



**ΕΛΛΗΝΙΚΟ ΑΝΟΙΧΤΟ ΠΑΝΕΠΙΣΤΗΜΙΟ**

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ**

Μεταπτυχιακή Εξειδίκευση στην Ανάπτυξη και Σχεδίαση Διάχυτων  
Συστημάτων Υπολογισμού

## ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Διδασκαλία Προγραμματισμού και Διάχυτος Υπολογισμός:  
Scratch – Mindstorms – LilyPad**

Μερκούρης Αλέξανδρος Α.Μ.: 73237

Επιβλέπων καθηγητής: Κωνσταντίνος Χωριανόπουλος

Πάτρα, Σεπτέμβριος 2014

Η παρούσα Διπλωματική Εργασία  
εκπονήθηκε στα πλαίσια των σπουδών  
για την απόκτηση του  
Μεταπτυχιακού Διπλώματος Ειδίκευσης  
που απονέμει το  
Τμήμα Θετικών Επιστημών & Τεχνολογίας  
του Ελληνικού Ανοικτού Πανεπιστημίου  
Τεχνολογία Υλικού και Λογισμικού:  
«Σχεδίαση και Ανάπτυξη Διάχυτων Συστημάτων Υπολογισμού»

**Όνοματεπώνυμο**  
**Επιβλέποντα**  
Κωνσταντίνος Χωριανόπουλος

**Όνοματεπώνυμο**  
**Μέλους 1**  
Γεώργιος Παλαιγεωργίου

**Όνοματεπώνυμο**  
**Μέλους 2**  
Αχιλλέας Καμέας

## **Ευχαριστίες**

Θέλω να ευχαριστήσω θερμά:

- Τον επιβλέποντα καθηγητή της διπλωματικής εργασίας μου κ. Κωνσταντίνο Χωριανόπουλο για την καθοδήγηση, τις πολύτιμες συμβουλές, τις χρήσιμες υποδείξεις του και τη βοήθειά του σε όλη τη διάρκεια της εκπόνησής της.
- Τον Δρ. Γιώργο Παλαιγεωργίου για τα εποικοδομητικά του σχόλια.
- Τους κ. Μπαλάφα Γ. και κ. Χατζημιχάλη Μ. για τη φιλολογική επιμέλεια της διπλωματικής εργασίας.
- Την οικογένεια μου και την Λίλα Παυλάκη για την κατανόηση τους και τη στήριξη που μου παρείχαν.

Μερκούρης Αλέξανδρος

Πάτρα, 2014

## Περίληψη

Μέσω της παρούσας διπλωματικής εργασίας επιδιώκεται, με τη διεξαγωγή μιας μελέτης περίπτωσης, να μελετηθούν και να συγκριθούν τα συναισθήματα των μαθητών, κατά την ενασχόληση τους με τρεις δημοφιλείς τεχνολογίες προγραμματισμού: την κλασική μέθοδο προγραμματισμού επιτραπέζιου υπολογιστή μέσω του περιβάλλοντος της Scratch, τον προγραμματισμό ρομπότ μέσω της πλατφόρμας ρομποτικής Lego Mindstorms και τον προγραμματισμό φυσικών αντικειμένων (e-textiles) μέσω της τεχνολογίας φορετού υπολογισμού LilyPad. Επίσης, αξιολογούνται τόσο η πρόθεση τους να ασχοληθούν με τον προγραμματισμό όσο και οι γνώσεις προγραμματισμού που αποκόμισαν κατά τη διάρκεια της μελέτης. Διερευνάται αν, όντως, οι τεχνολογίες προγραμματισμού φυσικών – απτών αντικείμενων υπερέχουν έναντι της κλασικής μεθόδου προγραμματισμού υπολογιστή.

Τα υποκείμενα της μελέτης ήταν 18 μαθητές και 18 μαθήτριες που φοιτούσαν στην Α' τάξη του Γυμνασίου, ενώ το πείραμα ήταν μια randomized within groups μελέτη με τη δημιουργία τριών ισάριθμων και ξεχωριστών ομάδων μαθητών και μαθητριών.

Σύμφωνα με τα ευρήματα της έρευνας ο προγραμματισμός ρομπότ, όντως, προκάλεσε θετικότερα συναισθήματα και αύξησε σε μεγαλύτερο βαθμό το ενδιαφέρον των μαθητών για προγραμματισμό σε σχέση με την κλασική μέθοδο προγραμματισμού. Ωστόσο η υπεροχή του προγραμματισμού φυσικών αντικειμένων e-textiles έναντι της κλασικής μεθόδου δεν επαληθεύτηκε, πλήρως, αφού δεν προέκυψε σημαντική στατιστική διαφορά στα συναισθήματα και στο ενδιαφέρον των μαθητών.

Επιπλέον το φυσικό αντικείμενο, είτε ρομπότ είτε e-textile, δεν επηρέασε καταλυτικά και στον αναμενόμενο βαθμό της επιδόσεις των μαθητών στην εκμάθηση των βασικών προγραμματιστικών εννοιών. Αναλύοντας τις επιδόσεις των μαθητών στην προγραμματιστική σκέψη με βάση το φύλο φάνηκε ότι τα κορίτσια της μελέτης υπερέχουν των αγοριών σε όλες τις κατηγορίες των αλγορίθμικών συνιστώσων που μελετήθηκαν.

Τέλος, αν και θεωρήθηκε ότι τα αγόρια θα έδειχναν μεγαλύτερο ενδιαφέρον στις δραστηριότητες ρομποτικής και τα κορίτσια στις δραστηριότητες φορετού υπολογισμού, στην πράξη και τα δύο φύλα έδειξαν ξεκάθαρη προτίμηση στον προγραμματισμό ρομπότ.

## **Λέξεις – κλειδιά**

Επιστήμη Υπολογιστών και Εκπαίδευση, προγραμματισμός, LilyPad Arduino, φορετός υπολογισμός, e-textiles, φύλο, παιδιά, Modkit, Scratch, Lego Mindstorms, κατασκευαστική θεωρία μάθησης.



# Περιεχόμενα

Περιεχόμενα.....	7
Κεφάλαιο 1 Εισαγωγή .....	8
Κεφάλαιο 2 Διδακτική του Προγραμματισμού και Διάχυτος Υπολογισμός.....	11
2.1 Παιδιά και Προγραμματισμός.....	11
2.2 Προγραμματισμός με φυσικά αντικείμενα .....	14
2.3 Κορίτσια και Προγραμματισμός.....	18
Κεφάλαιο 3 Μεθοδολογία.....	21
3.1 Επιλογή Υλικών και Προγραμματιστικών Εργαλείων .....	21
3.2 Σχεδιασμός Δραστηριοτήτων .....	24
3.3 Υποκείμενα και Σχεδιασμός Πειράματος .....	28
3.4 Εργαλεία Μέτρησης.....	29
3.4.1 Pre-test .....	29
3.4.2 Emotions – Test .....	31
3.4.3 Post – Test.....	32
3.4.4 Κριτήρια Αξιολόγησης .....	32
3.5 Διαδικασία .....	34
Κεφάλαιο 4 Παρουσίαση και Ανάλυση Δεδομένων Έρευνας.....	35
4.1 Έλεγχος Ισοδυναμίας Ομάδων .....	35
4.1.1 Έλεγχος Ισοδυναμίας με βάση την εισαγωγική μέθοδο προγραμματισμού .....	35
4.1.2 Έλεγχος Ισοδυναμίας με βάση το φύλο .....	36
4.2 Ανάλυση Αποτελεσμάτων σε σχέση με τη Μέθοδο Προγραμματισμού .....	37
4.2.1 Συναισθήματα Μαθητών .....	37
4.2.2 Ενδιαφέρον Μαθητών για Προγραμματισμό .....	40
4.2.3 Αξιολόγηση Δραστηριοτήτων .....	41
4.2.4 Επίδοση στην Προγραμματιστική Σκέψη .....	42
4.3 Ανάλυση Αποτελεσμάτων με βάση το φύλο .....	44
4.3.1 Συναισθήματα Αγοριών .....	44
4.3.2 Συναισθήματα Κοριτσιών .....	45
4.3.2 Συναισθήματα Αγοριών vs Κοριτσιών .....	46
4.3.3 Ενδιαφέρον Αγοριών – Κοριτσιών για Προγραμματισμό .....	47
4.3.4 Σύγκριση Επίδοσης Αγοριών – Κοριτσιών στην Προγραμματιστική Σκέψη .....	49
Κεφάλαιο 5 Συζήτηση – Συμπεράσματα .....	52
5.1 Προγραμματισμός με φυσικά αντικείμενα .....	52
5.2 Φύλο μαθητών και προγραμματισμός .....	54
5.4 Περιορισμοί .....	55
5.5 Συμπεράσματα .....	57
Βιβλιογραφία .....	58
Παράρτημα I Δραστηριότητες .....	63
ΧΡΙΣΤΟΥΓΕΝΝΙΑΤΙΚΟ ΔΕΝΤΡΟ ΜΕ ΦΩΤΑΚΙΑ .....	63
ΚΙΝΟΥΜΕΝΟ ROBOT NXT .....	80
ΜΟΝΟΧΡΩΜΑ ΦΩΤΑΚΙΑ LEDs ΠΟΥ ΑΝΑΒΟΣΒΗΝΟΥΝ.....	92

## Κεφάλαιο 1 Εισαγωγή

Η διδασκαλία και η εκμάθηση προγραμματισμού αποτελούν βασικά κομμάτια της εκπαίδευσης στον τομέα της επιστήμης των υπολογιστών. Ο προγραμματισμός, άλλωστε, έχει θεωρηθεί μια ευκαιρία για τους μαθητές να αναπτύξουν τις πνευματικές τους ικανότητες κατά την επίλυση απαιτητικών προβλημάτων [15].

Σε παγκόσμιο επίπεδο τα τελευταία χρόνια παρουσιάζεται μια αυξανόμενη τάση προσέλκυσης μαθητών από μικρή ηλικία σε διάφορες δραστηριότητες εκμάθησης προγραμματισμού. Στόχος είναι αφενός να αυξηθεί το ενδιαφέρον τους για την επιστήμη και την τεχνολογία και αφετέρου να ενθαρρυνθούν στο να παρακολουθήσουν στο μέλλον μαθήματα σχετικά με την επιστήμη των υπολογιστών. Θα καταστεί, έτσι, δυνατή η εμφύσηση των μαθητών από νωρίς στο «μικρόβιο» της τεχνολογίας, ώστε μελλοντικά, να μπορέσουν αυτοί να ακολουθήσουν τον τομέα της επιστήμης των υπολογιστών σε ακαδημαϊκό και επαγγελματικό επίπεδο, δεδομένου ότι στους συγκεκριμένους τομείς παρουσιάζεται έλλειμμα στο διαθέσιμο ανθρώπινο δυναμικό.

Νέες γλώσσες προγραμματισμού υπολογιστή για μαθητές, όπως η Scratch με το γραφικό περιβάλλον ανάπτυξης εφαρμογών, ήρθαν να πάρουν την θέση παλαιότερων γλωσσών, όπως η Logo. Τα πολύχρωμα γραφικά τους και ο εύκολος drag and drop τρόπος σύνθεσης προγραμμάτων έκαναν τον προγραμματισμό ελκυστικότερο και ευκολότερο για τα παιδιά, αφού πλέον για τη δημιουργία ενός προγράμματος δεν είναι απαραίτητη η εκμάθηση του αυστηρού συντακτικού μιας γλώσσας υψηλού επιπέδου.

Πέρα από τα φιλικά προς τους μαθητές γραφικά περιβάλλοντα προγραμματισμού, νέες διάχυτες τεχνολογίες προγραμματισμού δίνουν τη δυνατότητα στους μαθητές να ξεφύγουν από τον κλασικό προγραμματισμό επιτραπέζιου υπολογιστή. Αυτές οι διάχυτες τεχνολογίες τους επιτρέπουν να πειραματιστούν προγραμματίζοντας κινητά τηλέφωνα μέσω της τεχνολογίας App Inventor, ρομπότ μέσω της πλατφόρμας Lego Mindstorms, ακόμα και φυσικά αντικείμενα (e-textiles), όπως ρούχα και παιχνίδια, ενσωματώνοντας πάνω τους μικροεπεξεργαστές μέσω της τεχνολογίας φορετού υπολογισμού (wearable computing) του LilyPad. Οι μαθητές μπορούν, λοιπόν, να προγραμματίζουν φυσικά αντικείμενα, ακόμα και δικές τους δημιουργίες, που έχουν υπόσταση στον πραγματικό κόσμο και όχι μόνο να δημιουργούν και να προγραμματίζουν εικονικά αντικείμενα που τους παρουσιάζονται στον εικονικό κόσμο της οθόνης ενός επιτραπέζιου υπολογιστή.

Μέσω της παρούσας διπλωματικής εργασίας επιδιώκεται, με τη διεξαγωγή μιας μελέτης περίπτωσης, να μελετηθούν και να συγκριθούν τα συναισθήματα των μαθητών, κατά την ενασχόληση τους με τον προγραμματισμό φυσικών και εικονικών αντικειμένων. Επίσης, αξιολογούνται τόσο η πρόθεση τους να ασχοληθούν με τον προγραμματισμό όσο και οι γνώσεις προγραμματισμού που αποκόμισαν κατά τη διάρκεια της μελέτης.

Συγκεκριμένα, κατά τη διάρκεια της μελέτης οι μαθητές ήρθαν σε επαφή με τρεις δημοφιλείς τεχνολογίες προγραμματισμού:

1. Την κλασική μέθοδο προγραμματισμού επιτραπέζιου υπολογιστή μέσω του περιβάλλοντος της Scratch.
2. Τον προγραμματισμό ρομπότ μέσω της πλατφόρμας ρομποτικής Lego Mindstorms.
3. Τον προγραμματισμό φυσικών αντικειμένων (e-textiles) μέσω της τεχνολογίας φορετού υπολογισμού LilyPad.

Πολλοί ερευνητές στον τομέα της επιστήμης των υπολογιστών, βασιζόμενοι στη κατασκευαστική θεωρία μάθησης, έχουν την άποψη οι μαθητές αντιλαμβάνονται καλύτερα τις προγραμματιστικές έννοιες αλληλεπιδρώντας με απτά φυσικά αντικείμενα και χρησιμοποιώντας τις αισθήσεις τους. Θεωρούν ότι οι τεχνολογίες προγραμματισμού φυσικών αντικειμένων πλεονεκτούν έναντι της κλασικής μεθόδου προγραμματισμού [29, 38, 39, 41, 43]. Επιπλέον, κάποιοι ερευνητές πιστεύουν ότι οι τεχνολογίες προγραμματισμού φυσικών αντικειμένων δύνανται να εμπνεύσουν μεγαλύτερο αριθμό κοριτσιών να ακολουθήσουν την επιστήμη των υπολογιστών [10, 43, 68].

Στόχος της εργασίας, λαμβάνοντας υπόψη τα παραπάνω, είναι να δοθεί απάντηση στις εξής ερευνητικές ερωτήσεις – υποθέσεις:

- Ποια είναι η συναισθηματική στάση των μαθητών κατά την ενασχόληση τους με τις προαναφερθείσες τεχνολογίες προγραμματισμού; Επιπλέον, κατά πόσον ο προγραμματισμός φυσικών αντικειμένων, όπως ρομπότ και e-textiles, είναι πιο ευχάριστος για τους μαθητές από τον κλασικό τρόπο προγραμματισμού επιτραπέζιου υπολογιστή;
- Το φυσικό αντικείμενο, είτε είναι ρομπότ είτε είναι wearable e-textile, εμπνέει περισσότερο τους μαθητές σε σχέση με τον αμιγώς εικονικό κόσμο

που προσφέρει το γραφικό περιβάλλον μιας γλώσσας προγραμματισμού ενός κλασικού υπολογιστή; Μπορεί έτσι να αυξηθεί το ενδιαφέρον τους για ενασχόληση με δραστηριότητες προγραμματισμού;

- Ποιο ρόλο διαδραματίζει το φύλο του μαθητή; Τα αγόρια δείχνουν, πιθανόν, μεγαλύτερο ενδιαφέρον για τη ρομποτική σε σχέση με τα κορίτσια που, ίσως, εμφανίζουν μια μεγαλύτερη ροπή προς το φορετό υπολογισμό (wearable computing);
- Μέσω ποιας τεχνολογίας, τέλος, οι μαθητές δύνανται να αναπτύξουν αποδοτικότερα τις δεξιότητές τους στον προγραμματισμό; Ποιος είναι ο βαθμός εμπέδωσης στην πράξη τριών βασικών προγραμματιστικών συνιστωσών, δηλαδή της δομής ακολουθίας (sequence), της δομής επανάληψης (repeat) και της δομής επιλογής (if-else), κατά την ενασχόληση των μαθητών με καθεμία από τις παραπάνω τεχνολογίες προγραμματισμού;

## **Κεφάλαιο 2 Διδακτική του Προγραμματισμού και Διάχυτος Υπολογισμός**

### **2.1 Παιδιά και Προγραμματισμός**

Καθώς οι υπολογιστές διαδραματίζουν όλο και σημαντικότερο ρόλο στην καθημερινή ζωή των ανθρώπων, τόσο η εξοικείωση με αυτούς όσο και η ικανότητα προγραμματισμού τους αποκτούν βαρύνουσα σημασία. Μεγάλος αριθμός ερευνητών διεξάγουν έρευνες ώστε να ανακαλυφθούν νέες μέθοδοι που θα παροτρύνουν αρχάριους χρήστες, όπως είναι τα παιδιά, να ασχοληθούν με τον προγραμματισμό υπολογιστών [12]. Η εκμάθηση προγραμματισμού, άλλωστε, έχει θετικά και μετρήσιμα αποτελέσματα, όχι μόνο στα μαθηματικά και στη φυσική αλλά και στην καλλιέργεια των γλωσσικών δεξιοτήτων, καθώς και στη δημιουργικότητα των μαθητών [11, 14, 28]. Ενώ οι μαθητές, σε γενικές γραμμές, είναι πλέον αρκετά εξοικειωμένοι με τη χρήση υπολογιστικών μηχανών, όπως προσωπικών υπολογιστών, παιχνιδομηχανών και έξυπνων τηλεφώνων, δε συμβαίνει το ίδιο και με την ικανότητα τους στον προγραμματισμό. Μεταφορικά μπορεί να υποστηριχθεί ότι οι μαθητές γνωρίζουν «ανάγνωση», μιας και είναι αρκετά εξοικειωμένοι με τη χρήση υπολογιστή, αλλά δυσκολεύονται στη «γραφή», αφού η πλειοψηφία αυτών δεν έχει έρθει σε επαφή με τον προγραμματισμό· ακόμα και όταν οι μαθητές έρχονται σε επαφή με αυτόν μέσω του περιβάλλοντος μιας γλώσσας προγραμματισμού, δυσκολεύονται αρκετά στην εκμάθηση των προγραμματιστικών εννοιών [34].

Τα τελευταία χρόνια είχε παρατηρηθεί μια σημαντική πτώση στους αριθμούς των φοιτητών που ακολουθούν την επιστήμη των υπολογιστών [24, 65]. Ένας βασικός λόγος γι' αυτήν τη μείωση του ενδιαφέροντος θωρείται το γεγονός ότι οι προγραμματιστικές έννοιες είναι αρκετά αφηρημένες για να τις αντιλαμβάνονται εύκολα οι μαθητές [37]. Σύμφωνα με το Ινστιτούτο Έρευνας για την Ανώτατη Εκπαίδευση του Πανεπιστημίου της Καλιφόρνια, το ποσοστό των εισερχόμενων προπτυχιακών φοιτητών που επέλεγαν την επιστήμη των υπολογιστών είχε μειωθεί κατά 60 τις εκατό μεταξύ των ετών 2000 και 2004, ήταν δε μικρότερο απ' ό,τι ήταν στις αρχές τις δεκαετίας του 80 [65]. Επιπλέον, τα τελευταία χρόνια ο αριθμός των φοιτητών με πτυχίο στην επιστήμη των υπολογιστών είχε φθάσει στο κατώτερο σημείο του, σε επίπεδο σχεδόν 70 τις εκατό μικρότερο σε σχέση με τους αριθμούς των φοιτητών το 2000. Ανάλογες έρευνες [32] επιβεβαιώνουν τη μείωση του

ενδιαφέροντος για την επιστήμη των υπολογιστών και στο πλαίσιο της Δευτεροβάθμιας εκπαίδευσης στις Η.Π.Α.

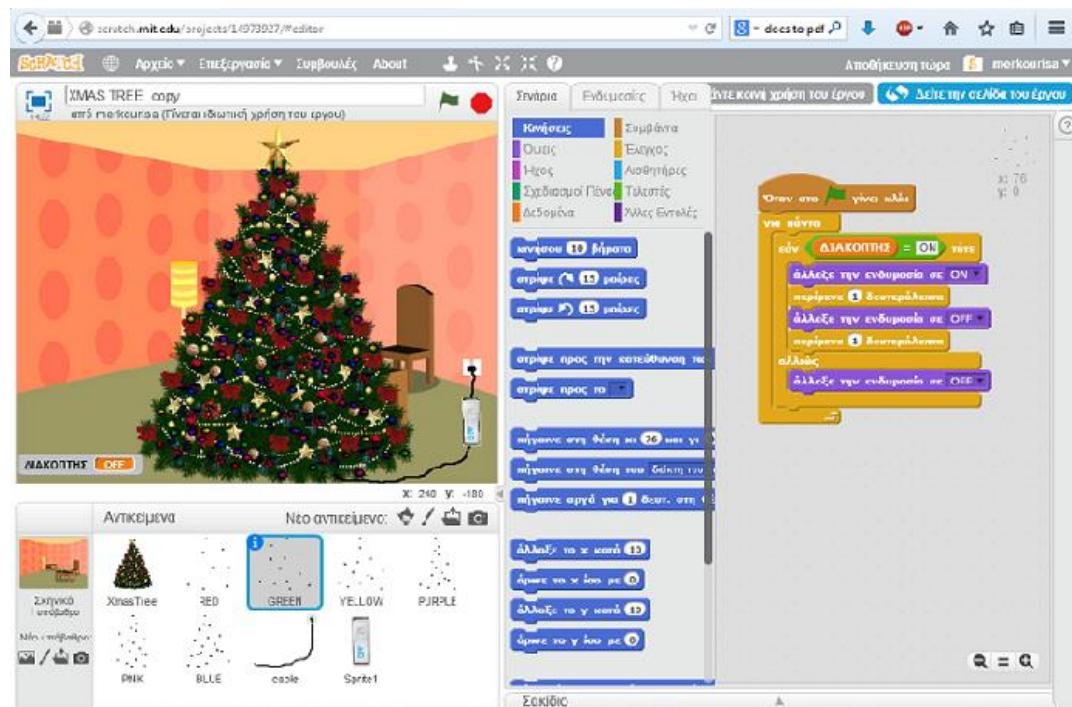
Αυτή η απότομη πτώση του ενδιαφέροντος για τον προγραμματισμό έρχεται σε αντίθεση με τα στοιχεία ζήτησης εργασίας που προέρχονται από το Αμερικάνικο Γραφείο Στατιστικών Εργασίας (US Bureau of Labor Statistics). Σύμφωνα με συγκεκριμένα δεδομένα [23], μόνο για τη δεκαετία 2006-2016 θα δημιουργηθούν 700.000 νέες θέσεις εργασίας στην Αμερική αποκλειστικά σχετιζόμενες με τον τομέα των υπολογιστικών συστημάτων. Ο συγκεκριμένος αριθμός είναι πολύ μεγαλύτερος από τον αναμενόμενο αριθμό πτυχίων σχετικών με την πληροφορική, που πρόκειται να χορηγηθούν στις Η.Π.Α. Γίνεται, λοιπόν, εύκολα αντιληπτό ότι οι Η.Π.Α. δε θα είναι σε θέση να καλύψουν από μόνες τους τη ζήτηση για εργασία στο συγκεκριμένο τομέα.

Μεγάλες προσπάθειες, πάντως, γίνονται σε διεθνές επίπεδο από διάφορους ερευνητές, ώστε να αντιμετωπιστεί και να αναστραφεί η συγκεκριμένη αρνητική κατάσταση. Βασικός στόχος αποτελεί η βελτίωση των προσφερόμενων μαθημάτων προγραμματισμού στους μαθητές, τόσο ως προς το περιεχόμενο όσο και ως προς την παιδαγωγική προσέγγιση [62]. Νέες μέθοδοι, εργαλεία και τεχνολογίες έχουν δημιουργηθεί ώστε να κεντρίσουν το ενδιαφέρον των μαθητών στην εκμάθηση προγραμματισμού [12]. Και όντως, τελευταία υπάρχουν αρκετές καταγεγραμμένες επιτυχίες τόσο ως προς την εύκολη προσέλκυση όσο και τη διατήρηση του ενδιαφέροντος των μαθητών στα μαθήματα της επιστήμης των υπολογιστών και, πιο συγκεκριμένα της εκμάθησης του προγραμματισμού.

Από τη δεκαετία του '60 έχει δημιουργηθεί ένας μεγάλος αριθμός γλωσσών προγραμματισμού, που απευθύνονται σε αρχάριους χρήστες και, πιο συγκεκριμένα, σε μαθητές. Μια από τις πρώτες σημαντικές γλώσσες προγραμματισμού υπολογιστή για μαθητές ήταν και η Logo [41]. Η συγκεκριμένη γλώσσα κατέχει ακόμα και σήμερα θέση και στο Ελληνικό εκπαιδευτικό σύστημα, καθώς είναι προτεινόμενη για χρήση στο πρόγραμμα σπουδών της Γ' τάξης του Γυμνασίου [33, 45] ως εργαλείο εκμάθησης προγραμματισμού. Αν και η γλώσσα Logo κατέστη πολύ δημοφιλής, αρκετές μελέτες απέτυχαν να καταδείξουν σημαντικά οφέλη και η κοινότητα Logo δεν ευδοκίμησε στο βαθμό που θα ήλπιζε [19, 42]. Πολλές από τις προκλήσεις που καλούνται οι μαθητές να αντιμετωπίσουν κατά την εκμάθηση προγραμματισμού μέσω της Logo οφείλονται στους περιορισμούς του προγραμματιστικού περιβάλλοντος της γλώσσας. Αυτό το εύθραυστο περιβάλλον αναγκάζει τους μαθητές να εστιάζουν αρκετά την προσοχή τους

στη σωστή σύνταξη του κώδικα, με αποτέλεσμα να μην επικεντρώνονται όσο θα έπρεπε στην σημασιολογία αυτού [15].

Νέας γενιάς γλώσσες προγραμματισμού όπως η Turtle Art [64], η StarLogo-TNG [61], η Alice [3], η StoryTelling Alice [13], και η Scratch [60], στηρίχθηκαν στις βασικές αρχές της γλώσσας Logo, ωστόσο έχουν υιοθετήσει ένα διαφορετικό τρόπο δημιουργίας προγραμμάτων. Η διαδικασία μοιάζει με την κατασκευή ενός παζλ στο οποίο ενώνονται κατάλληλα τα κομμάτια που το αποτελούν. Τα προγράμματα δημιουργούνται εύκολα με τη σύνδεση των διαφόρων οπτικών στοιχείων που αντιστοιχούν σε εντολές. Οι μαθητές για να συνθέσουν ένα πρόγραμμα απλά σύρουν και αφήνουν (drag and drop) τα οπτικά στοιχεία στα κατάλληλα σημεία. Το κλείδωμα των οπτικών στοιχείων ανάμεσά τους, δηλαδή η σύνθεση του προγράμματος, γίνεται με τέτοιον τρόπο που δεν επιτρέπεται στους μαθητές να κάνουν συντακτικά λάθη. Τα μόνα λάθη στα οποία μπορούν να υποπέσουν οι μαθητές, κατά τη διάρκεια της δημιουργίας ενός προγράμματος, είναι τα λογικά. Έτσι οι μαθητές επικεντρώνονται αποκλειστικά στη σημασιολογία του κώδικα και δεν αποσπώνται με την εκμάθηση του συντακτικού της γλώσσας, όπως συμβαίνει με την Logo. Αυτός ο εύκολος τρόπος δημιουργίας προγραμμάτων και τα πολύχρωμα γραφικά των οπτικών γλωσσών νέας γενιάς έχουν συμβάλει στο να γίνει ελκυστικότερος και πιο ενδιαφέρον ο προγραμματισμός ανάμεσα στους μαθητές.



Εικόνα 1: Περιβάλλον ανάπτυξη εφαρμογών της Scratch 2.0

Πιο συγκεκριμένα, μέσω του δημοφιλούς γραφικού περιβάλλοντος ανάπτυξης εφαρμογών για υπολογιστές της Scratch [46], οι μαθητές μπορούν ως επί το πλείστον να δημιουργούν και να προγραμματίζουν εικονικούς κόσμους, όπως παιχνίδια, ιστορίες και πολυμεσικές εφαρμογές για προσωπικούς υπολογιστές. Παράλληλα, μπορούν να μοιραστούν τα προγράμματά τους με άλλους χρήστες στην κοινότητα ανοιχτού λογισμικού της Scratch. Επίσης, με την τελευταία έκδοση της γλώσσας Scratch 2.0 οι επίδοξοι προγραμματιστές έχουν την επιπλέον δυνατότητα να συνδέσουν τα προγράμματά τους και με τον υλικό κόσμο, ελέγχοντας φυσικά αντικείμενα όπως το PicoBoard [57], ρομπότ μέσω της πλατφόρμας ρομποτικής Lego WeDo [40] και Lego Mindstorms [39], μικρόφωνα, κάμερες (Microsoft Kinect [44]) ακόμα και μικροεπεξεργαστές Arduino [8]. Διαπιστώνεται λοιπόν μια τάση σύνδεσης της Scratch με το φυσικό κόσμο.

Πέρα, βέβαια, από τα περιβάλλοντα προγραμματισμού προσωπικών υπολογιστών, αρχάριοι χρήστες τελευταία έχουν τη δυνατότητα μέσω της γλώσσας προγραμματισμού App Inventor [26] του MIT να προγραμματίσουν και smart-phones ή tablets με λειτουργικό σύστημα Android. Η δημιουργία εφαρμογών για κινητές συσκευές κεντρίζει σε μεγάλο βαθμό το ενδιαφέρον των μαθητών εφόσον έχουν τη δυνατότητα να συνθέσουν το δικό τους εξατομικευμένο λογισμικό για μια πολύτιμη γι' αυτούς συσκευή, όπως είναι το κινητό τους τηλέφωνο. Το App Inventor χρησιμοποιεί μια οπτική, κινούμενη από το γεγονός, γλώσσα προγραμματισμού που αμβλύνει τα προβλήματα που έχουν να κάνουν με το συντακτικό μιας γλώσσας υψηλού επιπέδου, όπως για παράδειγμα η JAVA, με αποτέλεσμα να είναι προσιτή στους αρχάριους χρήστες που κάνουν τα πρώτα τους βήματα στον προγραμματισμό. Παράλληλα, τα παραγόμενα προγράμματα έχουν αντίκτυπο σε καθημερινές προσωπικές συσκευές του χρήστη [18, 58].

## 2.2 Προγραμματισμός με φυσικά αντικείμενα

Σύμφωνα με τους Froebel, Montessori, Vygotsky και Piaget, τα παιδιά μαθαίνουν με βιωματικό τρόπο, είτε χρησιμοποιώντας τις αισθήσεις τους είτε μέσω φυσικής ή κοινωνικής αλληλεπίδρασης (hands on learning theory). Αυτή η βιωματική γνώση αποτελεί κρίσιμο παράγοντα για την πνευματική ανάπτυξη των παιδιών [25, 48, 49, 56, 67]. Τη συγκεκριμένη αρχή ενστερνίσθηκαν και πολλοί ερευνητές στον τομέα της επιστήμης των υπολογιστών, για να απλοποιήσουν τις αφηρημένες προγραμματιστικές έννοιες και να τις καταστήσουν φιλικότερες προς τους χρήστες. Με βάση τις απόψεις

τους, τα παιδιά αντιλαμβάνονται καλύτερα τις προγραμματιστικές έννοιες αλληλεπιδρώντας με απτά φυσικά αντικείμενα και χρησιμοποιώντας τις αισθήσεις τους [37]. Για αυτόν ακριβώς το λόγο, έχουν δημιουργηθεί ικανά προγραμματιστικά περιβάλλοντα και τεχνολογίες προγραμματισμού που στόχο έχουν την αλληλεπίδραση ή τον έλεγχο απτών φυσικών αντικειμένων και όχι μόνο εικονικών αντικειμένων που εμφανίζονται στην οθόνη ενός υπολογιστή ή ενός κινητού τηλεφώνου.

Ένας από τους πρώτους ερευνητές, ειδικότερα, που στηρίχθηκε στις συγκεκριμένες αρχές είναι ο Seymour Papert ο οποίος, τη δεκαετία του 1970, βασιζόμενος στην έρευνα του Piaget ανέπτυξε την κατασκευαστική θεωρία μάθησης (constructionism). Η θεωρία του Papert βασίζεται σε δυο βασικές έννοιες κατασκευής της γνώσης. Κατά πρώτον, τα παιδιά μαθαίνουν συμμετέχοντας ενεργητικά στην κατασκευή της νέας γνώσης, και όχι απλά λαμβάνοντας παθητικά τις πληροφορίες που τους παρέχονται. Κατά δεύτερον, η αποτελεσματική μάθηση λαμβάνει χώρα όταν ο μαθητής ασχολείται με την κατασκευή προσωπικών αντικειμένων που έχουν νόημα για τον ίδιο, όπως ρομπότ, παιχνίδια, φυσικά αντικείμενα, animation υπολογιστή. Ο Papert εφάρμοσε την συγκεκριμένη θεωρία για να αναπτύξει την γλώσσα προγραμματισμού Logo [41]. Χρησιμοποιώντας απλές εντολές μπορούσε να ελεγχθεί η κίνηση ενός ρομπότ που είχε την μορφή χελώνας. Στο συγκεκριμένο σύστημα, η έξοδος ήταν απτό φυσικό αντικείμενο ενώ η είσοδος παρέμενε αφηρημένη προγραμματιστική έννοια.

