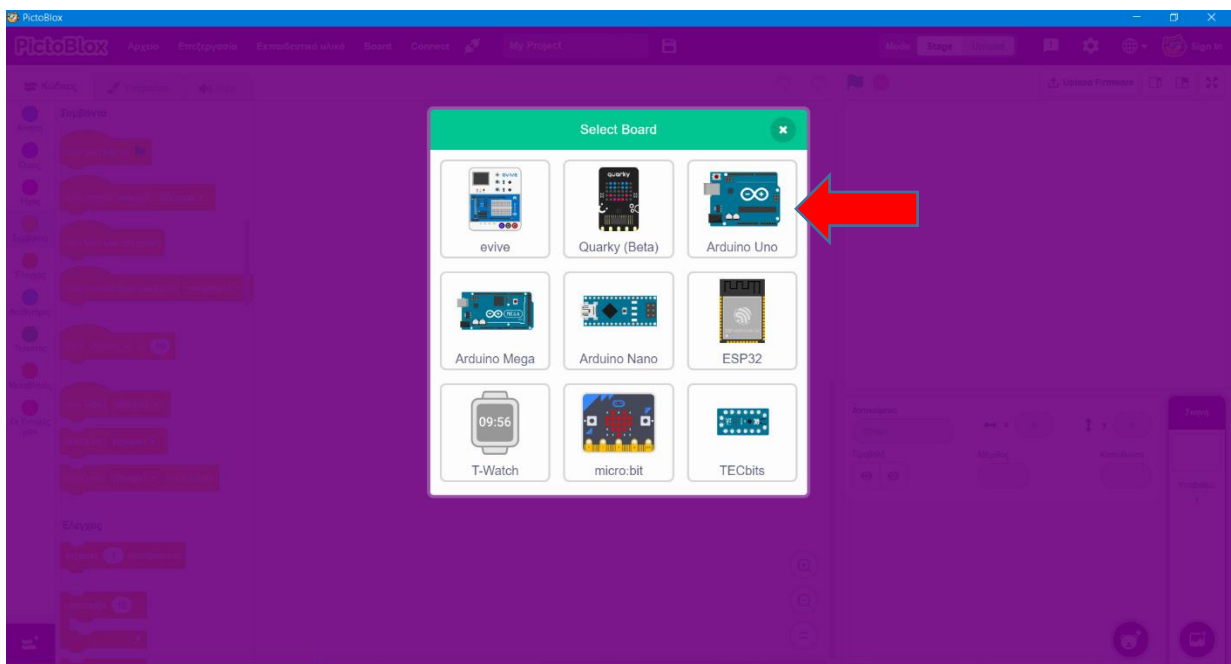
	<p>Εντολή η οποία ανιχνεύει αν πατηθεί κάποιο πλήκτρο του πληκτρολογίου.</p>
	<p>Εντολή η οποία ανιχνεύει αν πατηθεί κάποιο πλήκτρο του ποντικιού</p>
	<p>Εντολή η οποία αλλάζει το υπόβαθρο της σκηνής σε κάποιο άλλο.</p>
	<p>Εντολές αναπαραγωγής και παύσης κάποιου ήχου.</p>
	<p>Επιλογή δημιουργίας νέας μεταβλητής και εντολές διαχείρισής της.</p>
	<p>Μαθηματικός τελεστής πρόσθεσης.</p>
	<p>Μαθηματικός τελεστής αφαίρεσης.</p>
	<p>Μαθηματικός τελεστής πολλαπλασιασμού.</p>

	Μαθηματικός τελεστής διαίρεσης.
	Τελεστής τυχαίας επιλογής αριθμών μεταξύ καθορισμένου εύρους.
	Τελεστές αριθμητικής σύγκρισης, μεγαλύτερο, μικρότερο, ίσο.
	Λογικός τελεστής ΚΑΙ.
	Λογικός τελεστής Η.
	Λογικός τελεστής ΌΧΙ.
	Μαθηματικός τελεστής ο οποίος επιστρέφει το υπόλοιπο διαίρεσης δύο ακέραιων αριθμών.

Οι εντολές του παραπάνω πίνακα εμφανίζονται κατευθείαν με την έναρξη του PictoBlox ανεξαρτήτως του είδους του έργου που θα δημιουργήσουμε. Για να έχουμε στην διάθεση μας τις εντολές προγραμματισμού κάποιας εξωτερικής πλατφόρμας θα πρέπει αρχικά να την συνδέσουμε το Arduino Uno. Αρχικά από την γραμμή μενού επιλέγουμε το πεδίο «Board»

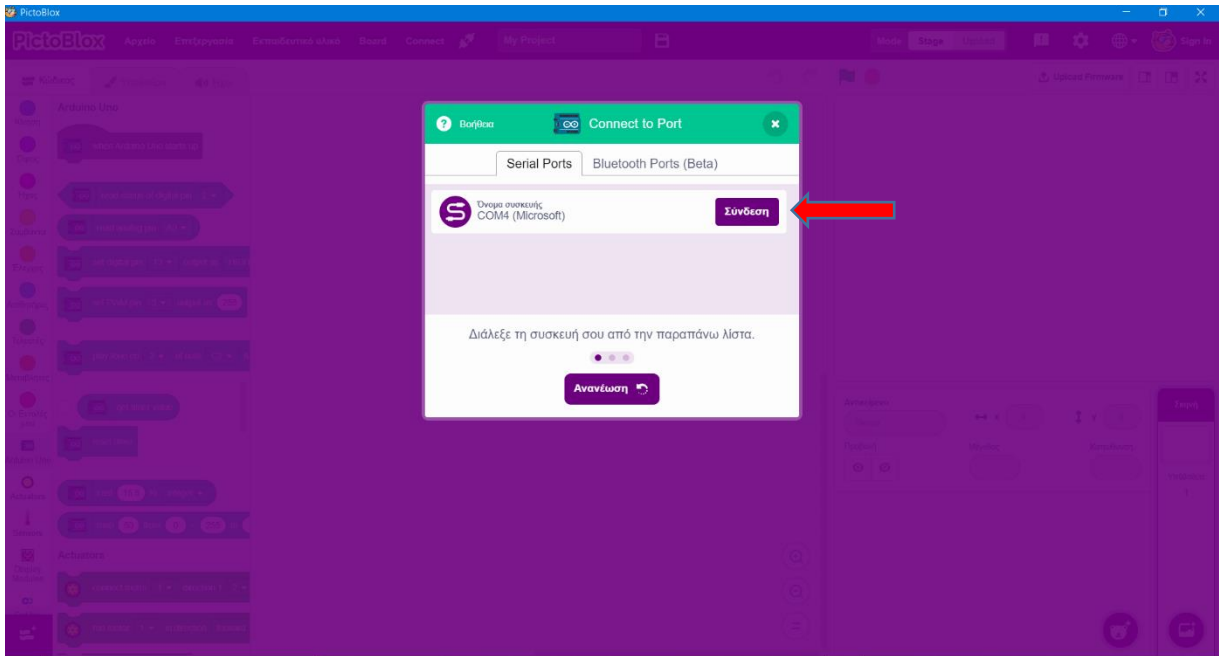


Στην συνέχεια από τις διαθέσιμες πλατφόρμες επιλέγουμε το ArduinoUno.

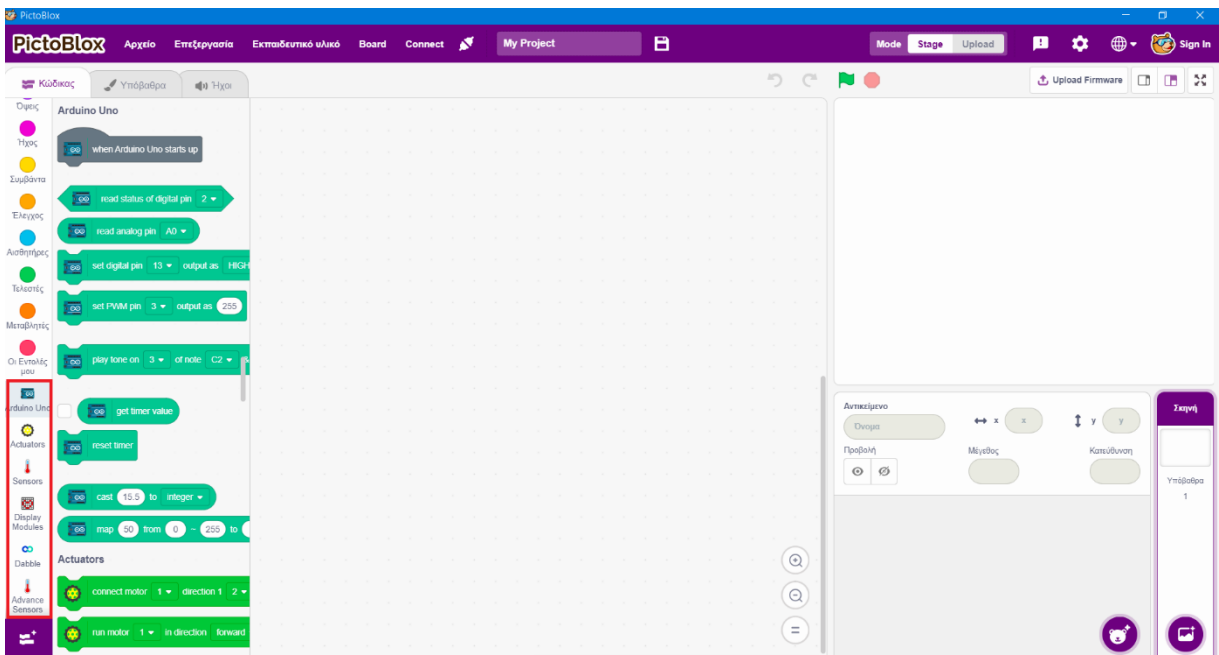




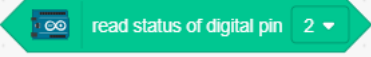

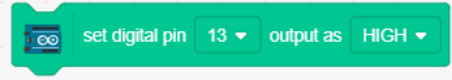
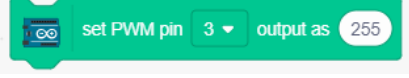
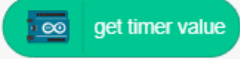

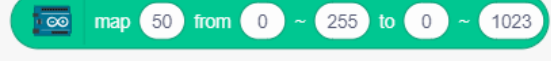

Τέλος, αφού η εφαρμογή εντοπίσει σε ποια θύρα του υπολογιστή μας είναι συνδεδεμένο το Arduino Uno επιλέγουμε το πεδίο «Σύνδεση».



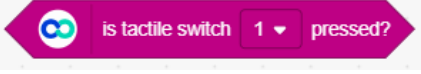
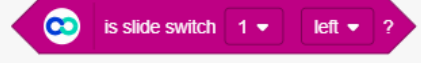



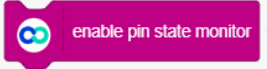



Μόλις πραγματοποιηθεί η σύνδεση, στην παλέτα εντολών έχουν προστεθεί οι επιπλέον κατηγορίες εντολών όπως φαίνεται στην παρακάτω εικόνα.



Το μειονέκτημα του PictoBloxείναι η μη μετάφραση στα ελληνικά των εντολών προγραμματισμού που αφορούν τις εξωτερικές πλατφόρμες εκπαιδευτικής ρομποτικής. Έτσι, προτείνεται να δοθούν μεταφρασμένες σε έντυπη μορφή στους μαθητές οι εντολές και η λειτουργία τους. Παρακάτω παρατίθεται ο πίνακας με τις βασικότερες εντολές και την λειτουργία που επιτελούν κατά τον προγραμματισμό του Arduino Uno.

	<p>Εντολή που διαβάζει την κατάσταση ενός από τους ψηφιακούς ακροδέκτες.</p>
	<p>Εντολή που διαβάζει την κατάσταση ενός από τους αναλογικούς ακροδέκτες.</p>
	<p>Θέτει την κατάσταση ενός ψηφιακού ακροδέκτη σε τιμή «HIGH» (περνάει ρεύμα) ή «LOW» (δεν περνάει ρεύμα)</p>
	<p>Θέτει τιμή εξόδου σε έναν ψηφιακό ακροδέκτη με λειτουργία PWM.</p>
	<p>Εντολή με την οποία λαμβάνουμε την ένδειξη του χρονιστή.</p>
	<p>Εντολή μηδενισμού του χρονιστή.</p>
	<p>Εντολή με την οποία αντιστοιχίζεται μια αναλογική τιμή σε κάποιον ακέραιο αριθμό.</p>
	<p>Διαβάζει την τιμή ενός αισθητήρα που έχει συνδεθεί σε κάποιον από τους αναλογικούς ακροδέκτες.</p>

	<p>Λαμβάνει την τιμή ενός αισθητήρα θερμοκρασία ή υγρασίας που έχει συνδεθεί σε ψηφιακό ακροδέκτη.</p>
	<p>Διαβάζει την τιμή ενός αισθητήρα που έχει συνδεθεί σε κάποιον από τους ψηφιακούς ακροδέκτες.</p>
	<p>Ανιχνεύει αν έχει πατηθεί διακόπτης τύπου «tactile».</p>
	<p>Ανιχνεύει την κατάσταση (αριστερά ή δεξιά) ενός διακόπτη τύπου switch.</p>
	<p>Παίρνει την τρέχουσα τιμή ενός ποτενσιόμετρου.</p>

	<p>Επιστρέφει την κατάσταση στην οποία βρίσκεται ένας ακροδέκτης.</p>
	
	<p>Εντολή εκκίνησης κινητήρα τύπου motor με επιλογή κατεύθυνσης και ταχύτητας κίνησης.</p>
	<p>Εντολή παύσης ή επανεκκίνησης κινητήρα τύπου motor.</p>

	<p>Θέτει έναν σερβοκινητήρα σε γωνία κίνησης.</p>
	<p>Ενεργοποιεί τον σερβοκινητήρα και ορίζει τον ψηφιακό ακροδέκτη που είναι συνδεδεμένος.</p>
	<p>Ενεργοποιεί τον κινητήρα τύπου και ορίζει τους ακροδέκτες που είναι συνδεδεμένος.</p>
	<p>Αρχικοποιεί μια οθόνη τύπου dotmatrix 8*8 και ορίζει τους ακροδέκτες που είναι συνδεδεμένη.</p>
	<p>Απεικονίζει ένα σχήμα φτιαγμένο από τις αντίστοιχες τελείες της οθόνης.</p>
	<p>Ενεργοποιεί ένα πεδίο της οθόνης.</p>

## **4.3 Φύλλα δραστηριοτήτων προγραμματισμού με Arduino**

### **4.3.1 Εισαγωγή και προετοιμασία**

Στο πλαίσιο της παρούσας εργασίας θα παρουσιαστούν 10 φύλλα εργασίας προγραμματισμού πάνω στην πλατφόρμα Arduino Uno. Το έναυσμα για την δημιουργία των συγκεκριμένων δραστηριοτήτων και κατ' επέκταση ολόκληρης της εργασίας δόθηκε από την ύπαρξη του απαραίτητου εξοπλισμού σε δύο Δημοτικά Σχολεία της Νέας Φιλαδέλφειας κατά το σχολικό έτος 2020-2021.

Πρώτος στόχος των συγκεκριμένων φύλλων εργασίας και της γενικότερης διδακτικής προσέγγισης είναι να διδαχτούν οι μαθητές τον προγραμματισμό με έναν τρόπο πιο διαδραστικό και πιο «χειροπιαστό» αφού θα βλέπουν μπροστά στα μάτια τους το αποτέλεσμα των προσπάθειών τους. Οι μαθητές εργάζονται σε ομάδες των τριών ή τεσσάρων ατόμων και μέσα από ένα συνεργατικό περιβάλλον εμπλέκονται ενεργητικά δραστηριότητες εκμάθησης βασικών εννοιών και δομών του προγραμματισμού και της ρομποτικής υλοποιώντας απλά έργα μέσω του Pictoblox.

Σαν δεύτερος στόχος στην παρούσα εργασία τέθηκε η εισαγωγή και εξοικείωση των μαθητών με μια πραγματική γλώσσα προγραμματισμού επιπλέον του Pictoblox. Έτσι στο τέλος κάθε φύλλου εργασίας οι μαθητές καλούνται να απαντήσουν σε απλά διαδραστικά quizμέσω των οποίων γίνεται η συσχέτιση ανάμεσα στα μπλοκ κώδικα του Pictoblox και σε πραγματικό κώδικα γραμμένο στην γλώσσα προγραμματισμού Javascript.

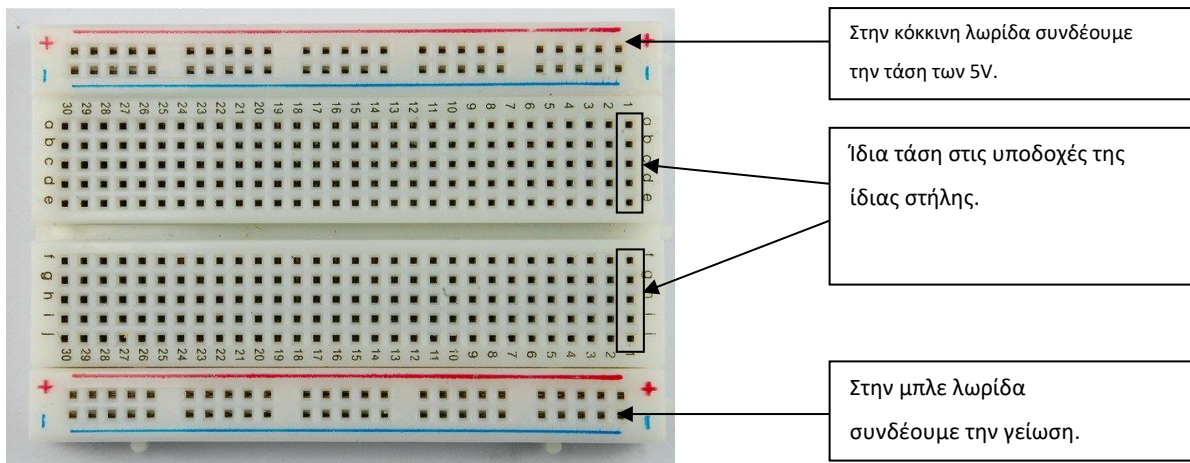
Σημειώνεται πως οι υλοποιήσεις των κυκλωμάτων του Arduino δίνονται έτοιμες στους μαθητές καθώς η διδασκαλία ηλεκτρονικών κυκλωμάτων ξεφεύγει από τα όρια της πρωτοβάθμιας εκπαίδευσης. Παρ' όλα αυτά γίνεται προσπάθεια να εξηγηθεί με απλά λόγια στους μαθητές η λειτουργία του κάθε εξαρτήματος. Οι αντιστάσεις παραδείγματος χάριν επεξηγούνται ως ένα εξάρτημα που το χρησιμοποιούμε για να μην περάσει περισσότερο ρεύμα απ' όσο χρειαζόμαστε.

Μέσα από την εργασία πάνω στα συγκεκριμένα φύλλα εργασίας εκτός της μάθησης του προγραμματισμού επιδιώκεται επίσης:

- Οι μαθητές να διαπιστώσουν τις δυνατότητες της πλατφόρμας Arduino Uno και να γίνει συζήτηση για τους μικροελεγκτές και τις εφαρμογές τους στην καθημερινότητα
- Να αναπτύξουν ομαδικό και συνεργατικό πνεύμα καθώς εμπλέκονται σε ομαδικά έργα και να βρίσκουν λύσεις σε τυχόν προβλήματα μέσα από ανταλλαγή απόψεων
- Να αποκτήσουν θετική στάση απέναντι στον προγραμματισμό αλλά και στην επιστήμη της Πληροφορικής γενικότερα.

Για την υλοποίηση των φύλλων εργασίας οι μαθητές εργάζονται, όπως προαναφέρθηκε, σε ομάδες των τριών ή τεσσάρων ατόμων. Οι ασκήσεις πραγματοποιούνται στο εργαστήριο πληροφορικής του σχολείου. Κάθε ομάδα έχει μπροστά της την πλατφόρμα του Arduino Uno συνδεδεμένη με υπολογιστή καθώς και την ηλεκτρονική διάταξη που απαιτείται από το κάθε φύλλο εργασίας. Ο υπεύθυνος εκπαιδευτικός πρέπει να έχει φροντίσει εκ των προτέρων να εγκατασταθούν τα απαραίτητα λογισμικά και τα προγράμματα οδήγησης για την επικοινωνία της πλακέτας με τον υπολογιστή.

Για την υλοποίηση των κυκλωμάτων προτείνεται να χρησιμοποιηθεί Breadboard. Το breadboard είναι μια πλαστική επιφάνεια με μικρές υποδοχές οι οποίες είναι συνδεδεμένες μεταξύ τους ανά πέντε. Σε αυτές τις υποδοχές συνδέονται οι ακροδέκτες των μονάδων του Arduino που απαρτίζουν το κύκλωμα. Όταν συνδεθεί ένα εξάρτημα σε μια υποδοχή, τότε είναι συνδεδεμένο σε σειρά με οτιδήποτε άλλο συνδεθεί στην ίδια **κάθετη** λωρίδα. Αυτό συμβαίνει λόγω των αγώγιμων μεταλλικών γραμμών που περιέχει. Σε κάθε λωρίδα του breadboard υπάρχουν 5 υποδοχές κάτι που σημαίνει ότι μπορούν να συνδεθούν μέχρι 5 εξαρτήματα σε κάθε μια και αυτό ισχύει ανεξαρτήτως του μεγέθους του breadboard.



Για την εισαγωγή στον προγραμματισμό «ενηλίκων» επιλέχθηκε στα πλαίσια της παρούσας εργασίας η γλώσσα Javascript. Ασφαλώς υπάρχουν και άλλες επιλογές αλλά η javascript είναι πιο απλή στην εκμάθηση της για μαθητές αυτής της ηλικίας. Επίσης είναι αντικειμενοστραφής γλώσσα κάτι που κάνει πιο ομαλή την μετάβαση από τα block-based περιβάλλοντα προγραμματισμού. Ο απευθείας προγραμματισμός της πλατφόρμας Arduino σε Javascript δεν είναι δυνατός οπότε πρέπει να γίνει η εγκατάσταση της κατάλληλης βιβλιοθήκης στον υπολογιστή μας. Αρχικά θα πρέπει να εγκατασταθεί στον υπολογιστή μας το επίσημο περιβάλλον προγραμματισμού του Arduino το Arduino IDE και το περιβάλλον προγραμματισμού Javascript Node.js. Στην συνέχεια θα πρέπει να ανεβάσουμε στην πλατφόρμα το πρωτόκολλο «StandardFirmdataPlus» που παρέχει το Arduino IDE. Αφού ανεβάσουμε το παραπάνω sketch πληκτρολογούμε σε περιβάλλον κονσόλας τις παρακάτω εντολές για την εγκατάσταση της βιβλιοθήκης «Johnny-Five».

```
npm init
npm install johnny-five
```

Για να δημιουργήσουμε μια νέα εφαρμογή σε javascript πληκτρολογούμε:

```
touch app.js
```

Για να εκτελέσουμε ένα πρόγραμμα Javascript που έχουμε γράψει σε κάποιον text editor αρκεί να πληκτρολογήσουμε σε περιβάλλον κονσόλας:

```
node app.js
```

Η εισαγωγή των μαθητών στην javascript θα πρέπει να γίνει με απλό και κατανοητό τρόπο χωρίς αναφορά σε πολύπλοκες δομές και σε αναλυτικούς κανόνες σύνταξης.