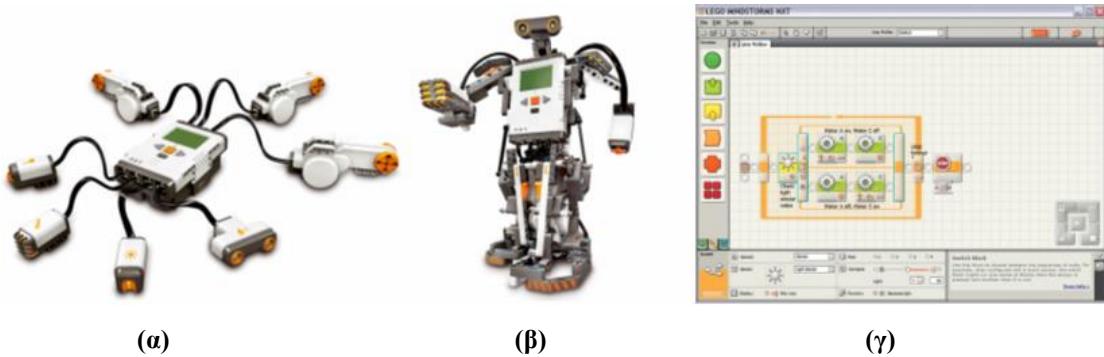
Προς αυτήν την κατεύθυνση κινείται και το σύστημα AlgoBlocks [29]. Η συγκεκριμένη γλώσσα προγραμματισμού αποτελείται από ένα σύνολο απτών δομικών στοιχείων προγραμματισμού μέσω των οποίων δύναται να ελεγχθεί η κίνηση ενός εικονικού αντικειμένου (υποβρυχίου) στην οθόνη ενός υπολογιστή. Ακολουθεί, δηλαδή, αντίθετη προσέγγιση από την Logo αφού η είσοδος αποτελείται από φυσικά αντικείμενα ενώ η έξοδος είναι εικονική.

To 1997 ο Ishii πρότεινε μια νέα διεπαφή χρήστη που την ονόμασε «**Tangible User Interface**» [30]. Η διεπαφή **TUI** χρησιμοποιεί φυσικά αντικείμενα για να αναπαραστήσει την ψηφιακή πληροφορία. Βασικός στόχος της είναι να γεφυρώθει το χάσμα ανάμεσα στο εικονικό και πραγματικό περιβάλλον. Ο χρήστης μέσω της διεπαφής TUI μπορεί πλέον να αλληλεπιδράσει φυσικότερα με την ψηφιακή πληροφορία μέσω των περιφερειακών του αισθήσεων και να τη διαχειριστεί άμεσα με την αφή του [37].

Μια από τις δημοφιλέστερες τεχνολογίες προγραμματισμού για παιδιά που περιλαμβάνουν αλληλεπίδραση με φυσικά αντικείμενα αποτελεί η πλατφόρμα

ρομποτικής LEGO Mindstorms [39] που πρωτοεμφανίστηκε στο εμπόριο το 1998. Η ιδέα χρήσης ρομπότ στην εκπαίδευση στηρίζεται στην προηγούμενη έρευνα του Papert. Μέσω της ρομποτικής, τα παιδιά ξεφεύγουν από τον παραδοσιακό τρόπο προγραμματισμού υπολογιστή. Ο Papert προσπάθησε σε συνεργασία με την εταιρία Lego να δημιουργήσει ένα περιβάλλον προγραμματισμού ρομπότ, γιατί πίστευε ότι τα παιδιά μπορούσαν να ταυτιστούν με τα ρομπότ που είναι φυσικές, απέξ αναπαραστάσεις του υπολογιστή και των προγραμμάτων του. Θεωρούσε, επίσης, ότι τα παιδιά αλληλεπιδρώντας με τα ρομπότ μπορούν να κατανοήσουν ευκολότερα αφηρημένες έννοιες. Η υλική φύση του ρομπότ, σύμφωνα με τον Papert, αποτελεί το κύριο πλεονέκτημα του.

Εξειδικεύοντας, η πλατφόρμα ρομποτικής LEGO Mindstorms αποτελείται από ένα ευφυές τουβλάκι-μικροεπεξεργαστή (NXT) μέσω του οποίου μπορούν να ελεγχθούν κινητήρες και αισθητήρες που είναι συνδεδεμένοι σε αυτό. Οι χρήστες χρησιμοποιώντας και δομικά στοιχεία LEGO έχουν τη δυνατότητα να κατασκευάσουν ρομπότ και, στη συνέχεια, μπορούν να τα προγραμματίσουν κατάλληλα ώστε να επιτελούν συγκεκριμένα καθήκοντα.



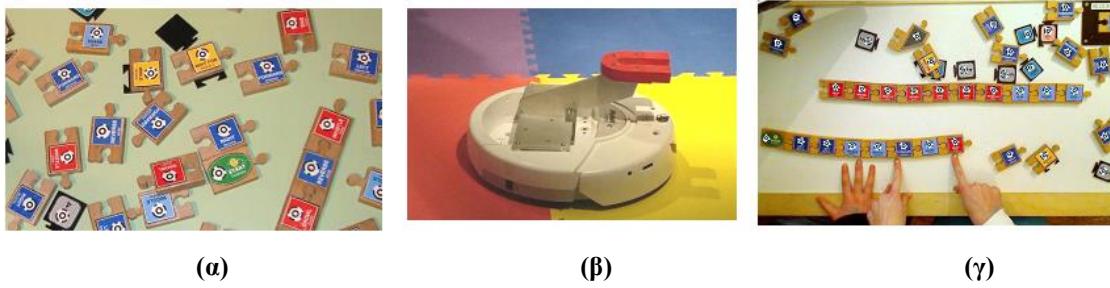
**Εικόνα 2:** (α) Ο μικροεπεξεργαστής NXT συνδεμένος με αισθητήρες και κινητήρες, (β) Ένα ρομπότ Lego Mindstorms και (γ) το περιβάλλον ανάπτυξης εφαρμογών NXT-G.

Η πλατφόρμα Mindstorms έχει χρησιμοποιηθεί ευρέως σε σχολικά πλαίσια για τη διδασκαλία έννοιών προγραμματισμού και μηχανικής [1, 20, 27, 52, 53, 59]. Υπάρχουν ερευνητές που υποστηρίζουν ότι τα ρομπότ συμβάλλουν σε σημαντικό βαθμό στην εκμάθηση βασικών προγραμματιστικών έννοιών [10, 36]. Έρευνα [16] που διεξήχθη με μαθητές Γυμνασίου αναφέρει θετικά αποτελέσματα τόσο στο ενδιαφέρον των μαθητών για το μάθημα όσο και στην επίτευξη των εκπαιδευτικών στόχων του μαθήματος. Υπάρχουν, βέβαια, και περιπτώσεις που τα συμπεράσματα ερευνών ήταν

ασαφή όσον αφορά στα εκπαιδευτικά κέρδη της χρήσης ρομπότ στην εκμάθηση προγραμματισμού [17, 50].

Έχοντας, λοιπόν, ως οδηγό τις προαναφερθείσες προσπάθειες, πολλοί ερευνητές άρχισαν να ασχολούνται με την ανάπτυξη διάχυτων απτών τεχνολογιών εκμάθησης προγραμματισμού. Σε αυτό τους διευκόλυνε και η ραγδαία ανάπτυξη της τεχνολογίας, αφού καλύτερα υλικά και εργαλεία ανάπτυξης, όπως RFID και μικροεπεξεργαστές, είναι διαθέσιμα.

Χαρακτηριστικό παράδειγμα αποτελεί η Tern [43], μια γλώσσα προγραμματισμού για μαθητές που υλοποιεί μια TUI διεπαφή χρήστη. Αποτελείται από μια συλλογή ξύλινων κομματιών που έχουν το σχήμα κομματιών πάζλ. Οι μαθητές συνδέοντας κατάλληλα τα κομμάτια δημιουργούν προγράμματα υπολογιστή, που έχουν υλική υπόσταση στον φυσικό κόσμο, με στόχο να ελέγξουν την κίνηση ενός ρομπότ.

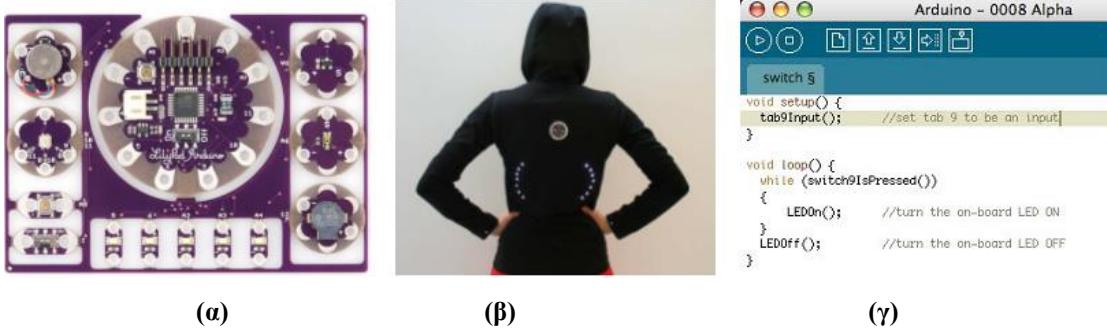


**Εικόνα 3:** (a) Κομμάτια που αντιστοιχούν σε εντολές της Tern (b) Πρόγραμμα γραμμένο σε Tern (γ) το i-Robot που εκτελούσε τις εντολές των προγραμμάτων

Άλλα παραδείγματα tangible γλωσσών προγραμματισμού αποτελούν τα Tangible Programming Blocks [63], SystemBlocks [54] και το FlowBlocks [55].

Πέρα, όμως, από τις tangible γλώσσες προγραμματισμού, τα τελευταία χρόνια παρατηρείται μια ταχεία ανάπτυξη τόσο στον τομέα του φορετού υπολογισμού (wearable computing) όσο και σε αυτόν του προγραμματισμού e-textiles. Νέες τεχνολογίες, όπως το LilyPad Arduino [38], έχουν καταστήσει το φορετό υπολογισμό πιο προσιτό στο ευρύτερο κοινό. Συγκεκριμένα, το LilyPad Arduino είναι ένα τεχνολογικό πακέτο ηλεκτρονικών εξαρτημάτων, που επιτρέπει σε αρχάριους να σχεδιάσουν και να δημιουργήσουν τα δικά τους διαδραστικά φορετά αντικείμενα (e-textiles) ενσωματώνοντας μικροεπεξεργαστές, αισθητήρες, κινητήρες, ηχεία και φωτάκια LED's σε φυσικά υφασμάτινα αντικείμενα. Η ενσωμάτωση αυτή γίνεται πολύ εύκολα ράβοντας τα διάφορα ηλεκτρονικά εξαρτήματα του πακέτου με τη χρήση αγώγιμης κλωστής η οποία συγκρατεί τα εξαρτήματα πάνω στο υφασμάτινο αντικείμενο, αλλά δρα ταυτόχρονα και σαν καλώδιο για την επικοινωνία ανάμεσα στον

μικροεπεξεργαστή και τα υπόλοιπα περιφερειακά ηλεκτρονικά εξαρτήματα. Από την στιγμή που οι χρήστες ενσωματώσουν τα διάφορα ηλεκτρονικά εξαρτήματα στο φυσικό αντικείμενο, δημιουργώντας έτσι ένα e-textile, έχουν την δυνατότητα να προγραμματίσουν τον μικροεπεξεργαστή – εγκέφαλο του LilyPad ελέγχοντας έτσι τις αλληλεπιδράσεις με τα υπόλοιπα εξαρτήματα.



**Εικόνα 4 (a) Η πλακέτα ανάπτυξης του LilyPad (β) LilyPad ενσωματωμένο σε μπλούζα (γ) Το περιβάλλον ανάπτυξης εφαρμογών Arduino.**

Η τεχνολογία φορετού υπολογισμού του LilyPad αποτέλεσε πηγή έμπνευσης για μια σειρά μαθημάτων που εστιάζουν στο φορετό υπολογισμό με σκοπό να διδάξουν τους μαθητές αντικείμενα σχετικά με την τεχνολογία και, ειδικότερα, τον προγραμματισμό [21, 38]. Οι μαθητές, σε αυτά τα μαθήματα, αρχικά σχεδιάζουν ρούχα ή άλλα υφασμάτινα αντικείμενα ενσωματώνοντας πάνω τους ηλεκτρονικά εξαρτήματα, όπως αισθητήρες και φωτάκια LED's, και στη συνέχεια έρχονται σε επαφή με τον προγραμματισμό δημιουργώντας προγράμματα που «τρέχουν» στις δημιουργίες τους. Μέσω, λοιπόν, των συγκεκριμένων δραστηριοτήτων οι μαθητές διδάσκονται αντικείμενα σχετικά με την τεχνολογία μέσα σε ένα φουτουριστικό, καινοτόμο και δημιουργικό πλαίσιο που προκαλεί σε μεγάλο βαθμό το ενδιαφέρον τους.

## 2.3 Κορίτσια και Προγραμματισμός

Καθώς η τεχνολογία γίνεται όλο και πιο διάχυτη στην καθημερινή ζωή των ανθρώπων, τόσο αυξάνεται και η ανάγκη αυτή να υποστηρίζει ισότιμα όλα τα μέλη της κοινωνίας [2, 13]. Για να διασφαλιστεί ότι η τεχνολογία καλύπτει ικανοποιητικά τις ανάγκες τις κοινωνίας, κρίνεται επιτακτική η ανάγκη για συμμετοχή ενός αντιπροσωπευτικού δείγματος του πληθυσμού στις διαδικασίες ανάπτυξης νέων τεχνολογιών. Αν και υπάρχουν πολλές κοινωνικές ομάδες του πληθυσμού που δεν εκπροσωπούνται επαρκώς στον τομέα της επιστήμης των υπολογιστών, οι γυναίκες

αποτελούν αναμφισβήτητα την πολυπληθέστερη [2, 66]. Ανισότητα ανάμεσα στα φύλα παρατηρείται και στον τομέα της εκπαίδευσης της επιστήμης των υπολογιστών, παρά τις σημαντικές προσπάθειες για εξισορρόπηση [10].

Παρά το γεγονός ότι στην παρούσα κατάσταση η πλάστιγγα γέρνει ποσοτικά εις βάρος των γυναικών σε σχέση με τους αριθμούς των αντρών που ασχολούνται με την επιστήμη των υπολογιστών, ο Kelleher υποστηρίζει [13] ότι τόσο το ενδιαφέρον για προγραμματισμό όσο και οι επιδόσεις σε αυτόν φαίνεται να εξαρτώνται από τις προηγούμενες εμπειρίες προγραμματισμού και τον χρόνο ενασχόλησης με αυτόν και όχι τόσο από το φύλο. Με βάση τα παραπάνω, πολλοί ερευνητές προσπαθούν να αναπτύξουν μεθόδους και εργαλεία που αφενός μεν θα κεντρίσουν το ενδιαφέρον των κοριτσιών όσον αφορά στην επιστήμη των υπολογιστών, αφετέρου δε θα τους προσφέρουν θετικές εμπειρίες όσον αφορά στην ενασχόληση με τον προγραμματισμό. Χαρακτηριστικό παράδειγμα τέτοιων εργαλείων αποτελεί η γλώσσα προγραμματισμού Storytelling Alice [13] καθώς και η τεχνολογία φορετού υπολογισμού LilyPad [38].

Η Storytelling Alice αποτελεί ένα από τα πρώτα ισχυρά βήματα προς ένα περιβάλλον προγραμματισμού το οποίο μπορεί να προσφέρει στα κορίτσια μια πρώτη θετική εμπειρία όσον αφορά τον προγραμματισμό υπολογιστή. Αυτές οι πρώτες θετικές εμπειρίες προγραμματισμού υπολογιστή δύνανται να εμπνεύσουν μεγαλύτερο αριθμό κοριτσιών να ακολουθήσουν την επιστήμη των υπολογιστών.

Αντίστοιχους στόχους και αποτελέσματα εμφανίζει και η διάχυτη τεχνολογία φορετού υπολογισμού LilyPad. Ερευνητές που έχουν υλοποιήσει και εφαρμόσει δραστηριότητες φορετού υπολογισμού [68], μέσω της τεχνολογίας LilyPad, έχουν καταλήξει στο συμπέρασμα ότι ο φορετός υπολογισμός είναι φιλικότερος προς τα κορίτσια σε σχέση με δραστηριότητες ρομποτικής, επειδή ενσωματώνει στοιχεία σχεδιασμού και μόδας, άρα μπορεί να κεντρίσει σε μεγαλύτερο βαθμό το ενδιαφέρον των κοριτσιών όσον αφορά στην επιστήμη των υπολογιστών και, ειδικότερα, τον προγραμματισμό. Ο Horn [43] καταλήγει σε ανάλογα συμπεράσματα, αφού με βάση τα ενθαρρυντικά αποτελέσματα συμμετοχής των κοριτσιών, κατά τη διάρκεια της μελέτης χρήσης της γλώσσας προγραμματισμού Tern, θεωρεί ότι οι tangible γλώσσες προγραμματισμού αποτελούν μια ενδιαφέρουσα προσέγγιση για τη δημιουργία δραστηριοτήτων προγραμματισμού υπολογιστή, ουδέτερων από άποψη φύλου. Η Beisser [10], από την πλευρά της, συμπεραίνει ύστερα από τη διεξαγωγή μιας μελέτης διάρκειας δύο χρονών με μαθητές δημοτικού ότι και τα δύο φύλα νοιώθουν εξίσου ικανοποιημένα από τη χρήση των τεχνολογιών Lego/Logo. Σύμφωνα με τα

αποτελέσματα της έρευνας, τα κορίτσια φαίνεται να εμφανίζουν σημαντικά κέρδη όσον αφορά στην αυτοεκτίμηση τους και την πίστη στις δυνατότητες σε σχέση με την επιστήμη των υπολογιστών και τη χρήση της τεχνολογίας. Παρά το γεγονός ότι τα αγόρια θεωρούν πως υπερέχουν σε σχέση με τα κορίτσια, στην πράξη τα τελευταία θεωρούν ότι δεν υστερούν εκτιμώντας ότι στο προσεχές μέλλον η ενασχόληση τους με την επιστήμη των υπολογιστών θα είναι ισότιμη και θα γίνεται κάτω από την ίδια βάση, με κοινή αφετηρία και ανάλογες προοπτικές.

Καταληκτικά, παρότι η υπεροχή των αγοριών στην επιστήμη των υπολογιστών θεωρείται από πολλούς δεδομένη, η κατάσταση μπορεί να εξισορροπηθεί, χρησιμοποιώντας τα κατάλληλα εργαλεία, με δεδομένο το ότι η θετική εμπειρία ενασχόλησης με τον προγραμματισμό είναι αυτή που, κυρίως, καθορίζει την επιτυχή ενσωμάτωση των κοριτσιών στον τομέα της τεχνολογίας των υπολογιστών.

## Κεφάλαιο 3 Μεθοδολογία

### 3.1 Επιλογή Υλικών και Προγραμματιστικών Εργαλείων

Για τη διεξαγωγή της μελέτης αρχικά έγινε προσεκτική επιλογή υλικού (hardware), δηλαδή της κατάλληλης τεχνολογίας προγραμματισμού, και λογισμικού (software), δηλαδή του κατάλληλου περιβάλλοντος ανάπτυξης εφαρμογών (IDE).

Οι τεχνολογίες προγραμματισμού με τις οποίες ασχολήθηκαν τα υποκείμενα της μελέτης ήταν: α) ο κλασικός προγραμματισμός υπολογιστή, β) ο προγραμματισμός ρομπότ και γ) η τεχνολογία φορετού υπολογισμού, προγραμματισμού φυσικών αντικειμένων e-textiles.

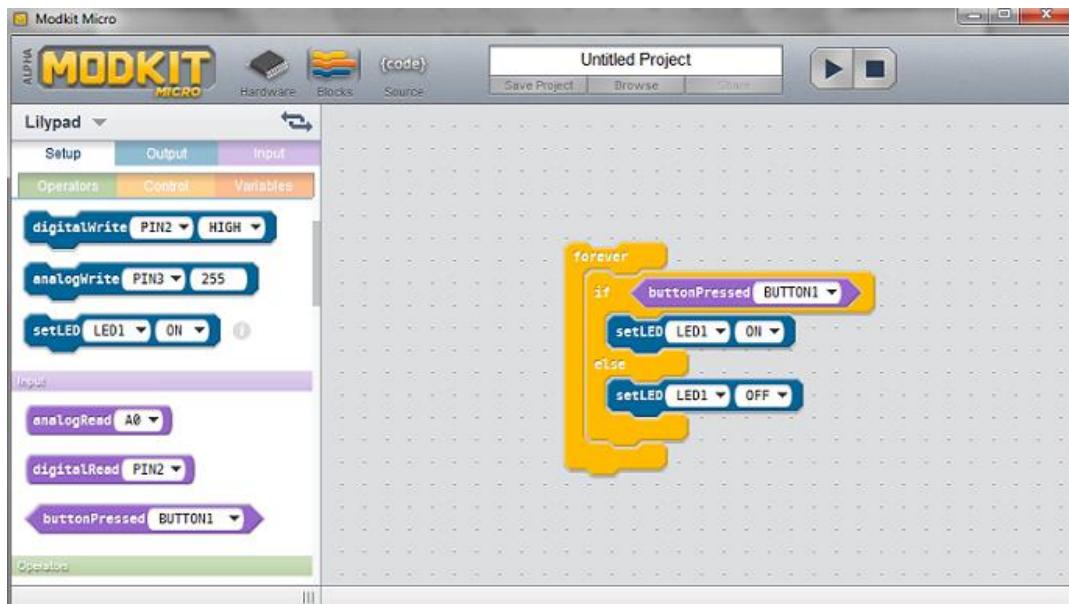
Στην κλασική μέθοδο προγραμματισμού χρησιμοποιήθηκε ένας επιτραπέζιος υπολογιστής. Όσον αφορά στον προγραμματισμό ρομπότ επιλέχθηκε το πακέτο ρομποτικής Lego Mindstorms [39], το οποίο δίνει τη δυνατότητα στους μαθητές εύκολα και γρήγορα να κατασκευάσουν τα δικά τους ρομπότ, ενώνοντας τα δομικά στοιχεία του πακέτου. Για τον προγραμματισμό φυσικών αντικειμένων e-textiles επιλέχθηκε η τεχνολογία φορετού υπολογισμού LilyPad. Σύμφωνα με την δημιουργό της συγκεκριμένης τεχνολογίας, Leah Buechley, στόχος ήταν να αναπτυχθεί μια τεχνολογία ανάλογη με αυτή της τεχνολογίας Lego Mindstorms [38].

Ωστόσο η δημιουργία e-textiles εμφανίζει κάποιες ιδιαίτερες προκλήσεις σε σχέση με την κατασκευή ρομπότ. Τα ρομπότ που κατασκευάζονται από κομμάτια Lego μπορούν εύκολα και γρήγορα να αποσυρμολογηθούν και να συναρμολογηθούν πάλι, δηλαδή υπάρχει ευκολία στην επαναληπτική διαδικασία σχεδίασης (iterative design). Η δημιουργία ενός e-textile, ωστόσο, απαιτεί από τους μαθητές να αφιερώσουν αρκετό χρόνο και προσεκτικό σχεδιασμό πριν να προχωρήσουν στην ενσωμάτωση των κυκλωμάτων στο φυσικά αντικείμενα. Παρόλο που η πλακέτα ανάπτυξης του LilyPad παρείχε έτοιμες συνδέσεις ανάμεσα στον μικροεπεξεργαστή και τα διάφορα εξαρτήματα (π.χ. φωτάκια LEDs, διακόπτες και αισθητήρες) ώστε οι μαθητές να ξεκινήσουν άμεσα τον προγραμματισμό της πλακέτας, οι πιλοτικές έρευνες που προηγήθηκαν της κύριας μελέτης έδειξαν ότι οι μαθητές αδυνατούσαν να καταλάβουν τον τρόπο λειτουργίας και σύνδεσης της πλακέτας. Ακριβώς για αυτό το λόγο, στην κύρια μελέτη η πλακέτα ανάπτυξης αποσυναρμολογήθηκε πλήρως και επιλέχθηκαν, αντί του αγώγιμου νήματος, καλώδια-κροκοδειλάκια (crocodile cables) για την εύκολη και γρήγορη σύνδεση των ηλεκτρικών κυκλωμάτων. Έτσι, αφενός οι μαθητές

αντιλαμβάνονταν καλύτερα τον τρόπο σύνδεσης των διαφόρων εξαρτημάτων με τον μικροεπεξεργαστή και αφετέρου τους δινόταν η δυνατότητα για εύκολο και γρήγορο iterative design, όπως γινόταν και στην περίπτωση των ρομπότ Lego.

Οσον αφορά στην επιλογή του κατάλληλου περιβάλλοντος ανάπτυξης εφαρμογών (IDE) δυο βασικές παράμετροι λήφθηκαν υπόψη. Πρώτον το περιβάλλον ανάπτυξης έπρεπε να είναι γραφικό, ώστε οι μαθητές να μπορούν εύκολα να προγραμματίσουν, απλά με drag-and-drop των εντολών, δίχως να χρειάζεται να αφιερώσουν χρόνο για την εκμάθηση του συντακτικού της γλώσσας. Δεύτερον τα περιβάλλοντα ανάπτυξης για κάθε μια από τις τρεις τεχνολογίες, έπρεπε να έχουν, όσο το δυνατόν, μεγαλύτερη ομοιότητα μεταξύ τους.

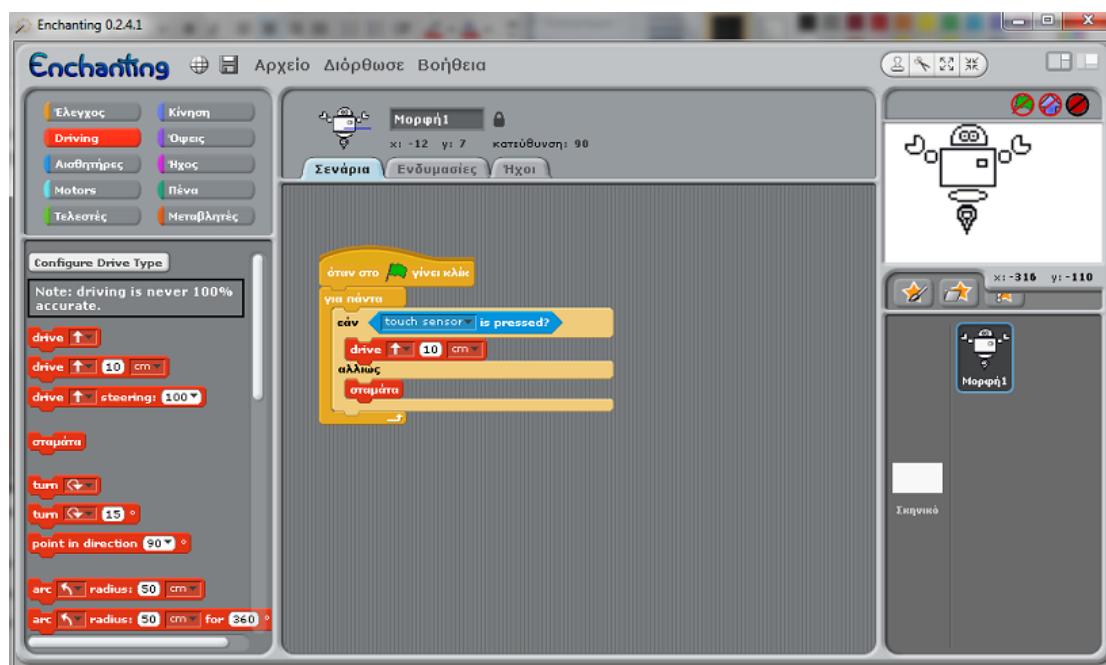
Αν και για τον επιτραπέξιο υπολογιστή υπάρχει μια πληθώρα από γραφικά περιβάλλοντα ανάπτυξης εφαρμογών που είναι διαθέσιμα ειδικά για τους μαθητές, στην περίπτωση του LilyPad βασικό περιβάλλον ανάπτυξης εφαρμογών αποτελεί το Arduino IDE [9]. Ωστόσο το συγκεκριμένο περιβάλλον δεν είναι γραφικό, με αποτέλεσμα να δυσκολεύει αρκετά τους μαθητές στην εκμάθηση του συντακτικού και κατ' επέκταση να είναι αρκετά δύσχρηστο για τη μελέτη. Εναλλακτική λύση για τον προγραμματισμό του LilyPad σε γραφικό περιβάλλον αποτελεί το Ardublock [7] που είναι μια επέκταση του Arduino IDE και το περιβάλλον του μοιάζει αρκετά με εκείνο του App Inventor [6]. Άλλες εναλλακτικές λύσεις για τον προγραμματισμό του Arduino LilyPad είναι το γραφικό περιβάλλον Amici [4] και το γραφικό περιβάλλον Modkit [47].



**Εικόνα 5:** Το περιβάλλον ανάπτυξης Modkit για τον προγραμματισμό του μικροεπεξεργαστή φορετού υπολογισμού LilyPad

Το τελευταίο, αν και βρίσκεται ακόμα σε στάδιο ανάπτυξης, εμφανίζει ένα περιβάλλον που μοιάζει σε μεγάλο βαθμό με το περιβάλλον του Scratch. Οι δημιουργοί του Modkit θέλησαν να φτιάξουν ένα γραφικό περιβάλλον παρόμοιο με το Scratch, μέσω του οποίου θα δίνεται η δυνατότητα σε αρχάριους χρήστες να προγραμματίσουν εύκολα μικροεπεξεργαστές Arduino [5].

Για τον προγραμματισμό του μικροεπεξεργαστή NXT βασικό γραφικό περιβάλλον ανάπτυξης εφαρμογών αποτελεί το NXT-G [51] που στηρίζεται στο περιβάλλον του LabView. Ο χρήστης γράφει το πρόγραμμα απλά σύροντας και αφήνοντας τα εικονίδια εντολών, δημιουργώντας έτσι μια αλυσίδα από εντολές που αποτελούν το πρόγραμμα. Οι χρήστες ωστόσο έχουν την δυνατότητα να προγραμματίσουν το ρομπότ με μια ενδιαφέρουσα παραλλαγή του Scratch που ονομάζεται Enchanting [22].



**Εικόνα 6 . Το περιβάλλον ανάπτυξης Enchanting για τον προγραμματισμό του ρομπότ Lego Mindstorms NXT.**

Το περιβάλλον ανάπτυξης του Enchanting είναι παρόμοιο με το Scratch 1.4 δίνοντας τη δυνατότητα στους μαθητές να ελέγχουν με εύκολο τρόπο τους αισθητήρες και κινητήρες του ρομπότ προγραμματίζοντας τον μικροεπεξεργαστή του NXT.

Με βάση τα παραπάνω φαίνεται ότι το περιβάλλον του Modkit και του Enchanting παρουσιάζουν μεγάλες ομοιότητες με το περιβάλλον του Scratch. Το γεγονός αυτό συνετέλεσε στην επιλογή των συγκεκριμένων γραφικών IDE ως βασικών εργαλείων, από πλευράς λογισμικού, για κάθε μια από τις τεχνολογίες προγραμματισμού που μελετήθηκαν κατά τη διάρκεια της ερευνάς.

Μέθοδος Προγραμματισμού	Υλικό Εκτέλεσης Προγραμμάτων	Λογισμικό Δημιουργίας Προγραμμάτων
Κλασικός	Επιτραπέζιος Υπολογιστής	Scratch 2.0
Ρομποτική	Lego Mindstorms NXT	Enchanting
e-textiles	LilyPad Development Board	Modkit

Πίνακας 1: Επιλογές υλικού – λογισμικού μελέτης

### 3.2 Σχεδιασμός Δραστηριοτήτων

Για κάθε μια μέθοδο προγραμματισμού δημιουργήθηκαν και οι αντίστοιχες δραστηριότητες που διαπραγματεύονταν οι μαθητές και μαθήτριες κατά τη διάρκεια της μελέτης. Βασική επιδίωξη κατά τη δημιουργία των δραστηριοτήτων ήταν να είναι όσο το δυνατόν παρόμοιες και ισοδύναμες μεταξύ τους για να είναι συγκρίσιμες. Επίσης η χρονική διάρκεια κάθε μιας από τις δραστηριότητες ήταν 45 λεπτά, ίση με μια διδακτική ώρα, για λόγους καλύτερης οργάνωσης της μελέτης και διαθεσιμότητας των μαθητών εφόσον η διεξαγωγή της πραγματοποιήθηκε κατά την κανονική διάρκεια λειτουργίας του σχολείου.

Ενώ θα μπορούσε το περιεχόμενο των δραστηριοτήτων να αφορά μόνο τον προγραμματισμό και την εκμάθηση βασικών προγραμματιστικών εννοιών μέσα από τα διαφορετικά περιβάλλοντα προγραμματισμού, αντί αυτού επιλέχθηκε πριν από την ενασχόληση με τον προγραμματισμό οι μαθητές προετοίμασαν κατάλληλα τα αντικείμενα, εικονικά ή πραγματικά, τα οποία στη συνέχεια προγραμμάτισαν. Εξάλλου οι τεχνολογίες προγραμματισμού που επιλέχθηκαν στη μελέτη δεν περιλαμβάνουν μόνο τον προγραμματισμό. Στην περίπτωση του Scratch οι χρήστες συνήθως διαμορφώνουν έναν εικονικό κόσμο από αντικείμενα, πριν προχωρήσουν στον προγραμματισμό. Στην περίπτωση της τεχνολογία Lego οι μαθητές πρέπει αρχικά να κατασκευάσουν ένα λειτουργικό ρομπότ, ενώ με την τεχνολογία LilyPad είναι απαραίτητη η δημιουργία ενός e-textile πριν την ενασχόληση με τον προγραμματισμό. Επιπρόσθετα το πόσο ευχάριστη είναι η κάθε τεχνολογία στους μαθητές επηρεάζεται σε μεγάλο βαθμό από το στάδιο της προετοιμασίας που προηγείται του προγραμματισμού.

Για τους παραπάνω λόγους κάθε δραστηριότητα χωρίστηκε σε δυο βασικά στάδια, της προετοιμασίας και του προγραμματισμού. Στο πρώτο στάδιο οι μαθητές

δημιούργησαν τα αντικείμενα τους είτε ήταν εικονικά είτε ήταν φυσικά, ενώ στο δεύτερο στάδιο ήρθαν σε επαφή με την εκμάθηση των προγραμματιστικών εννοιών.

Συγκεκριμένα, στην προετοιμασία της δραστηριότητας του επιτραπέζιου υπολογιστή, οι μαθητές χρησιμοποιώντας ένα έτοιμο πρόγραμμα στην Scratch, εμπλούτισαν το υπάρχον σκηνικό εισάγοντας και δημιουργώντας νέα εικονικά αντικείμενα (ήχους και εικόνες). Σαν υπόβαθρο του προγράμματος χρησιμοποιήθηκε ένα δωμάτιο σπιτιού, ενώ το μόνο αντικείμενο που ήταν διαθέσιμο ήταν ένα διακόπτης σε κατάσταση OFF. Οι μαθητές εισήγαγαν αρχικά ένα χριστουγεννιάτικο δέντρο και ένα χριστουγεννιάτικο τραγούδι. Στη συνέχεια ζωγράφισαν τα χριστουγεννιάτικα φωτάκια που αποτέλεσαν το εικονικό «αντικείμενο – μέσο», στο οποίο εμφανίζονταν τα αποτελέσματα των προγραμμάτων, που υλοποίησαν. Αφού οι μαθητές ολοκλήρωσαν το στήσιμο του σκηνικού, προχώρησαν στον προγραμματισμό.



Εικόνα 7: Σκηνικό δραστηριότητας στο περιβάλλον της Scratch

Στην προετοιμασία της δραστηριότητας που αφορούσε τη ρομποτική στους μαθητές δόθηκε ένα ρομπότ, το οποίο ωστόσο δεν ήταν ολοκληρωμένο.



**Εικόνα 8: Ρομπότ Mindstorms NXT που δόθηκε στους μαθητές στη δραστηριότητα ρομποτικής**

Το ρομπότ, που έμοιαζε με το κινηματογραφικό ρομπότ Wall-e, αποτελείτο από έναν μικροεπεξεργαστή NXT, δύο κινητήρες με ενσωματωμένες ρόδες για την κίνηση του ρομπότ, έναν αισθητήρα πίεσης και διάφορα δομικά στοιχεία Lego. Οι μαθητές σε πρώτη φάση τοποθέτησαν το κεφάλι του ρομπότ και μετά κάνοντας τις κατάλληλες διασυνδέσεις ανάμεσα στο μικροεπεξεργαστή NXT και τους κινητήρες – αισθητήρες, καθιστώντας το πλήρως λειτουργικό.

Τέλος στην προετοιμασία του φορετού υπολογισμού οι μαθητές χρησιμοποιώντας τα καλώδια-κροκοδειλάκια αρχικά να συνέδεσαν το μικροεπεξεργαστή του LilyPad με τρία σετ από μονόχρωμα LEDs και έναν διακόπτη. Ολοκλήρωσαν δηλαδή με επιτυχία τη δημιουργία ενός απλού ηλεκτρικού κυκλώματος. Στη συνέχεια τους δόθηκε ένα έτοιμο e-textile, πάνω στο οποίο εφαρμόστηκε το κύκλωμα που υλοποίησαν προηγουμένως. Ο βασικός λόγος για τη συγκεκριμένη επιλογή ήταν το μεγάλο χρονικό διάστημα, το οποίο θα απαιτείτο για την επιτυχή ενσωμάτωση των στοιχείων του LilyPad πάνω στο φυσικό αντικείμενο με τη χρήση αγώγιμης κλωστής.

Τα έτοιμα φυσικά αντικείμενα που δόθηκαν στους μαθητές ήταν μια σχολική τσάντα για τα αγόρια και earmuffs (αξεσουάρ που προστατεύουν τα αυτιά από τα κρύο) για τα κορίτσια. Τα συγκεκριμένα αντικείμενα σχεδιάστηκαν και κατασκευάστηκαν από μαθητές και μαθήτριες της Β' Γυμνασίου στην αρχική πιλοτική φάση.



**Εικόνα 9: Έτοιμα αντικείμενα που δόθηκαν στους μαθητές στη δραστηριότητα φορετού υπολογισμού.**

Κατά τη διάρκεια της πρώτης φάσης οι μαθητές μπορούσαν ελεύθερα να ζητούν επεξηγήσεις και τους παρεχόταν επιπλέον βοήθεια από τον συντονιστή της δραστηριότητας.

Το δεύτερο στάδιο της δραστηριότητας περιλάμβανε τον προγραμματισμό των δημιουργιών που διαχειρίστηκαν οι μαθητές στο πρώτο στάδιο. Ο στόχος ήταν η διδασκαλία και εκμάθηση τριών βασικών προγραμματιστικών εννοιών: της δομής ακολουθίας (sequence), της δομής επανάληψης (repeat) και τη δομής επιλογής (if-else). Για κάθε προγραμματιστική έννοια δόθηκαν οδηγίες στους μαθητές, για να δημιουργήσουν αρχικά κάποιο απλό πρόγραμμα, ώστε να κατανοήσουν την συγκεκριμένη έννοια, και στη συνέχεια να προχωρήσουν παραπέρα, δημιουργώντας ένα πιο σύνθετο πρόγραμμα για κάθε έννοια αξιοποιώντας αυτά που έμαθαν προηγουμένως. Οι αλγόριθμοι τους οποίους καλούνταν να υλοποιήσουν οι μαθητές ήταν οι ίδιοι, για κάθε τεχνολογία προγραμματισμού, ώστε να διατηρηθεί η ομοιογένεια και να μην επηρεαστούν τα υποκείμενα της μελέτης υπέρ της μίας ή της άλλης τεχνολογίας προγραμματισμού.

Στην κλασική μέθοδο προγραμματισμού υπολογιστή, οι μαθητές μέσω των προγραμμάτων τους προσπάθησαν να αναβοσβήσουν τα φωτάκια του χριστουγεννιάτικου δέντρου με συγκεκριμένο μοτίβο. Ο εικονικός διακόπτης, που προϋπήρχε στο σκηνικό, χρησιμοποιείτο στο σενάριο της δομής επιλογής. Ανάλογα με την κατάσταση στην οποία βρισκόταν (On ή Off) τα φωτάκια αναβόσβηναν ή έμεναν σβηστά. Στην περίπτωση του ρομπότ οι μαθητές προσπάθησαν μέσω των

προγραμμάτων να κινήσουν το ρομπότ με συγκεκριμένο τρόπο ενεργοποιώντας κατάλληλα τους κινητήρες οι οποίοι ήταν συνδεδεμένοι στον μικροεπεξεργαστή. Στην προκειμένη περίπτωση ο αισθητήρας πίεσης που ήταν συνδεδεμένος στο μικροεπεξεργαστή χρησιμοποιείτο στο σενάριο της δομής επιλογής, ανάλογα με το αν ήταν πιεσμένος ή όχι. Τέλος στην περίπτωση του φορετού υπολογισμού τα προγράμματα που έφτιαξαν οι μαθητές αλληλεπιδρούσαν με τα φωτάκια LEDs. Και στην περίπτωση αυτή ο διακόπτης που ήταν συνδεδεμένος με το μικροεπεξεργαστή αφορούσε το σενάριο το οποίο σχετιζόταν με τη δομή επιλογής. Αξίζει να σημειωθεί ότι στην περίπτωση του ρομπότ τα αποτελέσματα της εκτέλεσης των προγραμμάτων αφορούσαν κίνηση ενός φυσικού αντικειμένου (ρομπότ) σε αντίθεση με το Scratch που αφορούσε την αλλαγή του χρώματος ενός εικονικού αντικειμένου (φωτάκια χριστουγεννιάτικου δέντρου) και του LilyPad που αφορούσε την αλλαγή του χρώματος φυσικών αντικειμένων (φωτάκια LEDs). Αναλυτικά τα σενάρια που διαπραγματεύτηκαν οι μαθητές βρίσκονται στο Παράρτημα I.

### **3.3 Υποκείμενα και Σχεδιασμός Πειράματος**

Τα υποκείμενα της μελέτης ήταν μαθητές και μαθήτριες που φοιτούσαν στην Α' τάξη του 1<sup>ου</sup> Γυμνασίου Άνω Λιοσίων και συγκεκριμένα 36 μαθητές, 18 αγόρια και 18 κορίτσια. Η επιλογή τους έγινε τυχαία από το σύνολο των μαθητών της Α' τάξης του Γυμνασίου. Επειδή οι μαθητές, τόσο της Β' όσο και της Γ' τάξης του συγκεκριμένου δημόσιου σχολείου, είχαν ήδη διδαχθεί προγραμματισμό, μέσω του περιβάλλοντος της Scratch, σε προηγούμενες τάξεις του Γυμνασίου, για να υπάρχει ομοιογένεια στα υποκείμενα, ως συμμετέχοντες της κύριας μελέτης επιλέχθηκαν οι μαθητές και μαθήτριες της Α' τάξης με ηλικίες από 12 έως 13. Ωστόσο θα πρέπει να αναφερθεί ότι κατά τη διάρκεια των αρχικών πιλοτικών φάσεων της μελέτης κάποιοι από τους μαθητές και μαθήτριες της Β' τάξης χρησιμοποιήθηκαν με στόχο την αρχική αξιολόγηση και βελτίωση των δραστηριοτήτων. Επιπλέον πρέπει να σημειωθεί ότι οι μαθητές της αρχικής πιλοτικής φάσης της έρευνας συνέβαλλαν με τα σχόλια τους στην αξιολόγηση και τη βελτίωση των εργαλείων μέτρησης (ερωτηματολόγια, κριτήρια αξιολόγησης και συνεντεύξεις) που χρησιμοποιήθηκαν στην κύρια μελέτη.

Η πιλοτική φάση της έρευνας διήρκεσε σχεδόν δυο μήνες (Δεκέμβριο 2013 έως Ιανουάριο 2014), ενώ η κύρια μελέτη πραγματοποιήθηκε κατά τη διάρκεια του κανονικού ωρολογίου προγράμματος σπουδών του σχολείου και η συνολική διάρκεια της ήταν τρεις εβδομάδες κατά τον μήνα Φεβρουάριο του 2014.

Το πείραμα ήταν μια randomized within groups μελέτη με τη δημιουργία τριών ισάριθμων και ξεχωριστών ομάδων μαθητών και μαθητριών. Η πρώτη ομάδα, Scratch – First, αρχικά ήρθε σε επαφή με το περιβάλλον του Scratch ολοκληρώνοντας την αντίστοιχη δραστηριότητα. Η δεύτερη ομάδα, Lego – First, αρχικά ξεκίνησε με την τεχνολογία ρομποτικής Lego Mindstorms και τον προγραμματισμό ρομπότ μέσω του περιβάλλοντος ανάπτυξης εφαρμογών του Enchanting. Η τρίτη ομάδα, LilyPad – First, ξεκίνησε με την τεχνολογία φορετού υπολογισμού LilyPad και τον προγραμματισμό e-textiles μέσω του περιβάλλοντος του Modkit. Κάθε ομάδα αφού ολοκλήρωνε την αρχική της δραστηριότητα συνέχιζε με τις υπόλοιπες δυο τεχνολογίες προγραμματισμού. Με τον τρόπο αυτό όλοι οι μαθητές και μαθήτριες ήρθαν σε επαφή με διαφορετική σειρά και με τις τρεις τεχνολογίες προγραμματισμού της μελέτης. Οι μαθητές εργάζονταν συνεργατικά σε ομάδες των δυο ατόμων του ίδιου φύλου με στόχο την ολοκλήρωση των δραστηριοτήτων που σχεδιάστηκαν από τον ερευνητή.

Η ενασχόληση κάθε ομάδας μαθητών και μαθητριών με κάθε μια από τις τρεις διαφορετικές δραστηριότητες προγραμματισμού γινόταν σε τρεις διαφορετικές ημέρες στο χρονικό διάστημα μιας εβδομάδας.

## 3.4 Εργαλεία Μέτρησης

### 3.4.1 Pre-test

Πριν την ενασχόληση με κάθε μια από τις τρεις δραστηριότητες προγραμματισμού οι μαθητές κλήθηκαν να συμπληρώσουν ένα ερωτηματολόγιο σχετικά με την εμπειρία τους στην χρήση υπολογιστών, στον προγραμματισμό, στη ρομποτική και στην ηλεκτρονική. Μέσω του συγκεκριμένου ερωτηματολογίου αφενός αξιολογήθηκε η ομοιομορφία των τριών διαφορετικών ομάδων των μαθητών στη μελέτη και αφετέρου δημιουργήθηκε ένα σημείο αναφοράς μέσω του οποίου θα γίνονταν οι συγκρίσεις στη συμπεριφορά και τις προτιμήσεις των μαθητών ύστερα από την ενασχόληση τους με τις υπό μελέτη τεχνολογίες προγραμματισμού.

Το ερωτηματολόγιο ακολουθούσε την τεταρτοβάθμια κλίμακα (4 scale) Likert (καθόλου – λίγο – αρκετά – πολύ) [15], ενώ τα ερωτήματα που κλήθηκαν να απαντήσουν οι μαθητές αφορούσαν την εμπειρία χρήσης υπολογιστή, την εμπειρία τους στον προγραμματισμό, την εμπειρία τους στη ρομποτική και την εμπειρία τους στην ηλεκτρονική.

Αναλυτικά τα ερωτήματα που κλήθηκαν να απαντήσουν αρχικά οι μαθητές:

- Ως προς την εμπειρία χρήσης υπολογιστή [15]:
  - Είσαι καλός στο να χρησιμοποιείς τον υπολογιστή;
  - Βρίσκεις τους υπολογιστές δύσκολους στη χρήση;
  - Νοιώθεις άνετα όταν χρησιμοποιείς το ποντίκι;
  - Νοιώθεις άνετα όταν πληκτρολογείς σε έναν υπολογιστή;
- Ως προς την εμπειρία τους στον προγραμματισμό [68]:
  - Σε ενδιαφέρει η πληροφορική-επιστήμη των υπολογιστών;
  - Έχεις κάποια προηγούμενη εμπειρία προγραμματισμού;
  - Αν απάντησες θετικά στην προηγούμενη ερώτηση, τι είδους προγραμματισμού έχεις κάνει στο παρελθόν;
  - Πόσο δύσκολη θεωρείς ότι είναι η δημιουργία προγραμμάτων υπολογιστή;
  - Πόσες γνώσεις προγραμματισμού θεωρείς ότι έχεις;
- Ως προς την εμπειρία στα ηλεκτρικά κυκλώματα – ηλεκτρονική:
  - Σε ενδιαφέρει η ηλεκτρονική-δημιουργία ηλεκτρικών κυκλωμάτων;
  - Έχεις κάποια προηγούμενη εμπειρία στην δημιουργία ηλεκτρικών κυκλωμάτων ;
  - Αν απάντησες θετικά στην προηγούμενη ερώτηση, με τι είδους ηλεκτρικά κυκλώματα έχεις ασχοληθεί στο παρελθόν;
  - Πόσο δύσκολη θεωρείς την ενασχόληση με την ηλεκτρονική;
  - Πόσες γνώσεις ηλεκτρονικής θεωρείς ότι έχεις;
- Ως προς την εμπειρία στη ρομποτική:
  - Σε ενδιαφέρει η ρομποτική (κατασκευή-προγραμματισμός ρομπότ);
  - Έχεις κάποια προηγούμενη εμπειρία στη ρομποτική;
  - Αν απάντησες θετικά στην προηγούμενη ερώτηση, με τι είδους ρομπότ έχεις ασχοληθεί στο παρελθόν;
  - Πόσο δύσκολη θεωρείς την ενασχόληση με τη ρομποτική;
  - Πόσες γνώσεις ρομποτικής θεωρείς ότι έχεις;

Για τις ποσοτικές μετρήσεις μετά την ολοκλήρωση των τριών δραστηριοτήτων χρησιμοποιήθηκαν τρία εργαλεία.

### **3.4.2 Emotions – Test**

Για την αξιολόγηση των συναισθημάτων των μαθητών χρησιμοποιήθηκε ένα online εργαλείο επιλογής συναισθημάτων που κατασκευάστηκε για την παρούσα μελέτη και ακολουθούσε την πενταβάθμια κλίμακα (5 scale) Likert. Οι μαθητές είχαν την δυνατότητα να καταγράψουν με εύκολο και γρήγορο τρόπο τα συναισθήματα, τα οποία βίωσαν κατά την ενασχόληση τους με τις υπό εξέταση τεχνολογίες προγραμματισμού.

**Πως ένιωσες κατά τη διάρκεια της δραστηριότητας;**



**Εικόνα 10: Εργαλείο Επιλογής Συναισθημάτων για την αξιολόγηση των συναισθημάτων των παθητών στην χούση των τεχνολογιών προγραμματισμού.**

Στην πιλοτική φάση της μελέτης υπήρχε μεγαλύτερη ποικιλία συναισθημάτων, που μπορούσαν να επιλέξουν οι μαθητές, ωστόσο ύστερα από την αξιολόγηση των απαντήσεων της πιλοτικής φάσης προτιμήθηκαν για την κύρια μελέτη τα δημοφιλέστερα επιλεγμένα συναισθήματα καθώς και εκείνα που παρουσίαζαν μια τάση διαφοροποίησης σε σχέση με την ανεξάρτητη μεταβλητή της μελέτης (μέθοδος προγραμματισμού). Οι πέντε κατηγορίες συναισθημάτων που επιλέχθηκαν τελικά φαίνονται στην Εικόνα 10.

### 3.4.3 Post – Test

Επιπλέον με την ολοκλήρωση κάθε μιας από τις τρεις δραστηριότητες οι μαθητές καλούνταν να καταγράψουν την γνώμη τους συμπληρώνοντας ένα ερωτηματολόγιο Post – test.

Μέσω του συγκεκριμένου ερωτηματολογίου, που ακολουθούσε την τεταρτοβάθμια κλίμακα (4 scale) Likert (καθόλου – λίγο – αρκετά – πολύ), μελετήθηκε ο βαθμός ενδιαφέροντος των μαθητών για προγραμματισμό, και κατά πόσο θεωρούν οι ίδιοι ότι βελτιώθηκαν οι ικανότητες τους στον προγραμματισμό. Αναλυτικά τα ερωτήματα που κλήθηκαν να απαντήσουν οι μαθητές στο post – test παρουσιάζονται στον Πίνακα 2.

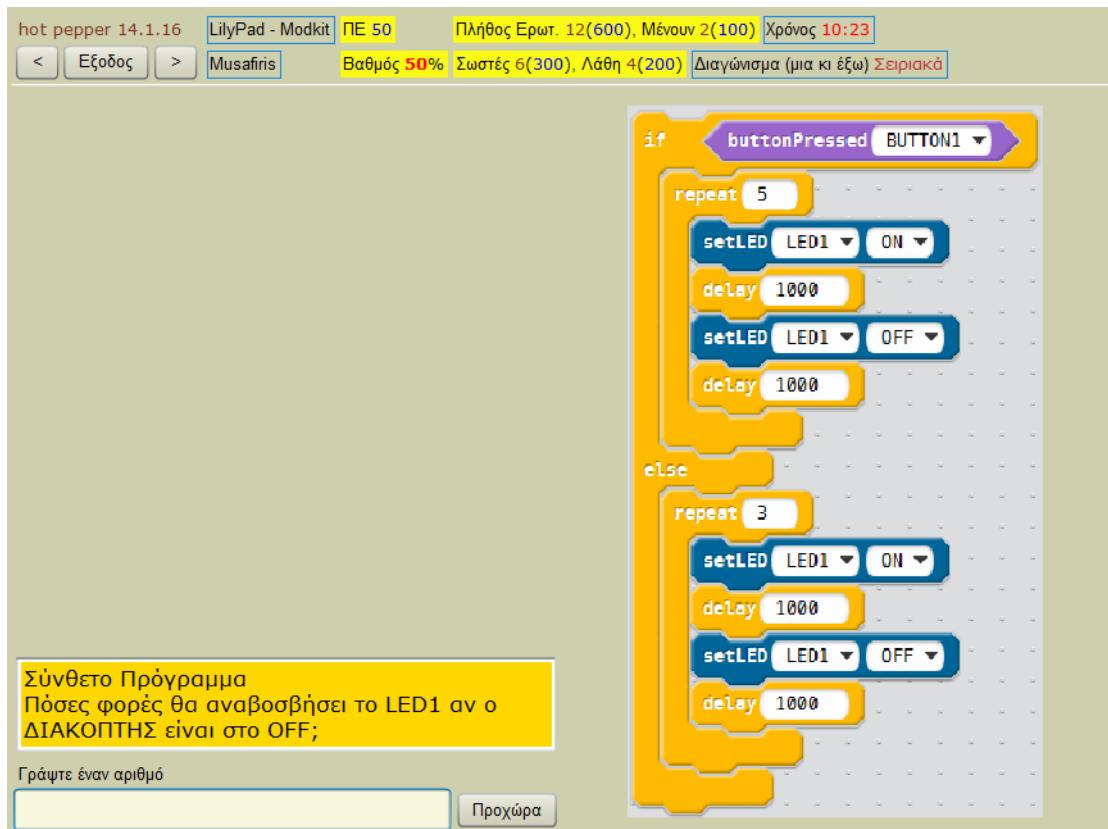
Εξαρτημένες Μεταβλητές	Ερωτήσεις
Ενδιαφέρον για Προγραμματισμό [68]	Θα σε ενδιέφερε να μάθεις προγραμματισμό στο μέλλον; Θα σε ενδιέφερε να παρακολουθήσεις μαθήματα προγραμματισμού στο μέλλον εξαιτίας της εμπειρίας σου στην συγκεκριμένη δραστηριότητα; Θα σε ενδιέφερε να δημιουργήσεις προγράμματα στο σπίτι στον ελεύθερο σου χρόνο;
Ικανότητα στον Προγραμματισμό [15]	Πόσο βελτιώθηκαν οι ικανότητες σου στον προγραμματισμό με τη συγκεκριμένη δραστηριότητα; Πόσο δύσκολη θεωρείς τώρα ότι είναι η δημιουργία προγραμμάτων υπολογιστή; Πόσες γνώσεις προγραμματισμού θεωρείς ότι έχεις μετά τη συγκεκριμένη δραστηριότητα; Είσαι καλός στη δημιουργία προγραμμάτων υπολογιστή;
Αξιολόγηση Δραστηριότητας	Συνολικά πόσο ενδιαφέρουσα θεωρείς ότι ήταν η συγκεκριμένη δραστηριότητα; Συνολικά πόσο δύσκολη θεωρείς ότι ήταν συγκεκριμένη δραστηριότητα; Πιστεύεις ότι η συγκεκριμένη δραστηριότητα αύξησε το ενδιαφέρον σου για προγραμματισμό; Συνολικά πόσο θα βαθμολογούσες την συγκεκριμένη δραστηριότητα;

Πίνακας 2: Ερωτήματα ερωτηματολογίου post – test

### 3.4.4 Κριτήρια Αξιολόγησης

Τρία ισοδύναμα online κριτήρια αξιολόγησης, ένα για κάθε τεχνολογία προγραμματισμού δημιουργήθηκαν με στόχο την αντικειμενικότερη αξιολόγηση των

μαθητών στην προγραμματιστική σκέψη και συγκεκριμένα στην εκμάθηση των τριών βασικών αλγορίθμικών συνιστωσών. Τα online κριτήρια δημιουργήθηκαν μέσω του εργαλείου hot pepper quiz creator [31]. Κάθε κριτήριο αποτελείτο από τέσσερις ομάδες ερωτήσεων. Μια ομάδα για τη δομή ακολουθίας, μια για τη δομή επανάληψης, μια για τη δομή επιλογής και τέλος μια ομάδα που αφορούσε σύνθετα προγράμματα με συνδυασμό των τριών υπό εξέταση αλγορίθμικών συνιστωσών. Κάθε ομάδα ερωτήσεων περιλάμβανε τρεις ερωτήσεις αξιολόγησης, επομένως οι συνολικές ερωτήσεις που καλούνταν οι μαθητές να απαντήσουν ήταν δώδεκα [69].



**Εικόνα 11: Ερώτηση στο σύνθετο πρόγραμμα σε Κριτήριο Αξιολόγησης των φορετού υπολογισμού LilyPad**

Τέλος με την ολοκλήρωση όλων των δραστηριοτήτων επιλέχθηκε δείγμα μαθητών να συμμετάσχει σε μια ημι-δομημένη συνέντευξη με τον ερευνητή. Τα ερωτήματα που κλήθηκαν να απαντήσουν οι μαθητές αυτοί αφορούσαν τα συναισθήματα τα οποία ένοιωσαν, τυχόν δυσκολίες που αντιμετώπισαν, τις εντυπώσεις και τις προτιμήσεις τους κατά την ενασχόληση τους με τις τρεις διαφορετικές τεχνολογίες προγραμματισμού. Οι συγκεκριμένες συνεντεύξεις βοήθησαν να εμπλουτιστούν ποιοτικά τα δεδομένα της μελέτης.

### 3.5 Διαδικασία

Το κυρίως μέρος της έρευνας ήταν αυστηρά προκαθορισμένο και αποτελούνταν από τα εξής στάδια:

- Όλες οι ομάδες ενημερώνονταν ότι θα συμμετέχουν σε μια ερεύνα της επιστήμης των υπολογιστών που σχετίζεται με τον προγραμματισμό.
- Όλοι οι μαθητές συμπλήρωναν ξεχωριστά το post test.
- Οι μαθητές ασχολούνταν με την πρώτη δραστηριότητα που αντιστοιχούσε στην ομάδα στην οποία ανήκαν. Δούλευναν συνεργατικά σε ομάδες των δυο ατόμων του ίδιου φύλου.
- Με την επιτυχή ολοκλήρωση της πρώτης δραστηριότητας συμπλήρωναν ξεχωριστά το Emotions – Test.
- Στην συνέχεια συμπλήρωναν ξεχωριστά το Post – Test.
- Τελευταίο ακολουθούσε το Κριτήριο Αξιολόγησης, ώστε να μην επηρεαστούν συναισθηματικά τα υποκείμενα της μελέτης.
- Σε επόμενη φάση τα υποκείμενα ασχολούνταν με τις υπόλοιπες δυο τεχνολογίες προγραμματισμού που απέμεναν και συμπλήρωναν ξεχωριστά κάθε φορά τα αντίστοιχα Emotions – Tests, Post – Tests και Κριτήρια Αξιολόγησης.
- Τέλος κάποιοι από τους μαθητές καλούνταν να απαντήσουν σε κάποιες ερωτήσεις στα πλαίσια μιας ημι-δομημένης συνέντευξης.

Με την επιτυχή ολοκλήρωση της κυρίας μελέτης τα ποσοτικά δεδομένα που συλλέχθηκαν κατά τη διάρκεια αυτής εισήχθησαν στο πρόγραμμα IBM SPSS Statistics 21.0 [35] για την περαιτέρω στατιστική επεξεργασία τους.

## Κεφάλαιο 4 Παρουσίαση και Ανάλυση Δεδομένων Έρευνας

### 4.1 Έλεγχος Ισοδυναμίας Ομάδων

#### 4.1.1 Έλεγχος Ισοδυναμίας με βάση την εισαγωγική μέθοδο προγραμματισμού

Δεδομένου ότι η επιλογή των μαθητών της μελέτης έγινε με τυχαίο τρόπο από το σύνολο των μαθητών της Α' τάξης αναμένεται ότι και οι τρεις ομάδες, Scratch – First, LilyPad – First και NXT – First, είναι μεταξύ τους ισοδύναμες. Για να επαληθευτεί αν αυτό όντως ισχύει, πριν από τη μελέτη οι μαθητές συμπλήρωσαν το ερωτηματολόγιο pre – test.

Αρχικά, λοιπόν, έγινε έλεγχος της κανονικότητας των απαντήσεων των τριών ομάδων της μελέτης και με βάση τα αποτελέσματα όλοι οι πληθυσμοί φάνηκε να ακολουθούν την κανονική κατανομή. Στη συνέχεια, για να ελεγχθούν οι μέσοι όροι των απαντήσεων και για τις τρεις ομάδες της έρευνας, έγινε εκτέλεση της μεθόδου ανάλυσης διακύμανσης ANOVA (ANalysis Of VAriance). Πρόκειται για μια διαδικασία ελέγχου του μέσου όρου δύο ή περισσότερων πληθυσμών. Όταν υπάρχει μια ανεξάρτητη μεταβλητή, χρησιμοποιείται το One Way ANOVA. Αν τα υπό εξέταση δείγματα είναι μεταξύ τους ανεξάρτητα, τότε εκτελείται το 1-Way Between Subjects ANOVA, ενώ, αν ανάμεσα στα υπό εξέταση δείγματα υπάρχει σχέση, εκτελείται το 1-Way Within Subjects ANOVA. Στην προκειμένη περίπτωση οι τρεις ομάδες θεωρήθηκαν ότι είναι ανεξάρτητες μεταξύ τους όποτε, για να ελεγχθεί η ισοδυναμία, εκτελέστηκε το 1-Way Between Subjects ANOVA.

Με βάση τα αποτελέσματα από την εκτέλεση της μεθόδου ANOVA υπήρξε σημαντική διαφορά μόνο **στο πόσο άνετα νοιώθουν όταν πληκτρολογούν στον υπολογιστή** οι μαθητές στα επίπεδα του  $p < .05$  για τις τρεις ομάδες [ $F(2, 33) = 5.112$ ,  $p = 0.012$ ]. Οι post hoc συγκρίσεις, που έγιναν για να εξακριβωθεί ανάμεσα σε ποιες ομάδες ακριβώς υπάρχει διαφορά, έδειξαν ότι η σημαντική στατιστική υφίστανται ανάμεσα στην ομάδα του Scratch – First ( $M = 2.58$ ,  $SD = 0.793$ ) και του NXT – First ( $M = 3.58$ ,  $SD = 0.515$ ). Με βάση τα παραπάνω θεωρήθηκε ότι γενικά οι τρεις ομάδες εμφανίζονταν να είναι ισοδύναμες μεταξύ τους, εκτός από την ευκολία χρήσης του πληκτρολογίου.

#### 4.1.2 Έλεγχος Ισοδυναμίας με βάση το φύλο

Στη συνέχεια συγκρίθηκαν οι μέσοι όροι των απαντήσεων στο pre – test των αγοριών με εκείνες των κοριτσιών.

	ΦΥΛΟ	N	Mean	Std. Deviation	Std. Error Mean	t	Sig. (2 - tailed)
COMPUTER USE	Αγόρια	18	2,89	,471	,111	,000	1,000
	Κορίτσια	18	2,89	,471	,111		
COMPUTER DIFFICULT	Αγόρια	18	2,17	,514	,121	3,280	,002
	Κορίτσια	18	1,61	,502	,118		
MOUSE USE	Αγόρια	18	3,56	,616	,145	,000	1,000
	Κορίτσια	18	3,56	,511	,121		
KEYBOARD USE	Αγόρια	18	2,72	,895	,211	-3,255	,003
	Κορίτσια	18	3,56	,616	,145		
CS INTEREST	Αγόρια	18	2,78	,943	,222	1,965	,058
	Κορίτσια	18	2,17	,924	,218		
PROGRAMMING EXPERIENCE	Αγόρια	18	1,22	,428	,101	,352	,0727
	Κορίτσια	18	1,17	,514	,121		
PROGRAMMING DIFFICULTY	Αγόρια	18	2,89	,676	,159	,227	,822
	Κορίτσια	18	2,83	,786	,185		
PROGRAMMING KNOW HOW	Αγόρια	18	1,89	,676	,159	2,832	,008
	Κορίτσια	18	1,33	,485	,114		
ELECTRONICS INTEREST	Αγόρια	18	2,61	1,145	,270	,661	,513
	Κορίτσια	18	2,39	,850	,200		
ELECTRONICS EXPERIENCE	Αγόρια	18	2,00	1,029	,243	1,279	,209
	Κορίτσια	18	1,61	,778	,183		
ELECTRONICS DIFFICULTY	Αγόρια	18	2,61	,608	,143	-1,604	,118
	Κορίτσια	18	2,94	,639	,151		
ELECTRONICS KNOW HOW	Αγόρια	18	2,06	,725	,171	1,653	,108
	Κορίτσια	18	1,67	,686	,162		
ROBOTICS INTEREST	Αγόρια	18	2,78	1,166	,275	,000	1,000
	Κορίτσια	18	2,78	1,166	,275		
ROBOTICS EXPERIENCE	Αγόρια	18	1,39	,698	,164	2,364	,030
	Κορίτσια	18	1,00	,000	,000		
ROBOTICS DIFFICULTY	Αγόρια	18	3,17	,707	,167	,216	,830
	Κορίτσια	18	3,11	,832	,196		
ROBOTICS KNOW HOW	Αγόρια	18	1,89	,900	,212	3,450	,002
	Κορίτσια	18	1,11	,323	,076		

Πίνακας 3: Μέσοι όροι απαντήσεων και αποτελέσματα independence T- Test στο pre –test.

Για να εξακριβωθεί αν υπήρχε σημαντική στατιστική διαφορά ανάμεσα στους μέσους όρους δύο πληθυσμών (αγόρια – κορίτσια) έγινε χρήση των T-Tests. Στην προκειμένη περίπτωση, όπου οι απαντήσεις των αγοριών ήταν ανεξάρτητες από εκείνες των κοριτσιών, χρησιμοποιήθηκαν τα independent T-Tests. Δεκαέξι independent T-Tests διεξήχθησαν για κάθε μια από τις υπό εξέταση κατηγορίες.