Προτείνεται να ακολουθηθεί η ίδια σειρά με την οποία γίνεται και η διδασκαλία του προγραμματισμού σε block-based προγραμματιστικό περιβάλλον. Με αυτό τον τρόπο από την δημιουργία αλγορίθμων θα περάσουμε στις επαναλήψεις και τις δομές ελέγχου για να καταλήξουμε στις μεταβλητές και τις συναρτήσεις η οποίες παίζουν κυρίαρχο στην javascript. Τα διαδικτυακά κουίζ του Kahoot μπορούν να προβάλλονται σε διαδραστικό πίνακα παράλληλα με την διδασκαλία κάθε έννοιας και αφού έχουν δημιουργηθεί τα προγράμματα στο PictoBlox.

#### **4.3.2 1<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΑΝΑΒΟΣΒΗΝΩΝΤΑΣ ΤΟ LED»**

##### **A. Διδακτικοί στόχοι**

- ✓ Εισαγωγή στην πλατφόρμα του Arduino και στον τρόπο διασύνδεσης των περιφερειακών.
- ✓ Εισαγωγή στον τρόπο προγραμματισμού του Arduino και την δημιουργία σεναρίων με το λογισμικό PictoBlox.
- ✓ Να κατανοήσουν τον τρόπο λειτουργίας των ψηφιακών pins.
- ✓ Να κατανοήσουν τον τρόπο λειτουργίας της δομή ελέγχου «περίμενε».
- ✓ Να μπορούν να αναβοσβήνουν το led με διαφορετική συχνότητα.

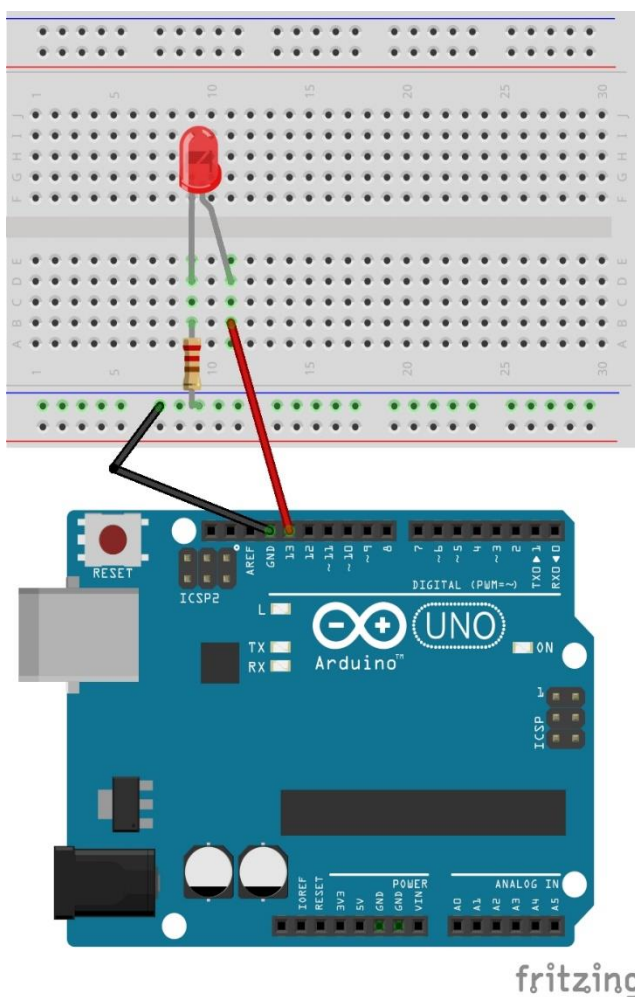
##### **B. Υλικά που θα χρειαστούν**

- Η πλακέτα Arduino Uno
- Breadboard
- Καλώδιο USB
- LED 5mm
- 1 αντίσταση 220 Ω
- 2 καλώδια





## Γ. Υλοποίηση διάταξης

Οι μαθητές έχουν μπροστά τους το παρακάτω κύκλωμα στο οποίο ένα κόκκινο led είναι συνδεδεμένο στην ψηφιακή έξοδο 13 του Arduino Uno.

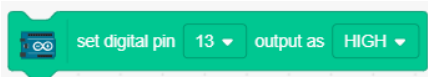


## Δ. Το πρόγραμμα

Οι μαθητές καλούνται να δημιουργήσουν το πρόγραμμα ώστε το led να αναβοσβήνει κάθε 1 δευτερόλεπτο. Κάθε σενάριο στο Pictoblox «πυροδοτείται» από κάποια εντολή-συμβάν. Για λόγους κατανόησης και συνέχειας μεταξύ των προγραμμάτων συστήνεται τα προγράμματα να ξεκινούν με την εντολή . Στην συνέχεια μπαίνουμε στο κύριο

σώμα του προγράμματος με την χρησιμοποιώντας την δομή επανάληψης . Με

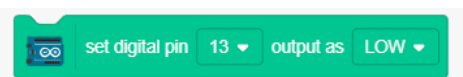
αυτό τον τρόπο οι εντολές που θα μπουν μέσα στο συγκεκριμένο μπλοκ θα εκτελούνται «για πάντα» δηλαδή για όσο τρέχει το πρόγραμμα μας. Είναι ευνόητο ότι όταν υπάρχει το συγκεκριμένο μπλοκ σε κάποιο πρόγραμμα του Pictoblox δεν μπορεί να ακολουθείται από κάποιο άλλο μπλοκ. Στην συνέχεια για να ανάψει το led θα πρέπει να θέσουμε την ψηφιακή έξοδο που είναι συνδεδεμένο στην τιμή «HIGH» με το ακόλουθο μπλοκ εντολής



. Στην συνέχεια θα πρέπει να εισάγουμε στο πρόγραμμα μας την καθυστέρηση του ενός δευτερολέπτου. Αυτό θα συμβεί χρησιμοποιώντας το



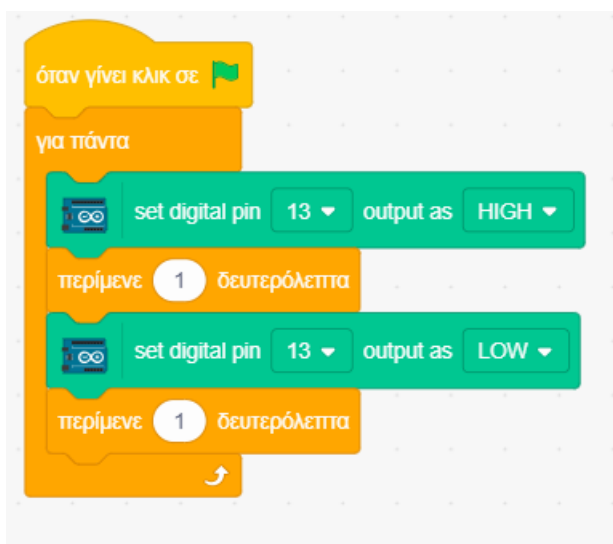
πλακίδιο από τις εντολές ελέγχου. Ακολούθως θέλουμε το led να σβήσει ξανά για 1 δευτερόλεπτο οπότε πρέπει να θέσουμε την ψηφιακή έξοδο του Arduino στην τιμή



«LOW» με το πλακίδιο και, τέλος, να εισάγουμε ξανά



το πλακίδιο της καθυστέρησης. Η ολοκληρωμένη μορφή του προγράμματος φαίνεται παρακάτω:



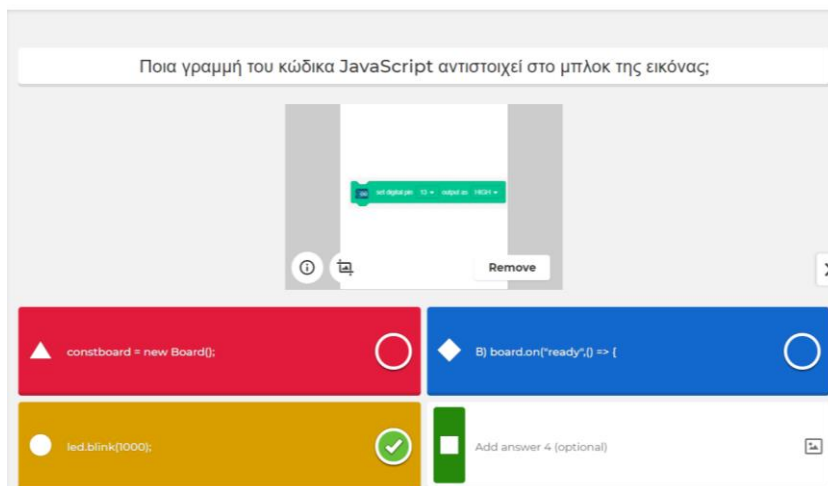
Εικόνα 10. Το πρόγραμμα του 1ου φύλλου εργασίας

## Ε. Συζήτηση του αποτελέσματος – μετατροπές

Αφού δημιουργήσουν το πρόγραμμα οι μαθητές καλούνται να περιγράψουν το αποτέλεσμα πάνω στην πλακέτα. Ακόμη μπορούν να κληθούν να πραγματοποιήσουν κάποιες μικρές μετατροπές όπως στον χρόνο αναμονής του led ή ακόμα και να αλλάξουν την ψηφιακή έξοδο πάνω στην οποία είναι συνδεδεμένο το led τόσο στο πρόγραμμα όσο και πάνω στο breadboard.

### ΣΤ. Απάντηση στο κουίζ

Στην συνέχεια οι μαθητές καλούνται να απαντήσουν το παρακάτω online quiz φτιαγμένο στην πλατφόρμα Kahootme ερωτήσεις πολλαπλής επιλογής που στόχο έχουν την συσχέτιση των μπλοκ του Pictoblox με τον κώδικα της Javascript: [Κουίζ 1ου Φύλλου εργασίας](#)



Εικόνα 11. Κουίζ 1ου Φύλλου Εργασίας

### 4.3.3 2<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΠΡΟΓΡΑΜΜΑΤΙΖΩ ΕΝΑΝ ΔΙΑΚΟΠΤΗ»

#### Α. Διδακτικοί στόχοι

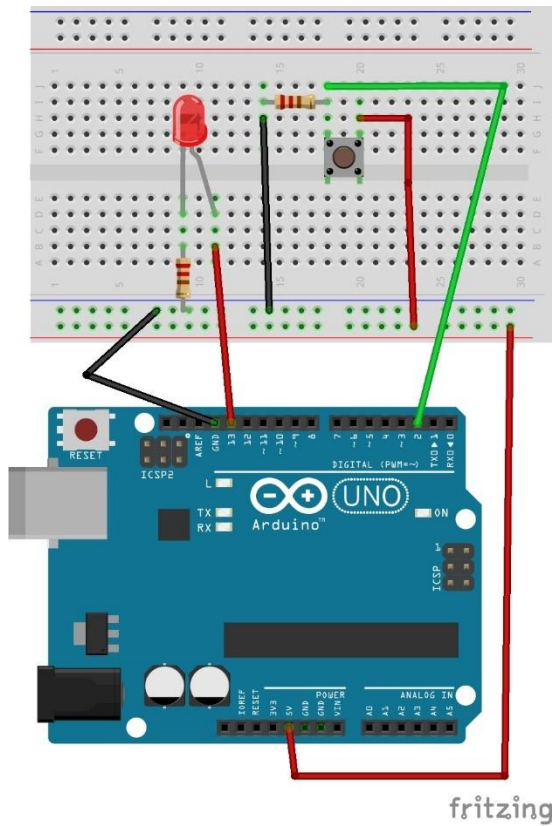
- ✓ Εισαγωγή στον τρόπο λειτουργίας ενός διακόπτη.
- ✓ Να κατανοήσουν την δομή ελέγχου «εάν.....τότε.....αλλιώς.....».
- ✓ Να μπορούν να προγραμματίσουν τον διακόπτη εάν συνδεθεί σε διαφορετικό pin.

## **B. Υλικά που θα χρειαστούν**

- Η πλακέτα Arduino Uno
- Breadboard
- Καλώδιο USB
- LED 5mm
- 1 αντίσταση 220 Ω
- 1 αντίσταση 10 kΩ
- 1 διακόπτης τύπου «Push Button»
- 6 καλώδια

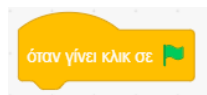
## **Γ. Υλοποίηση διάταξης**

Στο παρακάτω κύκλωμα επιπλέον του led που είναι συνδεδεμένος στην ψηφιακή έξοδο 13 του Arduino Uno, συνδέουμε έναν διακόπτη τύπου «push Button» μέσω του οποίου θα ανάβουμε και θα σβήνουμε το led.



#### Δ. Το πρόγραμμα

Όπως προαναφέρθηκε, επιλέγουμε να ξεκινάμε τα προγράμματά μας με την εντολή



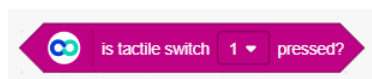
και να τοποθετούμε το κύριο σώμα του προγράμματος μέσα στο πλακίδιο



αέναης επανάληψης. Το Pictoblox σε αντίθεση με άλλα προγραμματιστικά περιβάλλοντα έχει ξεχωριστά πλακίδια για την λειτουργία εξαρτημάτων εισόδου όπως οι διακόπτες κάτι που δίνει μια επιπλέον ευκολία για τους μαθητές. Σε αυτή την άσκηση θα

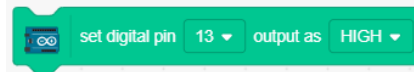


χρησιμοποιήσουμε το πλακίδιο ελέγχου. Θέλουμε λοιπόν αρχικά να ελέγχουμε εάν ο διακόπτης έχει πατηθεί συνεπώς στην πρώτη συνθήκη θα τοποθετήσουμε το πλακίδιο



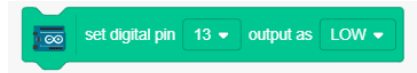
. Εάν η συγκεκριμένη συνθήκη είναι αληθής τότε η ψηφιακή

έξοδος 13 θα παίρνει την τιμή «HIGH»



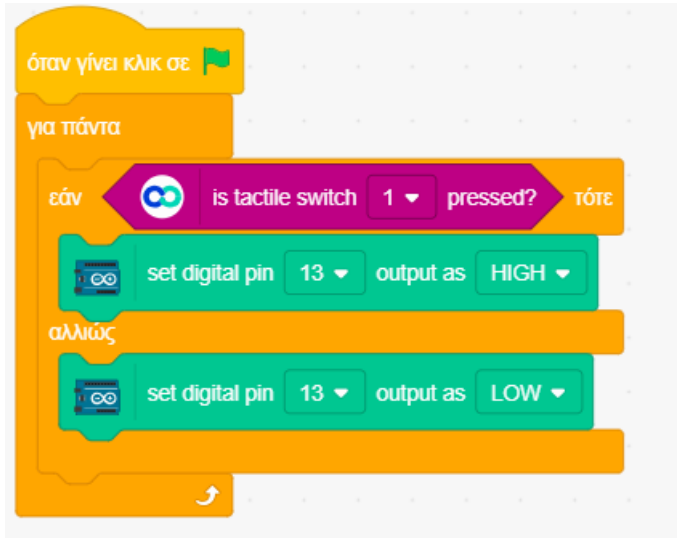
δηλαδή το led θα

ανάβει ενώ σε αντίθετη περίπτωση η ψηφιακή έξοδος 13 θα παίρνει την τιμή «LOW»



δηλαδή θα είναι σβηστό. Έτσι το ολοκληρωμένο σώμα του

προγράμματος θα είναι ως εξής:



Εικόνα 12. Το πρόγραμμα του 2ου φύλλου εργασίας

## Ε. Συζήτηση του αποτελέσματος – μετατροπές

Οι μαθητές καλούνται να περιγράψουν τι παρατηρούν με το πρόγραμμα που δημιούργησαν. Μπορεί να γίνει περαιτέρω συζήτηση για το πώς θα έπρεπε να διαμορφώσουμε το πρόγραμμα αν δεν υπήρχε στην διάθεσή μας αυτόσυιο το πλακίδιο του διακόπτη και έπρεπε εμείς να καθορίσουμε την ψηφιακή θύρα-είσοδο. Ακόμη μπορεί να ζητηθεί να συνδέσουν με προσοχή το led σε κάποια άλλη ψηφιακή έξοδο ώστε να εξοικειώνονται και με την δημιουργία διατάξεων.

## ΣΤ. Απάντηση στο κουίζ

Στην συνέχεια οι μαθητές απαντούν στο [online κουίζ δύο ερωτήσεων στο Kahoot](#). Στο πρώτο κουίζ πρέπει να εντοπίσουν στον κώδικα σε Javascript τα pins στα οποία είναι συνδεδεμένα τα στοιχεία του Arduino. Στο δεύτερο κουίζ εντοπίζουν τις συναρτήσεις με τις οποίες το led μας παραμένει αναμμένο.

Σύμφωνα με το παρακάτω κομμάτι κώδικα το led είναι συνδεδεμένο στο pin 13 και ο διακόπτης στο pin 8.

```
const led = new Led(13);  
  
board.on("ready", function() {  
  
  button = new five.Button({  
  
    board: board,  
  
    pin: 8,  
  
  });  
});
```

True  False

Ποιο ή ποια από τα παρακάτω κομμάτια κώδικα πιστεύεται ότι κρατάνε το led μας αναμμένο;

Find and insert media  
or drop an image here to upload

`button.on("down", function() {  
 led.on();  
});`

`button.on("hold", function() {  
 led.on();  
});`

`button.on("up", function() {  
 led.off();  
});`

Add answer 4 (optional)

Εικόνα 13. Κουίζ 2ου φύλλου εργασίας

#### 4.3.4 3<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΑΛΛΑΖΩ ΤΗΝ ΦΩΤΕΙΝΟΤΗΤΑ ΤΟΥ LED»

##### A. Διδακτικοί στόχοι

- ✓ Εισαγωγή στον τρόπο λειτουργίας ενός ποτενσιόμετρου.
- ✓ Να κατανοήσουν τον τρόπο λειτουργίας των αναλογικών pins.
- ✓ Να μπορούν να χρησιμοποιούν το πλακίδιο με την τιμή του ποτενσιόμετρου ως είσοδο για την τιμή της φωτεινότητας του led.

##### B. Υλικά που θα χρειαστούν

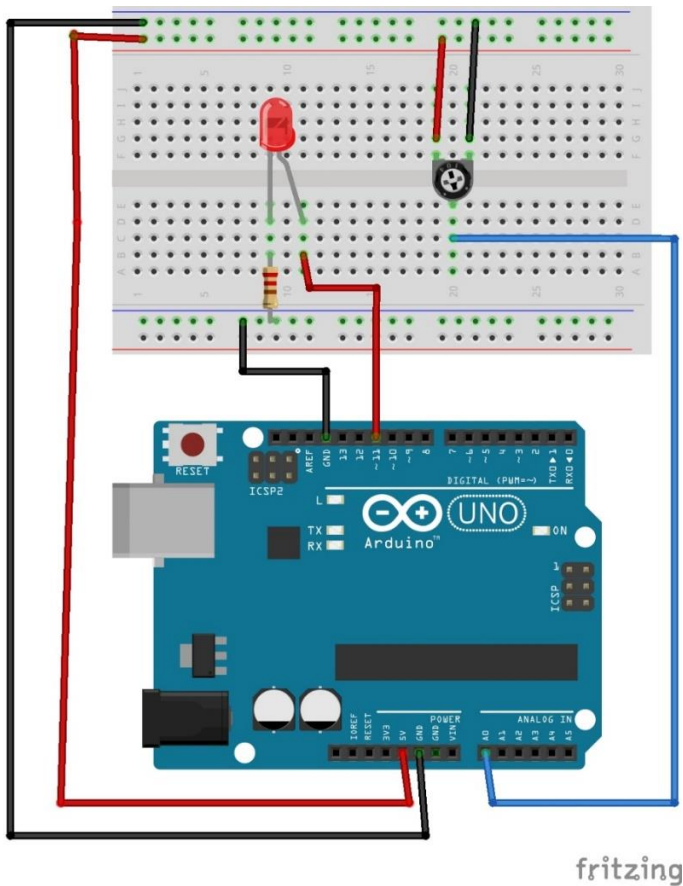
- Η πλακέτα Arduino Uno
- Καλώδιο USB
- Breadboard
- Led 5mm
- 1 αντίσταση 220Ω
- 1 ποτενσιόμετρο (10k Trimpot)
- 6 καλώδια

##### Γ. Υλοποίηση διάταξης

Οι μαθητές έχουν μπροστά τους το κύκλωμα της παρακάτω εικόνας. Σε αυτό, το led είναι συνδεδεμένο στην ψηφιακή έξοδο 11 του Arduino Uno ενώ το ποτενσιόμετρο συνδέεται στην αναλογική έξοδο A0 μέσω του μεσαίου του ακροδέκτη. Οι άλλοι δύο ακροδέκτες συνδέονται στην τροφοδοσία των 5V και στην γείωση αντίστοιχα. Το ποτενσιόμετρο είναι μια μεταβλητή αντίσταση η τιμή της οποίας μεταβάλλεται μέσω του μοχλού που υπάρχει στο πάνω μέρος του. Το συγκεκριμένο ποτενσιόμετρο είναι 10 κΩ κάτι που σημαίνει ότι μπορεί να δώσει τιμές από 0 Ω έως 10 ΚΩ. Γυρνώντας τον μοχλό αλλάζουμε την τιμή της αντίστασης και οι τιμές αυτές αποδίδουν τάση μεταξύ 0V και 5V. Αν χρησιμοποιούσαμε ένα βολτόμετρο θα βλέπαμε ότι τέρμα αριστερά η τάση είναι 0V ενώ τέρμα δεξιά 5V(ή το αντίθετο) έτσι στην αναλογική θύρα του Arduino θα έφτανε η τιμή 0 ή 1023 αντίστοιχα. Για την λειτουργία του led γίνεται χρήση της τεχνικής που ονομάζεται

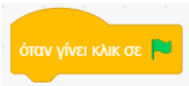


διαμόρφωση εύρους παλμού(Pulse Width Modulator, PWM). Η PWM είναι μια τεχνική η οποία μας δίνει αναλογικά αποτελέσματα με ψηφιακά μέσα καθώς εκμεταλλεύεται το χρονικό διάστημα που απαιτείται για να γίνει η εναλλαγή από το ανώτερο σημείο του παλμού στο χαμηλότερο. Το χρονικό διάστημα αυτό λέγεται πλάτος παλμού. Οι τιμές που μπορούν να δοθούν είναι από 0 έως 255. Το Arduino μπορεί να στείλει παλμούς PWM μόνο στους ακροδέκτες 3, 5, 6, 9, 10, και 11.



#### Δ. Το πρόγραμμα

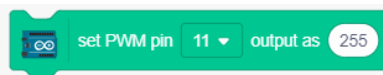
Όπως και στα προηγούμενα φύλλα εργασίας ξεκινάμε το πρόγραμμά μας με την

εντολή  και στην συνέχεια μπαίνουμε στο κύριο σώμα του προγράμματος

χρησιμοποιώντας της εντολή ελέγχου . Για την λειτουργία του led θα πρέπει να

γίνει αντιστοίχιση των αναλογικών τιμών του ποτενσιόμετρου με τις τιμές PWM της ψηφιακής εξόδου 11. Έτσι αρχικά παίρνουμε το πλακίδιο που αντιστοιχεί στην ψηφιακή

έξοδο που θα χρησιμοποιήσουμε



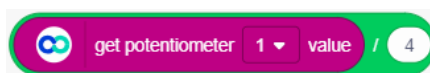
. Στο πεδίο της τιμής θα πρέπει να βάλουμε την τιμή η οποία αντιστοιχεί στο αναλογικό σήμα που λαμβάνουμε από το ποτενσιόμετρο. Έτσι αρχικά παίρνουμε το πλακίδιο το οποίο διαβάζει τις αναλογικές

τιμές του ποτενσιόμετρου

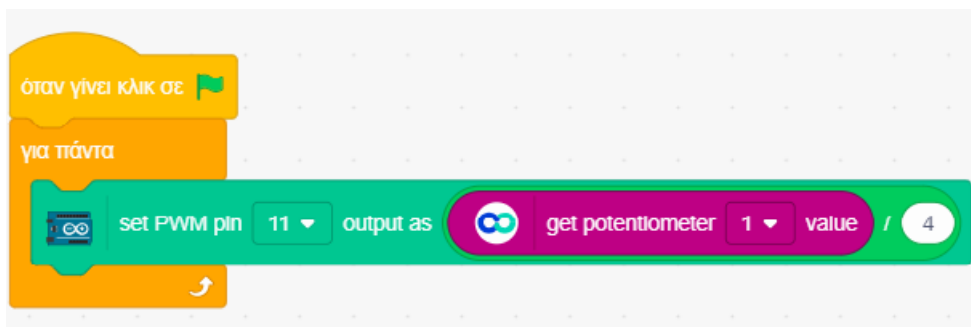


. Οι αναλογικές τιμές που δίνει το ποτενσιόμετρο κυμαίνονται από 0-1023 οπότε για την αντιστοίχιση αυτών των τιμών με τις PWM τιμές θα χρησιμοποιήσουμε έναν τελεστή διαίρεσης. Έτσι η τιμή του ποτενσιόμετρου

που θα αντιστοιχεί στην ψηφιακή έξοδο του led θα είναι



Το ολοκληρωμένο πρόγραμμα θα έχει ως εξής:



Εικόνα 14. Το πρόγραμμα του 3ου φύλλου εργασίας

## Ε. Συζήτηση του αποτελέσματος – μετατροπές


Οι μαθητές περιγράφουν το αποτέλεσμα πάνω στην πλατφόρμα και καλούνται να σκεφτούν παραδείγματα από την καθημερινή τους ζωή τα οποία προσομοιάζουν την χρήση ενός ποτενσιόμετρου. Επίσης με βάση το συγκεκριμένο φύλλο εργασίας μπορεί να γίνει μια πρώτη αναφορά στο τι είναι αναλογικό και τι ψηφιακό σήμα που τα συναντάμε και ποιες είναι οι διαφορές μεταξύ τους. Θα πρέπει να επισημανθεί εδώ ότι το κομμάτι της αντιστοίχισης των αναλογικών τιμών του ποτενσιόμετρου με τις PWM τιμές της ψηφιακής εξόδου αποτελεί ένα δυσνόητο κομμάτι για μαθητές πρωτοβάθμιας εκπαίδευση συνεπώς προτείνεται το κομμάτι με τον τελεστή διαίρεσης να δίνεται από τον εκπαιδευτικό.

## ΣΤ. Απάντηση στο Κουίζ

Οι μαθητές στην συνέχεια πρέπει να απαντήσουν στο [διαδικτυακό κουίζ](#) δύο ερωτήσεων σχετικών με τον κώδικα σε Javascript. Στην πρώτη ερώτηση πρέπει οι μαθητές να εντοπίσουν την γραμμή κώδικά στην οποία ορίζεται το ποτενσιόμετρο και ακροδέκτης στον οποίο είναι συνδεδεμένο. Πριν το δεύτερο κουίζ πρέπει να προηγηθεί προσπάθεια εισαγωγής των μαθητών στην έννοια της συνάρτησης και πως αυτή χρησιμοποιείται γενικά στις γλώσσες προγραμματισμού αλλά και ειδικότερα στην Javascript. Για μαθητές δημοτικού η έννοια της συνάρτησης αποτελεί ένα δυσνόητο φαινόμενο οπότε θα πρέπει να χρησιμοποιηθούν απλά παραδείγματα. Ακολούθως οι μαθητές απαντούν στο [κουίζ](#) όπου θα

πρέπει να αναγνωρίσουν που ξεκινά και που τελειώνει η συνάρτηση με την οποία ελέγχουμε την φωτεινότητα του Led μέσω του ποτενσιόμετρου.


Σε ποια από τα παρακάτω κομμάτια κώδικα ορίζουμε το ποτενσιόμετρο μας;



Find and insert media  
or drop an image here to upload

▲ varled = new five.Led(9);	◆ potentiometer = new five.Sensor({ pin: "A0",
● potentiometer.on("change",function(value)	■ Add answer 4 (optional)

Σε ποιες γραμμές του κώδικα πιστεύετε ότι αρχίζει και σταματάει η συνάρτηση που ρυθμίζει την φωτεινότητα του Led?



Remove

▲ 26 έως 42	◆ 19 έως 21
● 1 έως 5	■ Add answer 4 (optional)

Εικόνα 15. Το κουίζ του 3ου φύλλου εργασίας

#### **4.3.5 4<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΔΙΝΩ ΧΡΩΜΑΤΑ ΣΤΟ LED»**

##### **A. Διδακτικοί στόχοι**

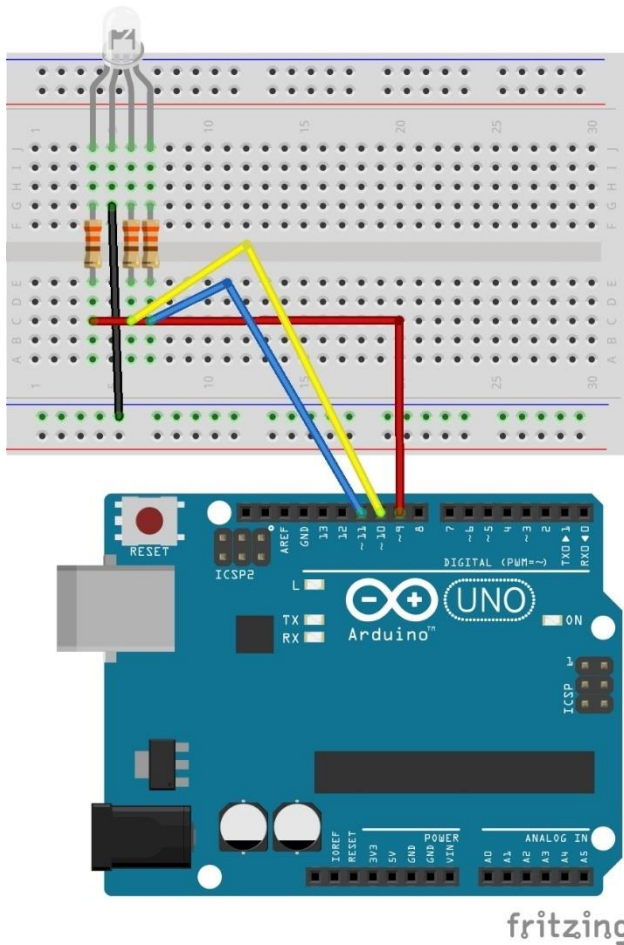
- ✓ Να κατανοήσουν την λειτουργία ενός RGB LED.
- ✓ Να μπορούν να δημιουργήσουν το κάθε χρώμα που διαθέτει το Led.
- ✓ Να μπορούν να αλλάξουν την συχνότητα εμφάνισης κάθε χρώματος.
- ✓ Εισαγωγή στο σύστημα RGB και στο πως μπορούν να δημιουργηθούν όλες οι αποχρώσεις μέσω αυτού.

##### **B. Υλικά που θα χρειαστούν**

- Η πλακέτα Arduino Uno
- Καλώδιο USB
- Breadboard
- RGB Led
- 3 αντιστάσεις 330Ω
- 5 καλώδια

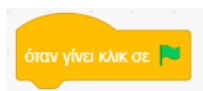
##### **Γ. Υλοποίηση διάταξης**

Στο κύκλωμα της παρακάτω εικόνας ένα RGB LED είναι συνδεδεμένο στις ψηφιακές εξόδους 9, 10 και 11 του Arduino Uno. Οι τρεις ακροδέκτες του RGB LED αντιστοιχούν σε ένα ξεχωριστό χρώμα ενώ ο τέταρτος ακροδέκτης είναι συνδεδεμένος στην γείωση. Πιο συγκεκριμένα ο ακροδέκτης που είναι συνδεδεμένος στην ψηφιακή έξοδο 9 αντιστοιχεί στο κόκκινο χρώμα, ο ακροδέκτης που είναι συνδεδεμένος στην ψηφιακή έξοδο 10 αντιστοιχεί στο πράσινο και ο ακροδέκτης που είναι συνδεδεμένος στην ψηφιακή έξοδο 11 στο μπλε.




## Δ. Το πρόγραμμα

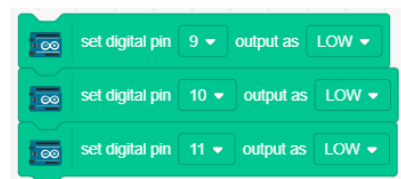
Στο παρόν φύλλο εργασίας θα προγραμματίσουμε το Arduino Uno να προβάλλει κατά σειρά στο RGB LED 7 βασικά χρώματα. Ξεκινάμε με την γνώριμη εντολή-συμβάν



και στην συνέχεια μπαίνουμε στο κύριο σώμα του προγράμματος

χρησιμοποιώντας το πλακίδιο με την εντολή ελέγχου . Αρχικά θέλουμε το LED μας να είναι τελείως σβηστό οπότε δεν πρέπει να περάνει ρεύμα από κανέναν ακροδέκτη συνεπώς όλες οι ψηφιακές έξοδοι θα είναι σε κατάσταση «LOW» δηλαδή δεν θα περνάει

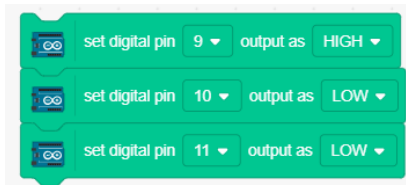
ρεύμα από αυτές. Οπότε θα χρησιμοποιήσουμε τα πλακίδια



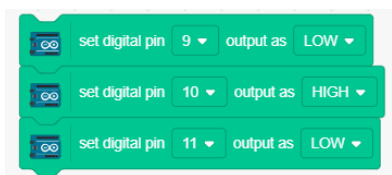
Θέλουμε το LED να παραμένει σε κάθε χρώμα για διάστημα 3 δευτερολέπτων οπότε μετά από κάθε μπλοκ προγράμματος που αναπαριστά το κάθε χρώμα θα χρησιμοποιούμε το

περίμενε 3 δευτερόλεπτα

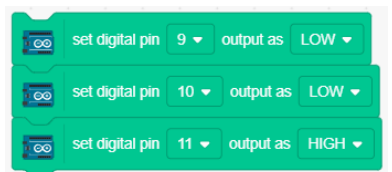
πλακίδιο με την εντολή ελέγχου «περίμενε». Στην συνέχεια θα δημιουργήσουμε το κόκκινο χρώμα οπότε η ψηφιακή έξοδος θα είναι σε κατάσταση «HIGH», δηλαδή θα περνάει ρεύμα από αυτή, ενώ οι υπόλοιπες δύο σε κατάσταση «LOW». Άρα το κομμάτι του προγράμματος που αναπαριστά το κόκκινο χρώμα θα είναι το εξής:



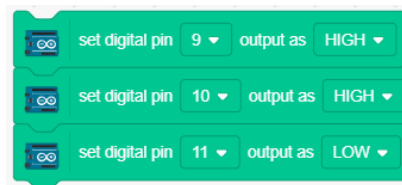
. Αντίστοιχα θα δημιουργήσουμε το πράσινο θέτοντας σε κατάσταση «HIGH» την έξοδο 10 και σε «LOW» τις άλλες δύο



, ενώ για μπλε θέτουμε ως «HIGH» μόνο την έξοδο 11

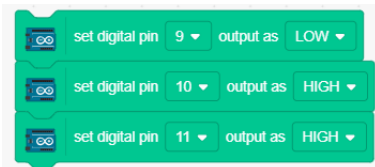


. Επόμενο χρώμα που θέλουμε να δημιουργήσουμε στο RGB LED είναι το κίτρινο για το οποίο απαιτείται ο συνδυασμός δύο χρωμάτων και συγκεκριμένα του κόκκινου και του πράσινου. Για αυτό τον λόγο πρέπει να δώσουμε ρεύμα και στους δύο αντίστοιχους ακροδέκτες ταυτόχρονα ώστε να πάρουμε το αποτέλεσμα που θέλουμε. Άρα θα θέσουμε σε κατάσταση «HIGH» τους ακροδέκτες 9 και 10. Το μπλοκ του προγράμματος

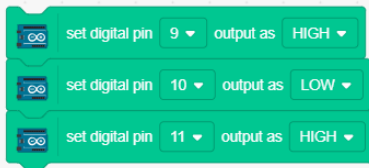


το οποίο αναπαριστά το κίτρινο χρώμα θα έχει ως εξής:

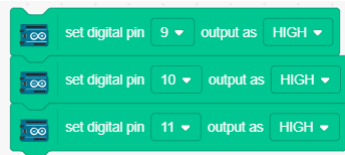
Στην συνέχεια θα κάνουμε το RGB LED να ανάψει σε γαλάζιο χρώμα. Για να το πετύχουμε αυτό θα πρέπει να δώσουμε ρεύμα στους ακροδέκτες 10 και 11 έτσι ώστε να συνδυάσουμε το πράσινο και το μπλε και να πάρουμε το χρώμα που θέλουμε. Άρα το μπλοκ του γαλάζιου



θα είναι: . Επόμενο χρώμα θα είναι το μωβ για το οποίο απαιτείται ο συνδυασμός του κόκκινου και του μπλε συνεπώς οι ακροδέκτες στους οποίους θα δώσουμε ρεύμα θα είναι ο 9 και ο 11. Έτσι λοιπόν για το μωβ θα έχουμε το μπλοκ



. Τελευταίο χρώμα που θα δώσουμε στο RGBLED μας θα είναι το λευκό. Το λευκό περιέχει όλα τα χρώματα του φάσματος οπότε θα πρέπει να δώσουμε ρεύμα και στους τρεις ακροδέκτες ταυτόχρονα δηλαδή και οι τρεις ψηφιακοί έξοδοι να είναι



σε κατάσταση «HIGH»: . Το ολοκληρωμένο πρόγραμμα με το οποίο το RGB LED θα ανάβει με 7 διαφορετικά χρώματα κάθε 3 δευτερόλεπτα θα έχει ως εξής:





Εικόνα 16. Το πρόγραμμα του 4ου φύλλου εργασίας

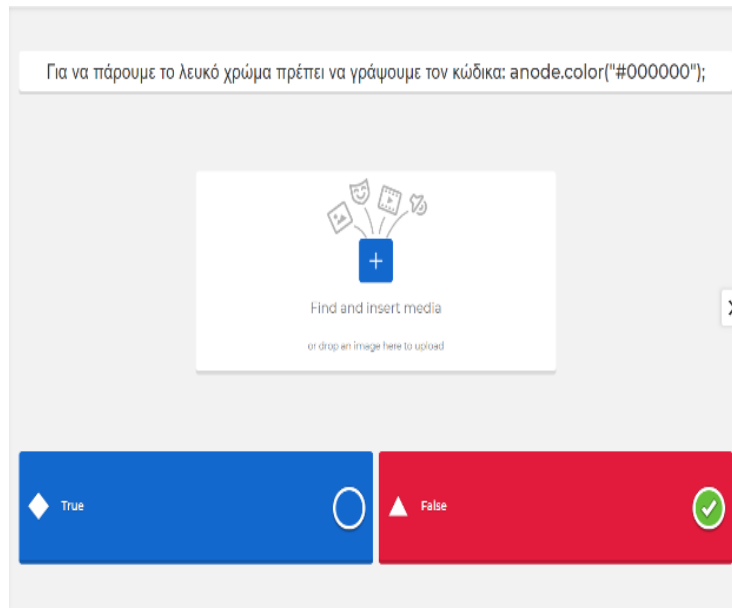
## Ε. Συζήτηση του αποτελέσματος – μετατροπές

Με το τέλος της άσκησης οι μαθητές εξηγούν το αποτέλεσμα του κώδικα και παροτρύνονται να κάνουν κάποια απλή μετατροπή όπως για παράδειγμα στον χρόνο που παραμένει το RGB Led σε κάθε χρώμα. Ακόμη γίνεται συζήτηση στην τάξη και καταιγισμός ιδεών για το πού και με ποιον τρόπο μπορεί να χρησιμοποιηθεί ένα τέτοιο πρόγραμμα σε εφαρμογές της καθημερινότητας. Εκτός από τις ιδέες των μαθητών ο εκπαιδευτικός μπορεί εκείνος να θέσει την ερώτηση για το πώς θα υλοποιούσαμε με βάση το συγκεκριμένο φύλλο εργασίας ένα φανάρι της τροχαίας.

## ΣΤ. Απάντηση στο κουίζ

Πριν δοθεί το κουίζ του συγκεκριμένου φύλλου εργασίας στους μαθητές θα πρέπει να γίνει εισαγωγή στο πρότυπο RGB και το πώς απεικονίζεται το κάθε χρώμα σε αυτό. Κάθε δυάδα ψηφίων αντιπροσωπεύει ένα από τα τρία βασικά χρώματα κόκκινο, μπλε και πράσινο κατά σειρά. Απλουστευμένα θα μπορούσαμε να εξηγήσουμε στους μαθητές ότι στο «00» είναι η πιο αχνή απόχρωση κάθε ενός από τα τρία χρώματα και το «FF» η πιο έντονη ώστε να μην χρειαστεί να μπούμε σε λεπτομέρειες σχετικά με το δεκαεξαδικό σύστημα που υπερβαίνει το επίπεδο της πρωτοβάθμιας εκπαίδευσης. Στην συνέχεια δίνεται στους μαθητές το [διαδικτυακό κουίζ](#) που αφορά την αντιστοίχιση των χρωμάτων από τα μπλοκ του PictoBlox στον κώδικα της Javascript.





Εικόνα 17. Το κουίζ του 4ου φύλλου εργασίας

#### 4.3.6 5<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΑΥΤΟΜΑΤΟΣ ΦΩΤΙΣΜΟΣ»

##### A. Διδακτικοί στόχοι

- ✓ Εισαγωγή στον τρόπο λειτουργίας μια φωτοαντίστασης.
- ✓ Να μπορούν να εισάγουν την τιμή της φωτοαντίστασης στην δομή ελέγχου «εάν.... Τότε.... Αλλιώς».
- ✓ Εισαγωγή στους μαθηματικούς τελεστές.
- ✓ Να μπορούν να αλλάζουν τις τιμές στον τελεστή για τις οποίες θα ανάβει το led.

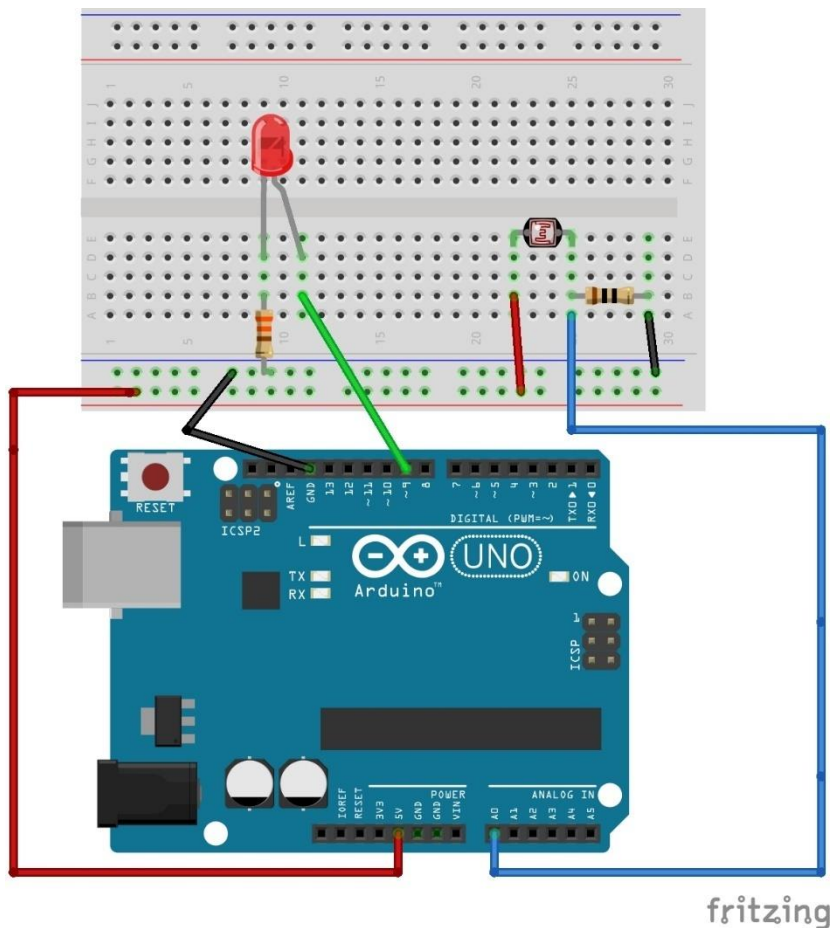
##### B. Υλικά που θα χρειαστούν

- Η πλακέτα Arduino Uno
- Καλώδιο USB
- Breadboard
- 1 Led 5mm

- 1 αντίσταση 330Ω
- 1 αντίσταση 10kΩ
- 1 φωτοαντίσταση
- 6 καλώδια

### **Γ. Υλοποίηση διάταξης**

Στην παρακάτω διάταξη μια φωτοαντίσταση είναι συνδεδεμένη στην αναλογική έξοδο A0 του Arduino Uno και ένα LED στην ψηφιακή έξοδο 9. Μέσω της φωτοαντίστασης αλλάζει η αντίσταση του κυκλώματος ανάλογα με την ποσότητα του φωτός που πέφτει πάνω στον αισθητήρα. Η φωτοαντίσταση συνδέεται σε αναλογική είσοδο καθότι οι τιμές που παίρνει δεν είναι διακριτές π.χ. «HIGH» ή «LOW» αλλά μπορεί να πάρει οποιαδήποτε τιμή από 0 έως 255.