Τα independent T-Tests έδειξαν σημαντική στατιστική διαφορά **στο πόσο δύσκολουν θεωρούν τους υπολογιστές στη χρήση** τα αγόρια ( $M = 2.17$ ,  $SD = .514$ ) και τα κορίτσια ( $M = 1.61$ ,  $SD = .502$ ).  $t(34) = 3.280$ ,  $p = .002$ , **στο πόσο άνετα νοιώθουν όταν πληκτρολογούν στον υπολογιστή** τα αγόρια ( $M = 2.72$ ,  $SD = .895$ ) και τα κορίτσια ( $M = 3.56$ ,  $SD = .616$ ).  $t(34) = -3.255$ ,  $p = .003$ , **στις γνώσεις προγραμματισμού που θεωρούν ότι έχουν τα αγόρια** ( $M = 1.89$ ,  $SD = .676$ ) και τα κορίτσια ( $M = 1.33$ ,  $SD = .485$ ).  $t(34) = 2.832$ ,  $p = .008$ , **στην εμπειρία ρομποτικής που θεωρούν ότι έχουν τα αγόρια** ( $M = 1.39$ ,  $SD = .698$ ) και τα κορίτσια ( $M = 1.00$ ,  $SD = .000$ ).  $t(34) = 2.364$ ,  $p = .030$  και **στις γνώσεις ρομποτικής που θεωρούν ότι έχουν τα αγόρια** ( $M = 1.89$ ,  $SD = .900$ ) και τα κορίτσια ( $M = 1.11$ ,  $SD = .323$ ).  $t(34) = 3.450$ ,  $p = .002$ .

Τα αγόρια της μελέτης, λοιπόν, θεώρησαν ότι οι υπολογιστές είναι δυσκολότεροι στη χρήση από ότι τα κορίτσια, ενώ τα κορίτσια ένοιωθαν πολύ πιο άνετα, όταν πληκτρολογούσαν στον υπολογιστή. Επίσης τα αγόρια ένοιωθαν μεγαλύτερη αυτοπεποίθηση στον προγραμματισμό, αφού πίστευαν ότι κατείχαν περισσότερες γνώσεις από ότι τα κορίτσια. Το ίδιο ίσχυε τόσο για τις γνώσεις όσο και την εμπειρία ρομποτικής όπου και εδώ τα αγόρια θεωρούσαν ότι έχουν περισσότερες. Το γεγονός ότι τα αγόρια έχουν την πεποίθηση ότι υπερέχουν τεχνολογικά, σε θέματα προγραμματισμού και ρομποτικής, έναντι των κοριτσιών συμπίπτει και με τις απόψεις της Beisser [10].

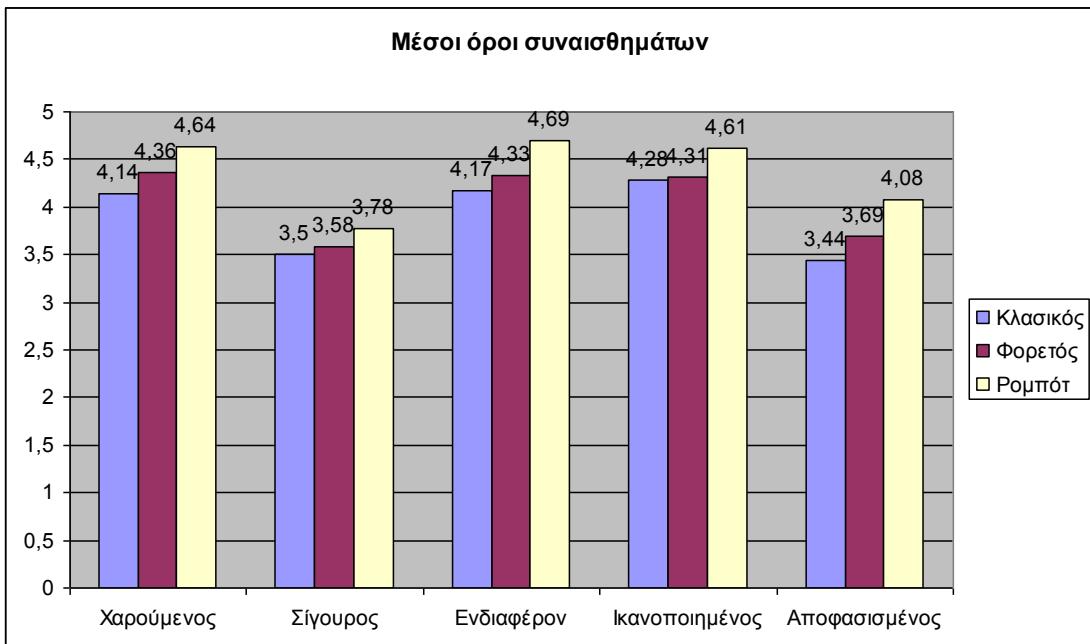
## 4.2 Ανάλυση Αποτελεσμάτων σε σχέση με τη Μέθοδο Προγραμματισμού

### 4.2.1 Συναισθήματα Μαθητών

Για να διερευνηθεί η υπόθεση ότι ο προγραμματισμός φυσικών αντικειμένων προκαλεί θετικότερα συναισθήματα από την κλασική μέθοδο προγραμματισμού, είναι απαραίτητο να αναλυθούν οι απαντήσεις των μαθητών, που συγκεντρώθηκαν μέσω του

emotions – test, σχετικά με τα συναισθήματα που θεώρησαν οι ίδιοι ότι ένοιωσαν για κάθε μια από τις τρεις μεθόδους προγραμματισμού (κλασικός, φορετός και ρομπότ).

Ο έλεγχος της κανονικότητας των απαντήσεων των μαθητών αναφορικά με τα συναισθήματα έδειξε ότι όλοι πληθυσμοί ακολουθούσαν την κανονική κατανομή. Οι μέσοι όροι για τις πέντε κατηγορίες συναισθημάτων που εξετάστηκαν στη μελέτη φαίνονται στο Γράφημα 1.



**Γράφημα 1: Μέσοι όροι συναισθημάτων ανά τεχνολογία προγραμματισμού**

Με βάση την αρχική εικόνα που δίνουν τα περιγραφικά στατιστικά για τα συναισθήματα των μαθητών ο προγραμματισμός ρομπότ φαίνεται να υπερτερεί των άλλων δύο τεχνολογιών προγραμματισμού και στις πέντε κατηγορίες συναισθημάτων, ενώ και ο προγραμματισμός αντικειμένων, μέσω της τεχνολογίας LilyPad φορετού υπολογισμού, υπερτερεί του κλασικού τρόπου προγραμματισμού μέσω του Scratch. Γενικά ο προγραμματισμός φυσικών αντικειμένων φαίνεται να προκαλεί θετικότερα συναισθήματα σε σχέση με την κλασική μέθοδο προγραμματισμού.

Για να εξακριβωθεί, ωστόσο, αν όντως υπάρχει σημαντική στατιστική διαφορά στα συναισθήματα των μαθητών κατά την ενασχόληση τους με τις τρεις τεχνολογίες προγραμματισμού, χρησιμοποιήθηκε και εδώ η μέθοδος ανάλυσης διακύμανσης ANOVA. Στην προκειμένη περίπτωση, οι πληθυσμοί της μελέτης ήταν τρεις όσες και οι τεχνολογίες προγραμματισμού και επειδή η ανεξάρτητη μεταβλητή ήταν μια, μέθοδος προγραμματισμού, χρησιμοποιήθηκε το 1-Way ANOVA. Επιπλέον επειδή οι υπό εξέταση πληθυσμοί δεν ήταν ανεξάρτητοι μεταξύ τους, μιας και η μελέτη ήταν μια

within – subjects μελέτη, και άρα συσχετίζονταν κατά κάποιο τρόπο, χρησιμοποιήθηκε η μέθοδος 1-Way Within Subjects ANOVA ή 1-Way Repeated Measures ANOVA.

Το 1-Way Within Subjects ANOVA το οποίο διεξήχθη για να μελετηθεί η επίδραση που είχε η μέθοδος προγραμματισμού στο συναίσθημα **Χαρούμενος** έδειξε ότι υπήρξε σημαντική στατιστική διαφορά, Wilks' Lambda = 0.745,  $F(2, 34)=5.81$ ,  $p=.007 < .05$ . Για να εξακριβωθεί ανάμεσα σε ποιες τεχνολογίες προγραμματισμού υπήρχαν σημαντικές διαφορές, διεξήχθησαν τρία ξεχωριστά paired sampled t-tests για τις post hoc συγκρίσεις. Το μόνο paired sampled t-tests που έδειξε ότι υπήρχε σημαντική στατιστική διαφορά στο συναίσθημα **Χαρούμενος** ήταν αυτό ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=4.14$ ,  $SD=0.867$ ) και στον προγραμματισμό ρομπότ ( $M=4.64$ ,  $SD=0.639$ ).  $t(35)=-3.416$ ,  $p=.002 < .017$ .

Το 1-Way Within Subjects ANOVA το οποίο διεξήχθη για το συναίσθημα **Σίγουρος** έδειξε ότι δεν υπήρχε σημαντική στατιστική διαφορά ανάμεσα στις τρεις τεχνολογίες προγραμματισμού, Wilks' Lambda = 0.929,  $F(2, 34)=1.309$ ,  $p=.283 > .05$ .

Το 1-Way Within Subjects ANOVA το οποίο διεξήχθη για το συναίσθημα **Ενδιαφέρον** έδειξε ότι υπήρχε σημαντική διαφορά, Wilks' Lambda = 0.766,  $F(2, 34)=5.183$ ,  $p=.011 < .05$ . Τρία paired sampled t-tests χρησιμοποιήθηκαν για τις post hoc συγκρίσεις ενώ το μόνο paired sampled t-tests που έδειξε ότι υπήρχε σημαντική στατιστική διαφορά στο συναίσθημα **Ενδιαφέρον** ήταν αυτό ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=4.17$ ,  $SD=1.056$ ) και στην ενασχόληση με τον προγραμματισμό ρομπότ ( $M=4.69$ ,  $SD=0.525$ ).  $t(35)=-2.927$ ,  $p=.006 < .017$ .

Το 1-Way Within Subjects ANOVA που διεξήχθη για το συναίσθημα **Ικανοποιημένος** έδειξε σημαντική στατιστική διαφορά, Wilks' Lambda = 0.819,  $F(2, 34)=3.745$ ,  $p=.034 < .05$ . Όπως και παραπάνω, τρία paired sampled t-tests χρησιμοποιήθηκαν για τις post hoc συγκρίσεις. Το μόνο paired sampled t-tests που έδειξε ότι υπήρχε σημαντική στατιστική διαφορά στο συναίσθημα **Ικανοποιημένος** ήταν αυτό ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=4.28$ ,  $SD=0.815$ ) και στον προγραμματισμό ρομπότ ( $M=4.61$ ,  $SD=0.645$ ).  $t(35)=-2.523$ ,  $p=.016 < .017$ .

Τέλος το 1-Way Within Subjects ANOVA που διεξήχθη για το συναίσθημα **Αποφασισμένος** εμφάνισε σημαντική διαφορά στο συναίσθημα αυτό, Wilks' Lambda = 0.577,  $F(2, 34)=12.466$ ,  $p=.000 < .05$ . Από τα τρία paired sampled t-tests που χρησιμοποιήθηκαν για τις post hoc συγκρίσεις το μόνο που εμφάνισε σημαντική στατιστική διαφορά ήταν αυτό ανάμεσα στον κλασικό τρόπο προγραμματισμού

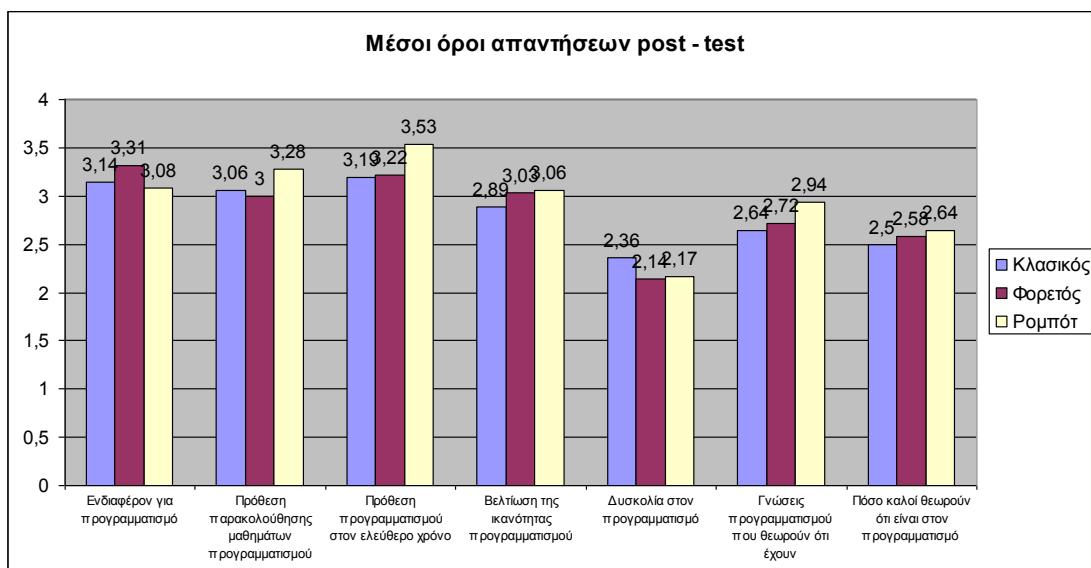
( $M=3.44$ ,  $SD=0.809$ ) και στον προγραμματισμό ρομπότ ( $M=4.08$ ,  $SD=0.692$ ).  $t(35)=-5.033$ ,  $p=.000 < .017$ .

Καταληκτικά, μέσω της μεθόδου ανάλυσης διακύμανσης, διαπιστώθηκε ότι ο προγραμματισμός ρομπότ συνολικά προκαλούσε θετικότερα συναισθήματα, στα τέσσερα από τα πέντε συναισθήματα της μελέτης, με σημαντική στατιστική διαφορά σε σχέση με τον κλασικό τρόπο προγραμματισμού. Παρόλο που ο φορετός προγραμματισμός εμφάνισε συγκριτικά καλύτερους μέσους όρους σε σχέση με τον κλασικό τρόπο προγραμματισμού, αυτή η διαφορά δεν ήταν στατιστικά σημαντική, με βάση τα ευρήματα που προέκυψαν από την εφαρμογή των 1-Way Within Subjects ANOVA και για τα πέντε συναισθήματα της μελέτης, εξαιτίας του μικρού δείγματος.

#### 4.2.2 Ενδιαφέρον Μαθητών για Προγραμματισμό

Για να διερευνηθεί η υπόθεση κατά πόσο το φυσικό αντικείμενο εμπνέει τους μαθητές και αυξάνει το ενδιαφέρον τους για προγραμματισμό, ήταν απαραίτητο να αναλυθούν οι απαντήσεις που έδωσαν οι μαθητές στο post – test.

Ο έλεγχος της κανονικότητας των απαντήσεων των μαθητών στο post test για κάθε μια από τις τρεις μεθόδους προγραμματισμού (κλασικός, φορετός και ρομπότ) έδειξε ότι όλοι οι πληθυσμοί ακολουθούσαν την κανονική κατανομή.



**Γράφημα 2: Μέσοι όροι απαντήσεων στο post – test ανά τεχνολογία προγραμματισμού**

Με βάση τα περιγραφικά στατιστικά των απαντήσεων του post test, όπως φαίνεται και στο Γράφημα 2, ο προγραμματισμός ρομπότ φάνηκε να πλεονεκτεί, τόσο ως προς

το φορετό προγραμματισμό, όσο και ως προς την κλασική μέθοδο προγραμματισμού. Επιπλέον ο φορετός προγραμματισμός φάνηκε να υπερτερεί του κλασικού.

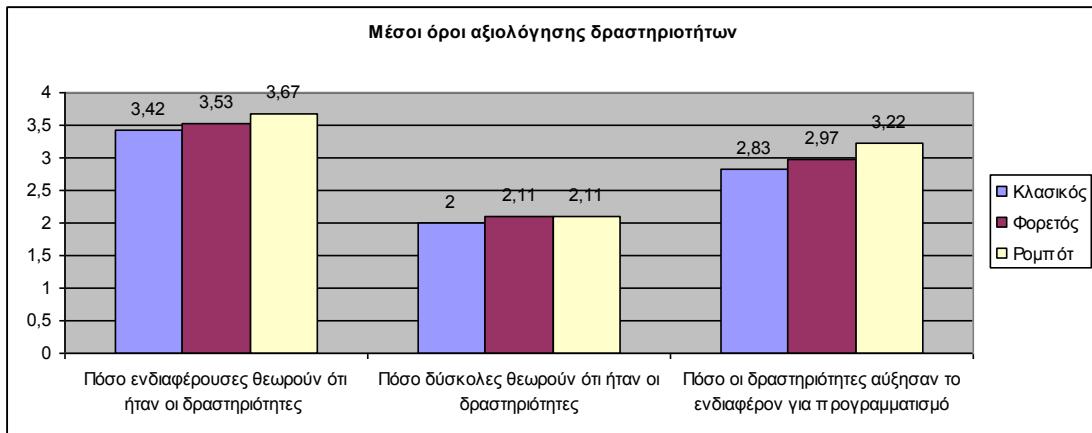
Τα 1-Way Within Subjects ANOVA έδειξαν ότι δεν υπάρχει σημαντική στατιστική διαφορά ανάμεσα στις τρεις τεχνολογίες προγραμματισμού στο **ενδιαφέρον για εκμάθηση προγραμματισμού**, Wilks' Lambda = 0.861,  $F(2, 34)=2.742$ ,  $p=.079 > .05$ , στην **πρόθεση παρακολούθησης μαθημάτων προγραμματισμού**, Wilks' Lambda = 0.868,  $F(2, 34)=2.584$ ,  $p=.090 > .05$ , στη **βελτίωση της ικανότητας προγραμματισμού**, Wilks' Lambda = 0.939,  $F(2, 34)=1.111$ ,  $p=.341 > .05$ , στη **δυσκολία προγραμματισμού**, Wilks' Lambda = 0.905,  $F(2, 34)=1.786$ ,  $p=.183 > .05$  και στο **πόσο καλοί θεωρούν οι μαθητές ότι είναι στον προγραμματισμό**, Wilks' Lambda = 0.962,  $F(2, 34)=0.671$ ,  $p=.518 > .05$ .

Σημαντική στατιστική διαφορά προέκυψε στην **πρόθεση προγραμματισμού στον ελεύθερο χρόνο**, Wilks' Lambda = 0.815,  $F(2, 34)=3.848$ ,  $p=.031 < .05$ , με τις post hoc συγκρίσεις να δείχνουν σημαντική στατιστική διαφορά ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=3.19$ ,  $SD=.749$ ) και στον προγραμματισμό ρομπότ ( $M=3.53$ ,  $SD=.560$ ).  $t(35)=-2.646$ ,  $p=.012 < .017$ . Επιπλέον σημαντική στατιστική διαφορά προέκυψε **στις γνώσεις προγραμματισμού που θεωρούν ότι έχουν οι μαθητές**, Wilks' Lambda = 0.737,  $F(2, 34)=6.066$ ,  $p=.006 < .05$ , με τις post hoc συγκρίσεις να δείχνουν πάλι σημαντική στατιστική διαφορά ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=2.64$ ,  $SD=.543$ ) και στον προγραμματισμό ρομπότ ( $M=2.94$ ,  $SD=.630$ ).  $t(35)=-3.494$ ,  $p=.001 < .017$ .

Συμπερασματικά, αν και ο προγραμματισμός φυσικών αντικειμένων εμφάνισε γενικά καλύτερους μέσους όρους από τον κλασικό τρόπο προγραμματισμού, σημαντική στατιστική διαφορά διαπιστώθηκε στην πρόθεση προγραμματισμού στον ελεύθερο χρόνο και στις γνώσεις προγραμματισμού που θεωρούσαν ότι έχουν οι μαθητές μόνο στην περίπτωση που το φυσικό αντικείμενο είναι ρομπότ.

#### 4.2.3 Αξιολόγηση Δραστηριοτήτων

Σύμφωνα με τους μέσους όρους (Γράφημα 3) η δραστηριότητα του προγραμματισμού ρομπότ φάνηκε πιο ενδιαφέρουσα σε σχέση με τις άλλες δύο δραστηριότητες ενώ οι μαθητές θεώρησαν ότι η συγκεκριμένη δραστηριότητα αύξησε περισσότερο το ενδιαφέρον τους για προγραμματισμό. Επίσης οι μαθητές είχαν την άποψη ότι και οι τρεις δραστηριότητες ήταν περίπου της ίδιας δυσκολίας.



**Γράφημα 3: Μέσοι όροι αξιολόγησης δραστηριοτήτων**

Τα 1-Way Within Subjects ANOVA, ωστόσο, έδειξαν ότι δεν υπάρχει σημαντική στατιστική διαφορά στο **πόσο ενδιαφέρουσες θεωρούν ότι είναι οι τρεις δραστηριότητες προγραμματισμού**, Wilks' Lambda = 0.901,  $F(2, 34)=1.867$ ,  $p=.170 > .05$  και στο **πόσο δύσκολες θεωρούν ότι ήταν οι δραστηριότητες**, Wilks' Lambda = 0.980,  $F(2, 34)=0.343$ ,  $p=.712 > .05$ .

Σημαντική στατιστική διαφορά προέκυψε μόνο στο **πόσο οι συγκεκριμένες δραστηριότητες αύξησαν το ενδιαφέρον τους για τον προγραμματισμό**, Wilks' Lambda = 0.762,  $F(2, 34)=5.323$ ,  $p=.010 < .05$ , με τις post hoc συγκρίσεις να δείχνουν σημαντική στατιστική διαφορά ανάμεσα στην αξιολόγηση της δραστηριότητας για τον κλασικό τρόπο προγραμματισμού ( $M=2.83$ ,  $SD=.737$ ) και στην αξιολόγηση της δραστηριότητας του προγραμματισμού ρομπότ ( $M=3.22$ ,  $SD=.722$ ).  $t(35)=-3.205$ ,  $p=.003 < .017$ .

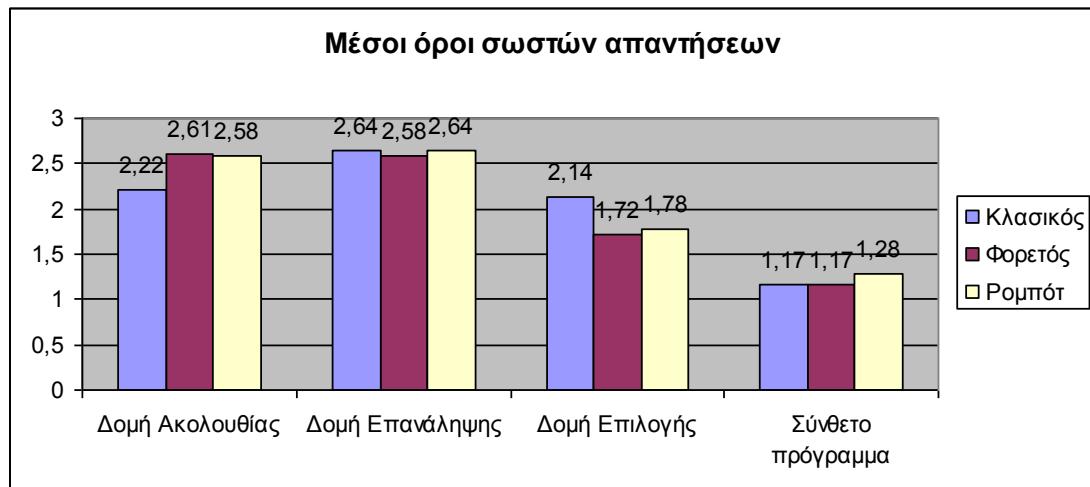
Φαίνεται λοιπόν ότι οι μαθητές αξιολόγησαν θετικότερα την δραστηριότητα που σχετιζόταν με τον προγραμματισμό ρομπότ, όσον αφορά την πρόθεση προγραμματισμού.

#### 4.2.4 Επίδοση στην Προγραμματιστική Σκέψη

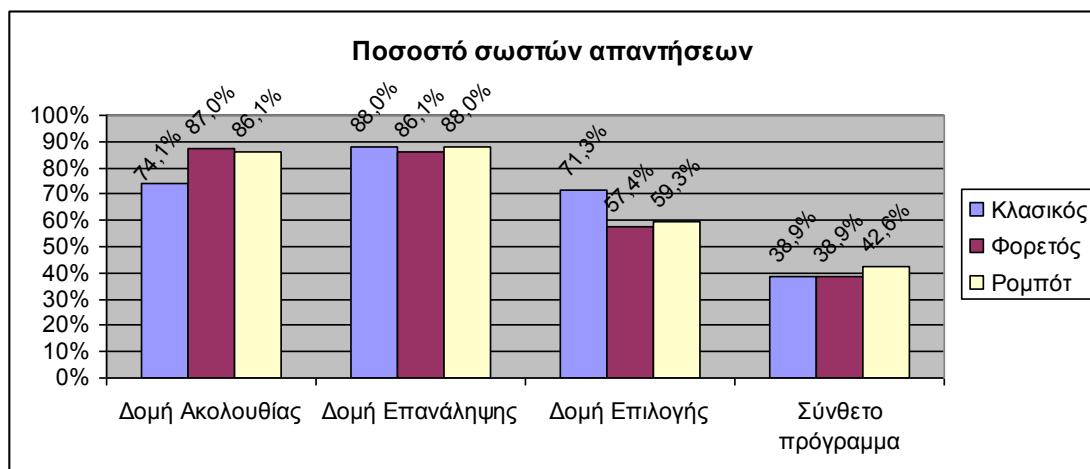
Για να διερευνηθεί το ερώτημα μέσω ποιας τεχνολογίας οι μαθητές έμαθαν καλύτερα προγραμματισμό ήταν απαραίτητο να εξεταστούν οι επιδόσεις τους στα κριτήρια αξιολόγησης στις υπό εξέταση αλγορίθμικές συνιστώσες: δομή επιλογής, δομή επανάληψης δομή επιλογής και σύνθετο πρόγραμμα (συνδυασμός των βασικών αλγορίθμικών συνιστωσών).

Ο έλεγχος της κανονικότητας των απαντήσεων στα κριτήρια αξιολόγησης των μαθητών στις βασικές αλγορίθμικές συνιστώσες, για κάθε μια από τις τρεις μεθόδους

προγραμματισμού (κλασικός, φορετός και ρομπότ), έδειξαν ότι όλοι οι πληθυσμοί ακολουθούσαν την κανονική κατανομή.



**Γράφημα 4: Μέσοι όροι σωστών απαντήσεων ανά αλγορίθμική συνιστώσα και ανά τεχνολογία προγραμματισμού**



**Γράφημα 5: Ποσοστό επί της εκατό σωστών απαντήσεων ανά αλγορίθμική συνιστώσα και ανά τεχνολογία προγραμματισμού**

Σύμφωνα με τους μέσους όρους, όπως φαίνεται στα Γραφήματα 4 και 5, παρατηρείται ότι οι μαθητές πήγαν καλύτερα στη δομή ακολουθίας στο φορετό προγραμματισμό, μετά ακολούθησε ο προγραμματισμός ρομπότ και τέλος ο κλασικός προγραμματισμός. Στη δομή επανάληψης δεν εμφανίζονταν μεγάλες διαφορές ανάμεσα στις τρεις μεθόδους προγραμματισμού. Στη δομή επιλογής, παραδόξως, οι μαθητές πήγαν καλύτερα στον κλασικό προγραμματισμό σε σχέση με τον προγραμματισμό φυσικών αντικειμένων, ενώ τέλος στο σύνθετο πρόγραμμα οι μαθητές είχαν περισσότερες σωστές απαντήσεις στο προγραμματισμό ρομπότ και μετά ακολούθησαν οι άλλες δύο τεχνολογίες

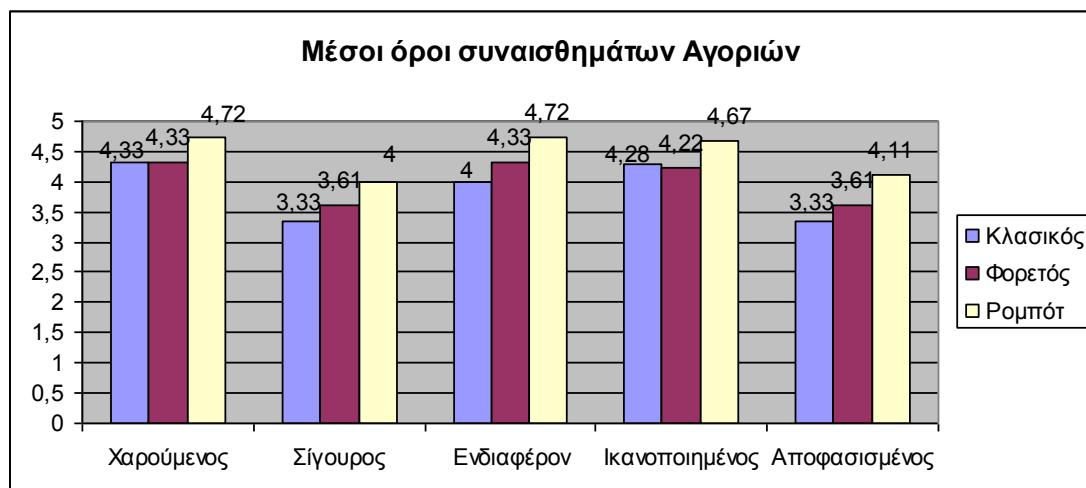
Παρά τις διαφορές στους μέσους όρους τα επαγωγικά στατιστικά δεν έδειξαν σημαντικές διαφορές στην επίδοση των μαθητών στις αλγορίθμικές συνιστώσες για κάθε μια από τις τεχνολογίες προγραμματισμού της μελέτης. Πιο συγκεκριμένα τα 1-Way Within Subjects ANOVA έδειξαν ότι δεν υπάρχει σημαντική στατιστική διαφορά στη **δομή ακολουθίας**, Wilks' Lambda = 0.922,  $F(2, 34)=1.447$ ,  $p=.249 > .05$ , στη **δομή επανάληψης**, Wilks' Lambda = 0.996,  $F(2, 34)=0.064$ ,  $p=.938 > .05$ , στη **δομή επιλογής**, Wilks' Lambda = 0.891,  $F(2, 34)=2.077$ ,  $p=.141 > .05$  και στο **σύνθετο πρόγραμμα**, Wilks' Lambda = 0.990,  $F(2, 34)=0.178$ ,  $p=.837 > .05$ .

Συμπερασματικά, φαίνεται ότι το φυσικό αντικείμενο δεν επηρέασε καταλυτικά και στον αναμενόμενο βαθμό, τις επιδόσεις των μαθητών στην εκμάθηση των προγραμματιστικών συνιστώσων, εξαιτίας, άλλωστε, και του μικρού δείγματος της μελέτης, αν και αξίζει να τονιστεί ότι υπήρχε μια τάση υπέρ του προγραμματισμού φυσικών αντικειμένων στη δομή ακολουθίας και αντιστοίχως μια υπέρ της κλασικής μεθόδου προγραμματισμού στη δομή επιλογής.

## 4.3 Ανάλυση Αποτελεσμάτων με βάση το φύλο

### 4.3.1 Συναισθήματα Αγοριών

Συμφώνα με την αρχική υπόθεση της μελέτης τα αγόρια αναμενόταν να έχουν θετικότερα συναισθήματα και μεγαλύτερο ενδιαφέρον προς τον προγραμματισμό ρομπότ.



**Γράφημα 6: Μέσοι όροι συναισθημάτων αγοριών ανά τεχνολογία προγραμματισμού**

Με βάση τα περιγραφικά στατιστικά, πράγματι, ο προγραμματισμός ρομπότ εμφάνισε θετικότερα συναισθήματα στα αγόρια και στις πέντε υπό εξέταση κατηγορίες συναισθημάτων, τόσο ως προς τον κλασικό τρόπο προγραμματισμού, όσο και ως προς

τον φορετό. Επίσης ο φορετός προγραμματισμός εμφάνισε καλύτερους μέσους όρους στα συναισθήματα Σίγουρος, Ενδιαφέρον και Αποφασισμένος σε σχέση με τον κλασικό τρόπο προγραμματισμού.

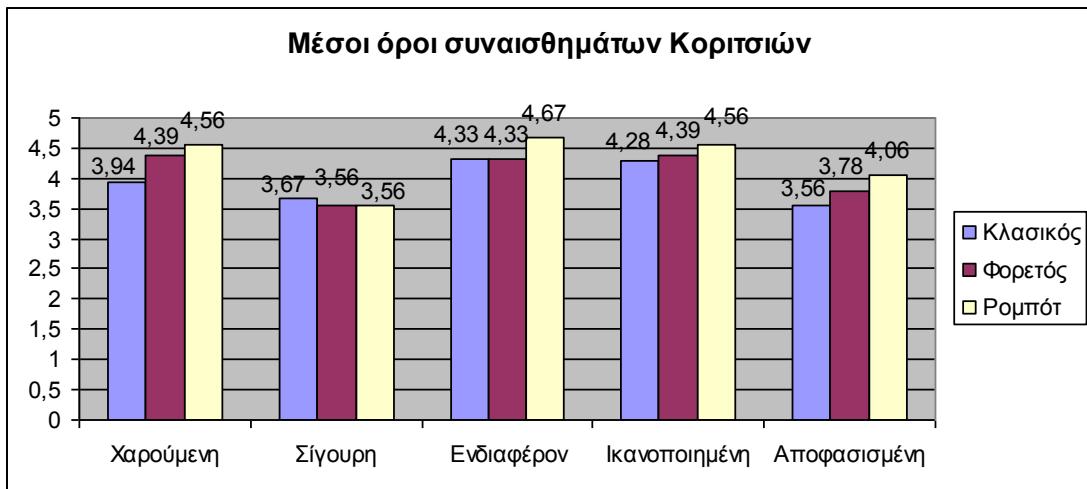
Τα 1-Way Within Subjects ANOVA που διεξήχθησαν, για τα αγόρια έδειξαν ότι δεν υπάρχει σημαντική στατιστική διαφορά ανάμεσα στις τρεις τεχνολογίες προγραμματισμού, για τα συναισθήματα **Χαρούμενος**, Wilks' Lambda = 0.791, F(2, 16)=2.119, p=.153 > .05, **Ενδιαφέρον**, Wilks' Lambda = 0.726, F(2, 16)=3.024, p=.077 > .05 και **Ικανοποιημένος**, Wilks' Lambda = 0.720, F(2, 16)=3.116, p=.072 > .05.