#### Δ. Το πρόγραμμα

Θα εκμεταλλευτούμε την ιδιότητα της φωτοαντίστασης να αλλάζει την αντίσταση του κυκλώματος ανάλογα με την φωτεινότητα που υπάρχει γύρω της ώστε να δημιουργήσουμε ένα αυτόματο σύστημα φωτός. Σε αυτό το αυτόματο σύστημα φωτός το LED θα ανάβει όταν η φωτεινότητα του χώρου πέσει κάτω από ένα συγκεκριμένο όριο. Όταν

ξεκινάμε θέλουμε το LED να είναι σβηστό οπότε μετά την αρχική εντολή

όταν γίνει κλικ σε

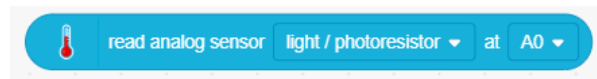
θα θέσουμε την ψηφιακή έξοδο 9 σε τιμή «LOW»



. Στην συνέχεια θέλουμε το πρόγραμμά μας να διαβάζει την τιμή της φωτοαντίστασης και να ελέγχει αν βρίσκεται πάνω ή κάτω από το όριο που έχουμε θέσει. Για τις ανάγκες της άσκησης θα θέσουμε ως όριο το 100. Για να το κάνουμε αυτό το PictoBlox μας δίνει στην κατηγορία «Sensors» ειδικά πλακίδια εντολών με τα οποία διαλέγουμε το είδος του

αισθητήρα καθώς και την είσοδο που το έχουμε συνδέσει. Το πλακίδιο που χρειαζόμαστε

για την συγκεκριμένη άσκηση είναι το



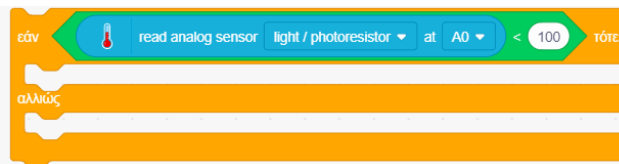
. Το συγκεκριμένο διαβάζει την τιμή της αναλογικής φωτοαντίστασης που είναι συνδεδεμένη στην αναλογική είσοδο A0. Τώρα πρέπει να χρησιμοποιήσουμε έναν μπλοκ ελέγχου το οποίο θα ελέγχει αν η τιμή της φωτοαντίστασης έχει ξεπεράσει το όριο που έχουμε θέσει. Συνεπώς



μέσα στη συνθήκη του μπλοκ ελέγχου

θα βάλουμε έναν τελεστή σύγκρισης

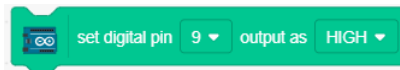
της τιμής του αισθητήρα με το όριο που έχουμε θέσει. Έτσι η συνθήκη ελέγχου θα πρέπει να



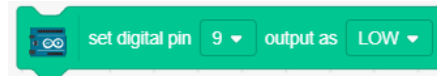
είναι η ακόλουθη:

. Αν η συνθήκη είναι

αληθής τότε το LED θα ανάβει

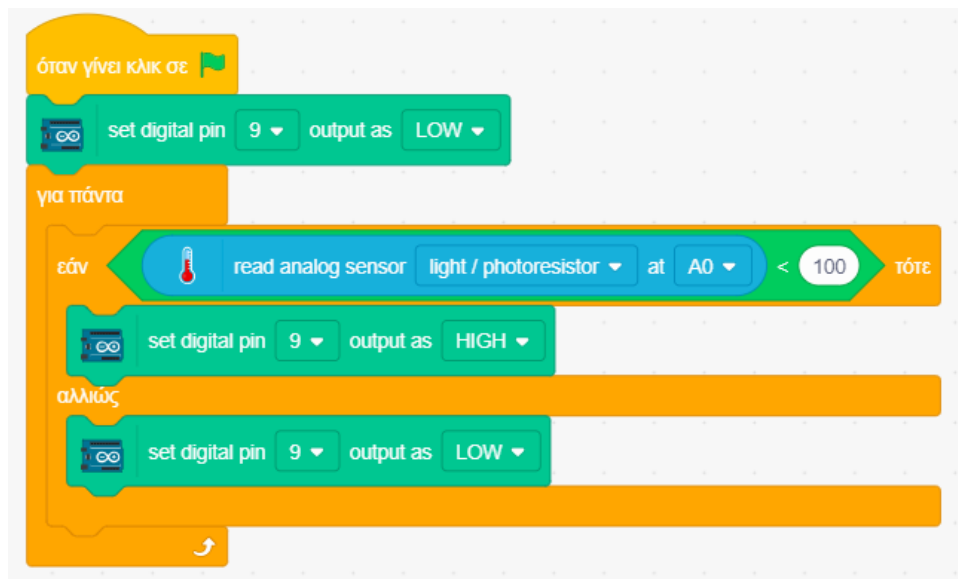


αλλιώς θα παραμένει σβηστό



. Άρα λοιπόν το ολοκληρωμένο μπλοκ για το πρόγραμμα

της άσκησης θα έχει ως εξής:



Εικόνα 18. Το πρόγραμμα του 5ου φύλλου εργασίας

## **Ε. Συζήτηση του αποτελέσματος – μετατροπές**

Αφού δημιουργηθεί ο κώδικας πραγματοποιείται συζήτηση στην τάξη για το αποτέλεσμα και για την πρακτική εφαρμογή που μπορεί να έχει. Μπορεί να ζητηθεί από τους μαθητές να αλλάξουν τιμή της μεταβλητής κάτω από την οποία ενεργοποιείται το led και να εξηγήσουν τι παρατηρούν. Για να γίνει ακόμα πιο κατανοητό το αποτέλεσμα και η εφαρμογή παροτρύνεται ο εκπαιδευτικός να πειραματιστεί με το πραγματικό περιβάλλον του σχολικού εργαστηρίου και να δημιουργεί συνθήκες σκοταδιού ώστε να ανάβουν αυτόματα τα led που έχουν προγραμματίσει οι μαθητές. Μια ακόμα πρόσθετη εφαρμογή που μπορούμε να δημιουργήσουμε είναι να εμπλέξουμε στο σενάριο μας την σκηνή και τα αντικείμενα του PictoBlox. Έτσι θα μπορούσαμε να αλλάζουμε το υπόβαθρο κάθε φορά που ανάβει το led ή ένα αντικείμενο της σκηνής να αλληλεπιδρά μαζί μας ως συνθήκη για την ενεργοποίησή του.

## **ΣΤ. Απάντηση στο κουίζ**

Το κουίζ για το 5<sup>ο</sup> φύλλο εργασίας θα εστιάζει στην έννοια της μεταβλητής και τον εντοπισμό της από τους μαθητές μέσα στον κώδικα της Javascript. Αρχικά πρέπει να εξηγηθεί στους μαθητές ότι στον προγραμματισμό πρέπει να ορίζουμε κάθε μεταβλητή στην αρχή ώστε να την χρησιμοποιήσουμε στην συνέχεια στο πρόγραμμα μας. Στο [πρώτο μέρος του διαδικτυακού κουίζ](#) οι μαθητές πρέπει να αναγνωρίσουν την γραμμή του κώδικα στην οποία δηλώνεται ο ακροδέκτης της φωτοαντίστασης. Στο [δεύτερο μέρος](#) θα πρέπει να εντοπίσουν σε ποιες γραμμές του κώδικα εκτελείται η συνάρτηση με την οποία ελέγχουμε το led. Στο [τρίτο μέρος](#) οι μαθητές θα πρέπει να εντοπίσουν την μεταβλητή η οποία δηλώνει την ένταση του φωτισμού και τις δομές ελέγχου για την ενεργοποίηση του led. Για να εισαχθούν ομαλά οι μαθητές στην έννοια της μεταβλητής προτείνεται ως πρώτο βήμα η χρήση ονομάτων που θυμίζουν πραγματικές λέξεις. Με αυτό τον τρόπο θα είναι πιο ευδιάκριτη η σημασία τους αλλά και η λειτουργία τους μέσα στον κώδικα. Τέλος, στο [τέταρτο κομμάτι](#) αναγνωρίζουν την συνθήκη ελέγχου με την οποία ανάβει το led.

Διαβάστε τον κώδικα της JavaScript και επιλέξτε τον ακροδέκτη στον οποίο έχει συνδεθεί η φωτααντίσταση.

```
function setup() {  
  createCanvas(100, 100);  
  background(255);  
  stroke(0);  
  strokeWeight(2);  
  rect(50, 50, 50, 50);  
  rect(50, 75, 50, 50);  
  rect(75, 50, 50, 50);  
  rect(75, 75, 50, 50);  
  rect(50, 50, 50, 75);  
  rect(75, 50, 50, 75);  
  rect(50, 75, 50, 75);  
  rect(75, 75, 50, 75);  
}
```

<input checked="" type="checkbox"/> pin A0	<input type="checkbox"/> pin 2
<input type="checkbox"/> pin A3	<input type="checkbox"/> pin 9

Παρατηρήστε τον κώδικα της JavaScript και επιλέξτε το σωστό όνομα της μεταβλητής της ένταση του φωτισμού.

```
function setup() {  
  createCanvas(100, 100);  
  background(255);  
  stroke(0);  
  strokeWeight(2);  
  rect(50, 50, 50, 50);  
  rect(50, 75, 50, 50);  
  rect(75, 50, 50, 50);  
  rect(75, 75, 50, 50);  
  rect(50, 50, 50, 75);  
  rect(75, 50, 50, 75);  
  rect(50, 75, 50, 75);  
  rect(75, 75, 50, 75);  
}
```

<input type="checkbox"/> console.log	<input checked="" type="checkbox"/> fos
<input type="checkbox"/> this.value	<input type="checkbox"/> "data"



Σε ποιες γραμμές του κώδικα JavaScript εκτελείται η συνάρτηση με την οποία ελέγχεται το led;

```

1 var five = new Five(1000, 100);
2 led4.φωτισματος.led;
3 led4 = new Five.Board();
4 var led = new Five.Led(5);
5 board.on("click", function() {
6   console.log("Board clicked");
7 });
8 console.log("new Five.Board()");
9 console.log("new Five.Led(5)");
10 console.log("led");
11 console.log("led.on()");
12 console.log("led.off()");
13 console.log("led.on()");
14 console.log("led.off()");
15 console.log("led.on()");
16 console.log("led.off()");
17 console.log("led.on()");
18 console.log("led.off()");
19 console.log("led.on()");
20 console.log("led.off()");
21 console.log("led.on()");
22 console.log("led.off()");
23 console.log("led.on()");
24 console.log("led.off()");
25 console.log("led.on()");
26 console.log("led.off()");
27 console.log("led.on()");
28 console.log("led.off()");
29 console.log("led.on()");
30 console.log("led.off()");
31 console.log("led.on()");
32 console.log("led.off()");
33 console.log("led.on()");
34 console.log("led.off()");
35 console.log("led.on()");
36 console.log("led.off()");
37 console.log("led.on()");
38 console.log("led.off()");
39 console.log("led.on()");
40 console.log("led.off()");
41 console.log("led.on()");
42 console.log("led.off()");
43 console.log("led.on()");
44 console.log("led.off()");
45 console.log("led.on()");
46 console.log("led.off()");
47 console.log("led.on()");
48 console.log("led.off()");
49 console.log("led.on()");
50 console.log("led.off()");
51 console.log("led.on()");
52 console.log("led.off()");
53 console.log("led.on()");
54 console.log("led.off()");
55 console.log("led.on()");
56 console.log("led.off()");
57 console.log("led.on()");
58 console.log("led.off()");
59 console.log("led.on()");
60 console.log("led.off()");
61 console.log("led.on()");
62 console.log("led.off()");
63 console.log("led.on()");
64 console.log("led.off()");
65 console.log("led.on()");
66 console.log("led.off()");
67 console.log("led.on()");
68 console.log("led.off()");
69 console.log("led.on()");
70 console.log("led.off()");
71 console.log("led.on()");
72 console.log("led.off()");
73 console.log("led.on()");
74 console.log("led.off()");
75 console.log("led.on()");
76 console.log("led.off()");
77 console.log("led.on()");
78 console.log("led.off()");
79 console.log("led.on()");
80 console.log("led.off()");
81 console.log("led.on()");
82 console.log("led.off()");
83 console.log("led.on()");
84 console.log("led.off()");
85 console.log("led.on()");
86 console.log("led.off()");
87 console.log("led.on()");
88 console.log("led.off()");
89 console.log("led.on()");
90 console.log("led.off()");
91 console.log("led.on()");
92 console.log("led.off()");
93 console.log("led.on()");
94 console.log("led.off()");
95 console.log("led.on()");
96 console.log("led.off()");
97 console.log("led.on()");
98 console.log("led.off()");
99 console.log("led.on()");
100 console.log("led.off()");

```

5 έως 17

9 έως 31

18 έως 30

Add answer 4 (optional)

Ποιο κομμάτι του κώδικα της JavaScript είναι το αντίστοιχο με αυτό της εικόνας;

```

1 board.on("click", function() {
2   console.log("Board clicked");
3 });
4 console.log("new Five.Board()");
5 console.log("new Five.Led(5)");
6 console.log("led");
7 console.log("led.on()");
8 console.log("led.off()");
9 console.log("led.on()");
10 console.log("led.off()");
11 console.log("led.on()");
12 console.log("led.off()");
13 console.log("led.on()");
14 console.log("led.off()");
15 console.log("led.on()");
16 console.log("led.off()");
17 console.log("led.on()");
18 console.log("led.off()");
19 console.log("led.on()");
20 console.log("led.off()");
21 console.log("led.on()");
22 console.log("led.off()");
23 console.log("led.on()");
24 console.log("led.off()");
25 console.log("led.on()");
26 console.log("led.off()");
27 console.log("led.on()");
28 console.log("led.off()");
29 console.log("led.on()");
30 console.log("led.off()");
31 console.log("led.on()");
32 console.log("led.off()");
33 console.log("led.on()");
34 console.log("led.off()");
35 console.log("led.on()");
36 console.log("led.off()");
37 console.log("led.on()");
38 console.log("led.off()");
39 console.log("led.on()");
40 console.log("led.off()");
41 console.log("led.on()");
42 console.log("led.off()");
43 console.log("led.on()");
44 console.log("led.off()");
45 console.log("led.on()");
46 console.log("led.off()");
47 console.log("led.on()");
48 console.log("led.off()");
49 console.log("led.on()");
50 console.log("led.off()");
51 console.log("led.on()");
52 console.log("led.off()");
53 console.log("led.on()");
54 console.log("led.off()");
55 console.log("led.on()");
56 console.log("led.off()");
57 console.log("led.on()");
58 console.log("led.off()");
59 console.log("led.on()");
60 console.log("led.off()");
61 console.log("led.on()");
62 console.log("led.off()");
63 console.log("led.on()");
64 console.log("led.off()");
65 console.log("led.on()");
66 console.log("led.off()");
67 console.log("led.on()");
68 console.log("led.off()");
69 console.log("led.on()");
70 console.log("led.off()");
71 console.log("led.on()");
72 console.log("led.off()");
73 console.log("led.on()");
74 console.log("led.off()");
75 console.log("led.on()");
76 console.log("led.off()");
77 console.log("led.on()");
78 console.log("led.off()");
79 console.log("led.on()");
80 console.log("led.off()");
81 console.log("led.on()");
82 console.log("led.off()");
83 console.log("led.on()");
84 console.log("led.off()");
85 console.log("led.on()");
86 console.log("led.off()");
87 console.log("led.on()");
88 console.log("led.off()");
89 console.log("led.on()");
90 console.log("led.off()");
91 console.log("led.on()");
92 console.log("led.off()");
93 console.log("led.on()");
94 console.log("led.off()");
95 console.log("led.on()");
96 console.log("led.off()");
97 console.log("led.on()");
98 console.log("led.off()");
99 console.log("led.on()");
100 console.log("led.off()");

```

board = new five.Board();  
var led = new five.Led(5);

if (form > 100) {  
 led.on() ;  
else {  
 led.off() ;  
}

Add answer 4 (optional)

Εικόνα 19. Το κουίζ του 5ου φύλλου εργασίας

#### 4.3.7 6<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΠΡΟΓΡΑΜΜΑΤΙΖΩ ΤΟΝ ΔΙΚΟ ΜΟΥ ΣΥΝΑΓΕΡΜΟ»

##### Α. Διδακτικοί στόχοι

- ✓ Να κατανοήσουν τον τρόπο λειτουργίας ενός ηχείου buzzer.
- ✓ Να κατανοήσουν και να μπορούν να εισάγουν την συνθήκη πατήματος ενός πλήκτρου μέσα σε μια δομή ελέγχου.

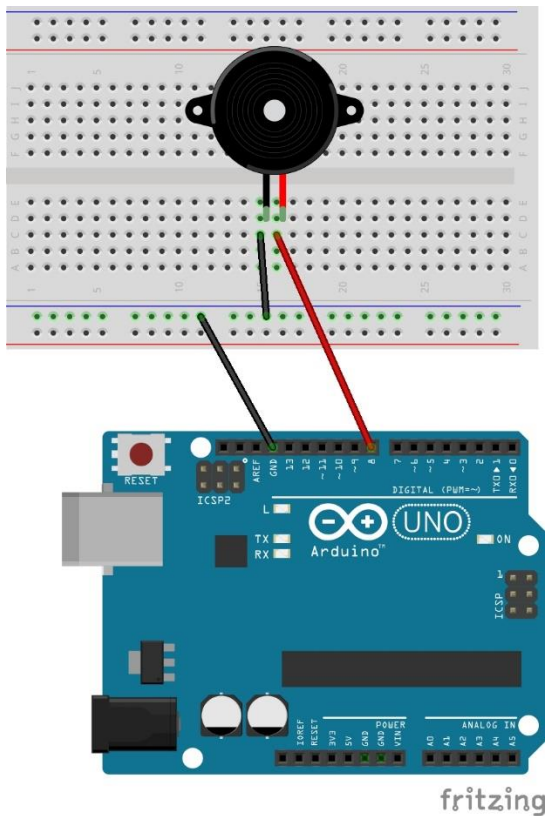
- ✓ Να κατανοήσουν την δομή ελέγχου «περίμενε.....ώσπου....».
- ✓ Να μπορούν να χρησιμοποιήσουν το πλακίδιο εντολής του buzzer αλλάζοντας τον τόνο και την διάρκεια του.
- ✓ Εισαγωγή στους λογικούς τελεστές.
- ✓ Να μπορούν να χρησιμοποιήσουν τον λογικό τελεστή «ένωσε».

## **B. Υλικά που θα χρειαστούν**

- Η πλακέτα Arduino Uno
- Καλώδιο USB
- Breadboard
- 1 Buzzer
- 2 καλώδια



## **Γ. Υλοποίηση διάταξης**

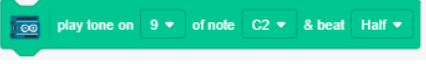
Στο παρακάτω κύκλωμα ένα ηχείο τύπου Buzzer είναι συνδεδεμένο στην ψηφιακή έξοδο 8 του Arduino. Το Buzzer είναι ένα μικρό μονοτονικό ηχείο σαν αυτά που συναντάμε για παράδειγμα σε υπολογιστές ή ξυπνητήρια. Μέσω αυτού μπορούμε να προγραμματίσουμε το Arduino Uno ώστε να παίζει κάποια μελωδία.