Σημαντική στατιστική διαφορά για τα αγόρια προέκυψε για το συναίσθημα **Σίγουρος**; Wilks' Lambda = 0.641, F(2, 16)=4.477, p=.029 < .0, με τις post hoc συγκρίσεις να δείχνουν σημαντική στατιστική διαφορά ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=3.33$ ,  $SD=1.029$ ) και στον προγραμματισμό ρομπότ ( $M=4.00$ ,  $SD=.907$ ),  $t(17)=-2.915$ ,  $p=.010 < .017$ , καθώς και στο συναίσθημα **Αποφασισμένος**, Wilks' Lambda = 0.418, F(2, 16)=11.124, p=.001 < .05, με τις post hoc συγκρίσεις να δείχνουν σημαντική στατιστική διαφορά ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=3.33$ ,  $SD=.686$ ) και στον προγραμματισμό ρομπότ ( $M=4.11$ ,  $SD=.758$ ),  $t(17)=-4.507$ ,  $p=.000 < .017$ .

#### 4.3.2 Συναισθήματα Κοριτσιών

Συμφώνα με την αρχική υπόθεση της μελέτης τα κορίτσια αναμενόταν να έχουν θετικότερα συναισθήματα και μεγαλύτερο ενδιαφέρον προς τον φορετό προγραμματισμό.

Με βάση τα περιγραφικά στατιστικά η συγκεκριμένη υπόθεση δεν φάνηκε να επαληθεύεται, αφού ο προγραμματισμός ρομπότ εμφάνισε θετικότερα συναισθήματα στα κορίτσια στις τέσσερις από τις πέντε υπό εξέταση κατηγορίες συναισθημάτων, τόσο ως προς τον κλασικό τρόπο προγραμματισμού, όσο και ως προς τον φορετό. Βέβαια, ο φορετός προγραμματισμός εμφάνισε καλύτερους μέσους όρους στα συναισθήματα **Χαρούμενη**, **Ικανοποιημένη** και **Αποφασισμένη** σε σχέση με τον κλασικό τρόπο προγραμματισμού. Ο κλασικός τρόπος προγραμματισμού υπερτερούσε των άλλων δύο μεθόδων μόνο στο συναίσθημα **Σίγουρη**.



**Γράφημα 7: Μέσοι όροι συναισθημάτων κοριτσιών ανά τεχνολογία προγραμματισμού**

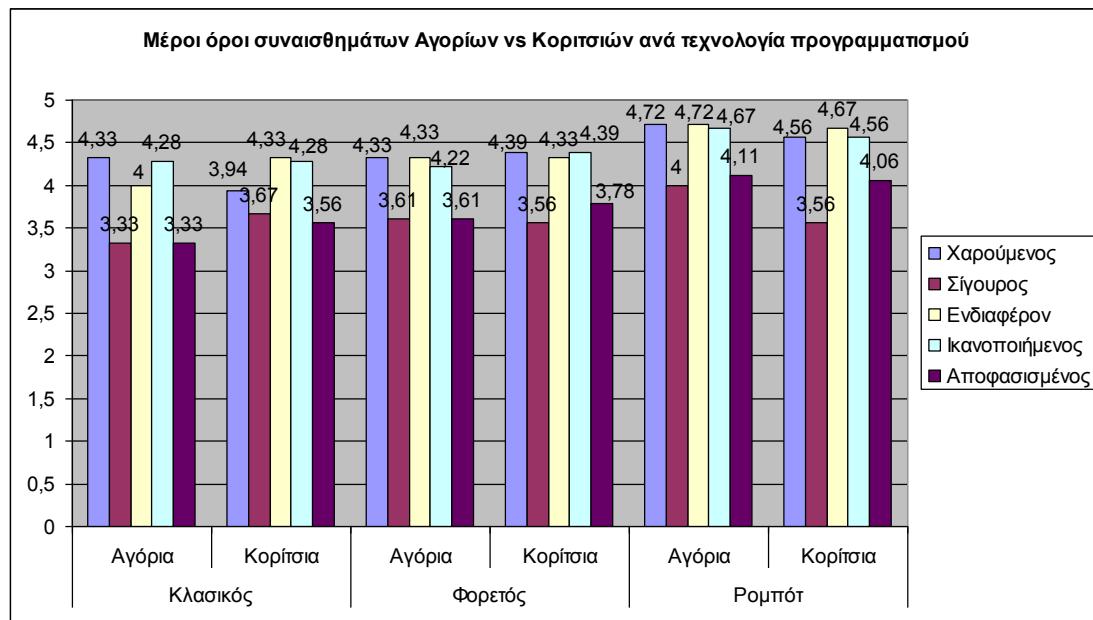
Όσον αφορά τα επαγωγικά στατιστικά, τα 1-Way Within Subjects ANOVA που διεξήχθησαν για τα κορίτσια έδειξαν ότι δεν υπάρχει σημαντική στατιστική διαφορά ανάμεσα στις τρεις τεχνολογίες προγραμματισμού, για τα συναισθήματα **Σίγουρη**, Wilks' Lambda = 0.981,  $F(2, 16)=0.151$ ,  $p=.861 > .05$ , **Ενδιαφέρον**, Wilks' Lambda = 0.799,  $F(2, 16)=2.008$ ,  $p=.0167 > .05$  και **Ικανοποιημένη**, Wilks' Lambda = 0.845,  $F(2, 16)=1.466$ ,  $p=.260 > .05$

Σημαντική στατιστική διαφορά για τα κορίτσια προέκυψε για το συναισθήμα **Χαρούμενη**, Wilks' Lambda = 0.482,  $F(2, 16)=8.591$ ,  $p=.003 < .05$ , με τις post hoc συγκρίσεις να δείχνουν σημαντική στατιστική διαφορά ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=3.94$ ,  $SD=.873$ ) και στον προγραμματισμό ρομπότ ( $M=4.56$ ,  $SD=.705$ ).  $t(17)=-4.267$ ,  $p=.001 < .017$ , καθώς και στο συναισθήμα **Αποφασισμένη**, Wilks' Lambda = 0.653,  $F(2, 16)=4.249$ ,  $p=.033 < .05$ , με τις post hoc συγκρίσεις να δείχνουν σημαντική στατιστική διαφορά ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=3.56$ ,  $SD=.922$ ) και στον προγραμματισμό ρομπότ ( $M=4.06$ ,  $SD=.639$ ).  $t(17)=-2.699$ ,  $p=.015 < .017$ .

#### 4.3.2 Συναισθήματα Αγοριών vs Κοριτσιών

Τέλος συγκρίθηκαν τα συναισθήματα των αγοριών σε σχέση με εκείνα των κοριτσιών τόσο συνολικά, όσο και ξεχωριστά για κάθε μια από τις τρεις τεχνολογίες. Στην προκειμένη περίπτωση χρησιμοποιήθηκε το independent samples T-test μιας και οι πληθυσμοί, απαντήσεις συναισθημάτων αγοριών vs κοριτσιών, ήταν ανεξάρτητοι μεταξύ τους. Τα πέντε independent samples T-test για κάθε ένα από τα πέντε

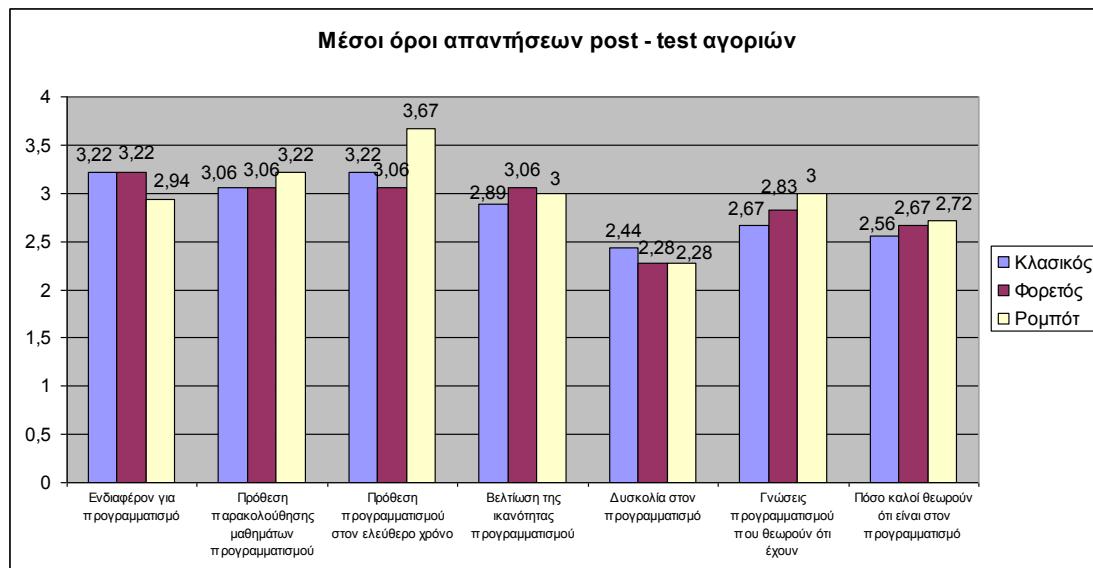
συναισθήματα της μελέτης δεν έδειξαν σημαντική στατιστική διαφορά ανάμεσα στα αγόρια και τα κορίτσια.



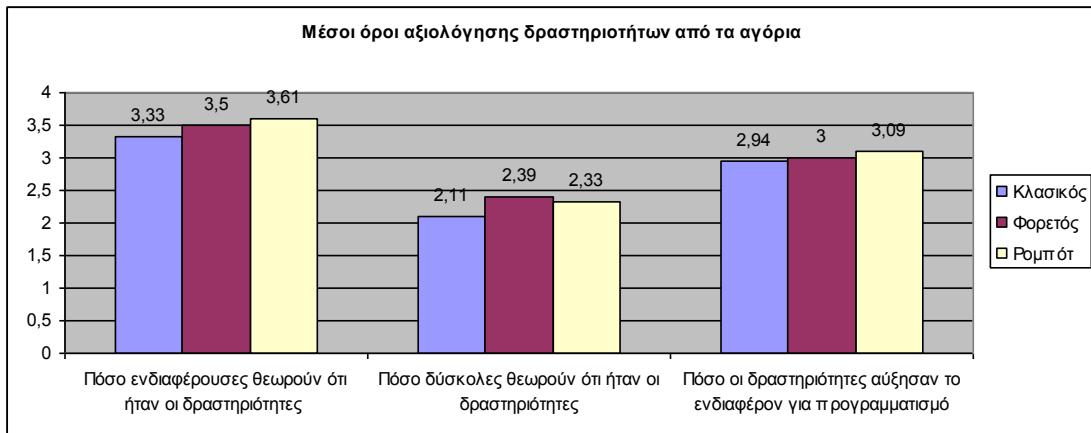
**Γράφημα 8: Μέσοι όροι συναισθημάτων Αγοριών vs Κοριτσιών ανά τεχνολογία προγραμματισμού**

Ομοίως, και στην περίπτωση σύγκρισης των συναισθημάτων των αγοριών σε σχέση με εκείνα των κοριτσιών για κάθε μια τεχνολογία ξεχωριστά δεν έδειξαν σημαντική στατιστική διαφορά.

#### 4.3.3 Ενδιαφέρον Αγοριών – Κοριτσιών για Προγραμματισμό

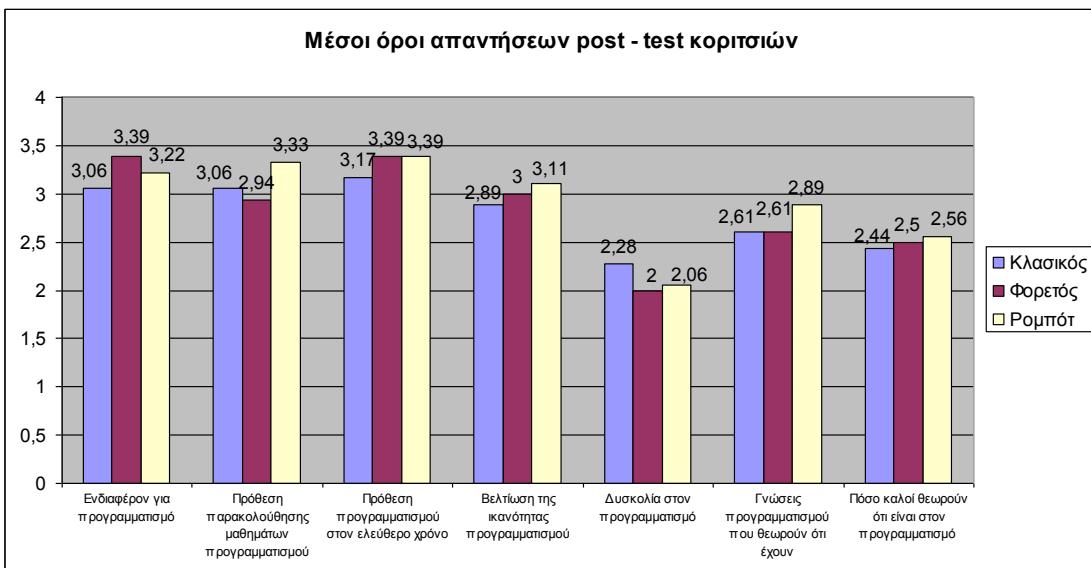


**Γράφημα 9: Μέσοι όροι απαντήσεων στο post – test ανά τεχνολογία προγραμματισμού για τα αγόρια**

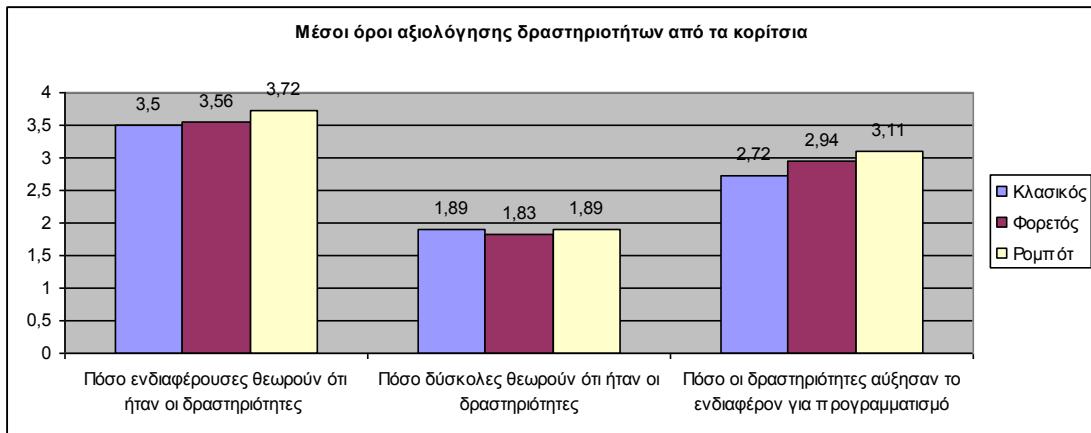


**Γράφημα 10: Μέσοι όροι αξιολόγησης δραστηριοτήτων από τα αγόρια**

Τα 1-Way Within Subjects ANOVA, για τα αγόρια, έδειξαν ότι υπήρχε σημαντική στατιστική διαφορά στην **πρόθεση προγραμματισμού στον ελεύθερο χρόνο**, Wilks' Lambda = 0.348,  $F(2, 16)=14.992$   $p=.000 < .05$ , με τις post hoc συγκρίσεις να δείχνουν διαφορά ανάμεσα τόσο ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=3.22$ ,  $SD=.732$ ) και στον προγραμματισμό ρομπότ ( $M=3.67$ ,  $SD=.485$ );  $t(17)=-3.063$ ,  $p=.007 < .017$ , όσο και ανάμεσα στον φορετό προγραμματισμό ( $M=3.06$ ,  $SD=.725$ ) και στον προγραμματισμό ρομπότ ( $M=3.67$ ,  $SD=.485$ );  $t(17)=-4.267$ ,  $p=.001 < .017$ . Στις υπόλοιπες υπό εξέταση κατηγορίες δεν υπήρξαν σημαντικές στατιστικές διαφορές, όσον αφορά τα αγόρια.



**Γράφημα 11: Μέσοι όροι απαντήσεων στο post – test ανά τεχνολογία προγραμματισμού για τα κορίτσια**



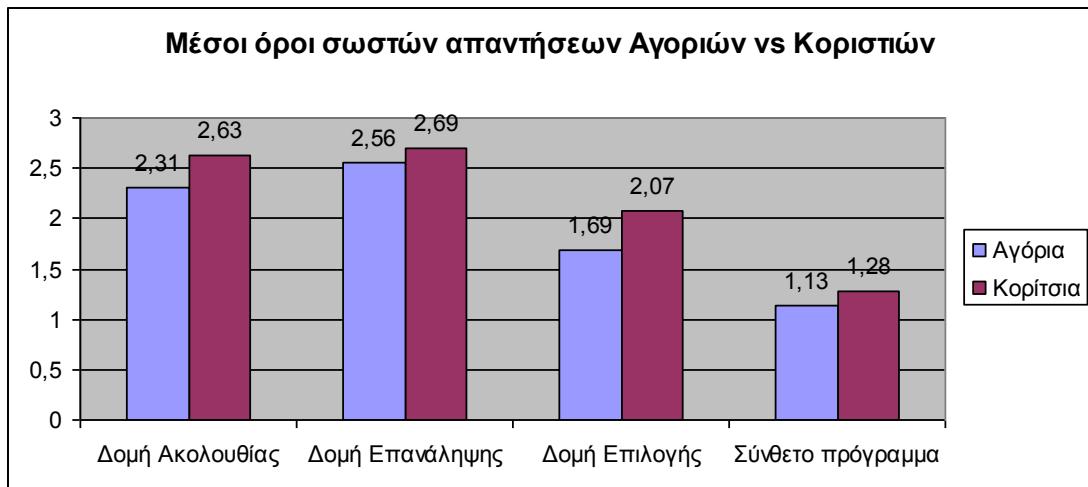
**Γράφημα 12: Μέσοι όροι αξιολόγησης δραστηριοτήτων από τα κορίτσια**

Τα 1-Way Within Subjects ANOVA για τα κορίτσια έδειξαν ότι υπήρξε σημαντική στατιστική διαφορά **πόσο οι συγκεκριμένες δραστηριότητες αύξησαν το ενδιαφέρον τους για τον προγραμματισμό**, Wilks' Lambda = 0.611, F(2, 16)=5.091 p=.019 < .05, : με τις post hoc συγκρίσεις να δείχνουν διαφορά ανάμεσα, τόσο ανάμεσα στον κλασικό τρόπο προγραμματισμού ( $M=2.72$ ,  $SD=.752$ ), όσο και στον προγραμματισμό ρομπότ ( $M=3.11$ ,  $SD=.832$ );  $t(17)=-3.289$ ,  $p=.004 < .017$ . Στις υπόλοιπες υπό εξέταση κατηγορίες δεν υπήρξαν σημαντικές στατιστικές διαφορές όσον αφορά τα κορίτσια.

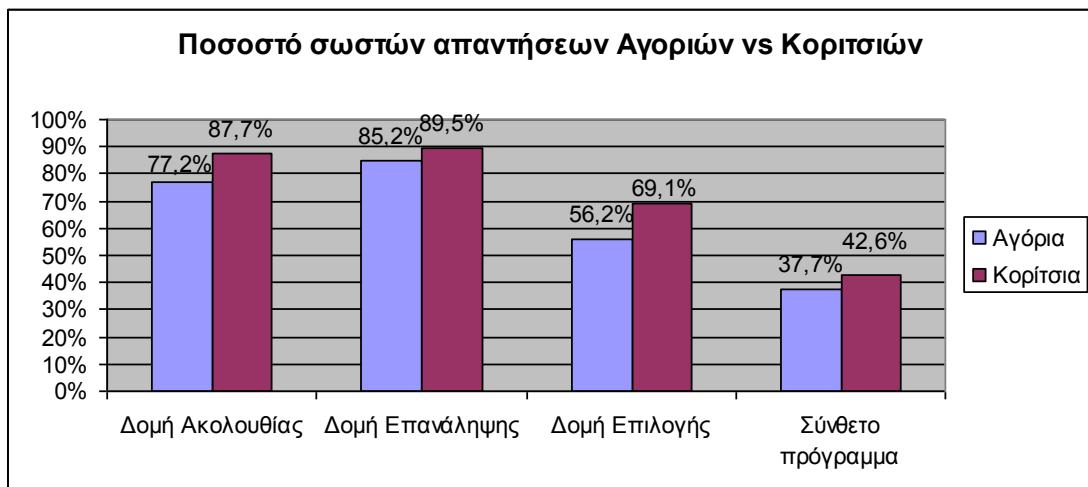
Τέλος συγκρίθηκαν οι απαντήσεις στο post test των αγοριών σε σχέση με εκείνα των κοριτσιών, τόσο συνολικά, όσο και ξεχωριστά για κάθε μια από τις τρεις τεχνολογίες. Στην προκειμένη περίπτωση χρησιμοποιήθηκε το independent samples T-test μιας και οι πληθυσμοί, απαντήσεις στο post test αγοριών vs κοριτσιών, ήταν ανεξάρτητοι μεταξύ τους. Τα independent samples T-test για κάθε μια από τις κατηγορίες του post test της μελέτης δεν έδειξαν σημαντική στατιστική διαφορά ανάμεσα στα αγόρια και τα κορίτσια, ενώ και στην περίπτωση σύγκρισης των απαντήσεων στο post test των αγοριών σε σχέση με εκείνα των κοριτσιών για κάθε μια τεχνολογία ξεχωριστά δεν υπήρξε σημαντική στατιστική διαφορά.

#### 4.3.4 Σύγκριση Επίδοσης Αγοριών – Κοριτσιών στην Προγραμματιστική Σκέψη

Σύμφωνα με τους μέσους όρους των απαντήσεων στα κριτήρια αξιολόγησης, τα κορίτσια τα κατάφεραν καλύτερα από τα αγόρια σε όλες τις υπό εξέταση αλγορίθμικές συνιστώσες.



**Γράφημα 13: Μέσοι όροι σωστών απαντήσεων αγοριών vs κοριτσιών ανά αλγορίθμική συνιστώσα**



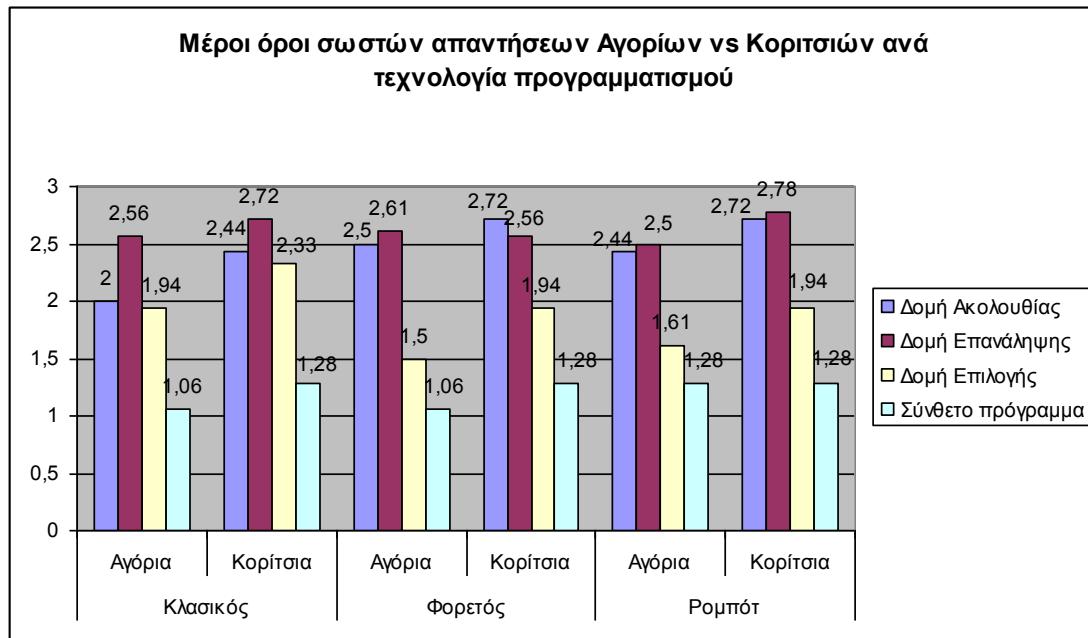
**Γράφημα 14: Ποσοστό επί της εκατό σωστών απαντήσεων αγοριών vs κοριτσιών ανά αλγορίθμική συνιστώσα**

Για να διαπιστωθεί ότι υπήρχε σημαντική στατιστική διαφορά χρησιμοποιήθηκε το independent samples T-test για κάθε μια από την αλγορίθμικές συνιστώσες μιας και οι πληθυσμοί, απαντήσεις στα κριτήρια αξιολόγησης αγοριών vs κοριτσιών, ήταν ανεξάρτητοι μεταξύ τους. Μόνο στην περίπτωση της **δομής επιλογής** υπήρξε σημαντική στατιστική διαφορά ανάμεσα στην επίδοση των αγοριών ( $M=1.69$ ,  $SD=.987$ ) και στην επίδοση των κοριτσιών ( $M=2.07$ ,  $SD=.929$ ).  $t(106) = -2.109$ ,  $p = 0.037$ .

Τα κορίτσια, λοιπόν, τα κατάφερναν καλύτερα σε όλες τις αλγορίθμικές συνιστώσες σε σχέση με τα αγόρια, αλλά στατιστικά σημαντική διαφορά υπήρχε στη δομή επιλογής. Διαφορά φαίνεται να υπήρχε και στη δομή ακολουθίας αλλά δεν ήταν τόσο σημαντική στατιστικά. Τα συγκεκριμένα αποτελέσματα καταδεικνύουν ότι στην

πράξη τα αγόρια δε φάνηκε να κατείχαν περισσότερες γνώσεις προγραμματισμού, όπως τα ίδια ανέφεραν, από τα κορίτσια, αλλά αντιθέτως οι τελευταίες ήταν αυτές που ελαφρώς υπερείχαν στην προγραμματιστική σκέψη.

Τέλος, συγκρίθηκαν οι επιδόσεις των αγοριών με εκείνες των κοριτσιών ανά προγραμματισμού συνιστώσα για κάθε μια τεχνολογία προγραμματισμού ξεχωριστά.



**Γράφημα 15: Μέσοι όροι σωστών απαντήσεων αγοριών vs κοριτσιών ανά αλγορίθμική συνιστώσα και ανά τεχνολογία προγραμματισμού**

Στην περίπτωση του κλασικού προγραμματισμού μέσω του περιβάλλοντος της Scratch με βάση του μέσους όρους τα κορίτσια είχαν καλύτερη επίδοση σε σχέση με τα αγόρια. Στην περίπτωση του φορετού προγραμματισμού μέσω της τεχνολογίας του LilyPad με βάση του μέσους όρους τα κορίτσια είχαν καλύτερη επίδοση σε σχέση με τα αγόρια στις τρεις από τις τέσσερις κατηγορίες και μόνο στην δομή επανάληψης τα αγόρια ήταν καλύτερα από τα κορίτσια στον φορετό υπολογισμό. Στον προγραμματισμό ρομπότ πάλι τα κορίτσια φάνηκε ότι είχαν καλύτερη επίδοση σε σχέση με την επίδοση των αγοριών στις τρεις από τις τέσσερις κατηγορίες ενώ στην αξιολόγηση του σύνθετου προγράμματος φάνηκε ότι είχαν την ίδια επίδοση.

Τα επαγωγικά στατιστικά, ωστόσο, με εφαρμογή των independent samples T-test, δεν έδειξαν σημαντική στατιστική διαφορά ανάμεσα στα αγόρια και στα κορίτσια για κάθε μια τεχνολογία προγραμματισμού ξεχωριστά.

## **Κεφάλαιο 5 Συζήτηση – Συμπεράσματα**

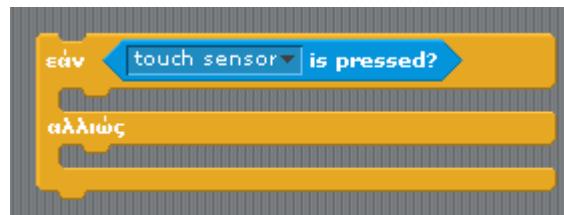
### **5.1 Προγραμματισμός με φυσικά αντικείμενα**

Η αρχική υπόθεση ότι ο προγραμματισμός φυσικών αντικειμένων προκαλεί θετικότερα συναισθήματα από τον προγραμματισμό με την κλασική μέθοδο φάνηκε από τα ευρήματα της έρευνας να επαληθεύεται πλήρως μόνο στην περίπτωση του ρομπότ. Οι μαθητές και οι μαθήτριες νοιώθουν εμφανώς θετικότερα συναισθήματα. Παράλληλα, το ρομπότ αφενός τους εμπνέει περισσότερο και αφετέρου αυξάνει σε μεγαλύτερο βαθμό το ενδιαφέρον τους για προγραμματισμό σε σχέση με τον αμιγώς εικονικό κόσμο που τους προσφέρει το περιβάλλον μια κλασικής γλώσσας προγραμματισμού. Ταυτόχρονα, οι μαθητές δηλώνουν πως θα ήθελαν να προγραμματίσουν το ρομπότ και στον ελεύθερο τους χρόνο. Παρά το γεγονός ότι και στην περίπτωση του φορετού προγραμματισμού εμφανίζεται μια τάση επαλήθευσης της αρχικής υπόθεσης, αυτή δεν ήταν τόσο σημαντική όσο θα αναμενόταν και στο βαθμό που καταγράφθηκε στην περίπτωση του ρομπότ. Το γεγονός αυτό είχε να κάνει και με τους περιορισμούς, κυρίως χρονικούς, των δραστηριοτήτων της μελέτης που θα αναλυθούν αργότερα.

Παρόλο που το φυσικό αντικείμενο δεν επηρεάζει καταλυτικά και στον αναμενόμενο βαθμό τις επιδόσεις των μαθητών στην εκμάθηση των προγραμματιστικών συνιστώσων, φάνηκε ότι διαμορφωνόταν η τάση οι μαθητές και οι μαθήτριες να τα καταφέρνουν καλύτερα στη δομή ακολουθίας και το σύνθετο πρόγραμμα, όσον αφορά στις τεχνολογίες προγραμματισμού φυσικών αντικειμένων, ενώ, παραδόξως, οι επιδόσεις τους στην κλασική μέθοδο προγραμματισμού ήταν καλύτερες στη δομή επιλογής. Στη δομή επανάληψης οι επιδόσεις ήταν σχεδόν ίδιες και για τις τρεις τεχνολογίες. Σύμφωνα δε με τους ίδιους τους μαθητές, όταν ρωτήθηκαν στις συνεντεύξεις γιατί πιστεύουν ότι πήγαν καλύτερα στον προγραμματισμό στη περίπτωση ρομπότ, αρκετοί δικαιολόγησαν την επίδοση αυτή λόγω του ότι το έβλεπαν να κινείται. Το αποτέλεσμα, δηλαδή, της εκτέλεσης των εντολών του προγράμματος μεταφραζόταν σε κίνηση του φυσικού αντικειμένου.

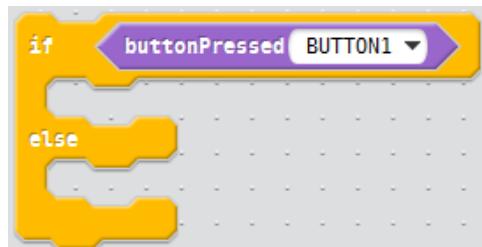
Αξίζει, ωστόσο, να εξεταστεί αναλυτικότερα η περίπτωση της δομής επιλογής, αφού θα ήταν αναμενόμενο οι μαθητές και μαθήτριες να πάνε και εκεί καλύτερα. Κατά την αξιολόγηση της συγκεκριμένης προγραμματιστικής συνιστώσας, στις τεχνολογίες προγραμματισμού φυσικών αντικειμένων σε σχέση με τον προγραμματισμό εικονικών

αντικειμένων, αυτό παραδόξως δε συνέβη. Για να δοθεί μια επαρκής εξήγηση για το συγκεκριμένο αποτέλεσμα, θα πρέπει να εξεταστούν λεπτομερέστερα οι δραστηριότητες που κλήθηκαν να υλοποιήσουν οι μαθητές, και ειδικότερα η ενότητα η οποία είχε να κάνει με τη δομή επιλογής. Κατά τη σύνταξη των εντολών της δομής επιλογής στην περίπτωση του προγραμματισμού ρομπότ μέσω του περιβάλλοντος ανάπτυξης εφαρμογών του Enchanting, γινόταν απευθείας χρήση της εντολής . Το συγκεκριμένο οπτικό στοιχείο αντιπροσώπευε τη συνθήκη που γινόταν αληθής, όταν πιεζόταν ο αισθητήρας πίεσης, και παρέμενε ψευδής, όταν ο αισθητήρας δεν ήταν πιεσμένος. Η συμπεριφορά της εντολής ήταν εξ αρχής προκαθορισμένη. Η σύνταξη της σύνθετης δομής επιλογής που χρησιμοποιήθηκε στις δραστηριότητες φαίνεται στην Εικόνα 12.



Εικόνα 12: Σύνθετη Δομή Επιλογής στο Enchanting

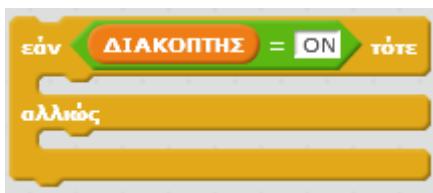
Το ίδιο συνέβαινε και στην περίπτωση του προγραμματισμού φυσικών αντικειμένων μέσω του περιβάλλοντος του Modkit. Το οπτικό στοιχείο που αντιπροσωπεύει τη συνθήκη ήταν έτοιμο , όπως και η συμπεριφορά της συνθήκης που ήταν εξ αρχής προκαθορισμένη, αφού η συνθήκη γινόταν αληθής, όταν ο διακόπτης ήταν στο ON, και ψευδής, όταν ο διακόπτης ήταν στο OFF. Η σύνταξη της σύνθετης δομής επιλογής στην προκειμένη περίπτωση φαίνεται στην παρακάτω εικόνα.



Εικόνα 13: Σύνθετη Δομή Επιλογής στο Modkit

Και στις δυο παραπάνω περιπτώσεις, οι εντολές που αντιπροσώπευαν τη συνθήκη ήταν ήδη διαθέσιμες και οι μαθητές απλά έπρεπε να τις τοποθετήσουν στην κατάλληλη θέση για να συντάξουν σωστά την εντολή επιλογής.