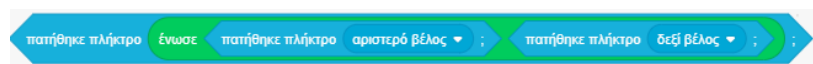
## Δ. Το πρόγραμμα

Θα χρησιμοποιήσουμε την λειτουργία του buzzer για να προγραμματίσουμε έναν αυτοσχέδιο συναγερμό προστασίας του υπολογιστή μας. Κάθε φορά που θα πατιέται κάποιο πλήκτρο ο «συναγερμός» θα ενεργοποιείται και θα σταματάει μόνο με συγκεκριμένο συνδυασμό πλήκτρων. Αρχικά λοιπόν θα δημιουργήσουμε την συνθήκη με την οποία θα ενεργοποιείται ο συναγερμός. Θα το κάνουμε χρησιμοποιώντας μέσα στο κύριο σώμα του

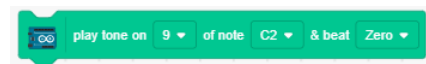
προγράμματος  την συνθήκη ελέγχου . Η συνθήκη θα ελέγχει αν πατήθηκε κάποιο πλήκτρο και σε αυτή την περίπτωση θα ενεργοποιείται ο ήχος του buzzer με

την εντολή . Το πρώτο πεδίο αντιστοιχεί στον ψηφιακό ακροδέκτη στον οποίο έχουμε συνδέσει το buzzer ενώ το δεύτερο και το τρίτο στον τόνο που θα ακούγεται και την διάρκειά του. Ενδεικτικά αναφέρουμε ότι το γράμμα C αντιστοιχεί στη νότα «ΝΤΟ», το γράμμα D στη νότα «ΡΕ» κ.ο.κ. Επόμενο βήμα είναι να κάνουμε τον

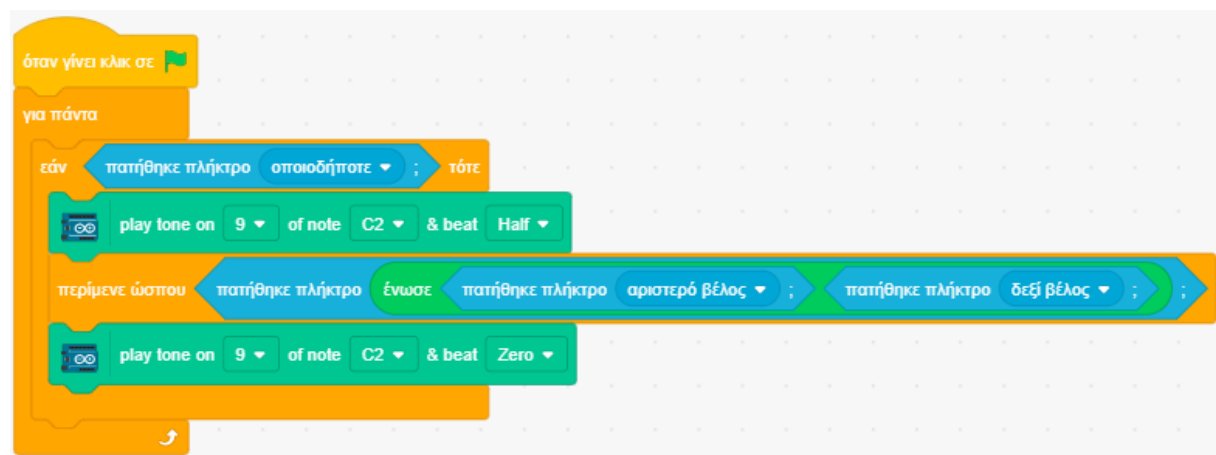
συναγερμό να απενεργοποιείται με συγκεκριμένο συνδυασμό δύο πλήκτρων. Για να το πετύχουμε θα χρησιμοποιήσουμε την δομή ελέγχου  μέσα στην οποία θα τοποθετήσουμε την συνθήκη για την συνδυασμό των πλήκτρων. Επειδή θέλουμε ο συνδυασμός να αποτελείται από δύο πλήκτρα ταυτόχρονα θα χρησιμοποιήσουμε τον τελεστή ένωσης  στον οποίο θα βάλουμε τα δύο πλήκτρα που θα αποτελούν τον συνδυασμό και θα τον τοποθετήσουμε **μέσα** σε ένα ακόμα πλακίδιο πατήματος πλήκτρου. Συνεπώς, η συνολική συνθήκη που θα ελέγχει το μπλοκ «περίμενε ώσπου» θα είναι η



. Όταν επαληθευτεί η συγκεκριμένη συνθήκη θα σταματάει ο «συναγερμός» με την εντολή



. Συνεπώς το ολοκληρωμένο μπλοκ του προγράμματος θα είναι όπως φαίνεται στην παρακάτω εικόνα:



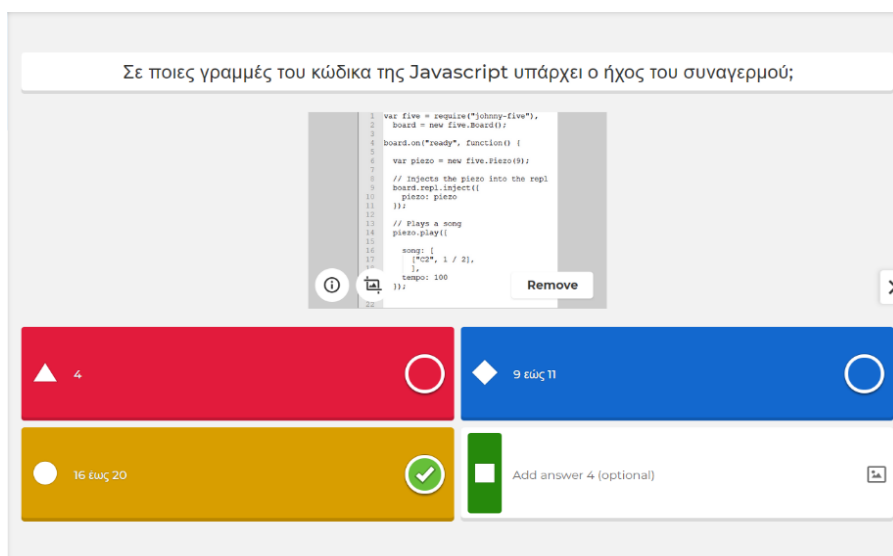
Εικόνα 20. Το πρόγραμμα του βου φύλλου εργασίας

## Ε. Συζήτηση του αποτελέσματος – μετατροπές

Μόλις ολοκληρωθεί η διαδικασία του προγραμματισμού, οι μαθητές καλούνται να περιγράψουν το αποτέλεσμα στην πράξη. Στην συνέχεια παροτρύνονται να προβούν σε μικρές μετατροπές και δοκιμές στο πρόγραμμα ώστε να πειραματιστούν με τον τόνο και την διάρκεια του ήχου του «συναγερμού». Ακόμη, μπορεί να γίνει συζήτηση στην τάξη για το που μπορεί να βρει πρακτική εφαρμογή στην καθημερινότητα ένα τέτοιο πρόγραμμα.

## ΣΤ. Απάντηση στο διαδικτυακό κουίζ

Η δημιουργία συνάρτησης στην Javascript οποία θα ελέγχει το πάτημα κάποιου πλήκτρου και στην συνέχεια τον συνδυασμό των πλήκτρων αποτελεί αρκετά δυσνόητη διαδικασία για αυτό το επίπεδο. Για τον λόγο αυτό ο κώδικας και το [διαδικτυακό κουίζ του kahoot](#) που τον συνοδεύει θα περιοριστούν στην εγκατάσταση και λειτουργία αποκλειστικά του buzzer.



Εικόνα 21. Το κουίζ του βου φύλλου εργασίας

### 4.3.8 7<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΕΛΕΓΧΩ ΕΝΑΝ ΣΕΡΒΟΚΙΝΗΤΗΡΑ»

#### Α. Διδακτικοί στόχοι

- ✓ Να κατανοήσουν τον τρόπο λειτουργίας και την χρησιμότητα ενός σερβοκινητήρα.
- ✓ Να μπορούν να ορίσουν τον σερβοκινητήρα, το ψηφιακό pin στο οποίο είναι συνδεδεμένος και την αρχική του θέση.
- ✓ Να μπορούν να εισάγουν τις εντολές αλλαγής γωνίας του σερβοκινητήρα ως συνθήκη σε δομές ελέγχου όπως η «εάν....τότε....αλλιώς...».
- ✓ Να μπορούν αν χρησιμοποιούν τις εντολές του σερβοκινητήρα σε συνδυασμό με άλλα περιφερειακά.

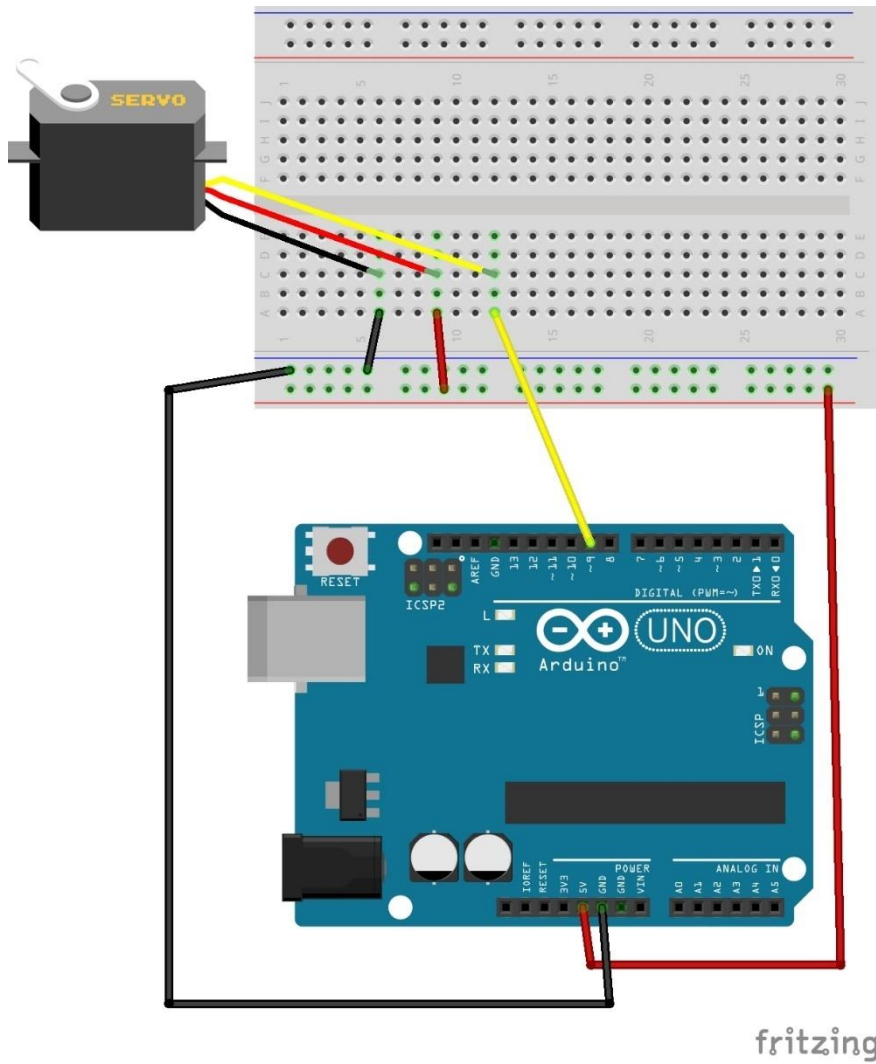
- ✓ Να είναι σε θέση να σκεφτούν και να προτείνουν πρακτικές κατασκευές με την χρήση του σερβοκινητήρα.

## **B. Υλικά που θα χρειαστούν**

- Η πλακέτα Arduino Uno
- Καλώδιο USB
- 1 Breadboard
- 1 Σερβοκινητήρας τύπου SG90
- 6 καλώδια

## **Γ. Υλοποίηση Διάταξης**

Στην παρακάτω διάταξη έχουμε συνδέσει στο Arduino Uno έναν σερβοκινητήρα τύπου SG90. Ο σερβοκινητήρας είναι ένας κινητήρας που έχει την δυνατότητα να περιστρέφει έναν άξονα από 0 έως 180 μοίρες. Στην άκρη του έχει τρία μικρά καλώδια χρώματος κόκκινου, κίτρινου και μαύρου. Το κίτρινο συνδέεται με μια ψηφιακή έξοδο και συγκεκριμένα με την ψηφιακή έξοδο 9, το κόκκινο με την τροφοδοσία των 5V ενώ το μαύρο συνδέεται με την γείωση.



#### Δ. Το πρόγραμμα

Για να γίνει πιο «χειροπιαστό» το αποτέλεσμα του προγράμματος θα συνδυάσουμε το κύκλωμα του παρόντος φύλλου εργασίας με το κύκλωμα του δεύτερου φύλλου εργασίας με το λαμπάκι led και τον διακόπτη. Με αυτό τον τρόπο θα προσομοιάσουμε την λειτουργία μια μπάρας διέλευσης αυτοκινήτων. Όταν πατάμε τον διακόπτη το led θα ανάβει και ο σερβοκινητήρας θα ανυψώνει την μπάρα για να περνάνε ελεύθερα τα αυτοκίνητα ενώ σε αντίθετη περίπτωση το led είναι σβηστό και η μπάρα θα είναι κλειστή. Ξεκινάμε όπως πάντα

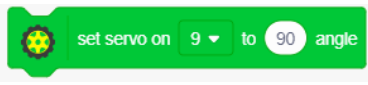
το πρόγραμμα μας με την εντολή





. Στην συνέχεια πρέπει να ορίσουμε τον σερβοκινητήρα και το pin στο οποίο είναι συνδεδεμένος. Στο κύκλωμά μας ο

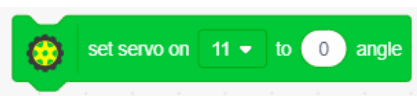
σερβοκινητήρας είναι συνδεδεμένος στην ψηφιακή έξοδο 9 οπότε η εντολή που θα

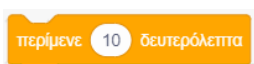
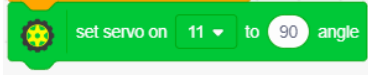
χρησιμοποιήσουμε θα είναι η  και επειδή θέλουμε αρχικά η μπάρα διέλευσης να είναι κατεβασμένη, δηλαδή σε οριζόντια θέση, η επόμενη εντολή θα

είναι . Στην συνέχεια μπαίνουμε στο κύριο σώμα του

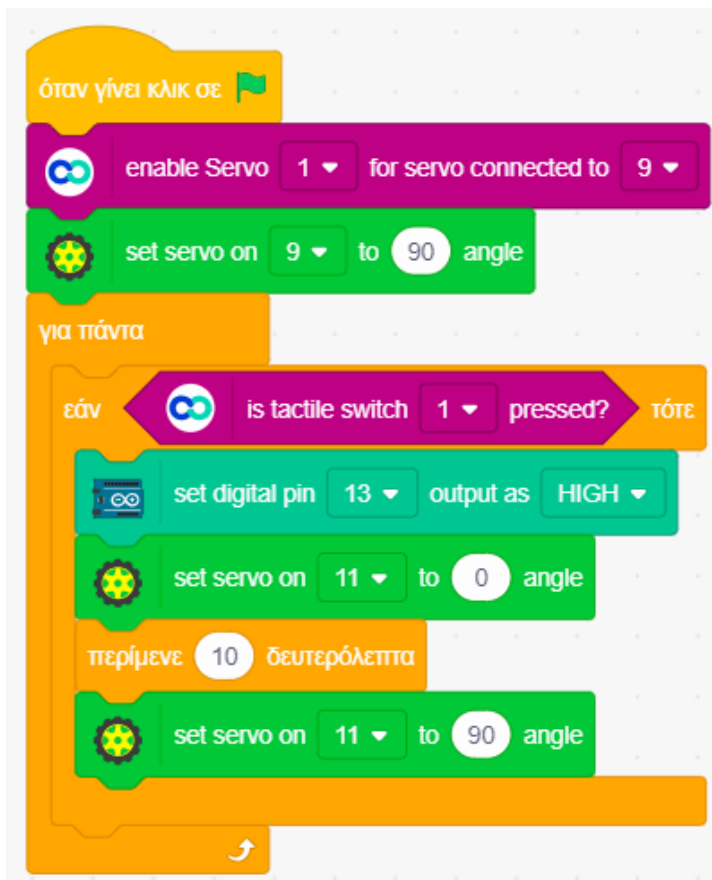
προγράμματος με την εντολή . Εκεί θέλουμε με την συνθήκη ελέγχου

 να ελέγχουμε αν έχει πατηθεί ο διακόπτης  και σε αυτή την περίπτωση θα ανάβει το led και η μπάρα διέλευσης θα σηκώνεται, δηλαδή ο σερβοκινητήρας θα στρέφεται σε κατακόρυφη θέση. Για να τεθεί σε κατακόρυφη θέση η μπάρα διέλευσης θα πρέπει η γωνία του σερβοκινητήρα να τεθεί στις 0 μοίρες δηλαδή η

εντολή θα διαμορφωθεί ως εξής: . Στην περίπτωση που η μπάρα διέλευσης σηκωθεί θέλουμε να μείνει όρθια για συγκεκριμένο χρονικό διάστημα για παράδειγμα 10 δευτερόλεπτα και στην συνέχεια να κλείνει ξανά οπότε θα χρησιμοποιήσουμε

τις εντολές  . Το ολοκληρωμένο μπλοκ του προγράμματος του σερβοκινητήρα σε συνδυασμό με το κύκλωμα και το πρόγραμμα του δεύτερου φύλλου εργασίας θα διαμορφωθεί ως εξής:





Εικόνα 22. Το πρόγραμμα του 7ου φύλλου εργασίας

## Ε. Συζήτηση του αποτελέσματος – μετατροπές

Αφού δημιουργήσουμε το πρόγραμμα προχωράμε στην συζήτηση με τους μαθητές για το πώς θα μπορούσε να γίνει η κατασκευή της μπάρας διέλευσης και στην συνέχεια επιχειρείται στην πράξη. Προτείνεται αρχικά στους μαθητές να υλοποιηθεί με κάποιο πλαστικό καλαμάκι και ακολουθεί από εκείνους καταγισμός ιδεών και προτάσεων για άλλους τρόπους υλοποίησης. Σαν επόμενο θέμα συζήτησης στην τάξη τίθεται οι επιπλέον εφαρμογές που μπορεί να έχει ένας σερβοκινητήρας και ο τρόπος για να τον προγραμματίσουμε σε αυτές. Μια επίσης απλή εφαρμογή που θα μπορούσε να υλοποιηθεί με σερβοκινητήρα και είναι πολύ οικεία στους μαθητές είναι η τραμπάλα μια παιδικής χαράς.

## ΣΤ. Διαδικτυακό κουίζ

Στο διαδικτυακό κουίζ οι μαθητές καλούνται να απαντήσουν σε δύο ερωτήσεις. Στην [πρώτη ερώτηση](#) πρέπει να εντοπίσουν σε ποιες γραμμές εντός του κώδικα σε JavaScript ορίζουμε τον σερβοκινητήρα. Στην [δεύτερη ερώτηση](#) οι μαθητές θα πρέπει να επιλέξουν την σωστή συνάρτηση σύμφωνα με το ερώτημα που τους τίθεται.

The image shows two screenshots of an online quiz interface. The top screenshot displays a question: "Σε ποιες γραμμές του κώδικα JavaScript ορίζεται ο σερβοκινητήρας μας;" (In which lines of the JavaScript code is our servo motor defined?). A code editor shows a snippet of JavaScript code with line numbers 1 through 24. Below the code editor are four answer options: a red button labeled "5 έως 9", a blue button labeled "1 έως 4", a yellow button labeled "12 έως 16", and a green button labeled "Add answer 4 (optional)". The bottom screenshot displays a question: "Επιλέξτε τις συνάρτησεις με τις οποίες η μπάρα διέλευσης είναι ανοιχτή και το led αναμμένο." (Select the functions with which the slider bar is open and the led is lit). A code editor shows a snippet of JavaScript code with line numbers 1 through 10. Below the code editor are four answer options: a red button with a code snippet and a green checkmark, a blue button with a code snippet and a green checkmark, a yellow button with a code snippet and a white circle, and a green button labeled "Add answer 4 (optional)".

Εικόνα 23. Το κουίζ του 7ου φύλλου εργασίας

#### 4.3.9 8<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΜΕΤΡΩ ΤΗΝ ΘΕΡΜΟΚΡΑΣΙΑ»

##### A. Διδακτικοί στόχοι.

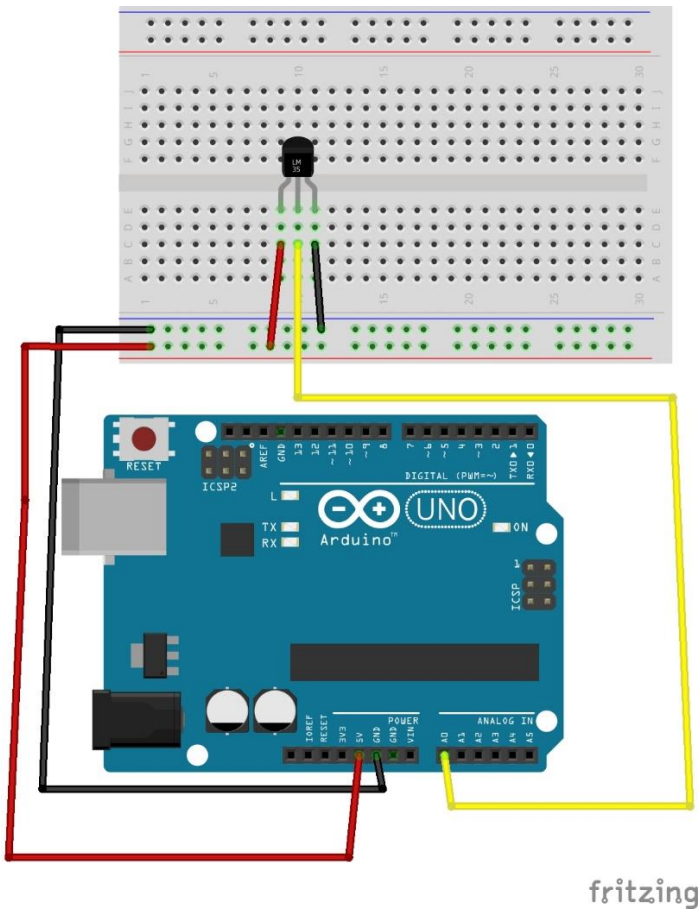
- ✓ Να κατανοήσουν τον τρόπο λειτουργίας ενός αισθητήρα θερμοκρασίας.
- ✓ Εισαγωγή στην έννοια της μεταβλητής.
- ✓ Να μπορούν να δημιουργήσουν και να ορίσουν μια νέα μεταβλητή.
- ✓ Να μπορούν να εισάγουν την αναλογική τιμή του αισθητήρα ως τιμή της μεταβλητής που όρισαν.
- ✓ Να μπορούν να χρησιμοποιήσουν την μεταβλητή σε συνδυασμό με άλλα αντικείμενα του προγραμματιστικού περιβάλλοντος PictoBlox.

##### B. Υλικά που θα χρειαστούν

- Η πλακέτα Arduino Uno
- Καλώδιο USB
- Breadboard
- 1 αισθητήρας θερμοκρασίας τύπου LM35
- 5 καλώδια

##### Γ. Υλοποίηση Διάταξης

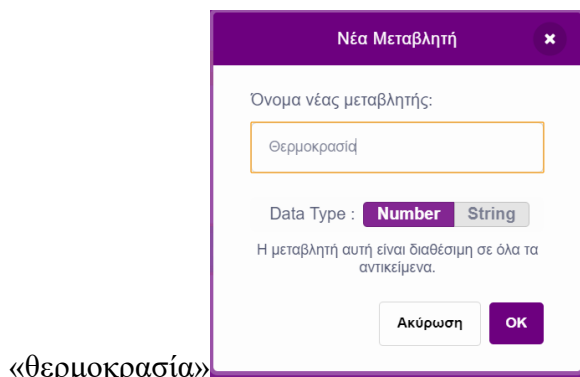
Στο κύκλωμα της άσκησης έχουμε συνδέσει στο Arduino Uno έναν αναλογικό αισθητήρα θερμοκρασία τύπου LM35. Ο συγκεκριμένος αισθητήρας μπορεί να μετρήσει θερμοκρασίες από 0 έως 150 βαθμούς κελσίου με ακρίβεια  $\pm 1^{\circ}\text{C}$ . Η τάση εξόδου του μεταβάλλεται κατά 30 mV/0. Ο LM35 έχει τρεις ακροδέκτες από τους οποίους ο ακροδέκτης που βρίσκεται δεξιά συνδέεται στην γείωση, ο μεσαίος ακροδέκτης συνδέεται σε μια αναλογική είσοδο (για την άσκηση έχουμε επιλέξει την είσοδο A0) και ο αριστερός ακροδέκτης στην τροφοδοσία των 5V.




fritzing

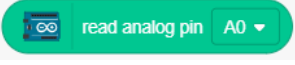
#### Δ. Το πρόγραμμα

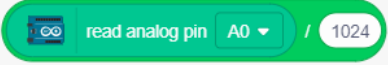
Όπως είδαμε και στην διάταξη παραπάνω ο LM35 είναι συνδεδεμένος στον αναλογικό ακροδέκτη A0 συνεπώς πρόκειται για έναν αναλογικό αισθητήρα. Συνεπώς για να πάρουμε και να προβάλλουμε την θερμοκρασία σε βαθμούς Κελσίου θα πρέπει να μετατρέψουμε την αναλογική τιμή που παίρνουμε από τον αισθητήρα σε ψηφιακή. Για τον λόγο αυτό θα δημιουργήσουμε μια καινούρια μεταβλητή την οποία θα ονομάσουμε

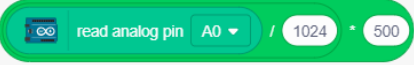



«θερμοκρασία». Στην συνέχεια θα πρέπει να ορίσουμε την

συγκεκριμένη μεταβλητή . Σύμφωνα με τις οδηγίες του κατασκευαστή του αισθητήρα η φόρμουλα για να μετατραπεί η αναλογική τιμή σε ψηφιακή και σε βαθμούς κελσίου είναι:  $val = \text{analogRead}(\text{tempPin}); \text{float}mv = (val/1024.0)*5000;$   $\text{float}cel = mv/10;$  συνεπώς θα πρέπει να χρησιμοποιήσουμε τους τελεστές για να ορίσουμε την μεταβλητή. Αρχικά θα χρησιμοποιήσουμε το μπλοκ με το οποίο παίρνουμε τις τιμές του

αναλογικού pin A0 . Στην συνέχεια χρησιμοποιώντας τον τελεστή διαίρεσης θα διαιρούμε την τιμή που παίρνει με την τιμή 1024.