Αντίθετα στην περίπτωση του προγραμματισμού μέσω του περιβάλλοντος της scratch δεν υπήρχε έτοιμη εντολή που να αντιπροσωπεύει τη συνθήκη η οποία θα χρησιμοποιείτο στη δομή επιλογής. Διαθέσιμη ήταν μόνο η μεταβλητή **ΔΙΑΚΟΠΗΣ** με την τιμή να αλλάζει από ON σε OFF κάθε φορά που οι μαθητές έκαναν κλικ πάνω στο διακόπτη ο οποίος είχε τοποθετηθεί στο σκηνικό. Στην συνέχεια, οι μαθητές έπρεπε να συντάξουν μόνοι τους τη συνθήκη  , ώστε να καθορίσουν την συμπεριφορά της και να μπορέσουν μετά να τη χρησιμοποιήσουν στη δομή επιλογής.



Εικόνα 14: Σύνθετη Δομή Επιλογής στο Scratch

Γίνεται λοιπόν αντιληπτό ότι, ενώ στις περιπτώσεις του περιβάλλοντος του Enchanting και του Modkit η συμπεριφορά της συνθήκης ήταν εξ αρχής προκαθορισμένη, στην περίπτωση της Scratch οι μαθητές έπρεπε να καθορίσουν μόνοι τους τη συμπεριφορά της συνθήκης συντάσσοντας κατάλληλα την εντολή. Το γεγονός αυτό, κατά πάσα πιθανότητα, συνετέλεσε στο να αντιληφθούν καλύτερα οι μαθητές τη λογική της δομής επιλογής και κατ' επέκταση να έχουν καλύτερες επιδόσεις στα κριτήρια αξιολόγησης στον κλασικό προγραμματισμό υπολογιστή μέσω του προγραμματιστικού περιβάλλοντος της Scratch σε σχέση, τόσο με τον προγραμματισμό ρομπότ, όσο και τον φορετό προγραμματισμό. Αναδεικνύεται, λοιπόν, η σημασία που μπορεί να έχει το περιβάλλον ανάπτυξης εφαρμογών μιας γλώσσας και ο τρόπος σύνταξης των εντολών στη διαδικασία κατανόησης των προγραμματιστικών εννοιών, πέρα από το αν τα αντικείμενα που προγραμματίζουν οι μαθητές είναι φυσικά ή εικονικά.

## 5.2 Φύλο μαθητών και προγραμματισμός

Αν και θεωρήθηκε ότι τα αγόρια θα έδειχναν μεγαλύτερο ενδιαφέρον στη ρομποτική και τα κορίτσια στο φορετό προγραμματισμό, στην πράξη η συγκεκριμένη

υπόθεση δεν επαληθεύτηκε για την περίπτωση των κοριτσιών, αφού και τα δύο φύλα έδειξαν ξεκάθαρη προτίμηση στον προγραμματισμό ρομπότ. Το γεγονός αυτό, ίσως, είχε να κάνει με τους περιορισμούς, κυρίως τους χρονικούς, στο σχεδιασμό των δραστηριοτήτων της μελέτης με αποτέλεσμα να μην προκύψουν εμφανείς διαφορές στις προτιμήσεις των μαθητών.

Παράλληλα, επιβεβαιώθηκε από τα ευρήματα της έρευνας, ότι όντως το φυσικό αντικείμενο, το ρομπότ σε μεγαλύτερο βαθμό και τα e-textiles σε μικρότερο, αυξάνει σημαντικά το ενδιαφέρον των κοριτσιών για προγραμματισμό και τους δημιουργεί θετικότερα συναισθήματα.

Αναφορικά, τέλος, με τις επιδόσεις των υποκειμένων της έρευνας στην προγραμματιστική σκέψη, παρόλο που τα αγόρια θεωρούσαν ότι υπερέχουν τεχνολογικά και θα περίμενε κανείς, συμφώνα με την επικρατούσα αντίληψη, να πάνε καλύτερα στις ασκήσεις αξιολόγησης προγραμματισμού, στην πράξη τα κορίτσια υπερτερούν των αγοριών σε όλες τις κατηγορίες των αλγορίθμικών συνιστώσων, ιδιαίτερα στη δομή επιλογής, που μελετήθηκαν κατά τη διάρκεια της μελέτης.

## 5.4 Περιορισμοί

Η συντριπτική πλειοψηφία των μαθητών και μαθητριών της μελέτης δεν είχε προηγούμενη εμπειρία προγραμματισμού, όπως φάνηκε και από τα ευρήματα του pre – test. Μέσω, λοιπόν, των συγκεκριμένων δραστηριοτήτων οι μαθητές ήρθαν για πρώτη φορά σε επαφή με τον προγραμματισμό. Ωστόσο οι χρονικοί περιορισμοί είχαν ως αποτέλεσμα τη δημιουργία δραστηριοτήτων οι οποίες δεν ξεπερνούσαν χρονικά τη μια ώρα για κάθε μια τεχνολογία προγραμματισμού. Οι μαθητές, επομένως, πήραν μόνο μια μικρή γεύση από τις δυνατότητες κάθε μιας τεχνολογίας. Στην περίπτωση της πλατφόρμας ρομπότικης Lego Mindstorms δεν ήρθαν σε επαφή με όλους τους διαθέσιμους αισθητήρες του ρομπότ, δεν τους δόθηκε η δυνατότητα να σχεδιάσουν και να κατασκευάσουν πιο πολύπλοκες προσωπικές κατασκευές ή να δημιουργήσουν πιο προχωρημένα προγράμματα, ώστε τα ρομπότ τους να συμπεριφέρονται με πιο έξυπνο τρόπο. Αντίστοιχα στην τεχνολογία φορετού οι μαθητές δεν γνώρισαν όλους τους αισθητήρες που είναι διαθέσιμοι στο LilyPad, δεν ασχολήθηκαν με τον σχεδιασμό και την ενσωμάτωση των ηλεκτρονικών στοιχείων σε δικά τους αντικείμενα, δεν δημιούργησαν πιο προχωρημένα προγράμματα. Προηγούμενες μελέτες [38, 68], άλλωστε, που αφορούσαν την τεχνολογία LilyPad, με θετικότερα αποτελέσματα, είχαν μεγαλύτερη χρονική διάρκεια έτσι ώστε οι μαθητές να γνωρίσουν σε βάθος τη

συγκεκριμένη τεχνολογία και να τους δοθεί η δυνατότητα να δημιουργήσουν τα δικά τους προσωπικά αντικείμενα, e-textiles, πριν ασχοληθούν με τον προγραμματισμό. Τέλος στο Scratch δεν τους δόθηκε η δυνατότητα να δημιουργήσουν πιο προχωρημένα προγράμματα όπως είναι τα παιχνίδια, να μοιραστούν τις δημιουργίες τους ή να εξερευνήσουν και να αναμίξουν δημιουργίες άλλων ατόμων μέσω της δικτυακής πλατφόρμας διαμοιρασμού αρχείων του Scratch. Θα πρέπει, επίσης, να ληφθεί υπ' όψιν ότι αρκετές φορές ενώ ένα αντικείμενο ή μια τεχνολογία στην αρχή μπορεί να ενθουσιάσει τα παιδιά προκαλώντας τους θετικά συναισθήματα μετά από κάποιο εύλογο χρονικό διάστημα να τους φαίνεται βαρετό και να χάσουν το ενδιαφέρον τους για αυτό. Το γεγονός, λοιπόν, ότι οι μαθητές δεν ασχολήθηκαν πολύ χρόνο με τις υπό μελέτη τεχνολογίες προγραμματισμού δεν μας επέτρεψε να σχηματίσουμε μια ξεκάθαρη γνώμη για την αντοχή που έχουν οι συγκεκριμένες τεχνολογίες στο πέρασμα του χρόνου και τα συναισθήματα που βιώνουν μακροπρόθεσμα οι μαθητές. Είναι απαραίτητη, λοιπόν, η δημιουργία στο μέλλον κατάλληλων δραστηριοτήτων μεγαλύτερης χρονικής διάρκειας και ποικιλίας ως προς το περιεχόμενο.

Επιπλέον περιορισμοί αποτέλεσαν οι δημογραφικοί παράγοντες, όπως είναι η ηλικία των υποκειμένων και η βαθμίδα εκπαίδευσης. Η μελέτη επικεντρώθηκε αποκλειστικά σε μαθητές και μαθήτριες της Α' τάξης του Γυμνασίου, ηλικίας 12 – 13 χρονών. Μελλοντικά θα μπορούσαν να εξεταστούν τα συναισθήματα των μαθητών σε μεγαλύτερο εύρος ηλικιών, τόσο στην πρωτοβάθμια όσο και στην δευτεροβάθμια εκπαίδευση.

Όσον αφορά την επίδοση στην προγραμματιστική σκέψη, και εδώ κρίνεται απαραίτητη η διεξοδικότερη αξιολόγηση των μαθητών και μαθητριών στις αλγορίθμικές συνιστώσες. Η επανάληψη, άλλωστε, και η διάρκεια έχουν βαρύνουσα σημασία στη διαδικασία της μάθησης. Αφενός, λοιπόν, θα πρέπει να δοθεί περισσότερος χρόνος στους μαθητές και στις μαθήτριες να αφομοιώσουν τις προγραμματιστικές έννοιες και αφετέρου κρίνεται απαραίτητη η χρήση επιπλέον εργαλείων για την αντικειμενικότερη και πιο εκτεταμένη αξιολόγηση των μαθητών, πέρα από τα κριτήρια αξιολόγησης.

Τέλος, εξαιτίας χρονικών περιορισμών της μελέτης δεν συμπεριλήφθη στις υπό εξέταση τεχνολογίες προγραμματισμού και η δημοφιλής τεχνολογία προγραμματισμού κινητών συσκευών App Inventor [26]. Ωστόσο θεμιτό θα ήταν να συμπεριληφθεί σε εκτενέστερη μελλοντική έρευνα.

## 5.5 Συμπεράσματα

Και οι τρεις τεχνολογίες προγραμματισμού που εξετάστηκαν κατά τη διάρκεια της μελέτης αποτελούν σημαντικά εργαλεία στην προσπάθεια προσέλκυσης των μαθητών στην επιστήμη των υπολογιστών. Μέσω αυτών οι μαθητές μπορούν να έρθουν σε επαφή με τον προγραμματισμό με ευχάριστο και καινοτόμο τρόπο. Ο προγραμματισμός ρομπότ μέσω της πλατφόρμας Lego Mindstorms, σύμφωνα με τα ευρήματα της έρευνας, εμφάνισε ξεκάθαρα τα πιο ελπιδοφόρα μηνύματα ως προς τα συναισθήματα των μαθητών και το ενδιαφέρον για προγραμματισμό. Παρόλο που ο φορετός προγραμματισμός, μέσω της τεχνολογίας LilyPad, δεν επαλήθευσε πλήρως τις προσδοκίες που υπήρχαν αρχικά, φαίνεται να εμφανίζει μια τάση υπεροχής έναντι της κλασικής μεθόδου προγραμματισμού. Ωστόσο για να υπάρξει ξεκάθαρη επιβεβαίωση αυτής της τάσης χρειάζονται επιπλέον προσπάθειες μελέτης στο μέλλον.

Τα παραπάνω ευρήματα μπορούν να ωφελήσουν τους εκπαιδευτικούς, συνδράμοντας τους στο σχεδιασμό αποτελεσματικών δραστηριοτήτων προγραμματισμού, στοχευόμενων στο φυσικό αντικείμενο, οι οποίες θα διατηρούν αμείωτο το ενδιαφέρον των μαθητών, ανεξάρτητα από το φύλο τους. Βέβαια, κατά πόσο μπορούν οι συγκεκριμένες τεχνολογίες να ενσωματωθούν στο ευρύτερο πρόγραμμα σπουδών του εκπαιδευτικού συστήματος της Ελλάδας εξαρτάται, πλέον, και σε μεγάλο βαθμό από τα χρήματα που είναι απαραίτητο να διατεθούν για την απόκτηση τους. Αν και, όπως είδαμε από τα αποτελέσματα της μελέτης, η πλατφόρμα ρομποτικής Lego Mindstorms έδωσε τα πιο ελπιδοφόρα μηνύματα, βασικό μειονέκτημα που δεν της επιτρέπει να χρησιμοποιηθεί ευρέως στο εκπαιδευτικό σύστημα της χώρας είναι το υψηλό κόστος αγοράς τους εξοπλισμού (350 € για το βασικό πακέτο ρομποτικής). Από την άλλη μεριά η τεχνολογία φορετού υπολογισμού LilyPad σε σχέση με την τεχνολογία Lego Mindstorms εμφανίζει πολύ χαμηλότερο κόστος αγοράς εξοπλισμού (50 € για τη βασική πλατφόρμα ανάπτυξης του LilyPad). Το συγκεκριμένο πλεονέκτημα καθώς και η δημιουργία εργαλείων ανάπτυξης εφαρμογών φιλικών προς τους μαθητές, όπως το Modkit, δημιουργούν τις κατάλληλες προϋπόθεσης για την αξιοποίηση της σε μεγαλύτερη κλίμακα στο χώρο της εκπαίδευσης.

## Βιβλιογραφία

- [1] A. Nagchaudhuri, G. Singh, M. Kaur, and S. George, "LEGO robotics products boost student creativity in precollege programs at UMES," *Proc. 32nd Annual ASEE/IEEE Frontiers in Education Conference (FIE'02)* 2002, pp. S4D- 1-S4D-6.
- [2] AAUW, *Tech-Savvy: Educating Girls in the New Computer Age*. American Association of University Women Educational Foundation, Washington, DC, USA, 2000.
- [3] Alice <http://www.alice.org>
- [4] Amici  
<http://dimeb.informatik.uni-bremen.de/eduwear/category/development-software/>
- [5] Amon Millner and Edward Baafi. 2011. Modkit: blending and extending approachable platforms for creating computer programs and interactive objects. In *Proceedings of the 10th International Conference on Interaction Design and Children* (IDC '11). ACM, New York, NY, USA, 250-253.
- [6] App Inventor <http://appinventor.mit.edu/explore/>
- [7] Ardublock <http://blog.ardublock.com/>
- [8] Arduino <http://www.arduino.cc/>
- [9] Arduino IDE <http://arduino.cc/en/main/software>
- [10] Beisser, S. R. (Jan 2006). *An Examination of Gender Differences in Elementary Constructionist Classrooms Using Lego/Logo Instruction*. Computers in the Schools, 22, 7- 19.
- [11] Bers, M. (2008). Blocks to Robots: Learning with Technology in the Early Childhood Classroom. *Teachers College Press*, NY, NY.
- [12] Caitlin Kelleher and Randy Pausch. 2005. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Comput. Surv.* 37, 2 (June 2005), 83-137.
- [13] Caitlin Kelleher, Randy Pausch, and Sara Kiesler. 2007. Storytelling alice motivates middle school girls to learn computer programming. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '07). ACM, New York, NY, USA, 1455-1464.

- [14] Clements, D.H. The future of educational computing research: the case of computer programming. *Information Technology in Childhood Education*, (1999), 147-179.
- [15] Colleen M. Lewis. 2010. How programming environment shapes perception, learning and goals: logo vs. scratch. In *Proceedings of the 41st ACM technical symposium on Computer science education* (SIGCSE '10). ACM, New York, NY, USA, 346-350.
- [16] Dagilelis, V., Sartatzemi, M. & Kagani, K. (2005). Teaching (with) Robots in Secondary Schools: some new and not-so-new Pedagogical problems. In ICALT'05 - *Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies*.
- [17] Daniel, C., & Cliburn, D. C. (2006). Experiences with LEGO MINDSTORMS throughout the Undergraduate Computer Science Curriculum. *CA 36th ASEE/IEEE Frontiers in Education Conference*, San Diego, 1-6.
- [18] David Wolber. 2011. App inventor and real-world motivation. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (SIGCSE '11). ACM, New York, NY, USA, 601-606.
- [19] diSessa, A. (2000). Changing Minds: Computers, Learning, and Literacy. MIT Press:Cambridge, MA.
- [20] E. Wang and R. Wang, "Using Legos and RoboLab (LabVIEW) with elementary school children," *Proc. 31st Annual ASEE/IEEE Frontiers in Education Conference (FIE' 01)*, 2001, pp. T2E-T11.
- [21] EduWear [dimeb.informatik.uni-bremen.de/eduwear/](http://dimeb.informatik.uni-bremen.de/eduwear/)
- [22] Enchanting <http://enchanting.robotclub.ab.ca/tiki-index.php>
- [23] Figueroa, E. and Woods, R. (2007). Employment Outlook: 2006-2016. Industry output and employment projections to 2016. *Monthly Labor Review*, 130(11):53-85
- [24] Fraunheim, E. Students Saying No to Computer Science. C | Net News.com, August 11, 2004, <http://news.cnet.com/2100-1022-5306096.html>
- [25] Froebel, F. 1826. *On the Education of Man* (Die Menschenerziehung), Keilhau/Leipzig: Wienbrach.
- [26] H. Abelson. App Inventor for Android. *Google Research Blog*, <http://googleresearch.blogspot.com/2009/07/app-inventor-for-android.html>, July 31, 2009.

- [27] H. H. Lund and L. Pagliarini, "RoboCup Jr. with LEGO MINDSTORMS," *Proc. IEEE International Conference on Robotics and Automation (ICRA'00)*, 2000, pp. 813-819 vol.1.
- [28] Haugland, S.W. The Effect of Computer Software on Preschool Children's Developmental Gains. *Journal of Computing in Childhood Education*, 3(1), (1992), 15-30.
- [29] Hideyuki Suzuki and Hiroshi Kato. 1995. Interaction-level support for collaborative learning: AlgoBlock—an open programming language. In *The first international conference on Computer support for collaborative learning (CSCL '95)*, John L. Schnase and Edward L. Cunnias (Eds.). L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 349-355.
- [30] Hiroshi Ishii and Brygg Ullmer. 1997. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems (CHI '97)*. ACM, New York, NY, USA, 234-241.
- [31] Hot Pepper Quiz Creator <http://users.sch.gr/kodulis/quiz/>
- [32] <http://professionals.collegeboard.com/data-reports-research/ap>
- [33] [http://www.pi-schools.gr/content/index.php?lesson\\_id=1&ep=65&c\\_id=267](http://www.pi-schools.gr/content/index.php?lesson_id=1&ep=65&c_id=267)
- [34] [http://www.ted.com/talks/mitch\\_resnick\\_let\\_s\\_teach\\_kids\\_to\\_code](http://www.ted.com/talks/mitch_resnick_let_s_teach_kids_to_code)
- [35] IBM SPSS <http://www-01.ibm.com/software/analytics/spss/>
- [36] Karantarou, A., Tahos, N., Aliisis, D. (2005). Introduction to the basic principles and structures of programming using LEGO Mindstorms robot instructions. *Proceedings of the 3rd Pan-Hellenic conference "Informatics didactics"*, University of Peloponnisos, Korinthos.
- [37] Lau Wing Yiu. 2011. Exploring children's usage on tangible computational construction platforms : hands-on learning through functionality, crafts and stories. *The Hong Kong Polytechnic University*.
- [38] Leah Buechley, Mike Eisenberg, Jaime Catchen, and Ali Crockett. 2008. The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. ACM, New York, NY, USA, 423-432.
- [39] Lego Mindstorms  
<http://www.lego.com/en-us/mindstorms/?domainredir=mindstorms.lego.com>

- [40] Lego WeDo <https://education.lego.com/en-us/lesi/elementary/wedo>
- [41] LOGO Foundation  
<http://el.media.mit.edu/logo-foundation/resources/index.html>
- [42] Mannila, L., Peltomaki, M., & Salakoski, T. (2006). What About a Simple Language? Analyzing the Difficulties in Learning to Program. *Computer Science Education*, 16(3), 211-227.
- [43] Michael S. Horn and Robert J. K. Jacob. 2007. Tangible programming in the classroom with tern. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems* (CHI EA '07). ACM, New York, NY, USA, 1965-1970.
- [44] Microsoft Kinect <http://www.microsoft.com/en-us/kinectforwindows/>
- [45] MicroWords Logo <http://www.microworlds.com/>
- [46] Mitchel Resnick, John Maloney, Andrés Monroy-Hernández, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, and Yasmin Kafai. 2009. Scratch: programming for all. *Commun. ACM* 52, 11 (November 2009), 60-67.
- [47] Modkit <http://www.modkit.com/>
- [48] Montessori Materials. <http://www.montessoriedutoys.com/>
- [49] Montessori, M 1912. The Montessori Method. New York Frederick Stokes Co.
- [50] Myles McNally, Michael Goldweber, Barry Fagin, and Frank Klassner. 2006. Do lego mindstorms robots have a future in CS education?. In *Proceedings of the 37th SIGCSE technical symposium on Computer science education* (SIGCSE '06). ACM, New York, NY, USA, 61-62.
- [51] NXT-G <http://www.legoengineering.com/program/nxt-g/>
- [52] O. Gerovich, R. P. Goldberg, and I. D. Donn, "From science projects to the engineering bench," in *IEEE Robotics & Automation Magazine*. vol. 10, 2003, pp. 9-12
- [53] O. Gerovich, R. P. Goldberg, I. D. Bonn, A. Viswanathan, and R. H. Taylor, "Surgical robot competition - Introducing engineering in medicine to pre-college students," *Proc. ASEE Annual Conference*, Montreal, Que., Canada, 2002, pp. 11547-11559.
- [54] Oren Zuckerman and Mitchel Resnick. 2004. Hands-on modeling and simulation of systems. In *Proceedings of the 2004 conference on Interaction design and children: building a community* (IDC '04). ACM, New York, NY, USA, 157-158.

- [55] Oren Zuckerman, Tina Grotzer, and Kelly Leahy. 2006. Flow blocks as a conceptual bridge between understanding the structure and behavior of a complex causal system. In *Proceedings of the 7th international conference on Learning sciences* (ICLS '06). International Society of the Learning Sciences 880-886.
- [56] Piaget, J. 1954. *The Construction Reality in the Child*. Routledge and Kegan Paul Ltd. (Originally published in 1937)
- [57] PicoBoard <http://www.picocricket.com/picoboard.html>
- [58] R. Morelli, T. de Lanerolle, P. Lake, N. Limardo, E. Tamotsu, and C. Uche. Can android app inventor bring computational thinking to k-12? *Computer Science Department*, Trinity College, Hartford, CT, USA, HFOSS, 2010
- [59] S. H. Kim and J. W. Jeon, "Educating C Language using LEGO mindstorms Robotic Invention System 2.0," *Proc. IEEE International Conference on Robotics and Automation (ICRA'06)*, 2006, pp. 715-720.
- [60] Scratch <http://scratch.mit.edu/>
- [61] StarLogo TNG <http://education.mit.edu/projects/starlogo-tng>
- [62] Stephen Cooper, Wanda Dann, and John Harrison. 2010. A k-12 college partnership. In *Proceedings of the 41st ACM technical symposium on Computer science education* (SIGCSE '10). ACM, New York, NY, USA, 320-324.
- [63] Timothy S. McNerney. 2004. From turtles to Tangible Programming Bricks: explorations in physical language design. *Personal Ubiquitous Comput.* 8, 5 (September 2004), 326-337.
- [64] Turtle Art <http://turtleart.org/>
- [65] Vegso, J. 2005. Interest in CS as a Major Drops Among Incoming Freshmen. *Computing Research News*, Vol 17/ No. 3.
- [66] Vegso, J. Drop in CS Bachelor's Degree Production. *Computing Research News*, 18(2), (2006).
- [67] Vygotsky, L.S. 1978. *Mind in society: The development of higher psychological processes*. Cambridge, MA: Harvard University Press.
- [68] Winnie W.Y. Lau, Grace Ngai, Stephen C.F. Chan, and Joey C.Y. Cheung. 2009. Learning programming through fashion and design: a pilot summer course in wearable computing for middle school students. *SIGCSE Bull.* 41, 1 (March 2009), 504-508.
- [69] Κριτήρια Αξιολόγησης: <http://users.sch.gr/merkourisa/quiz>

## Παράρτημα Ι Δραστηριότητες

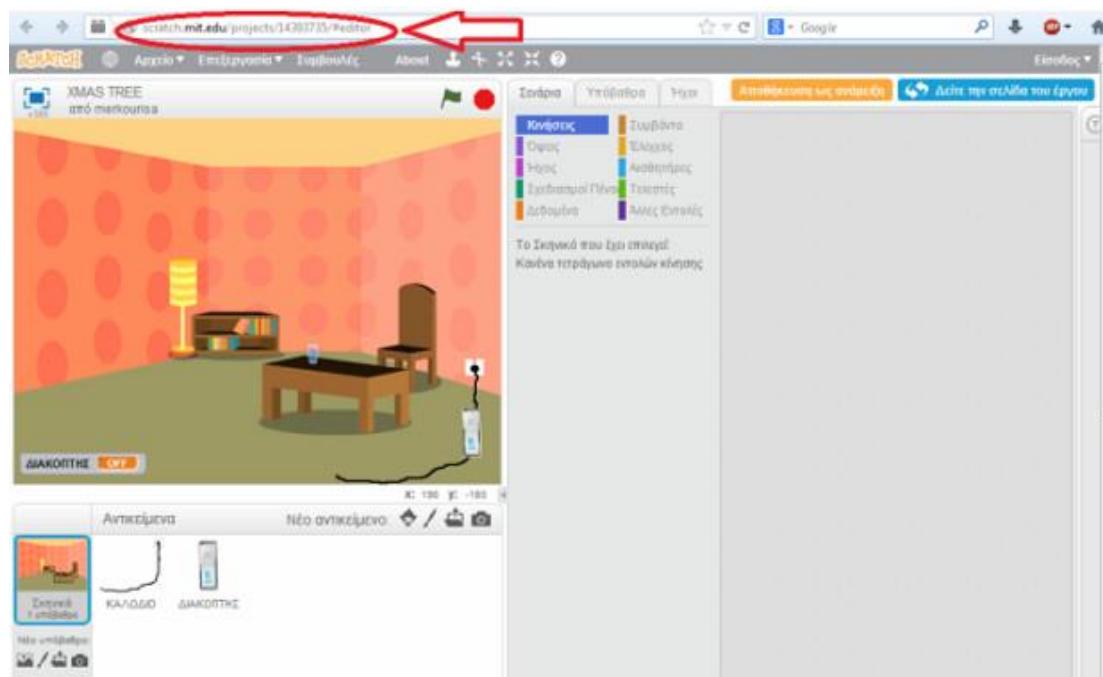


## ΤΙ ΧΡΕΙΑΖΕΣΑΙ

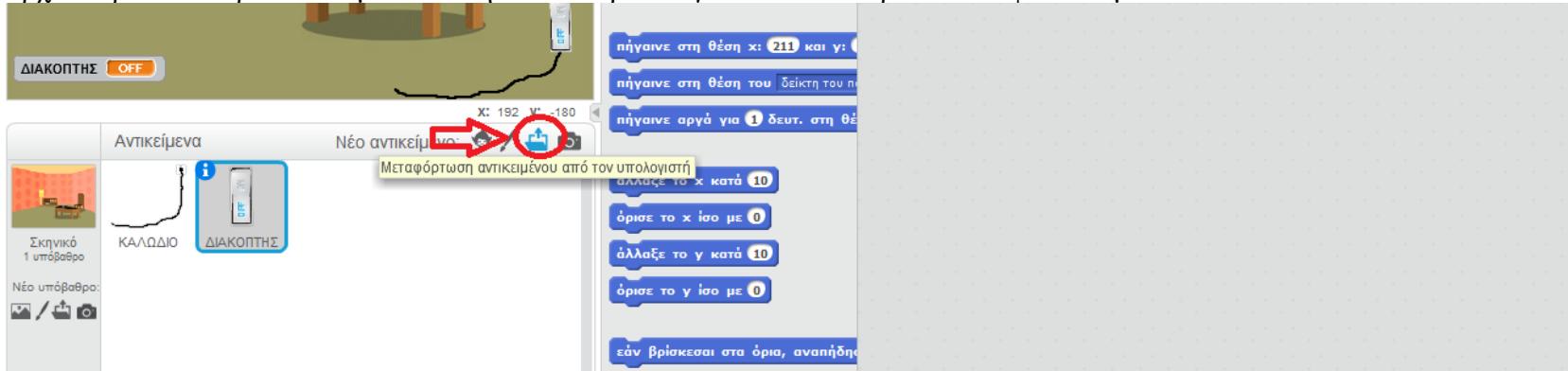
- Το πρόγραμμα XMAS TREE.
- Μια μορφή για το χριστουγεννιάτικο δέντρο.
- Το τραγούδι Jingle Bells.
- Να δημιουργήσεις τα φωτάκια που θα αναβοσβήνουν πάνω στο δέντρο.

## ΕΤΟΙΜΑΣΟΥ

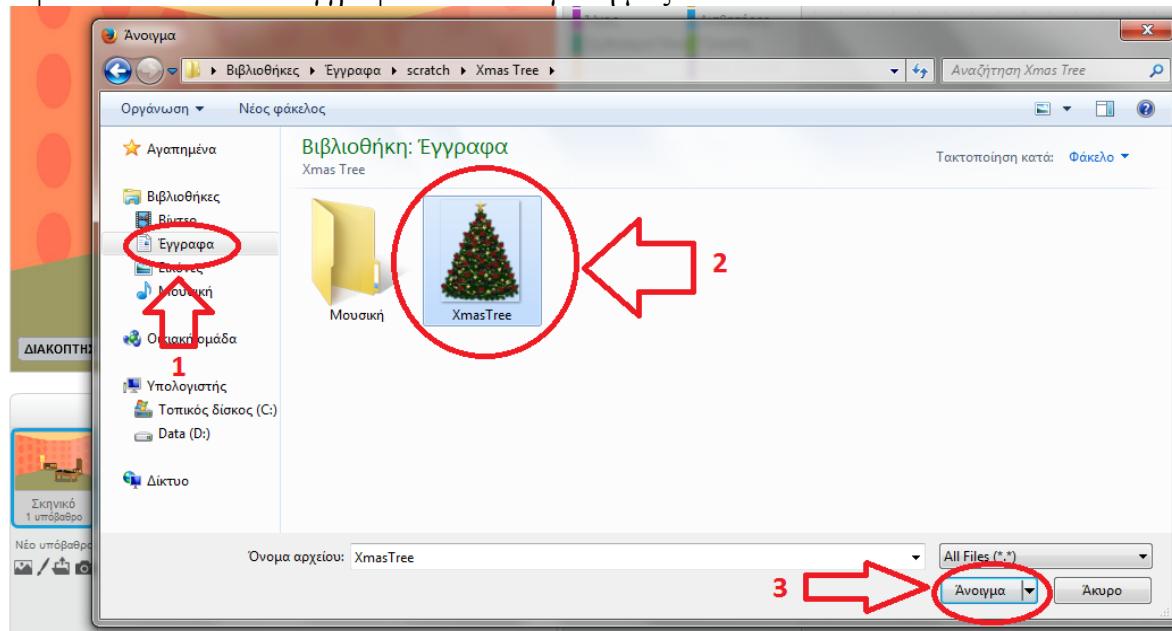
1. Πήγαινε στη σελίδα <http://scratch.mit.edu/projects/14393735/#editor> (**CTRL + κλικ** πάνω στο σύνδεσμο) για να ανοίξει το πρόγραμμα με το οποίο θα ασχοληθείς:



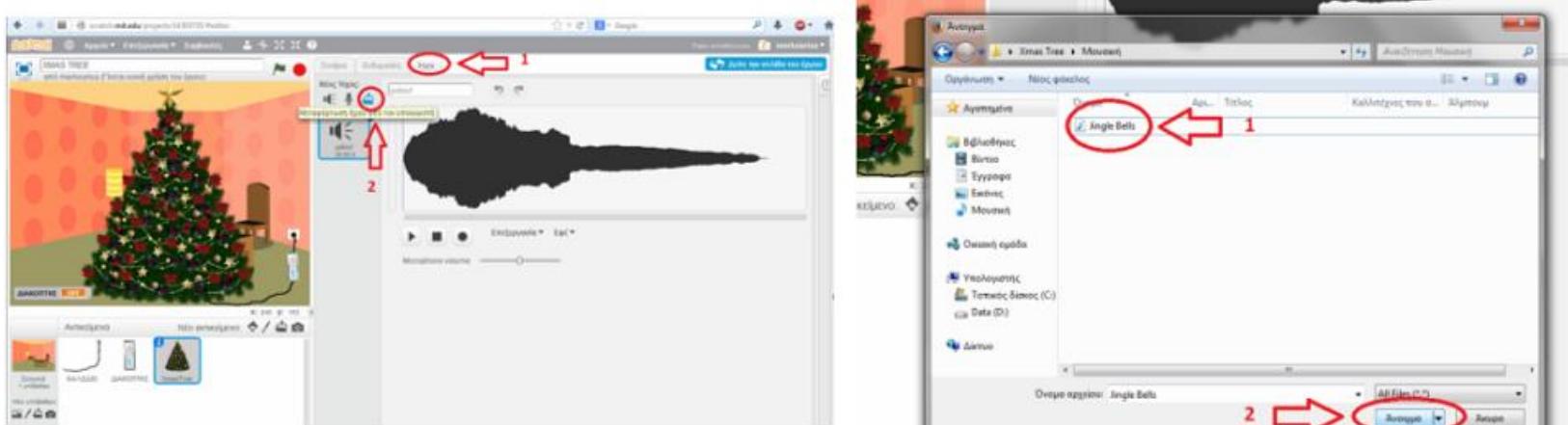
2. Αρχικά πρέπει να προσθέσουμε στο Σκηνικό το Χριστουγεννιάτικο δέντρο που θα φωτίσουμε.