. Στην συνέχεια η τιμή αυτή πολλαπλασιάζεται με το 5000(τάση σε mV) και η τιμή που προκύπτει διαιρείται με το 10, οπότε αρκεί να τοποθετήσουμε το προηγούμενο μπλοκ σε έναν τελεστή πολλαπλασιασμού με την τιμή 500

. Έτσι έχει οριστεί πλήρως η μεταβλητή που ονομάσαμε

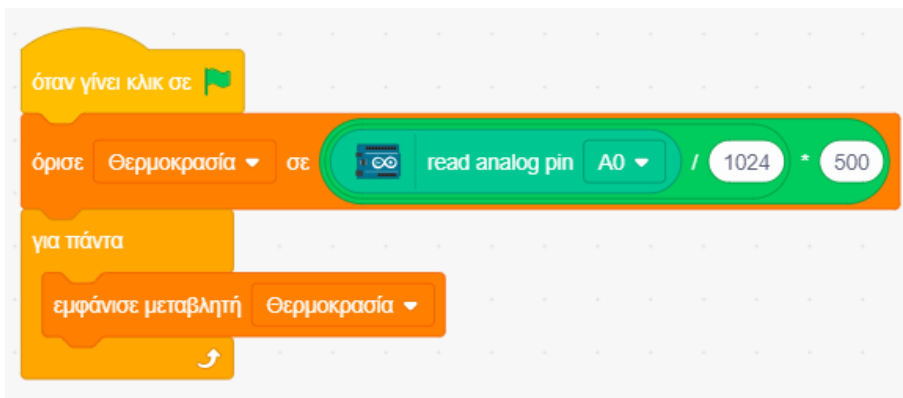
«Θερμοκρασία» . Στην συνέχεια και στο κύριο μέρος του προγράμματος θα πρέπει να «ανακοινώνεται» η θερμοκρασία. Για να το πετύχουμε αυτό θα χρησιμοποιήσουμε δύο εναλλακτικές. Στην πρώτη η τιμή της θερμοκρασίας θα είναι μόνιμα εμφανής στο τερματικό (σκηνή) ενώ στην δεύτερη θα εμφανίζεται με μήνυμα ανά διάστημα που θα ορίσουμε. Για την πρώτη περίπτωση θα

χρησιμοποιήσουμε το μπλοκ  ενώ για την δεύτερη θα

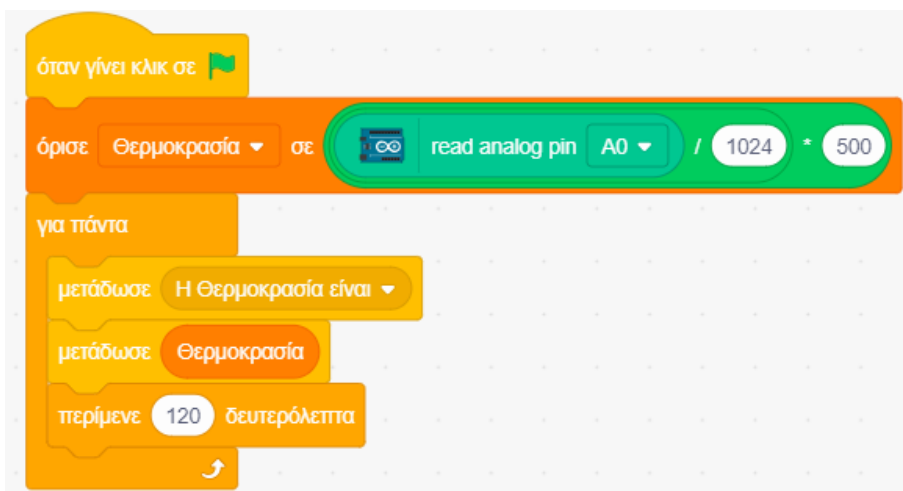
χρησιμοποιήσουμε τα μπλοκ συμβάντος   ακολουθούμενα από την

εντολή . Άρα οι δύο εναλλακτικές του προγράμματος για τον

αισθητήρα θερμοκρασίας θα είναι οι εξής:



Εικόνα 24. Το 1ο πρόγραμμα του 8ου φύλλου εργασίας



Εικόνα 25. Το 2ο πρόγραμμα του 8ου φύλλου εργασίας

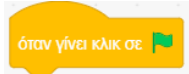


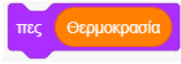
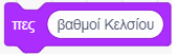

## Ε. Συζήτηση του αποτελέσματος – Μετατροπές

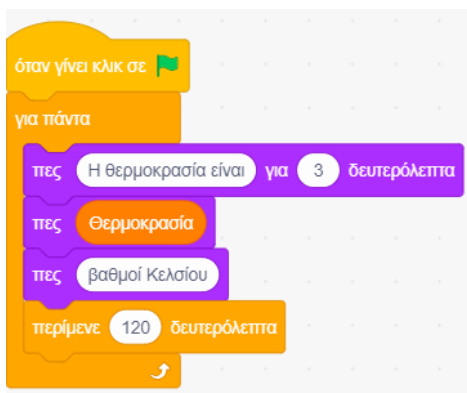
Στην συνέχεια ακολουθεί συζήτηση με τους μαθητές στην τάξη για τις εφαρμογές στην καθημερινότητα ενός αισθητήρα θερμοκρασίας και το πώς θα μπορούσαμε να τον χρησιμοποιήσουμε στο σχολικό περιβάλλον. Μια ακόμη εναλλακτική προσέγγιση θα ήταν να προβάλλουμε την θερμοκρασία θα χρησιμοποιώντας ένα αντικείμενο ως μέσο

ανακοίνωσης. Εισάγουμε λοιπόν το έτοιμο αντικείμενο



. Ξεκινώντας με την ίδια

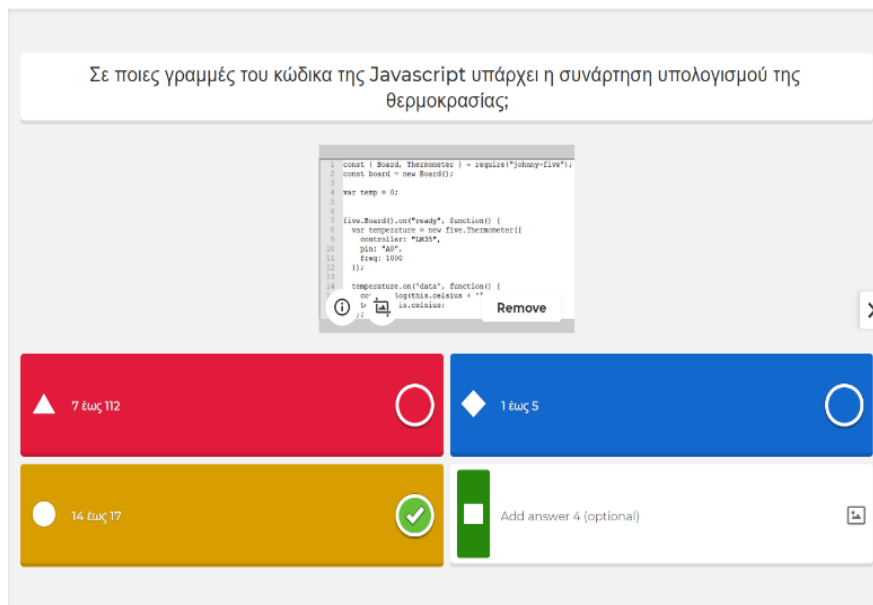
εντολή  περνάμε στο κύριο σώμα του προγράμματος  όπου θα τοποθετήσουμε τα πλακίδια με τα οποία θα μας «λέει» την θερμοκρασία. Συνεπώς η πρώτη πρόταση θα είναι  και θα ακολουθεί η τιμή της θερμοκρασίας. Για να γίνει αυτό θα πρέπει στο μπλοκ της εντολής όψεως «πες» να εισάγουμε την μεταβλητή «θερμοκρασία» που ορίσαμε νωρίτερα. Συνεπώς το μπλοκ θα είναι . Για να είναι πιο ολοκληρωμένο εισάγουμε ακόμα ένα μπλοκ «πες» με την φράση «βαθμοί Κελσίου» . Τέλος θα εισάγουμε το μπλοκ  ώστε να περάσει ένα χρονικό διάστημα πριν πάρουμε μια νέα τιμή θερμοκρασίας.



Εικόνα 26. Εναλλακτικό πρόγραμμα αισθητήρα θερμοκρασίας

## ΣΤ. Απάντηση στο διαδικτυακό κουίζ

Στην συνέχεια οι μαθητές απαντούν στο [κουίζ](#) του kahoot που αφορά τον εντοπισμό της συνάρτησης υπολογισμού της θερμοκρασίας μέσα στον κώδικα της Javascript. Η βιβλιοθήκη «jshy-five» της javascript που χρησιμοποιείται στην παρούσα εργασία παρέχει έτοιμη την συνάρτηση υπολογισμού θερμοκρασίας σε βαθμούς Κελσίου κάτι που απλοποιεί τον κώδικά μας.



Εικόνα 27. Κουίζ 8ου φύλλου εργασίας

#### 4.3.10 9<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΒΡΙΣΚΩ ΤΗΝ ΑΠΟΣΤΑΣΗ ΑΠΟ ΕΜΠΟΔΙΑ»

##### A. Διδακτικοί στόχοι

- ✓ Να κατανοήσουν τον τρόπο λειτουργίας ενός αισθητήρα υπερήχων.
- ✓ Να μπορούν να ορίσουν τον αισθητήρα και να δημιουργήσουν την μεταβλητή για την μέτρηση της απόστασης από αντικείμενο.
- ✓ Να μπορούν να χρησιμοποιήσουν τον τελεστή στρογγυλοποίησης για την προβολή της απόστασης σε εκατοστά.
- ✓ Να μπορούν να χρησιμοποιήσουν την εντολή «μετάδωσε» για την προβολή στο τερματικό της απόστασης που μέτρησε ο αισθητήρας.
- ✓ Να είναι σε θέση να προτείνουν πρακτικές κατασκευές στις οποίες μπορεί να χρησιμοποιηθεί ένας τέτοιος αισθητήρας.

##### B. Υλικά που θα χρειαστούν

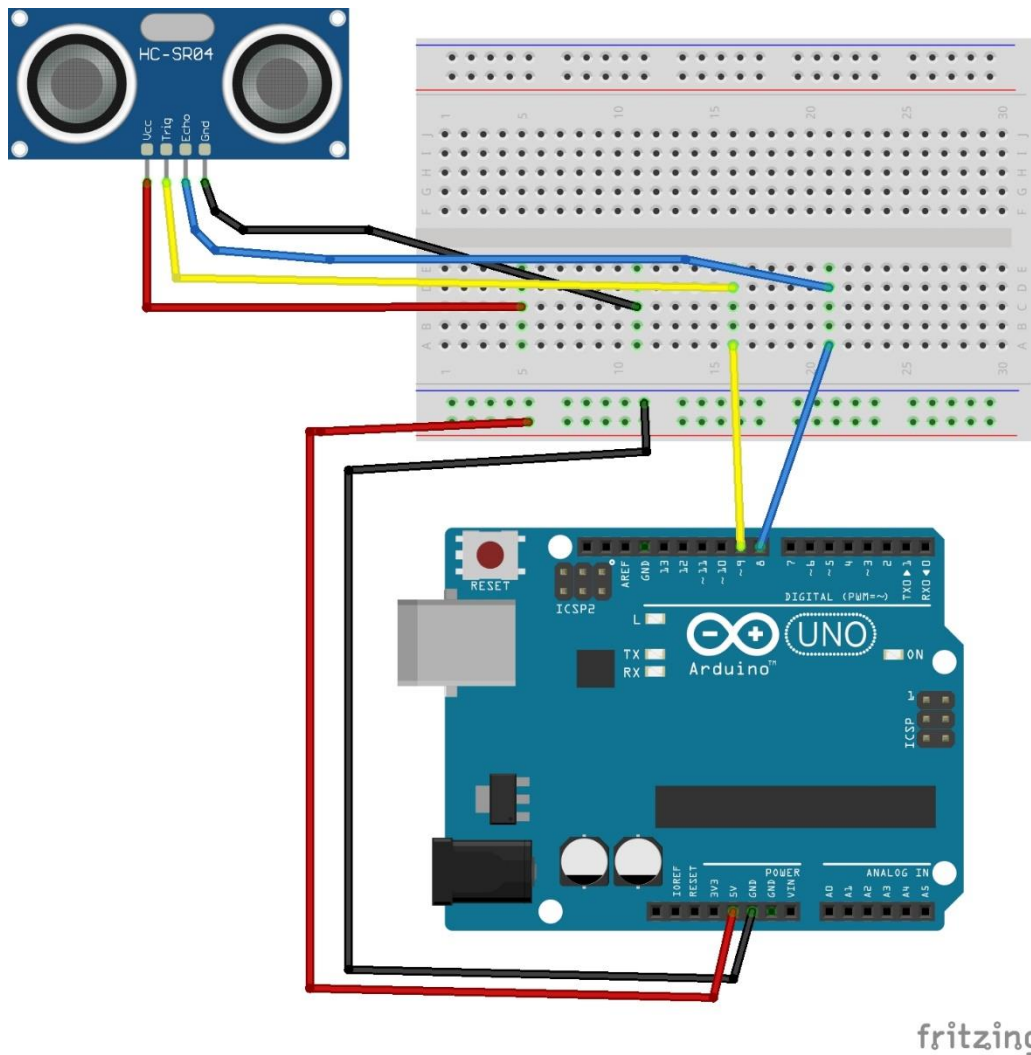
- Η πλακέτα Arduino Uno
- Καλώδιο USB



- Breadboard
- 1 αισθητήρας υπερήχων τύπου HC-SR04
- 8 καλώδια

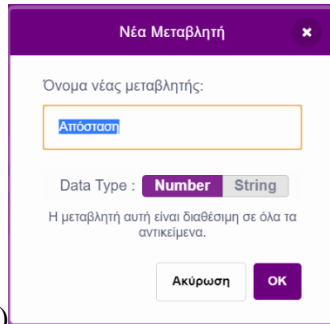
### **Γ. Υλοποίηση διάταξης**

Στο παρόν φύλλο εργασίας έχουμε συνδέσει στο Arduino Uno έναν αισθητήρα υπερήχων τύπου HC-SR04. Οι αισθητήρες υπερήχων αποτελούν ένα πολύ χρήσιμο περιφερειακό εξάρτημα καθώς χρησιμοποιούνται για να μετράμε μεταβολές απόστασης. Ο HC-SR04 εκπέμπει έναν υπέρηχο συχνότητας 40kHz ο οποίος μεταδίδεται στον αέρα και αν κατά την διάδοση του συναντήσει κάποιο εμπόδιο επιστρέφει πίσω στον αισθητήρα. Έτσι υπολογίζοντας τον χρόνο διάδοσης και την ταχύτητα του ήχου προκύπτει η απόσταση. Για την σύνδεσή του στο Arduino χρησιμοποιεί 4 ακροδέκτες. Ο πρώτος ακροδέκτης συνδέεται στην τάση των 5V, ο δεύτερος και ο τρίτος σε όποια ψηφιακή έξοδο θέλουμε και ο τέταρτος ακροδέκτης συνδέεται στην γείωση. Για την συγκεκριμένη άσκηση έχουμε επιλέξει τις ψηφιακές εξόδους 8 και 9.





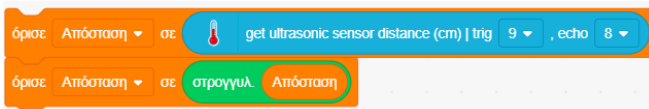
#### Δ. Το πρόγραμμα


Όπως αναφέρθηκε παραπάνω ο αισθητήρας υπερήχων είναι ένα πολύ χρήσιμο περιφερειακό καθώς μας επιτρέπει να μετράμε την απόσταση από κάποιο αντικείμενο. Αρχικά λοιπόν θα πρέπει να ορίσουμε στο πρόγραμμά μας την μεταβλητή αυτή την οποία θα ονομάσουμε «Απόσταση». Από το μενού των μεταβλητών επιλέγουμε «Δημιουργία Μεταβλητής» και πληκτρολογούμε το όνομα της και επιλέγουμε ότι ο τύπος των δεδομένων



της θα είναι αριθμοί(number). Στην συνέχεια προχωράμε στον ορισμό της επιλέγοντας το ειδικό πλακίδιο που μας παρέχει το PictoBloxia να διαβάζει τα δεδομένα του αισθητήρα υπερήχων. Στην διάταξη του κυκλώματος έχουμε συνδέσει την άκρη trigστον ψηφιακό ακροδέκτη 8 και την άκρη echoστον ψηφιακό ακροδέκτη 9 οπότε το

πλακίδιο θα διαμορφωθεί ως εξής: . Στην συνέχεια θα χρησιμοποιήσουμε ένα ακόμη πλακίδιο χρησιμοποιώντας τον τελεστή στρογγυλοποίησης ώστε να παίρνουμε την απόσταση σε ακέραιο αριθμό . Έτσι ο ορισμός της μεταβλητής θα αποτελείται από δύο εντολές και θα έχει ως εξής:



. Στην συνέχεια θα περάσουμε στο κύριο σώμα του προγράμματος όπου θα ανακοινώνεται η απόσταση που μετράει ο αισθητήρας. Χρησιμοποιώντας την εντολή «μετέδωσε» και δύο φορές τον τελεστή ένωσης θα σχηματίσουμε το μήνυμα που θέλουμε. Αυτό θα είναι . Στην συνέχεια θα περιμένει ένα δευτερόλεπτο πριν ανακοινώσει τη νέα τιμή. Συνεπώς το ολοκληρωμένο πρόγραμμα για τον αισθητήρα απόστασης θα είναι ως εξής:



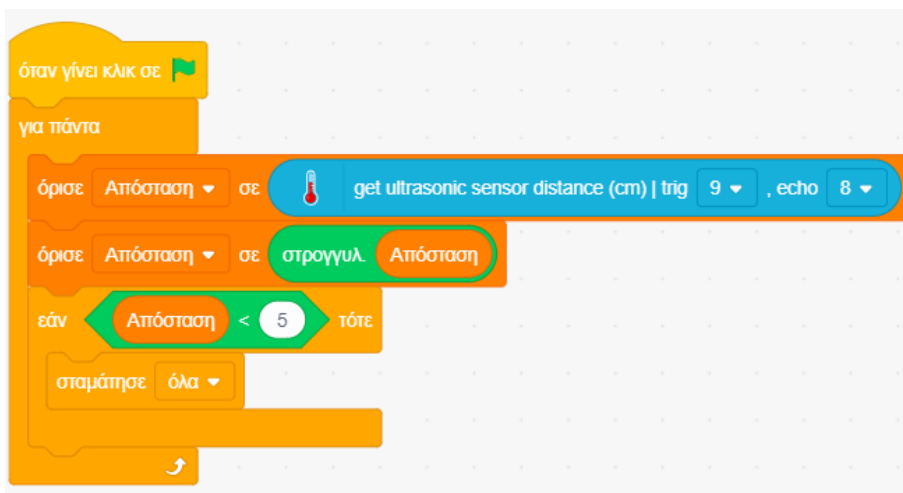
Εικόνα 28. Το πρόγραμμα του 9ου φύλλου εργασίας.

## Ε. Συζήτηση του αποτελέσματος – μετατροπές

Αφού ολοκληρώσουμε το πρόγραμμα ακολουθεί συζήτηση με τους μαθητές για το που μπορεί να βρει πρακτική εφαρμογή ένας τέτοιος αισθητήρας. Αφού οι μαθητές καταθέσουν τις ιδέες τους προτείνεται στον εκπαιδευτικό να προβάλλει βίντεο από κατασκευές εκπαιδευτικών ρομπότ. Φτάνοντας στο σημείο όπου το ρομπότ σταματάει μόλις εντοπίσει κάποιο εμπόδιο αναλύει την χρησιμότητα του αισθητήρα υπερήχων και ρωτάει τους μαθητές πως φαντάζονται ότι υλοποιείται ο προγραμματισμός ώστε να επιτυγχάνεται το αποτέλεσμα του βίντεο. Μια απλή υλοποίηση με βάση το πρόγραμμα που δημιουργήσαμε είναι η ακόλουθη. Η πρώτη μετατροπή που θα κάνουμε είναι να περάσουμε τον ορισμό της μεταβλητής μέσα στο κύριο σώμα του προγράμματος ώστε να διαβάζει συνεχώς την τιμή



του αισθητήρα. Στην συνέχεια θα χρησιμοποιήσουμε την δομή ελέγχου και έναν τελεστή σύγκρισης. Στην μία πλευρά του τελεστή θα μπει η μεταβλητή της απόστασης και στην άλλη κάποιος μικρός ακέραιος αριθμός κάτω από τον οποίο θα σταματάει η ρομποτική κατασκευή. Οπότε η μετατροπή του αρχικού προγράμματος θα είναι η παρακάτω:



Εικόνα 29. Τροποποίηση του προγράμματος του 9ου φύλλου εργασίας

## ΣΤ. Απάντηση στο διαδικτυακό κουίζ

Το διαδικτυακό κουίζ του Kahoot αφορά [πρώτον](#) τον εντοπισμό στον κώδικα το πεδίο όπου ορίζεται ο αισθητήρας και [δεύτερον](#) την γραμμή στην οποία προβάλλεται η απόσταση που μετράει ο αισθητήρας.

Σε ποιες γραμμές του κώδικα Javascript ορίζεται ο αισθητήρας υπερήχων;

```

1 const { Board, Proximity } = require("johnny-five");
2 const board = new Board();
3
4 board.on("ready", () => {
5   const proximity = new Proximity({
6     controller: "HC-SR04",
7     pin: 8
8   });
9
10
11   proximity.on("change", () => {
12     const { distance, } = proximity;
13     console.log("Proximity: " +
14       distance.log() + " cm : ", controller);
15     console.log("-----");
16   });
17
18 });

```

1 έως 2

11 έως 17

4 έως 8

Add answer 4 (optional)

Ποια γραμμή του κώδικα Javascript προβάλλει την απόσταση που μετρά ο αισθητήρας μας;

```

1 const { board, proximity } = require("johnny-five");
2 const board = new Board();
3
4 board.on("ready", () => {
5   const proximity = new Proximity({
6     controller: "HC-SR04",
7     pin: 8
8   });
9
10
11   proximity.on("change", () => {
12     const { distance, } = proximity;
13     console.log("Proximity: " +
14       distance.log() + " cm : ", controller);
15     console.log("-----");
16   });
17
18 });

```

7

6

11

14

Εικόνα 30. Κουίζ 9ου φύλλου εργασίας

#### 4.3.11 10<sup>ο</sup> ΦΥΛΛΟ ΕΡΓΑΣΙΑΣ – «ΠΡΟΓΡΑΜΜΑΤΙΖΩ ΕΝΑ ΜΟΤΕΡΑΚΙ» Α. Διδακτικοί στόχοι

- ✓ Να κατανοήσουν πως λειτουργεί ένα μοτεράκι τύπου DC.
- ✓ Να μπορούν να ορίζουν το μοτεράκι και τα pins στα οποία έχει συνδεθεί.
- ✓ Να χρησιμοποιούν τις εντολές διαχείρισης του DC Motor ως συνθήκες σε δομές ελέγχου.
- ✓ Να μπορούν να μεταβάλλουν την ταχύτητα περιστροφής του κινητήρα.
- ✓ Να είναι σε θέση να προτείνουν πρακτικές κατασκευές στις οποίες θα μπορούσε να χρησιμοποιηθεί ένα τέτοιο μοτεράκι.

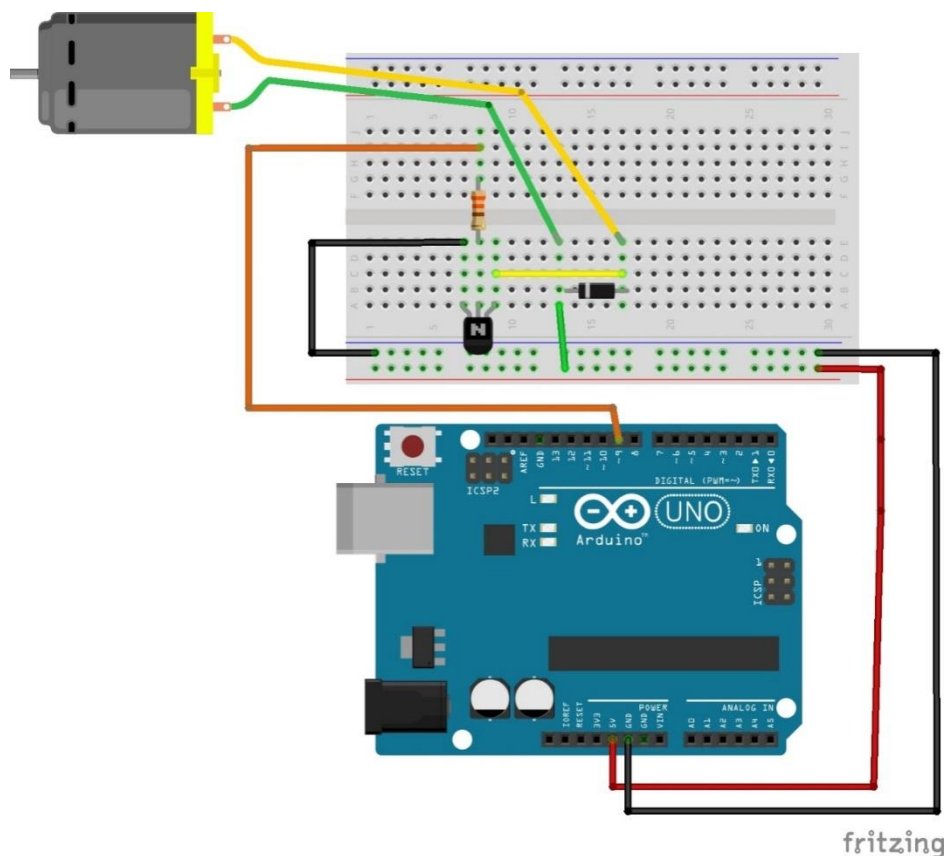
## **B. Υλικά που θα χρειαστούν**

- Η πλακέτα Arduino Uno
- Καλώδιο USB
- Breadboard
- 1 μοτεράκι τύπου DC
- 1 αντίσταση 330Ω
- 1 δίοδος τύπου 1N4148
- 1 τρανζίστορ τύπου NPN
- 6 καλώδια

## **Γ. Υλοποίηση διάταξης**

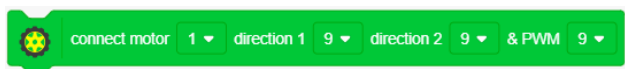
Σε αυτό το φύλλο εργασίας θα συνδέσουμε στο Arduino Uno έναν κινητήρα τύπου DC motor και θα ελέγχουμε την ένταση με την οποία αυτός περιστρέφεται. Για την σύνδεση του θα χρειαστεί να χρησιμοποιήσουμε ένα τρανζίστορ. Το τρανζίστορ ή κρυσταλλοτρίοδος είναι μια διάταξη ημιαγωγών στερεάς κατάστασης. Το βασικό του χαρακτηριστικό είναι ότι μπορεί να λειτουργεί ως μεταβλητή ωμική αντίσταση σε ένα κύκλωμα καθώς μπορεί να ρυθμίζει την ροή του ηλεκτρικού ρεύματος που απορροφά ανάλογα με την τάση που δέχεται. Το Arduino μπορεί να παρέχει στις ψηφιακές του εξόδους ρεύμα το πολύ 40mA. Οι

περισσότεροι κινητήρες όμως χρειάζονται περισσότερο ρεύμα ή και τάση για να λειτουργήσουν έτσι το τρανζίστορ λειτουργεί σαν ψηφιακός διακόπτης αφήνοντας το Arduino να ελέγξει την ροή ρεύματος σε εξαρτήματα που θέλουν περισσότερο ρεύμα. Τα τρανζίστορ έχουν 3 ακροδέκτες που ονομάζονται αντίστοιχα βάση, συλλέκτης και εκπομπός. Ακόμη για την συγκεκριμένη διάταξη θα χρειαστούμε μια δίοδο ώστε να μην καταστραφεί το τρανζίστορ από ενδεχόμενη απότομη αλλαγή τάσης.



#### Δ. Το πρόγραμμα

Θα δημιουργήσουμε ένα πρόγραμμα με το οποίο θα ελέγχουμε κάποιες λειτουργίες του κινητήρα με το πληκτρολόγιό μας. Αρχικά πρέπει να δηλώσουμε την σύνδεση του κινητήρα χρησιμοποιώντας το ειδικό μπλοκ



που παρέχει η εφαρμογή. Στην συνέχεια στο κύριο μέρος του προγράμματος ο κινητήρας θα κινείται προς τα εμπρός με την μέγιστη ταχύτητα και εμείς είτε θα τον σταματάμε είτε θα μειώνουμε την ταχύτητά περιστροφής του,

χρησιμοποιώντας το πληκτρολόγιο. Συνεπώς θα πρέπει να χρησιμοποιήσουμε την δομή



ελέγχου μέσα στην οποία θα τοποθετήσουμε ακόμα δύο συνθήκες ελέγχου για το πάτημα των πλήκτρων. Για το σταμάτημα θα ορίσουμε το πλήκτρο «space»



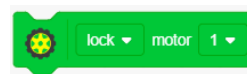
ενώ για να μειωθεί η ταχύτητα περιστροφής του κινητήρα θα

ορίσουμε το πλήκτρο «κάτω βέλος»



. Οι εντολές με τις οποίες

σταματάει ο κινητήρας και μειώνεται η ταχύτητα του είναι οι

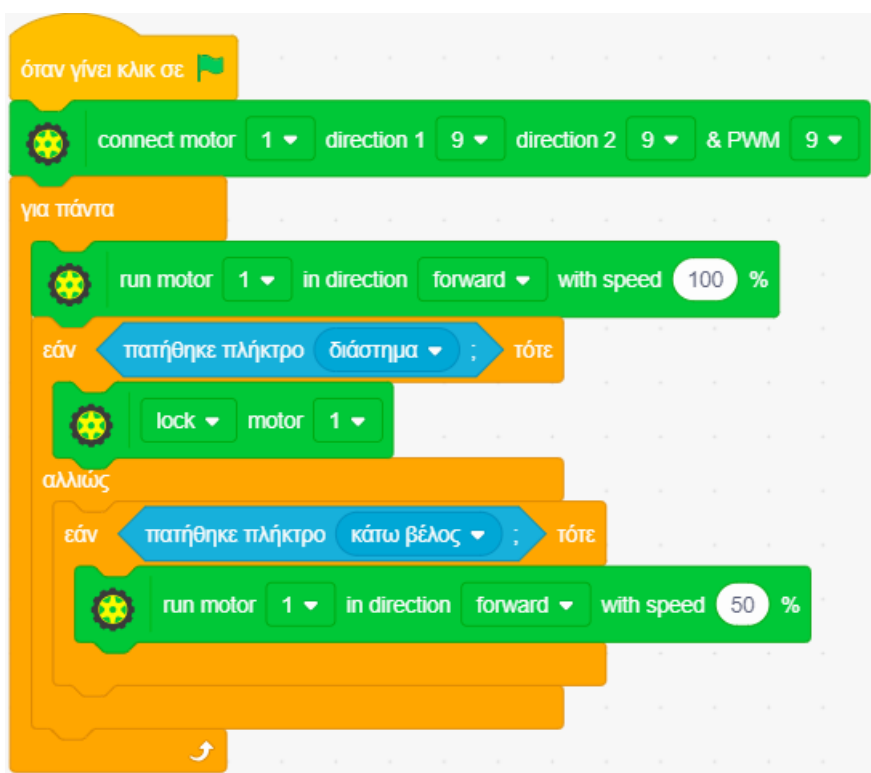


και



αντίστοιχα. Συνεπώς το ολοκληρωμένο

πρόγραμμα για την λειτουργία του κινητήρα τύπου «DC Motor» θα έχει ως εξής:

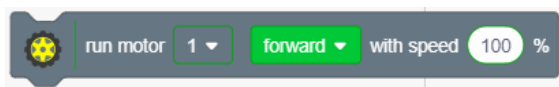


Εικόνα 31. Πρόγραμμα 10ου φύλλου εργασίας

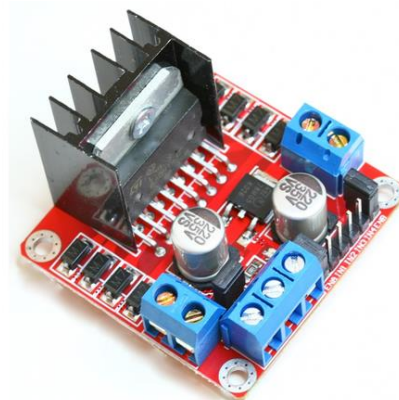


## Ε. Συζήτηση του αποτελέσματος – Μετατροπές

Αφού ολοκληρωθεί η δημιουργία του προγράμματος οι μαθητές καλούνται να αναφέρουν περιπτώσεις όπου θα μπορούσε να έχει εφαρμογή ένας τέτοιος κινητήρας και με ποιο τρόπο θα μπορούσαν να υλοποιηθούν. Στην συνέχεια ο εκπαιδευτικός με την βοήθεια βίντεο και εικόνων από το διαδίκτυο περιγράφει την διαδικασία κατασκευής και λειτουργίας ενός ρομποτικού οχήματος το οποίο θα κινείται σε μια επιφάνεια και θα σταματάει μόλις συναντήσει κάποιο εμπόδιο. Με αυτό τον τρόπο μπορούν να συνδυαστούν τα προγράμματα του 9<sup>ου</sup> και 10<sup>ου</sup> φύλλου εργασίας για την υλοποίηση μιας τέτοιας ρομποτικής κατασκευής. Για τον σκοπό αυτό υπάρχει το shield του Arduino Uno με την ονομασία L298N. Στο συγκεκριμένο shield μπορούν να συνδεθούν δύο κινητήρες οι οποίοι θα βρίσκονται δεξιά και αριστερά αντίστοιχα ώστε να προσομοιώνεται η κίνηση ενός οχήματος. Επίσης για τον σκοπό αυτό το PictoBlox παρέχει ειδικό πλακίδιο διαχείρισης των κινητήρων του shield.



Εικόνα 32. Πλακίδιο διαχείριση πολλαπλών κινητήρων



Εικόνα 33. Shield για διασύνδεση κινητήρων

## ΣΤ. Απάντηση στο διαδικτυακό κουίζ

Ο κώδικας της Javascript αποτελείται από τον ορισμό του κινητήρα και τις συναρτήσεις με τις οποίες αυτός κινείται με την μέγιστη ταχύτητα, με το 50% της μέγιστης και τέλος την συνάρτηση με την οποία ακινητοποιείται. Οι τιμές που μπορεί να πάρει η

ταχύτητα του κινητήρα κυμαίνονται από 0 έως 255. Το [πρώτο μέρος](#) του κουίζ αφορά τον ορισμό του κινητήρα στο πρόγραμμα ενώ στο δεύτερο ενώ στο [δεύτερο](#) οι μαθητές καλούνται να βάλουν σε σειρά τα κομμάτια του κώδικα αφού προηγηθεί σχετική συζήτηση σχετικά με την δήλωση των μεταβλητών και την σειρά που θέλουμε να τοποθετήσουμε τις συναρτήσεις κίνησης του κινητήρα.

Ο ορισμός του κινητήρα μας γίνεται στις γραμμές 4 έως 8 του κώδικα Javascript

```

1 const {Board, Motor} = require("johnny-five");
2 const board = new Board();
3
4 board.on("ready", () => {
5   // Create a new 'motor' hardware instance.
6   const motor = new Motor({
7     pin: 9
8   });
9
10  motor.on('start', function() {
11    speed = 255;
12    motor.fwd(speed);
13  });
14
15  motor.on('slow', function() {
16    speed = 120;
17    motor.fwd(speed);
18  });
19
20  motor.on('stop', function() {
21    motor.stop();
22  });

```

True False

B I X<sub>2</sub> X<sup>2</sup> Ω f(x)

Βάλτε στην σωστή σειρά τα κομμάτια του κώδικα σύμφωνα με όσα συζητήσαμε στην τάξη.

```

1 motor.on("slow", function() {
2   speed = 120;
3   motor.fwd(speed);
4 });
5
6 motor.on("start", function() {
7   speed = 255;
8   motor.fwd(speed);
9 });
10
11 board.on("ready", () => {
12   // Create a new 'motor' hardware instance.
13   const motor = new Motor({
14     pin: 9
15   });
16
17   motor.on('start', function() {
18     speed = 255;
19     motor.fwd(speed);
20   });
21
22   motor.on('slow', function() {
23     speed = 120;
24     motor.fwd(speed);
25   });
26
27   motor.on('stop', function() {
28     motor.stop();
29   });
30 });

```

3 2 1 4

Εικόνα 34. Κουίζ 10ου Φύλλου Εργασίας

## **Βασικά Συμπεράσματα και προτάσεις για μελλοντική έρευνα**

Είναι αναντίρρητο γεγονός πλέον ότι ο προγραμματισμός και η εκπαιδευτική ρομποτική πρέπει να αποτελούν κομμάτι της εκπαίδευσης από την πρωτοβάθμια εκπαίδευση. Το Αναλυτικό Πρόγραμμα Σπουδών του υπουργείου Παιδείας, ωστόσο, δεν περιέχει ξεκάθαρη πρόβλεψη παρά μόνο σε μάθημα εκτός της Πληροφορικής («Εργαστήρια Δεξιότητων») που εισήχθη το 2021 και μόνο ως υποενότητα αυτού («Ρομποτική – STEM»). Επιπροσθέτως η διδασκαλία του δεν έχει ανατεθεί στους εκπαιδευτικούς πληροφορικής. Παρ' όλα αυτά, έχουν αναπτυχθεί πολλές προσπάθειες εισαγωγής στην πρωτοβάθμια εκπαίδευση του προγραμματισμού και της εκπαιδευτικής ρομποτικής από μεμονωμένους εκπαιδευτικούς πληροφορικής και ανεξάρτητους ερευνητές. Μια από αυτές επιχειρείται και στην παρούσα εργασία με ένασμμα τον προ υπάρχων εξοπλισμό Arduino Uno σε δύο δημοτικά σχολεία της Νέας Φιλαδέλφειας Αττικής κατά το σχολικό έτος 2020-2021.

Παρά τους περιορισμούς της πανδημίας κατά το σχολικό έτος 2020-2021 που κατέστησε πρακτικά αδύνατη την διδασκαλία πέραν των τεσσάρων πρώτων ασκήσεων που περιλαμβάνονται στην εργασία, το σύνολο των δέκα φύλλων εργασίας αποτελούν μια πλήρη εισαγωγή στον προγραμματισμό και την εκπαιδευτική ρομποτική μέσα από έναν τρόπο συνεργατικό και ψυχαγωγικό ώστε να κρατά το ενδιαφέρον των μαθητών αυτής της ηλικίας. Περιλαμβάνονται σε αυτά όλες οι βασικές έννοιες προγραμματισμού όπως δομές ελέγχου, επανάληψης, μεταβλητές ακόμα και πιο προχωρημένες προγραμματιστικές έννοιες όπως η εμφωλευμένη επιλογή. Ακόμη περιλαμβάνεται η εισαγωγή και διδασκαλία των βασικών στοιχείων της πλατφόρμας Arduino και των περιφερειακών της όπως λαμπάκια led, διάφοροι αισθητήρες, πλήκτρα, και κινητήρες.

Για τον προγραμματισμό της πλατφόρμας στα πλαίσια της παρούσας εργασίας επιλέχθηκε το λογισμικό PictoBlox της STEMPedia. Η επιλογή αυτή έγινε για να δοθεί μια

ακόμη επιλογή σε όποιον εκπαιδευτικό επιλέξει να εφαρμόσει την διδασκαλία προγραμματισμού με Arduino στην πρωτοβάθμια εκπαίδευση εκτός των ήδη δημοσιευμένων με προγράμματα όπως το Ardublock και το S4A(Scratch for Arduino). Το περιβάλλον του PictoBlox παρόμοιο με εκείνο του Scratch 3 αποτελεί μια οικεία αίσθηση για τους μαθητές ενώ προσφέρει όλες τις δυνατότητες που χρειάζονται για τον προγραμματισμό του Arduino. Το μειονέκτημα με την μη μετάφραση των σχετικών εντολών προτείνεται να καλυφθεί μοιράζοντας στους μαθητές τον πίνακα με την μετάφραση κάθε εντολής.

Τα φύλλα εργασίας απαιτούν από μία έως δύο διδακτικές ώρες το καθένα κάτι που αποτελεί σημαντικό πλεονέκτημα ώστε να εξασφαλίζεται η διδακτική και παιδαγωγική συνέχεια. Η πληροφορική στο δημοτικό διδάσκεται μία ώρα εβδομαδιαίως οπότε καθίσταται πρακτικά δύσκολη η υλοποίηση ασκήσεων που απαιτούν παραπάνω διδακτικές ώρες. Με αυτό τον τρόπο εκπληρώνεται ο στόχος της εισαγωγής σε βασικά στοιχεία προγραμματισμού και εκπαιδευτικής ρομποτικής μέσα από συνεργατικές ασκήσεις. Υλοποιώντας ο εκπαιδευτικός το σύνολο των φύλλων εργασίας που παρουσιάζεται στην εργασία έχει στην διάθεση του μια σειρά μαθημάτων που καλύπτουν κατά μέσο όρο περίπου 15 διδακτικές εβδομάδες. Για το μάθημα των ΤΠΕ στο δημοτικό όπου το Αναλυτικό Πρόγραμμα Σπουδών δεν είναι συγκεκριμένο και δεν υπάρχει βιβλίο του μαθήματος ένα σύνολο μαθημάτων που καλύπτει ένα τέτοιο χρονικό διάστημα αποτελεί μια ενδιαφέρουσα εναλλακτική λύση στα χέρια ενός εκπαιδευτικού πληροφορικής.

Παρότι το Αναλυτικό Πρόγραμμα Σπουδών για την Πληροφορική στο δημοτικό δεν δίνει συγκεκριμένες κατευθύνσεις για την διδασκαλία του προγραμματισμού και της εκπαιδευτικής ρομποτικής, η σειρά ασκήσεων της παρούσας εργασίας σχετίζεται άμεσα με τα όσα αναφέρει το Διαθεματικό Πλαίσιο Προγράμματος Σπουδών για τις Νέες Τεχνολογίες στην εκπαίδευση και παράλληλα δεν ξεφεύγει από τις οδηγίες του υπάρχοντος ΑΠΣ. Ακόμη θέτει μια σειρά από διδακτικούς στόχους. Πιο συγκεκριμένα:

- Να κατανοήσουν οι μαθητές την αναγκαιότητα και την ουσία του προγραμματισμού.
- Να μπορούν να περιγράψουν τι είναι ο προγραμματισμός και η εκπαιδευτική ρομποτική.
- Να περιγράψουν την λειτουργία των εντολών της εφαρμογής PictoBlox.

- Να αναγνωρίζουν και να χειρίζονται την πλατφόρμα του Arduino Uno και τα βασικά περιφερειακά του.
- Να αναπτύσσουν ομαδικό και συνεργατικό πνεύμα μέσα από την ενεργή τους ενασχόληση στις δραστηριότητες των φύλλων εργασίας.
- Να προβληματίζονται και αναζητούν λύσεις και εφαρμογές μέσα από τον προγραμματισμό και την ρομποτική.
- Να δημιουργεί ευχάριστα συναισθήματα στους μαθητές μέσα από την ψυχαγωγική φύση των δραστηριοτήτων.