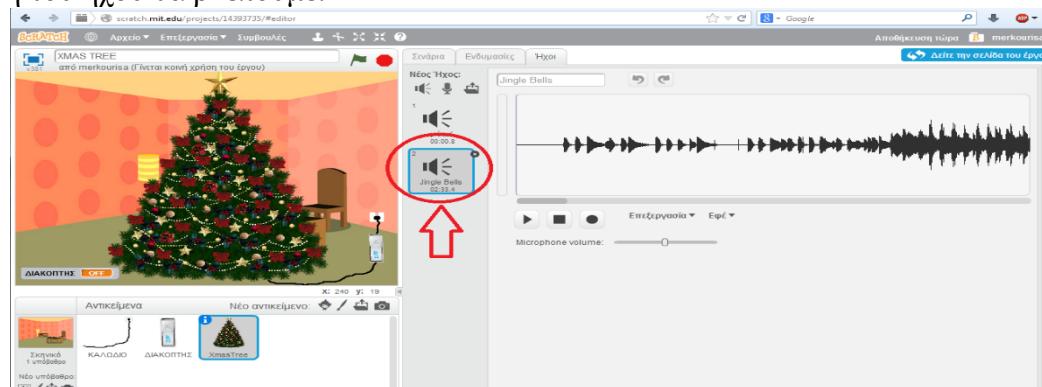
- Η μορφή βρίσκεται στο φάκελο scratch στα Έγγραφα του υπολογιστή μας



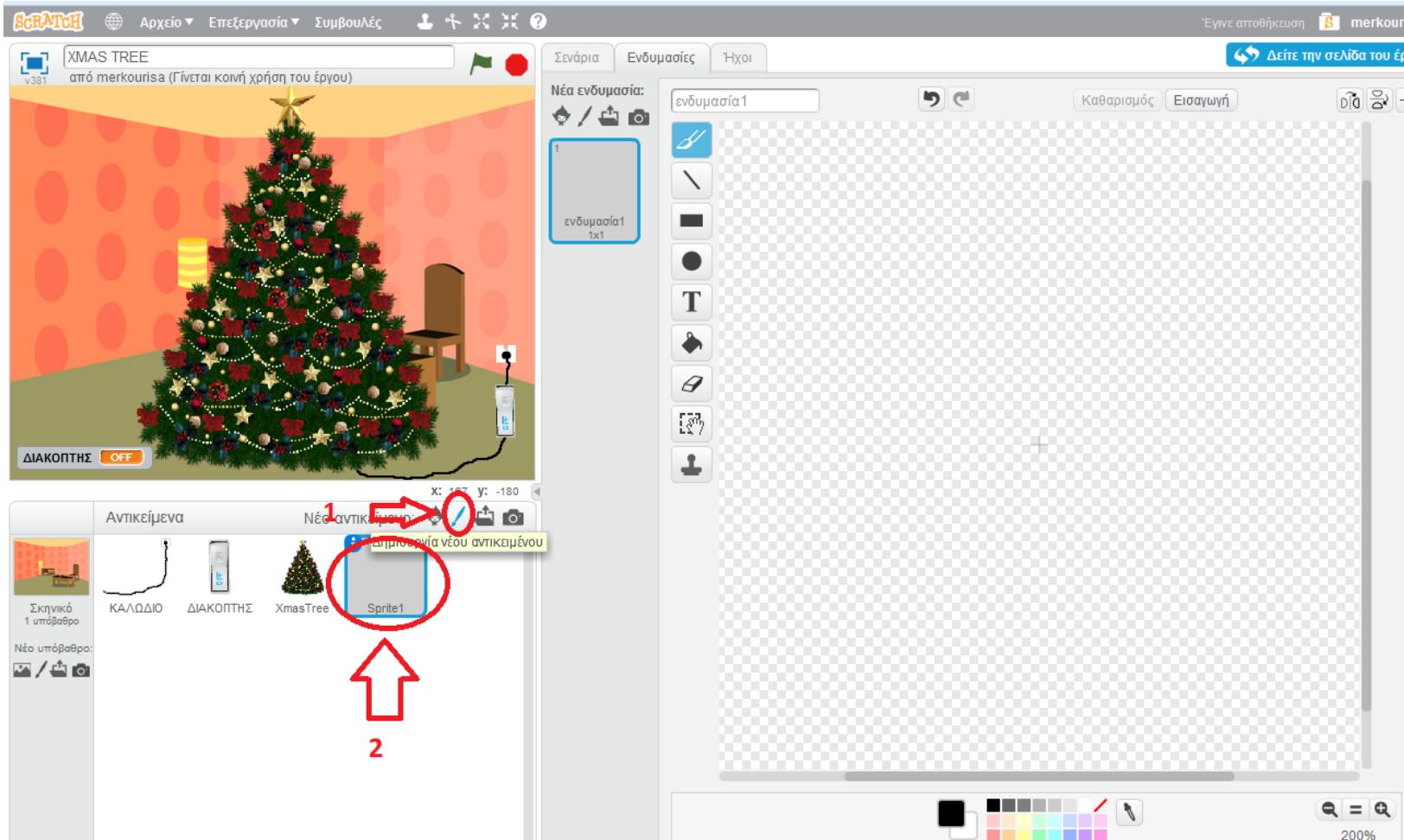
- Αφού προσθέσαμε το δέντρο πρέπει να το μετακινήσουμε στην σωστή θέση ώστε να φαίνεται ότι συνδέεται με το καλώδιο που θα φωτίζει τα φωτάκια.
3. Στην συνέχεια θα πρέπει να προσθέσουμε τον ήχο με το τραγούδι Jingle Bells. Επιλέγουμε το τραγούδι από τον φάκελο scratch\Μουσική που βρίσκεται στα Έγγραφα μου.



- Μετά την επιτυχή προσθήκη του ήχου θα βλέπουμε:



4. Τώρα πρέπει να δημιουργήσουμε τα φωτάκια που θα στολίσουν το δέντρο. Ας ξεκινήσουμε με τα κόκκινα φωτάκια.
- Επιλέγουμε Δημιουργία νέου Αντικειμένου.



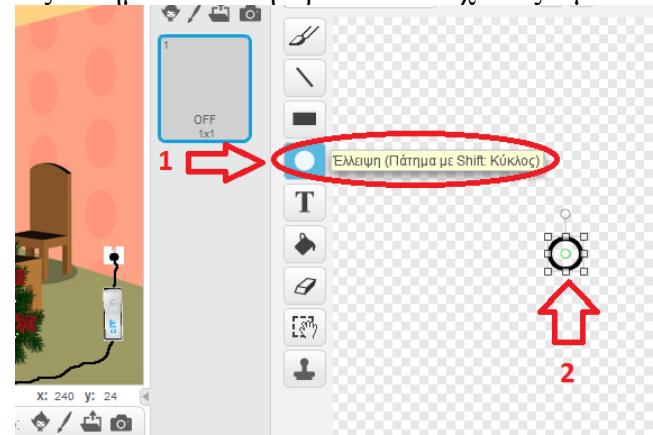
- Μετονομάζουμε την μορφή Sprite1 σε RED



και την ενδυμασία1 σε OFF.

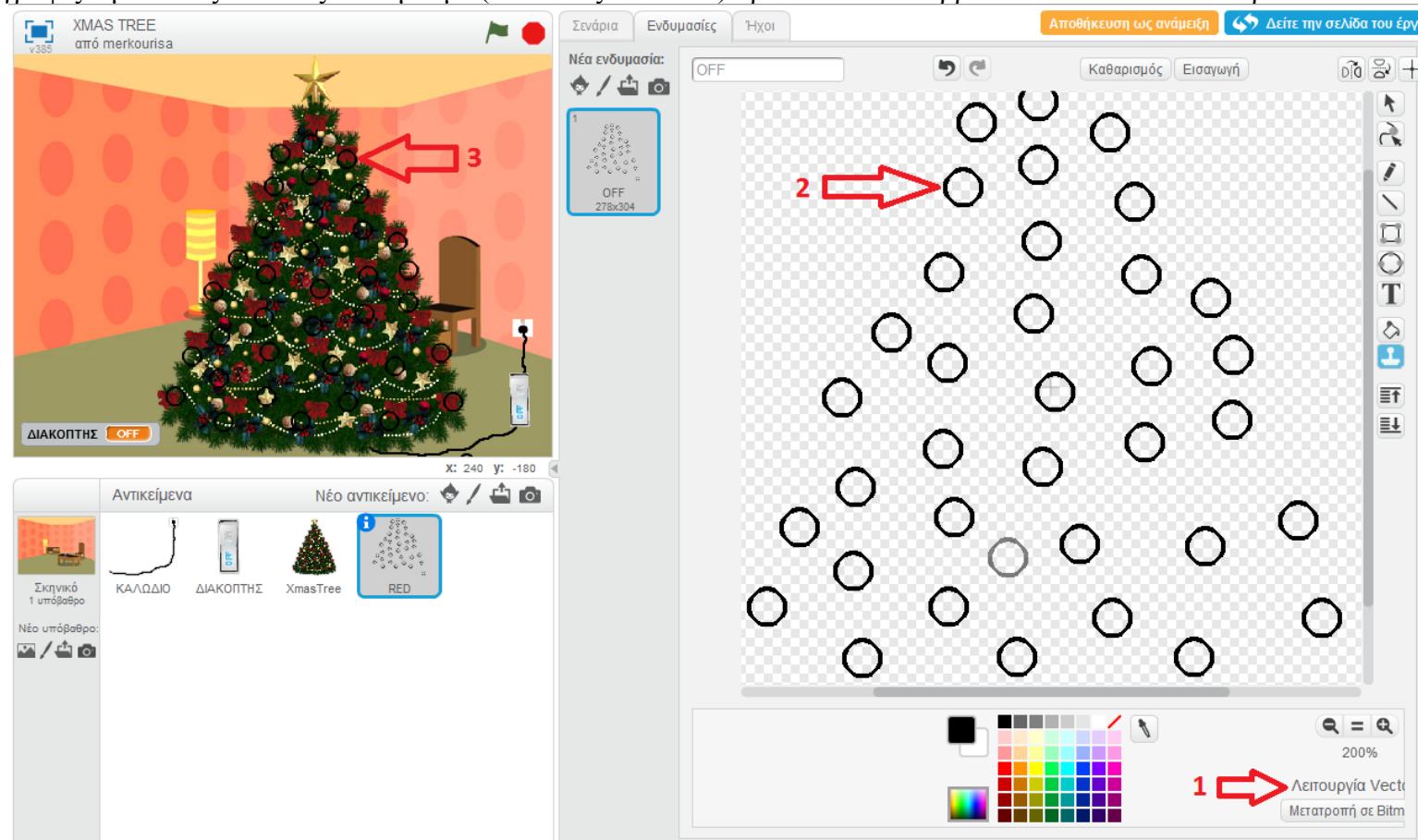


- Τώρα πρέπει να ζωγραφίσουμε το περίγραμμα από τα φωτάκια.
- Επιλέγουμε το εργαλείο Έλλεψη. Κρατώντας πατημένο το πλήκτρο SHIFT σχεδιάζουμε ένα κύκλο για το πρώτο φωτάκι.

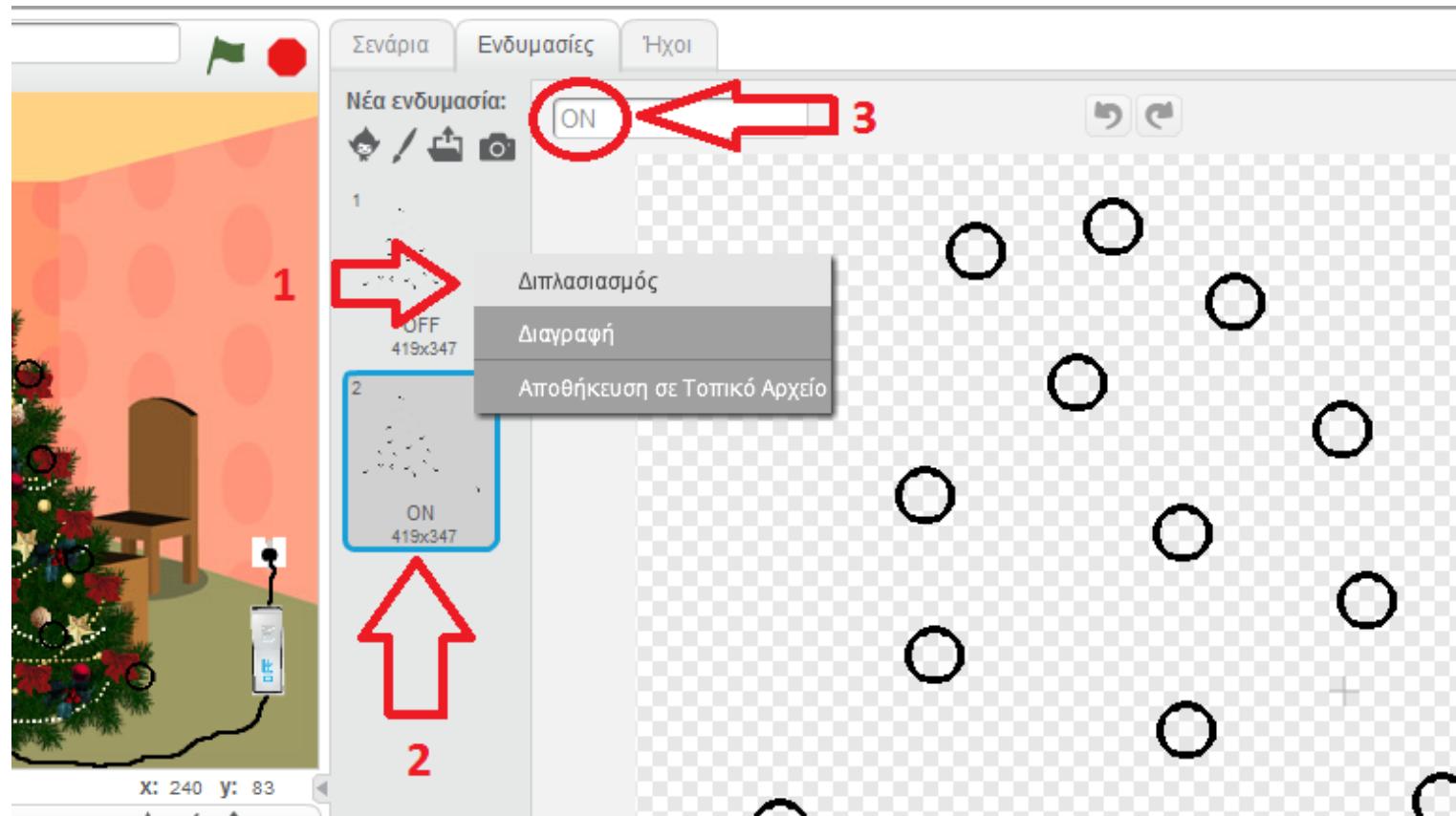


- Τα υπόλοιπα φωτάκια θα είναι αντίγραφο του πρώτου.

- Επιλέγουμε **Μετατροπή σε Vector** ώστε να μπούμε σε **Λειτουργία Vector**. Παρατηρούμε ότι ο κύκλος που σχεδιάσαμε νωρίτερα έχει σχηματιστεί στο Σκηνικό.
- Επιλέγουμε τον σχεδιασμένο κύκλο και πατάμε ταυτόχρονα στο πληκτρολόγιο **CTRL-C** για να κάνουμε αντιγραφή το αρχικό σχήμα.
- Στην συνέχεια **CTRL-V** για να επικολλήσουμε ένα νέο κύκλο.
- Ζωγραφίζουμε όσους κύκλους επιθυμούμε (πατώντας **CTRL-V**) αρκεί να όλοι να βρίσκονται πάνω στο δέντρο.

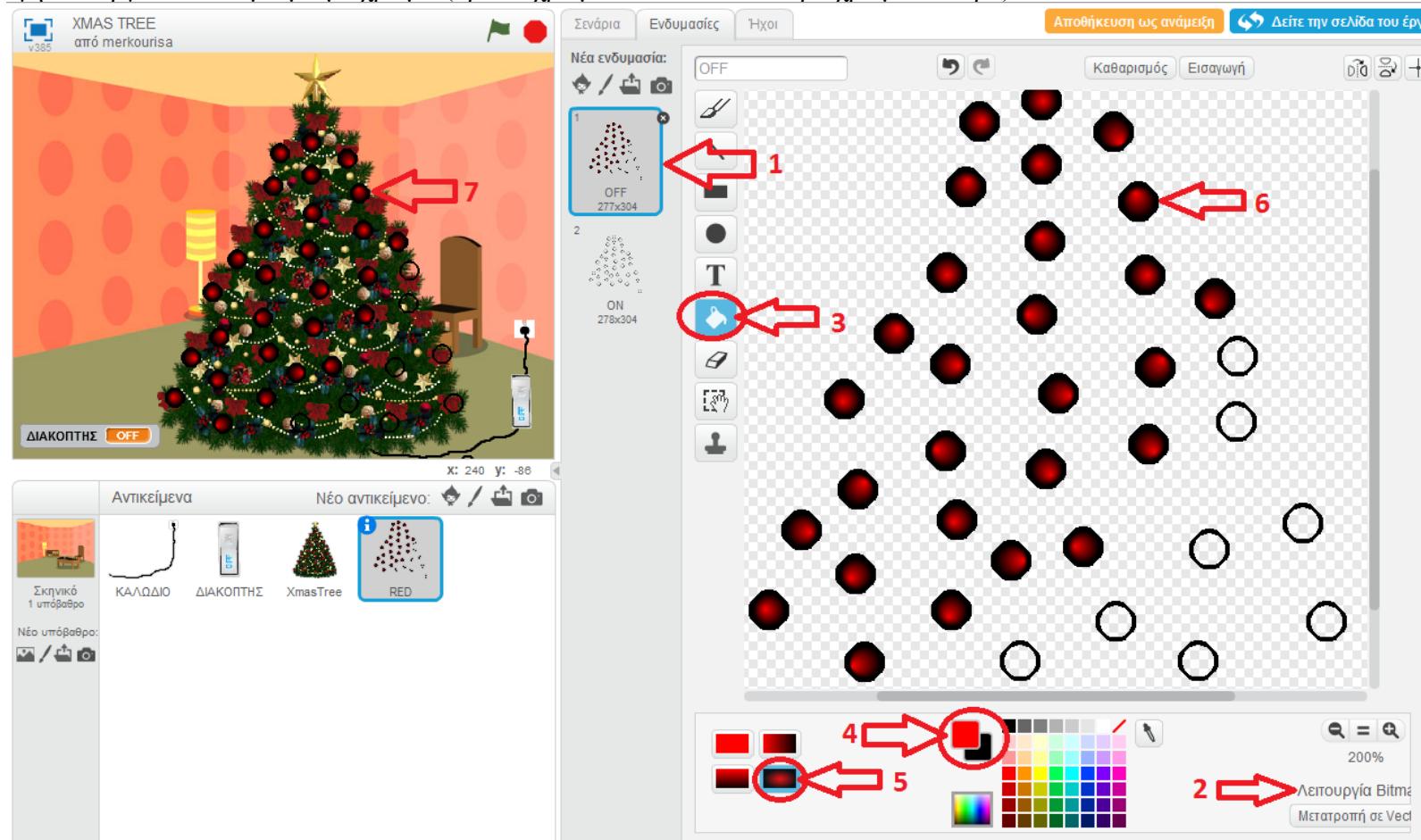


- Αφού ζωγραφίσουμε όλους του κύκλους επιλέγουμε διπλασιασμός στην ενδυμασία OFF κάνοντας δεξί κλικ πάνω της.
- Μια νέα ενδυμασία OFF2 δημιουργείται ακριβώς ίδια με την OFF.
- Την μετονομάζουμε σε ON.



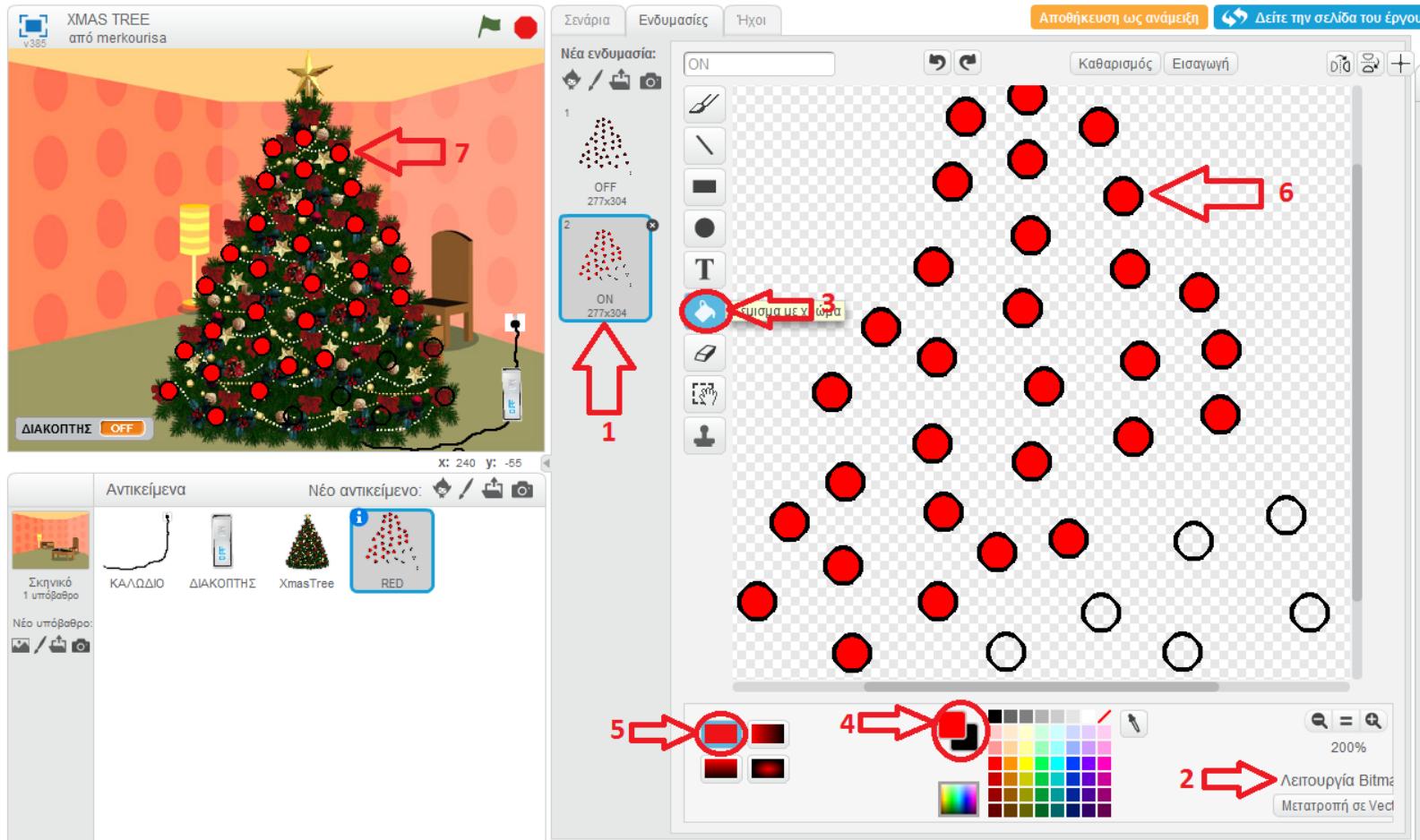
- Πάμε τώρα να γεμίσουμε με χρώμα τους κύκλους που θα αποτελούν τα φωτάκια τόσο στην ενδυμασία OFF όσο και στην ON.

- Επιλογή Ενδυμασίας OFF
- Μετατροπή σε Bitmap
- Επιλογή του εργαλείου Γέμισμα με χρώμα (πρώτο χρώμα Κόκκινο –δεύτερο χρώμα Μαύρο)



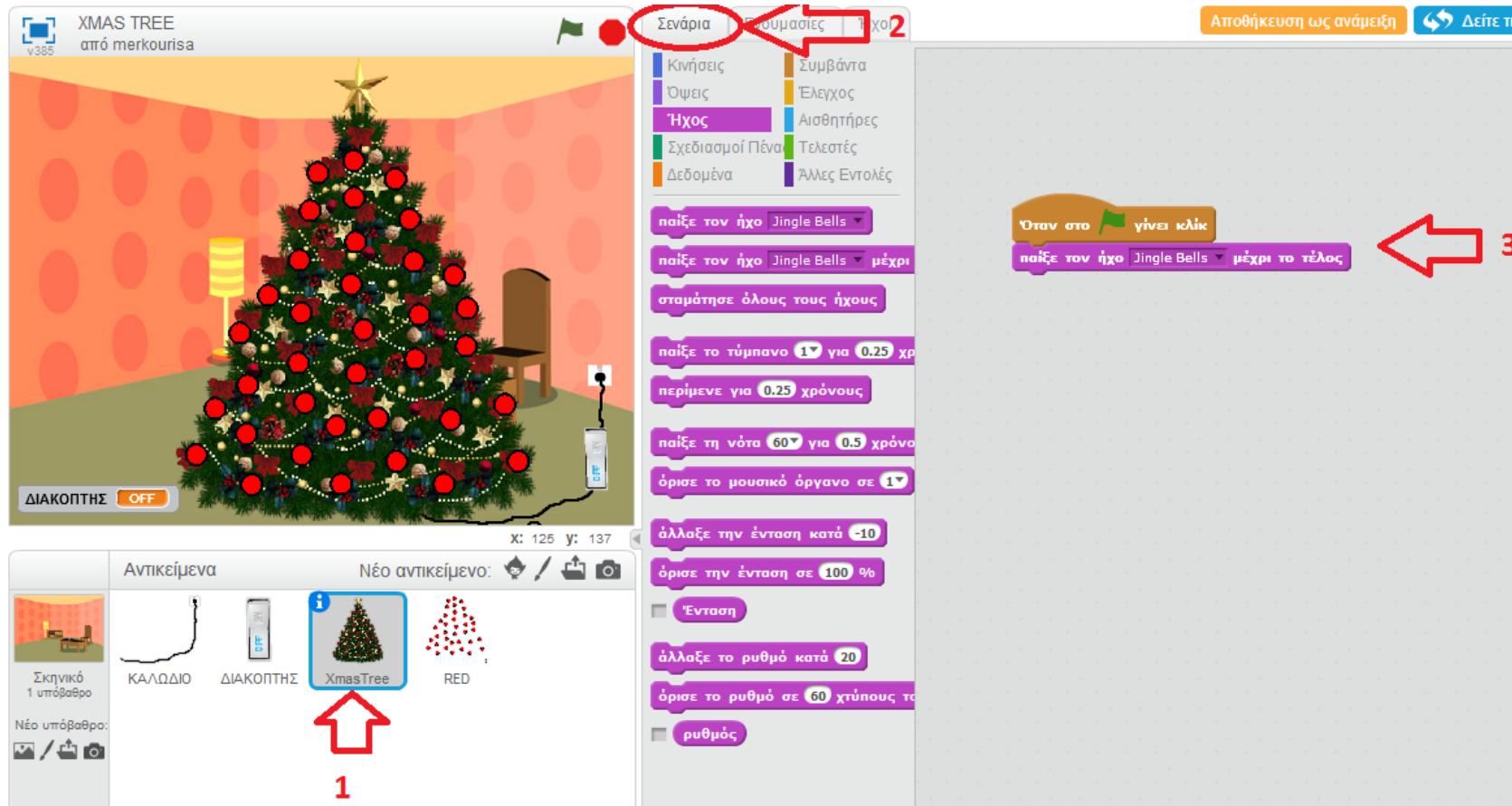
- Γέμισε όλους τους κύκλους με χρώμα

- Επιλογή Ενδυμασίας ON
- Μετατροπή σε Bitmap
- Επιλογή του εργαλείου Γέμισμα μόνο με Κόκκινο χρώμα.



- Γέμισε όλους τους κύκλους με Κόκκινο χρώμα

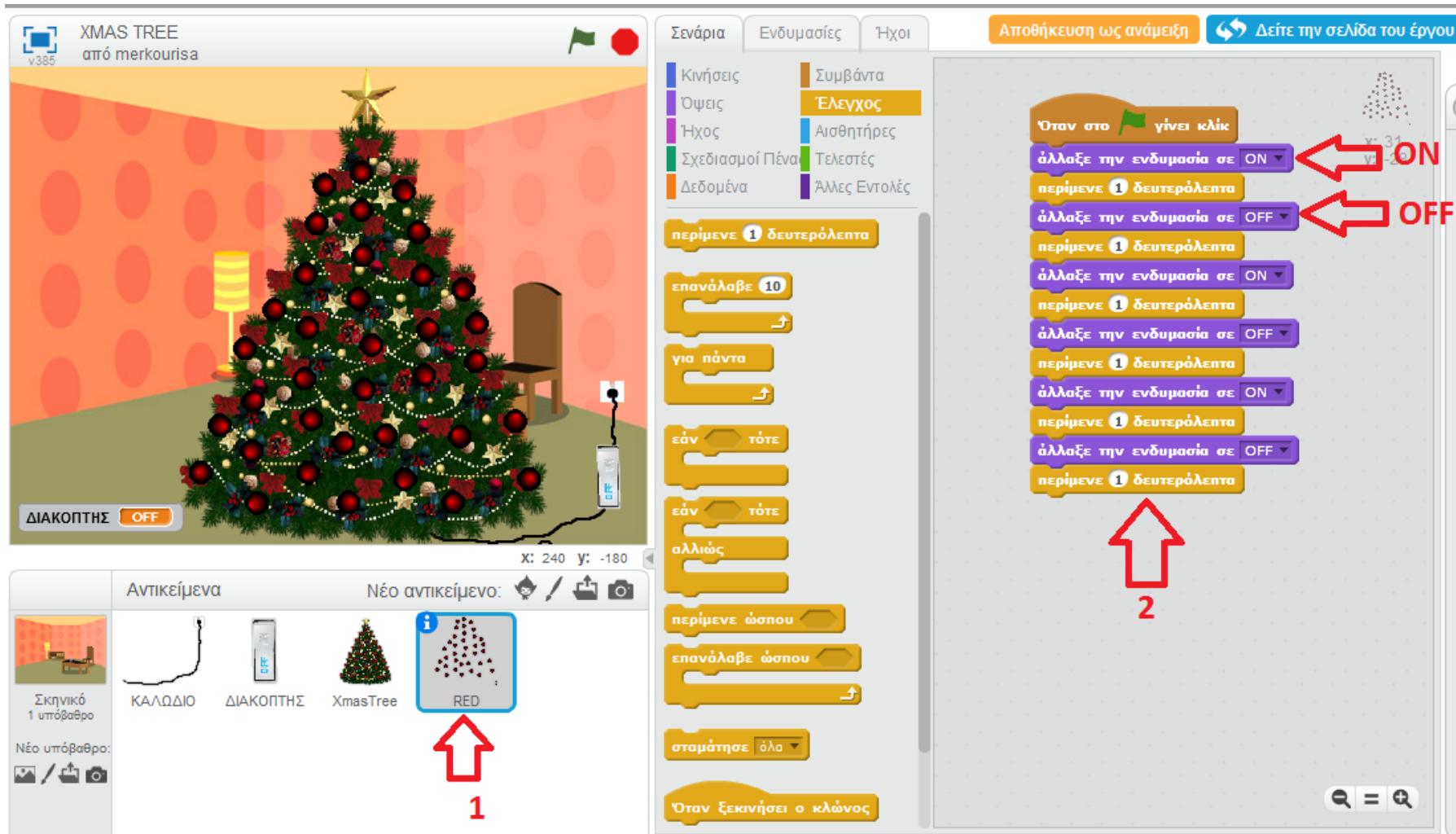
## **ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ**



## ТРЕЕЕ ТО ПРОГРАММА

Κάνε κλικ στη σημαία  ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω. Τι συμβαίνει;

## ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



## ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ

Κάνε κλικ στη σημαία  ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω.

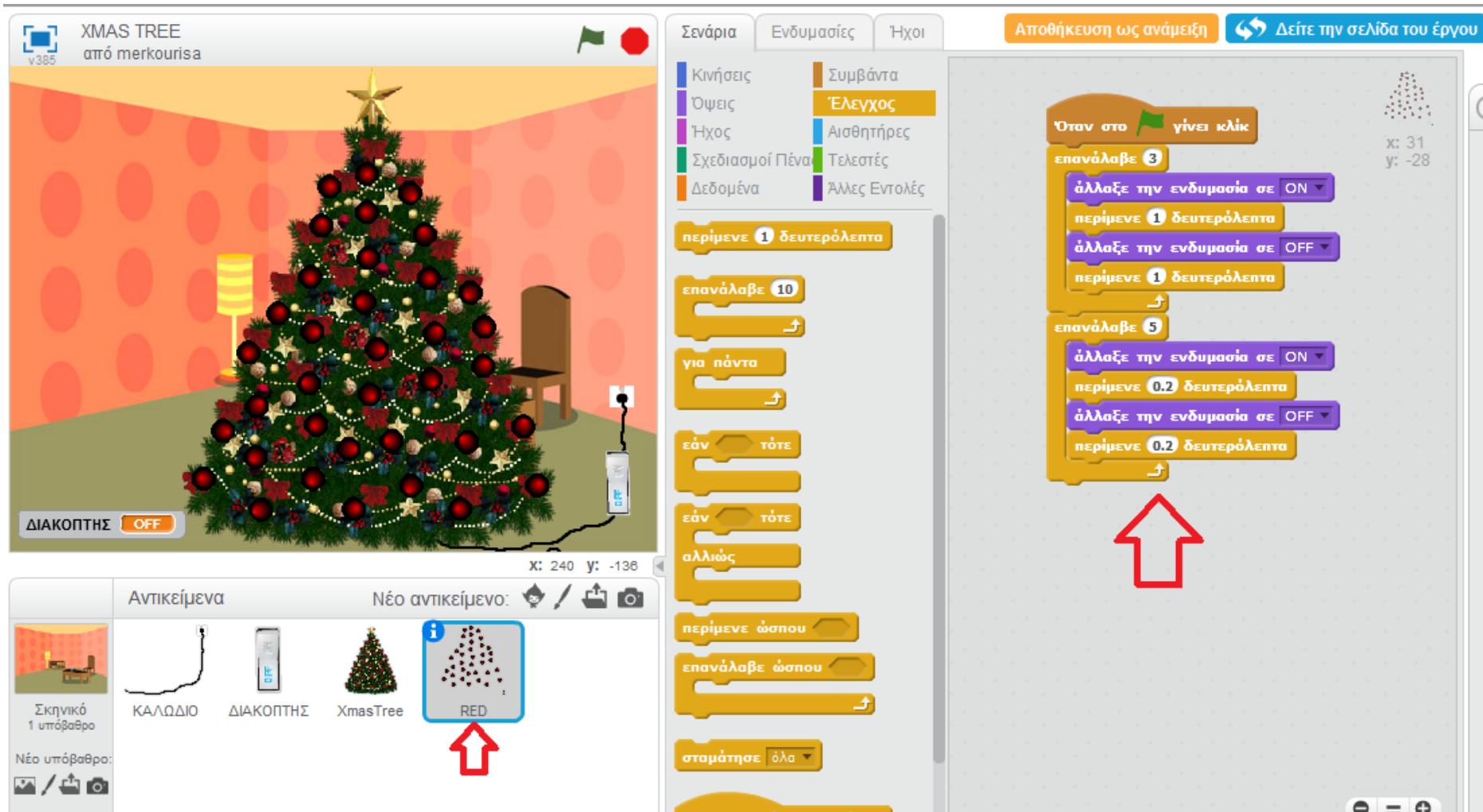
Τι συμβαίνει; Πόσες φορές αναβοσβήνουν τα λαμπάκια; Κάθε πότε σβήνουν και ανοίγουν τα λαμπάκια;

## ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ

Κάνε τα φωτάκια να αναβοσβήσουν δυο επιπλέον φορές.