Ο τομέας της εκπαιδευτικής ρομποτικής είναι ραγδαία αναπτυσσόμενος και όλο και περισσότεροι εκπαιδευτικοί επιχειρούν να τον εντάξουν στην διδακτική τους διαδικασία. Έρευνες στην διεθνή βιβλιογραφία (Karplancali & Demirkol, 2017) έχουν δείξει ότι οι μαθητές που έχουν διδαχθεί προγραμματισμό σε block-based περιβάλλοντα στις τελευταίες τάξεις του Δημοτικού κατανοούν πιο γρήγορα στην συνέχεια δομές της Java ή της C#. Η συγκεκριμένη εργασία δίνει το έναυσμα και την κατεύθυνση σε εκπαιδευτικούς που δεν έχουν πρότερη ενασχόληση με την εκπαιδευτική ρομποτική και την πλατφόρμα Arduino. Ακόμη για τον προγραμματισμό της πλατφόρμας παρουσιάζει ένα λογισμικό που δεν έχει χρησιμοποιηθεί ευρέως μέχρι τώρα και το οποίο μπορεί να χρησιμοποιηθεί και για άλλες δημοφιλείς πλατφόρμες ρομποτικής διατηρώντας έτσι μια διδακτική συνέχεια για τους μαθητές.

Ασφαλώς υπάρχουν πολλές ακόμα πτυχές του προγραμματισμού με Arduino που θα μπορούσαν να διερευνηθούν σε μελλοντικές εργασίες. Ένα κομμάτι που προσφέρει μεγάλο εύρος είναι το ποιες απλές κατασκευές ρομποτικής βασισμένες στο Arduino μπορούν να κατασκευαστούν από τους μαθητές στα πλαίσια του μαθήματος. Σε μια τέτοια έρευνα εκτός από τον προγραμματισμό της πλατφόρμας απαιτούνται και οδηγίες κατασκευής οι οποίες θα πρέπει να είναι απλές ώστε να είναι κατανοητές από τους μαθητές και ταυτόχρονα να υλοποιούν μια λειτουργική και ρεαλιστική κατασκευή. Ένα ακόμη πεδίο έρευνας στο οποίο έγινε εισαγωγή και στην παρούσα εργασία είναι η μετάβαση από τον block-based προγραμματισμό σε προγραμματισμό με πραγματικό κώδικα. Η εισαγωγή των μαθητών του δημοτικού σε απλές λειτουργίες μια πραγματικής γλώσσας προγραμματισμού μπορεί με μια πρώτη ματιά να φαντάζει δύσκολη αλλά δεν είναι κάτι το ακατόρθωτο αν γίνει με δομημένο και κατανοητό τρόπο. Κοινός τόπος κάθε

μελλοντικής έρευνας θα πρέπει να είναι το ότι οι μαθητές «αγκαλιάζουν» οποιαδήποτε δραστηριότητα ξεφεύγει από την πεπατημένη του σχολικού προγράμματος, είναι ομαδική και η γνώση έρχεται μέσα από την κατασκευή και το «χειροπιαστό» αποτέλεσμα.

## ΠΑΡΑΡΤΗΜΑ

### Κώδικες σε JavaScript

#### Κώδικας 1<sup>ο</sup> Φύλλου Εργασίας

```
const { Board, Led } = require("johnny-five");

const board = new Board();

board.on("ready", () => {

const led = new Led(13); // orismos led sto pin 13

led.blink(1000); // anavosvinei kathe 1000ms

});
```

#### Κώδικας 2<sup>ο</sup> Φύλλου Εργασίας

```
var five = require("johnny-five"),

board, button, led;

board = new five.Board();

const led = new Led(13); // orismos led sto pin 13

board.on("ready", function() {

button = new five.Button({ // synarthsh orismoy button

board: board,

pin: 8,

holdtime: 1000,
```

```

invert: false

});

board.repl.inject({

button: button

});

button.on("down", function() { // synartisi gia to patima

led.on();

});

button.on("hold", function() {

led.on();

});

button.on("up", function() {

led.off();

});

});

```

### **Κώδικας 3<sup>ο</sup> Φύλλου εργασίας**

```

const five = require ("jonhy-five");

const board = new five.Board();

let potentiometer;

board.on("ready", function() {
var led = new five.Led(9); // orismos led sto pin 9
Potentiometer = new five.Sensor ({ // orismos potentiometrou

```



```

pin: "A0", // sto analog pin A0
freq: 250
});
board.repl.inject({
pot: potentiometer
});

// antistoixisi timwn potentiometer me times fwteinothtas

potentiometer.on("change", function(value) {
var brightValue = five.Fn.constrain.(five.Fn.map(value, 0 ,
1023, 0, 255), 0 , 255);
var percentValue=
five.Fn.constrain.(five.Fn.map(brightValue, 0 , 255, 0, 100),
0 , 100);
console.log(`> Led Brightness: ${percentValue}%`);
// 0 -255
});
});

```

### **Κώδικας 4<sup>ο</sup> Φύλλου εργασίας**

```

const { Board, Led } = require("johnny-five");

const board = new Board();

board.on("ready", () => {

const anode = new Led.RGB({ // orismos led sta pin 9, 10, 11

pins: {

red: 9,

```

```
green: 10,  
blue: 11  
},  
  
isAnode: true  
  
});  
  
anode.on();  
  
anode.color("#FF0000"); // kokkino xrwma  
  
anode.blink(3000); // menei anameno gia 3 deuterolepta  
  
anode.color("#00FF00"); // prasino xrwma  
  
anode.blink(3000);  
  
anode.color("#0000FF"); // mple xrwma  
  
anode.blink(3000);  
  
anode.color("#FFFF00"); // kitrino xrwma  
  
anode.blink(3000);  
  
anode.color("#00FFFF"); // kyano xrwma  
  
anode.blink(3000);  
  
anode.color("#FF00FF"); // mwv xrwma  
  
anode.blink(3000);  
  
anode.color("#FFFFFF"); // leuko xrwma  
  
anode.blink(3000);  
  
});
```

## Κώδικας 5<sup>ο</sup> Φύλλου εργασίας

```
var five = require("johnny-five"),

board, photoresistor, led;

board = new five.Board();

var led = new five.Led(9); // ορισμος led sto pin 9

board.on("ready", function() { // ορισμος photoantistasis

photoresistor = new five.Sensor({

pin: "A0", // sto analogiko pin A0

freq: 250

});

//synartisi leitoyrgias led otan h fwteinothta sthn

photoresistor kseperasei mia timi

photoresistor.on("data", function() {

console.log(this.value);

var fos = this.value

if (fos > 100) {

led.on() }

else {

led.off() }

});

});
```

### **Κώδικας 6<sup>ο</sup> Φύλλου Εργασίας**

```
var five = require("johnny-five"),

board = new five.Board();

board.on("ready", function() {

var piezo = new five.Piezo(9); // orismos hxeiou sto pin 9

board.repl.inject({

piezo: piezo

});

piezo.play({ // synartisi melwdias toy hxeiou

song: [

["C2", 1 / 2],

],

tempo: 100 // ruthmos melwdias

});
```

### **Κώδικας 7<sup>ο</sup> Φύλλου εργασίας**

```
var five = require("johnny-five");

var board = new five.Board();

board, button, led;

const led = new Led(13); // orismos led sto pin 13

// synartisi orismou button sto pin 8

board.on("ready", function() {
```

```

button = new five.Button({

board: board,

pin: 8,

holdtime: 1000,

invert: false

});

var servo = new five.Servo(9); // orismos servokinthrhra

servo.min();

servo.max();

servo.center();

});

button.on("down", function() { // sunarthsh kinshs

led.on();

servo.move(0);

});

button.on("hold", function() {

led.on();

servo.move(0);

});

button.on("up", function() { // sunarthsh paushs servo

```

```
led.off();

servo.center();

});

});

});

};
```

### **Κώδικας 8<sup>ο</sup> Φύλλου Εργασίας**

```
const { Board, Thermometer } = require("johnny-five");

const board = new Board();

var temp = 0; // orismos metavliths thermokrasias

five.Board().on("ready", function() {

var temperature = new five.Thermometer({ // orismos aisthitira

controller: "LM35",

pin: "A0", // sto analog pin A0

freq: 1000

});

temperature.on('data', function() { // sunartisi provolis

console.log(this.celsius + '°C');

temp = this.celsius;

});
```

### Κώδικας 9<sup>ο</sup> Φύλλου Εργασίας

```
const { Board, Proximity } = require("johnny-five");

const board = new Board();

board.on("ready", () => {                                     //orismos aisthitira

const proximity = new Proximity({

controller: "HCSR04",

pin: 8                                                         // sto pin 8

});

// synartisi provolis apostasis

proximity.on("change", () => {

const {centimeters,} = proximity;

console.log("Proximity: ");

console.log("  cm  : ", centimeters);

console.log("-----");

});

});
```

### Κώδικας 10<sup>ο</sup> Φύλλου Εργασίας

```
const {Board, Motor} = require("johnny-five");

const board = new Board();

board.on("ready", () => {                                     // orismos motor sto pin 9

const motor = new Motor({
```

```
pin: 9

});

motor.on('start', function() { // synartisi ekinisis

speed = 255;

motor.fwd(speed);

});

motor.on('slow', function() { // synartisi epivradinsis

speed = 120;

motor.fwd(speed);

});

motor.on ('stop', function() { // synaritisi pausis

motor.stop();

});
```



## Βιβλιογραφία

- Papert, S. (1993). *The Children's Machine: Rethinking School in the Age of the Computer*.
- Atmatzidou, S., & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences.
- Barr, V., & Stephenson, C. (2011). *Bringing Computational thinking to K-12*.
- Bitesize. (χ.χ.). *Introduction to computational thinking*. Ανάκτηση από BBC:  
<https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1>
- Chatzinikolakis, G., & Papadakis, S. (2015, Ιανουάριος). Motivating K-12 students learning fundamental Computer Science concepts with App Inventor. *IEEE Xplore*.
- Denning, P. (2009). *The profession of IT Beyond computational thinking*.
- Edison. (2021, Σεπτέμβριος). *EdPython*. Ανάκτηση από Ιστοσελίδα του Edison:  
<https://meetedison.com/robot-programming-software/edpy/>
- Eguchi, A. (2014). Why Robotics in Education? - Robotics as a Learning Tool for Educational Revolution. *Society for Information Technology & Teacher Education International Conference*, (σσ. 94-95). Jacksonville, Florida, United States.
- Fokides, E. (2017, Νοέμβριος). Students Learning to Program by Developing Games: Results of Year-Long Project in Primary School Settings. *Journal of Information Technology Education: Research*, σσ. 475-505.
- Gedik, N., Cetin, M., & Koca, C. (2017). Examining the Experiences of Preschoolers on Programming via Tablet Computers . *Mediterranean Journal of Humanities*, σσ. 193-203.
- Gioulekas, F., & Katsaros, P. (2020). *School of Informatics Aristotle University of Thessaloniki*. Ανάκτηση από Programmin Arduino with Scratch (S4A):  
<https://www.csd.auth.gr/images/ekdiloseis/STEMART.pdf>

- Gonzalez, H., & Kuenzi, J. (2102, Αύγουστος 1). STEM Education: A Primer. *Congression Research Service*.
- Google. (2020). *Google for Education*. Ανάκτηση από Exploring Computational Thninking: <https://edu.google.com/resources/programs/exploring-computational-thinking/>
- <https://meetedison.com/barcodes/>. (2021). *Edison*. Ανάκτηση από Ιστοσελίδα της Edison: <https://meetedison.com/barcodes/>
- <https://meetedison.com/robot-programming-software/>. (2021). *Edison*. Ανάκτηση από Ιστοσελίδα του Edison: <https://meetedison.com/robot-programming-software/edblocks/>
- <https://meetedison.com/robot-programming-software/>. (2021). *Edison EdScratch*. Ανάκτηση από Ιστοσελίδα του Edison: <https://meetedison.com/robot-programming-software/edscratch/>
- [https://www.csteachers.org/page/CSTA\\_Standards\\_2016](https://www.csteachers.org/page/CSTA_Standards_2016). (χ.χ.). Ανάκτηση από [https://www.csteachers.org/page/CSTA\\_Standards\\_2016](https://www.csteachers.org/page/CSTA_Standards_2016)
- Hutchison, A., Nadolny, L., & Estapa, A. (2015). *Using Coding Apps to Support Literacy Instruction and Develop Coding Literacy*.
- Kalelioglu, F. (2015, Νοέμβριος). A new way of teaching programming skills to K-12 students: Code.org. *ELSEVIER - Computers in Human Behavior*, σσ. 200-210.
- Kalelioglu, F., & Gulbahar, Y. (2013, Μάιος). The Effect of Instructional Techniques on Critical Thinking and Critical Thinking Dispositions in Online Discussion. *Educational Technology & Society*, σσ. 248-258.
- Kelley, T., & Knowles, J. (2016, Ιούλιος 19). A conceptual framework for integrated STEM education. *International Journal of STEM Education*, 3.
- Kraetzschmar, G. (2009). *Educational Robotics: On the Role of Robotics in Learning and Education*.
- LEGO. (2021, Αύγουστος). *LEGO Mindstorms EV3*. Ανάκτηση από Ιστοσελίδα του ελεγκτή EV3 της LEGO: <https://www.lego.com/en-gr/product/ev3-intelligent-brick-45500>

- Makerspaces. (χ.χ.). Ανάκτηση από 15 Arduino Uno Breadboard Projects:  
<https://www.makerspaces.com/15-simple-arduino-uno-breadboard-projects/>
- Malan, D. (2018). *Computational Thinking & Scratch - Intro to Computer Science - Harvard's CS50*.  
Ανάκτηση από <https://www.youtube.com/watch?v=F0WoVEr0-44&t=186s>
- Mobjork, M. (2010, Οκτώβριος). Consulting versus participatory transdisciplinarity: A refined classification of transdisciplinary research. *Futures*, 42, σσ. 866-873.
- Morelli, R., de Laneroll, T., Lake, P., Limardo, N., Tamotsu, E., & Uche, C. (2018, Νοέμβριος). Can Android App Inventor Bring Computational Thinking to K-12?  
[http://hfoss.org/uploads/docs/appinventor\\_manuscript.pdf](http://hfoss.org/uploads/docs/appinventor_manuscript.pdf).
- Morrison, J. (2006). *TIES STEM Education Monograph Series: Attributes of STEM Education*.
- Nicolescu, B. (2010). Methodology of Transdisciplinary- Levels of Reality, Logic of the Included Middle and Complexity. *Transdisciplinary Journal of Engineering & Science*, σσ. 19-38.
- Padlipsky, S. (2018, Δεκέμβριος). *Digital Commons @ Cal Poly*. Ανάκτηση από Using Offline Activities to Enhance Online Cybersecurity Education:  
<https://digitalcommons.calpoly.edu/theses/1956/>
- Papert, S. (1991). *Νοητικές θύελλες: Παιδιά, ηλεκτρονικοί υπολογιστές και δυναμικές ιδέες Μετάφραση Αίγλη Σταματίου*. Αθήνα: Οδυσσέας.
- Perdikuri, K. (2014, Οκτώβριος). Students' Experiences from the use of MIT App Inventor in classroom. *PCI' 14: Proceedings of the 18th Panhellenic Conference on Informatics*, (σσ. 1-6). Athens.
- Pila, S., Alade, F., Sheehan, K., Lauricella, A., & Wartella, E. (2019, Ιανουάριος). Learning to code via tablet applications: An evaluation of Daisy the Dinosaur and Kodable as learning tools for young children. *Computers & Education*, σσ. 52-62.
- Ruan, L., Patton, E., & Tissenbaum, M. (2017). Evaluations of Programming Complexity in App Inventor. *Conference Proceedings of International Conference on Computational Thinking Educational 2017. Hong Kong: The Education University of Hong Kong* (σσ. 2-5). The Education University of Hong Kong.

- Smith, P., Hartley, T., & Mehdi, Q. (2013, Οκτώβριος). C# interpreter and unity 3D for educational programming games. *IEEE Xplore*, σσ. 41-47.
- Stohlman, M., Moore, T., & Roehrig, G. (2012, Απρίλιος). Considerations for Teaching Integrated STEM Education. *Journal of Pre-College Engineering Education Research (J-PEER)*, 2.
- Vasquez, J., Comer, M., & Sneider, C. (2013). *STEM Lesson Essentials, Grades 3-8: integrating Science, Technology, Engineering, and Mathematics*.
- Wing, J. (2006). *Computational thinking. Communication of the ACM*.
- Wing, J. (2010). *Computational thinking: What and why?*
- Γεωργιτζίκη, Ν. (2019). *Δημιουργώ με το Arduino και προγραμματίζω με το Ardublock*. Λιβαδειά.
- Γκίνης, Δ. (2005, Μάϊος). Ανάκτηση από <http://www.unipi.gr/faculty/dghinis/ts/diaf5.pdf>
- Γριζιώτη, Μ. (2020). *Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών, Φιλοσοφική Σχολή, Παιδαγωγικό Τμήμα Δημοτικής Εκπαίδευσης*. Ανάκτηση από ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ Ο προγραμματισμός ως πτυχή του ψηφιακού αλφαριθμητισμού και της καλλιέργειας της υπολογιστικής σκέψης: η περίπτωση της ανάπτυξης και διασκευής ψηφιακών παιχνιδιών: <https://freader.ekt.gr/eadd/index.php?doc=48514&lang=el#page/44/mode/2up>
- Γριζιώτη, Μ., Ξένος, Μ., & Κυνηγός, Χ. (2016). *Ενίσχυση του ενδιαφέροντος των μαθητών για το STEM μέσω της εκπαιδευτικής Ρομποτικής*. Αθήνα: Εργαστήριο Εκπαιδευτικής Τεχνολογίας Φ.Σ ΕΚΠΑ.
- Εισαγωγή στις εκπαιδευτικές εφαρμογές των τεχνολογιών της πληροφορίας και των επικοινωνιών*. (2004). Εκδόσεις Νέων Τεχνολογιών.
- Ένωση Πληροφορικών Ελλάδας. (2006). *Μελέτη Επισκόπησης της Πληροφορικής στην Ελλάδα*.
- Ζωγόπουλος, Ε. (2001). *Νέες Τεχνολογίες και Μέσα στην Εκπαιδευτική Διαδικασία*. Κλειδάριθμος.
- Θεοδόση, Α. (2021). Διπλωματική Εργασία Η ΕΦΑΡΜΟΣΙΜΟΤΗΤΑ ΤΩΝ ΠΡΟΓΡΑΜΜΑΤΩΝ STEM ΣΤΗΝ ΠΡΩΤΟΒΑΘΜΙΑ ΕΚΠΑΙΔΕΥΣΗ: ΜΙΑ ΕΜΠΕΙΡΙΚΗ ΜΕΛΕΤΗ. Πάτρα.

Ινστιτούτο Εκπαιδευτικής Πολιτικής. (2020). *ΟΔΗΓΙΕΣ ΓΙΑ ΤΗ ΔΙΑΧΕΙΡΙΣΗ ΤΗΣ ΥΛΗΣ ΓΙΑ ΤΟ ΜΑΘΗΜΑ «ΤΕΧΝΟΛΟΓΙΕΣ ΠΛΗΡΟΦΟΡΙΑΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ (Τ.Π.Ε)» ΤΟΥ ΔΗΜΟΤΙΚΟΥ ΣΧΟΛΕΙΟΥ ΓΙΑ ΤΟ ΣΧΟΛΙΚΟ ΕΤΟΣ 2020-2021.*

Καψιμάλη, Β. (χ.χ.). Ανάκτηση από <http://synedrio.pekap.gr/praktika/6o/ergasies/12-kapsimali.pdf>

Κομής, Β. (2004). *Εισαγωγή στις εκπαιδευτικές εφαρμογές των τεχνολογιών της πληροφορίας και των επικοινωνιών.* Εκδόσεις Νέων Τεχνολογιών.

Κομής, Β. (2005). *Εισαγωγή στην διδακτική της πληροφορικής.* Εκδόσεις Κλειδάριθμος.

Μαθηματικών, Ο. Ε. (χ.χ.). *STEMeducation.* Ανάκτηση από Ιστοσελίδα του Οργανισμού Εκπαιδευτικής Ρομποτικής Επιστήμης Τεχνολογίας & Μαθηματικών:  
<https://stem.edu.gr/%CF%84%CE%AF-%CE%B5%CE%AF%CE%BD%CE%B1%CE%B9-stem/>

Μαυραγάνης, Κ. (2015, Μάιος 23). Ανάκτηση από Ελληνική έκδοση της Hufffigton Post:  
[https://www.huffingtonpost.gr/2015/05/23/plhroforikh\\_n\\_7309422.html](https://www.huffingtonpost.gr/2015/05/23/plhroforikh_n_7309422.html)

Μπάρας, Ι., & Βασιλόπουλος, Γ. (2014). Διδάσκοντας προγραμματισμό με την χρήση Εκπαιδευτικής Ρομποτικής: Learning by doing. *8ου Πανελλήνιο Συνέδριο Καθηγητών Πληροφορικής «Η Πληροφορική στην Εκπαίδευση».*, (σσ. 1-6). Βόλος.

Πουλάκης, Ε. (2015). *Προγραμματίζοντας με τον μικροελεγκτή Arduino.* Ηράκλειο.

Σμυρναίου, Ζ. (2014). *Θεωρίες Μάθησης.* Ανάκτηση από <https://docplayer.gr/11132939-Paidagogika-enotita-g-didaktiki-mathisi-kai-didaskalia-zaharoyla-smyrnaioy-sholi-filosofias-tmima-paidagogikis-kai-psychologias.html>

Στυλιάρης, Γ., & Δήμου, Β. (2015). *Διδακτική της Πληροφορικής.*

Τερπένη, Ε. (2013). *Η Πληροφορική στην Α'/βάθμια Εκπαίδευση.* Καστοριά.

Τρακοσας, Δ. (2019). *Μελέτη Εφαρμογών Εκμάθησης Προγραμματισμού στην Πρωτοβάθμια Εκπαίδευση, Διπλωματική Εργασία, Πανεπιστήμιο Μακεδονίας.*

Τρομπούκη, Α. (2019). Μεταπτυχιακή Διπλωματική Εργασία Η ΕΚΠΑΙΔΕΥΤΙΚΗ ΡΟΜΠΟΤΙΚΗ ΩΣ ΠΑΡΑΓΟΝΤΑΣ ΜΕΤΑΒΟΛΗΣ ΤΩΝ ΣΤΑΣΕΩΝ ΤΩΝ ΜΑΘΗΤΩΝ ΑΠΕΝΑΝΤΙ ΣΤΙΣ ΕΠΙΣΤΗΜΕΣ STEM: ΕΚΤΙΜΗΣΕΙΣ ΓΟΝΕΩΝ.

ΦΕΚ 1324. (2016).

Ψυχάρης, Σ., & Καλοβρέκτης, Κ. (2018). Διδακτική & Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM & ΤΠΕ. Εκδόσεις Τζιόλα.

Ψυχάρης, Σ., & Καλοβρέκτης, Κ. (2018). Διδακτική και Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM και ΤΠΕ. Εκδόσεις ΤΖΙΟΛΑ.

Ψυχάρης, Σ., & Καλοβρέκτης, Κ. (2018). Εκπαιδευτική Ρομποτική. Στο Διδακτική & Σχεδιασμός Εκπαιδευτικών Δραστηριοτήτων STEM & ΤΠΕ (σσ. 394-410). Εκδόσεις ΤΖΙΟΛΑ.