Που θα μπορούσες να χρησιμοποιήσεις την εντολή ;

## ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



## ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ

Κάνε κλικ στη σημαία  ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω.

Πόσες φορές αναβοσβήνουν τα λαμπάκια αργά και πόσες φορές γρήγορα;

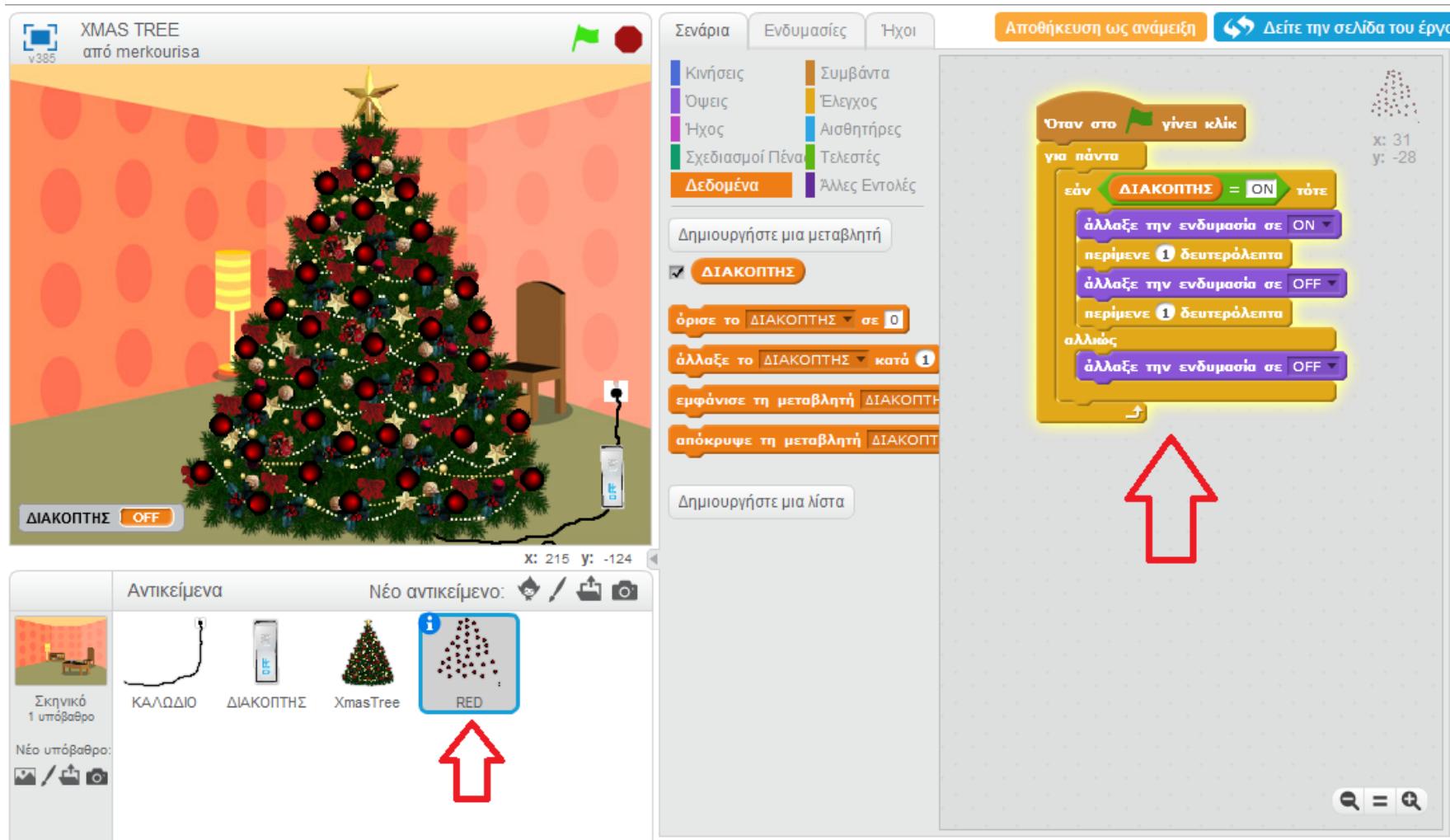
## ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ

Κάνε τα λαμπάκια να αναβοσβήσουν 6 φορές αργά (με 1 δευτερόλεπτο καθυστέρηση) και μετά 7 φορές γρήγορα (0.2 δευτερόλεπτα καθυστέρηση).

Κάνε τα λαμπάκια να αναβοσβήνουν **συνέχεια** με τον παραπάνω ρυθμό (6 αργά και 7 γρήγορα) χρησιμοποιώντας κατάλληλα την εντολή:



## ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



## ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ

Κάνε κλικ στη σημαία  ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω. Στην συνέχεια κάνε κλικ στον διακόπτη.

Τι συμβαίνει όταν ο διακόπτης είναι στο ON και τι συμβαίνει όταν ο διακόπτης είναι στο OFF;

## ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ

Αλλαξε το πρόγραμμα ώστε:

- όταν ο διακόπτης είναι στο ON τα φωτάκια να αναβοσβήνουν αργά (ανά 1 δευτερόλεπτο)
- ενώ όταν ο διακόπτης είναι στο OFF να αναβοσβήνουν γρήγορα (ανά 0.2 δευτερόλεπτα)

## KINOYMEMO ROBOT NXT



## ΤΙ ΧΡΕΙΑΖΕΣΑΙ

- Ένα μικροεπεξεργαστή NXT.



- Δυο κινητήρες NXT.



- Έναν αισθητήρα πίεσης (switch button).



- Δομικά στοιχεία NXT



και 3 καλώδια σύνδεσης.



## ΕΤΟΙΜΑΣΟΥ

5. Φτιάξε το ρομπότ.

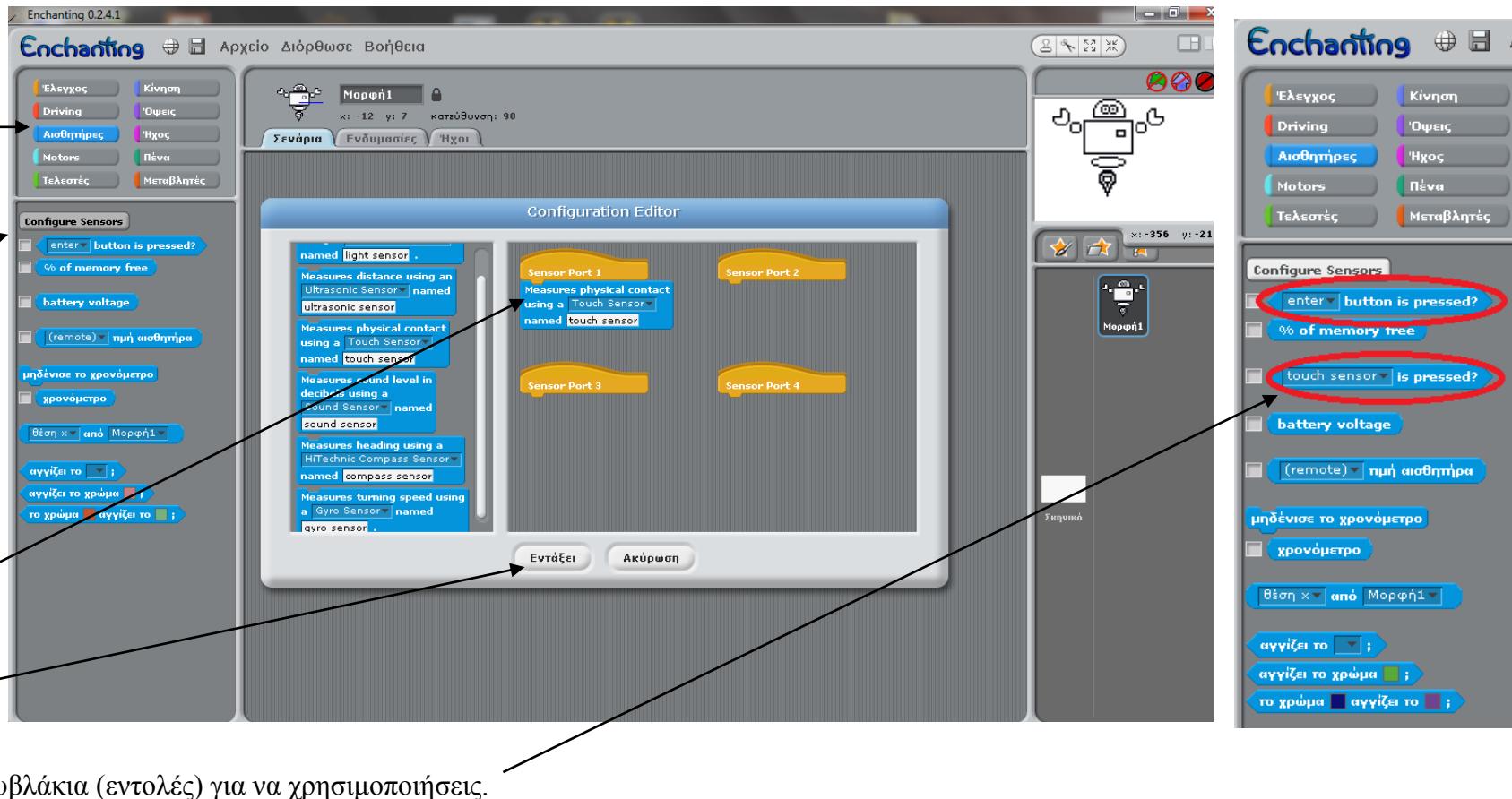
- Ένωσε με το καλώδιο τον κινητήρα που είναι συνδεδεμένος στην δεξιά ρόδα στην θύρα C του μικροεπεξεργαστή NXT.
- Ένωσε με το καλώδιο τον κινητήρα που είναι συνδεδεμένος στην αριστερή ρόδα στην θύρα B του μικροεπεξεργαστή NXT.
- Ένωσε τον αισθητήρα πίεσης στην θύρα 1 του μικροεπεξεργαστή NXT.

6. Σύνδεσε το ρομπότ με τον υπολογιστή μέσω του καλωδίου USB.



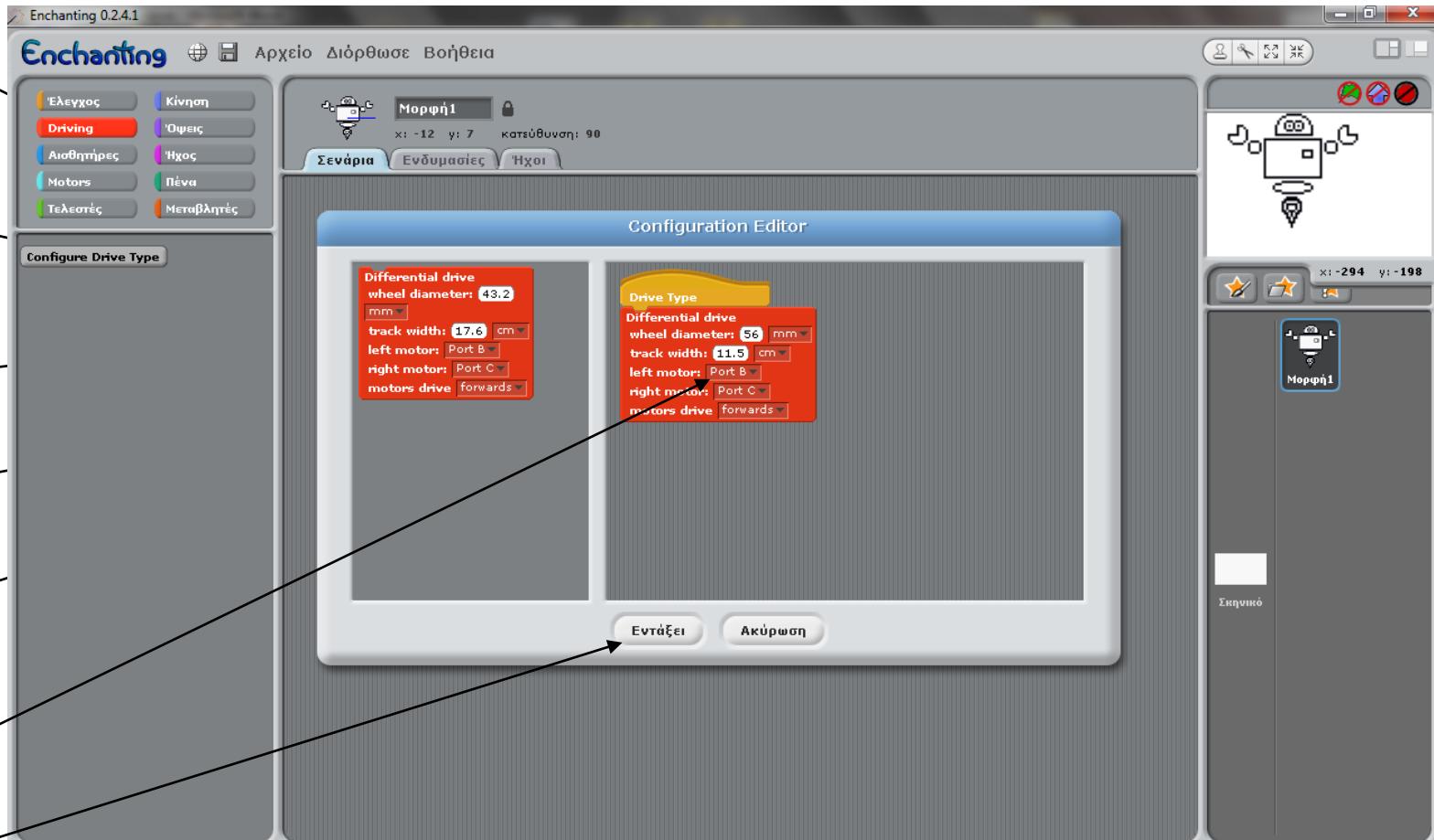
7. Κάνε διπλό κλικ στο εικονίδιο  στην επιφάνεια εργασία για να ανοίξεις το πρόγραμμα Enchanting.

8. Πως θα πούμε στο Enchanting ότι έχουμε συνδέσει στο NXT έναν αισθητήρα πίεσης στην θύρα 1.

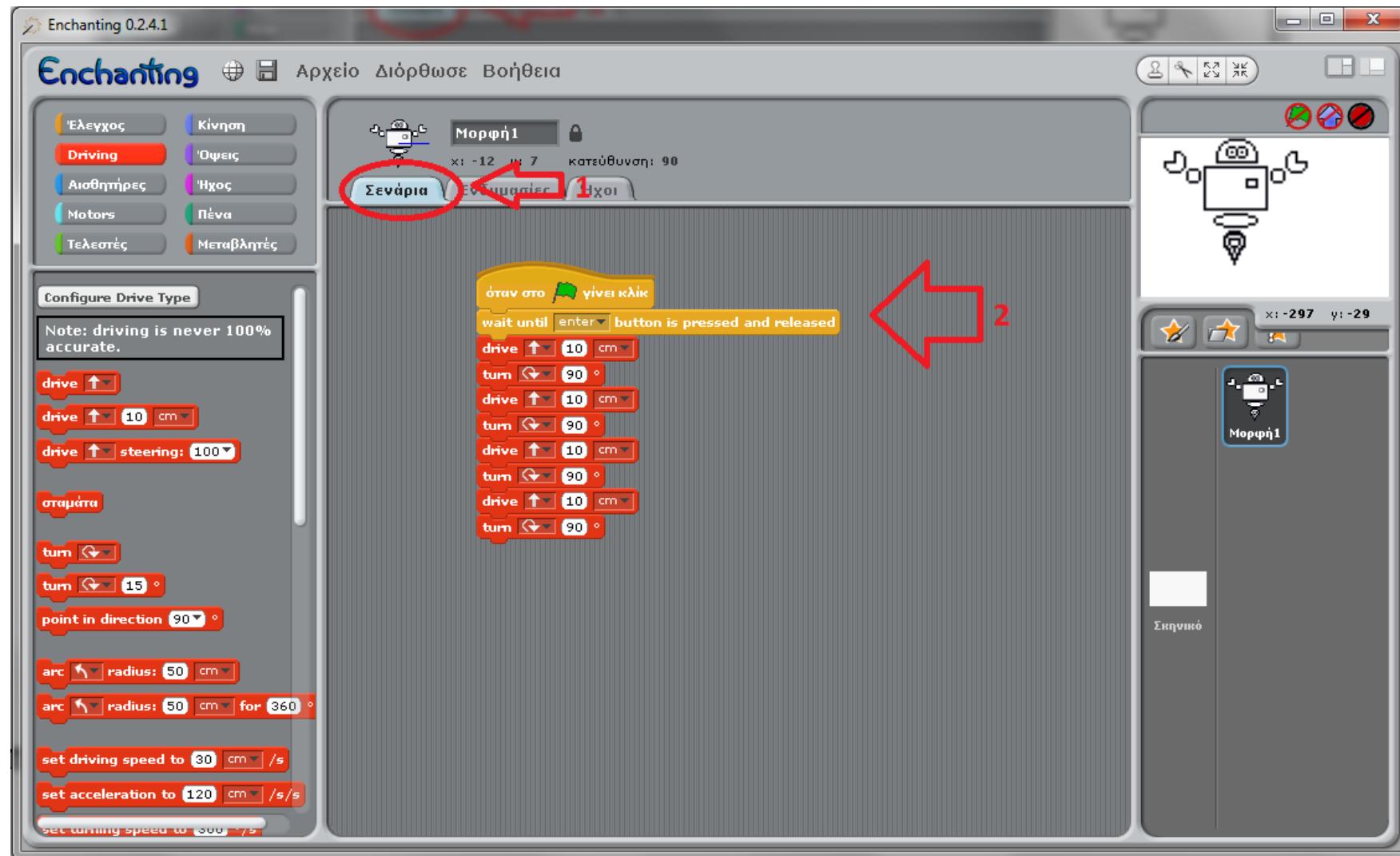


9. Πως θα πούμε στο Enchanting ότι έχουμε συνδέσει στο NXT δύο κινητήρες στις θύρες B και C.

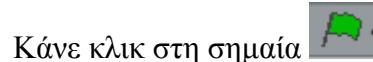
- Κάνε κλικ στο “Driving”.
- Κάνε κλικ στο κουμπί “Configure Drive Type”. Ένα νέο παράθυρο θα ανοίξει.
- Σύρε το κόκκινο τουβλάκι και άφησε το κάτω από το “Drive Type”.
- Όρισε την διάμετρο της ρόδας (wheel diameter) σε 56 mm.
- Όρισε το μήκος του άξονα (track width) σε 11.5 cm.
- Όρισε τον αριστερό κινητήρα στο “Port B” και τον δεξί στο “Port C”.
- Πάτα το “Εντάξει”.
- Τώρα υπάρχουν νέα τουβλάκια (εντολές) κίνησης για να χρησιμοποιήσεις.



## **ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ**



## ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



Κάνε κλικ στη σημαία ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω στο ρομπότ.

Στην συνέχεια πίεσε το κίτρινο κουμπί enter που βρίσκεται πάνω στο μικροεπεξεργαστή NXT.

Τι συμβαίνει; Πως κινείται το ρομπότ;



Το ρομπότ όντως κινείται σε διαδρομή που σχηματίζει τετράγωνο;

Είναι σωστά ρυθμισμένο; Σημείωσε ότι ποτέ δεν υπάρχει 100% ακρίβεια στην κίνηση.

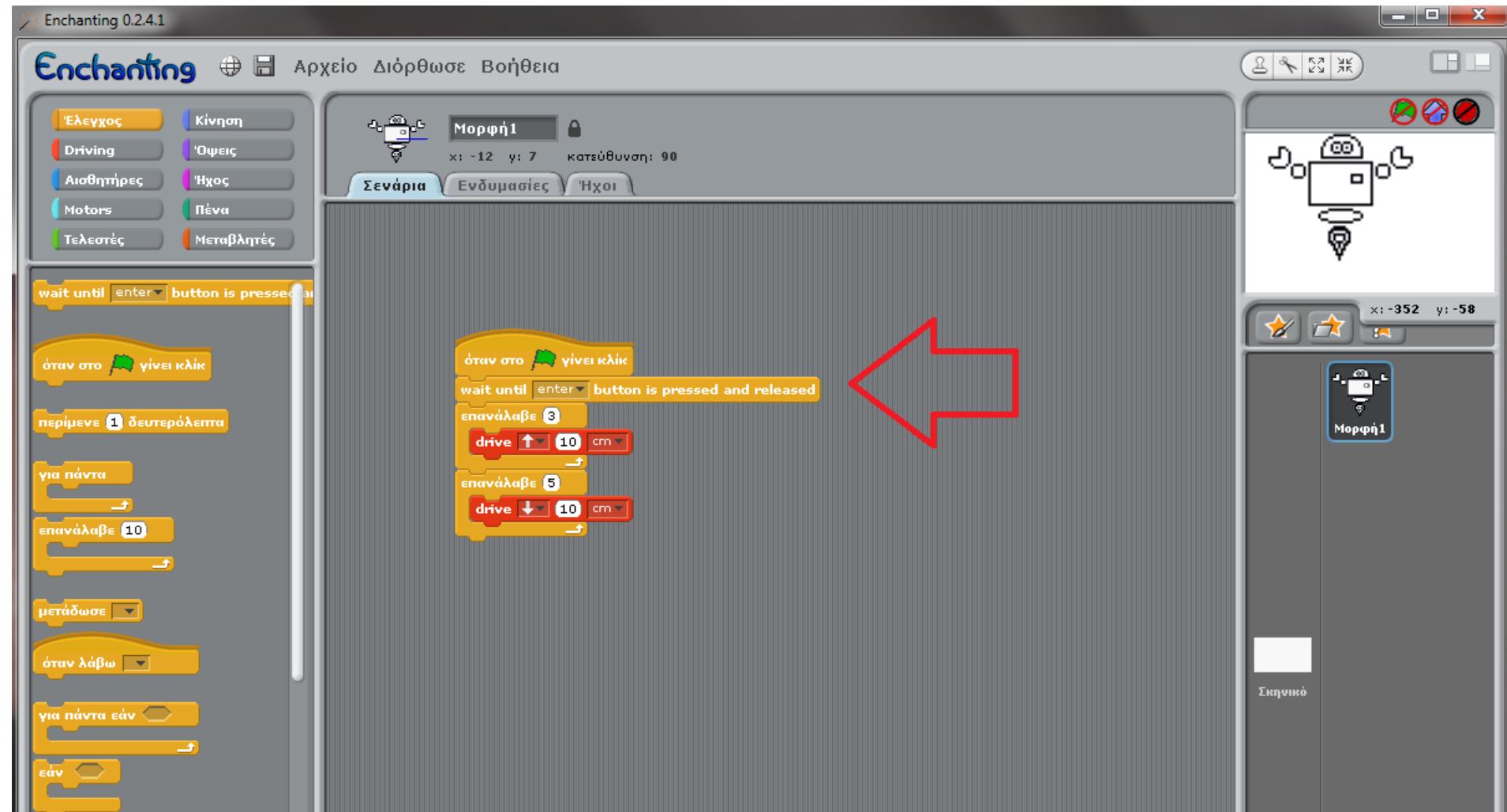
## ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ

Φτιάξε το πρόγραμμα ώστε το ρομπότ να κινηθεί:

- 10 cm μπροστά.
- να στρίψει 90° δεξιόστροφα (στην φορά κίνησης των δεικτών του ρολογιού )
- να κινηθεί 20 cm μπροστά.
- να στρίψει 90° αριστερόστροφα (αντίθετα με την φορά κίνησης των δεικτών του ρολογιού )
- να κινηθεί 10 cm μπροστά.

Που θα μπορούσες να χρησιμοποιήσεις την εντολή **πυχαία επιλογή από 10 μέχρι 20** ;

## ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



## ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ

Κάνε κλικ στη σημαία  ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω στο ρομπότ.

Στην συνέχεια πίεσε το κίτρινο κουμπί enter που βρίσκεται πάνω στο μικροεπεξεργαστή NXT.



Τι συμβαίνει; Πώς κινείται το ρομπότ;

Πόσα βήματα κάνει μπροστά και πόσα πίσω;

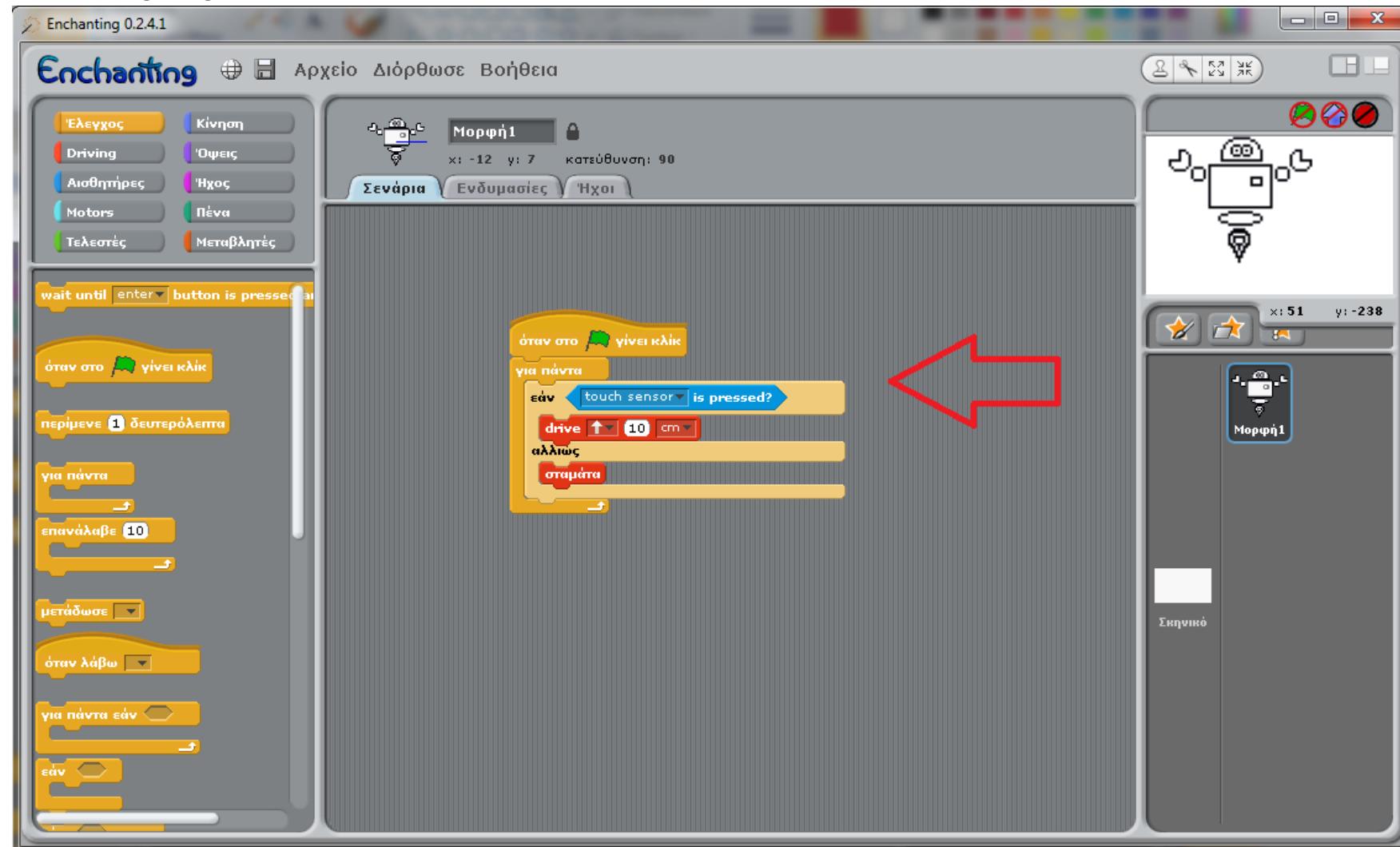
## ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ

Θα παρούσες να αναγκάσεις το ρομπότ αρχικά να κινηθεί 4 βήματα μπροστά και στην συνέχεια 6 βήματα προς τα πίσω. Το κάθε βήμα θα είναι 10 cm (εκατοστά).

Κάνε το ρομπότ να ακολουθεί **συνέχεια** την παραπάνω διαδρομή (4 βήματα μπροστά και 6 πίσω) χρησιμοποιώντας κατάλληλα την εντολή για πάντα:



## ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



## ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ

Κάνε κλικ στη σημαία  ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω στο ρομπότ.

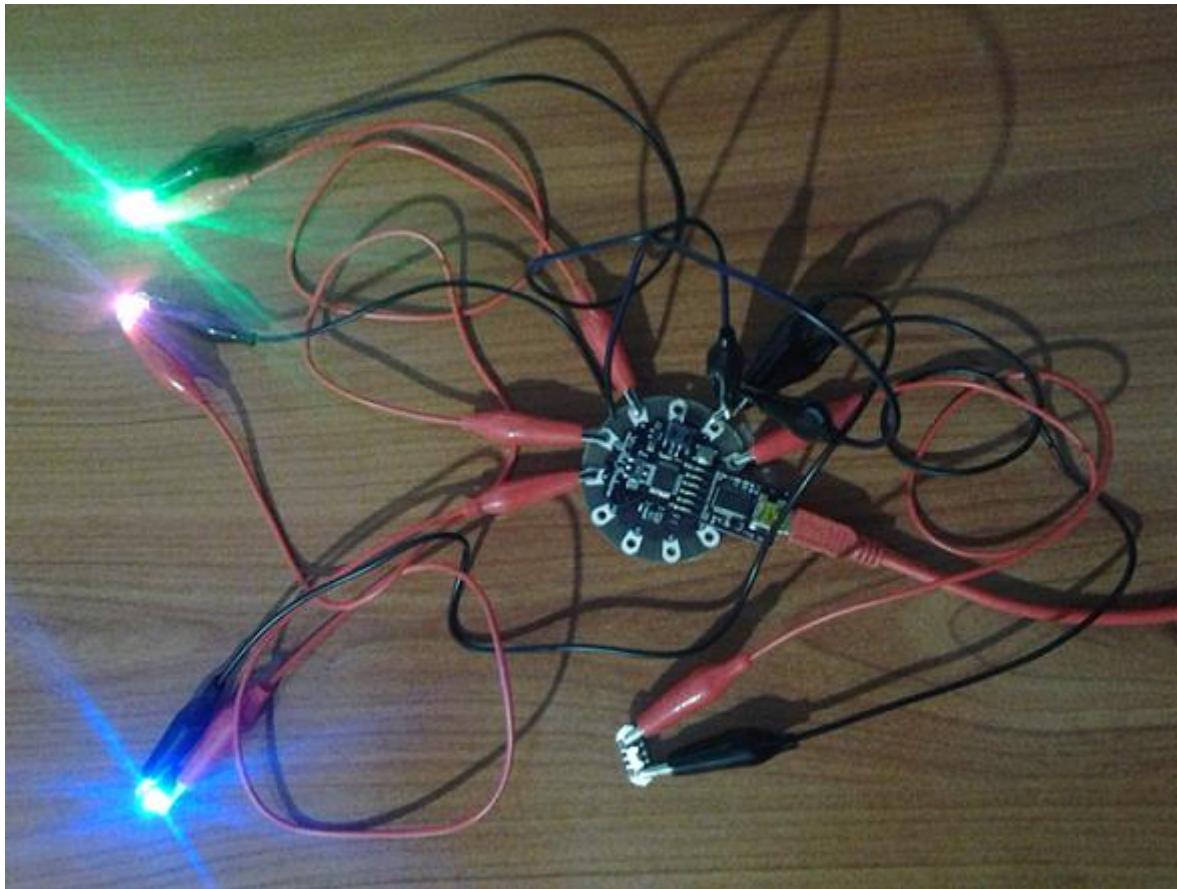
Τι συμβαίνει όταν πιέζεις τον αισθητήρας πίεσης και τι συμβαίνει όταν δεν τον πιέζεις;

## ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ

Αλλαξε το πρόγραμμα ώστε

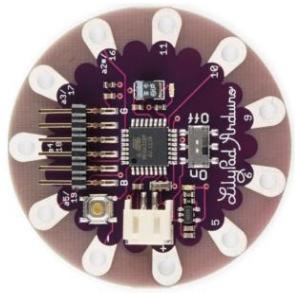
- όταν πιέσεις τον αισθητήρα το ρομπότ να κινείται 20 εκατοστά, να στρίβει  $90^\circ$  μοίρες και να κινείται άλλα 10 εκατοστά.
- όταν δεν πιέζεις τον αισθητήρα πίεσης το ρομπότ να στρίβει  $1^\circ$  μοίρα.

**ΜΟΝΟΧΡΩΜΑ ΦΩΤΑΚΙΑ LEDs ΠΟΥ ΑΝΑΒΟΣΒΗΝΟΥΝ**



## ΤΙ ΧΡΕΙΑΖΕΣΑΙ

- Ένα μικροεπεξεργαστή LilyPad.



- Τρία μονόχρωμα φωτάκια LED's συνδεδεμένα στις θύρες 5, 6 και 9 του μικροεπεξεργαστή.



- Έναν διακόπτη (switch) συνδεδεμένο στη θύρα A5 του μικροεπεξεργαστή.

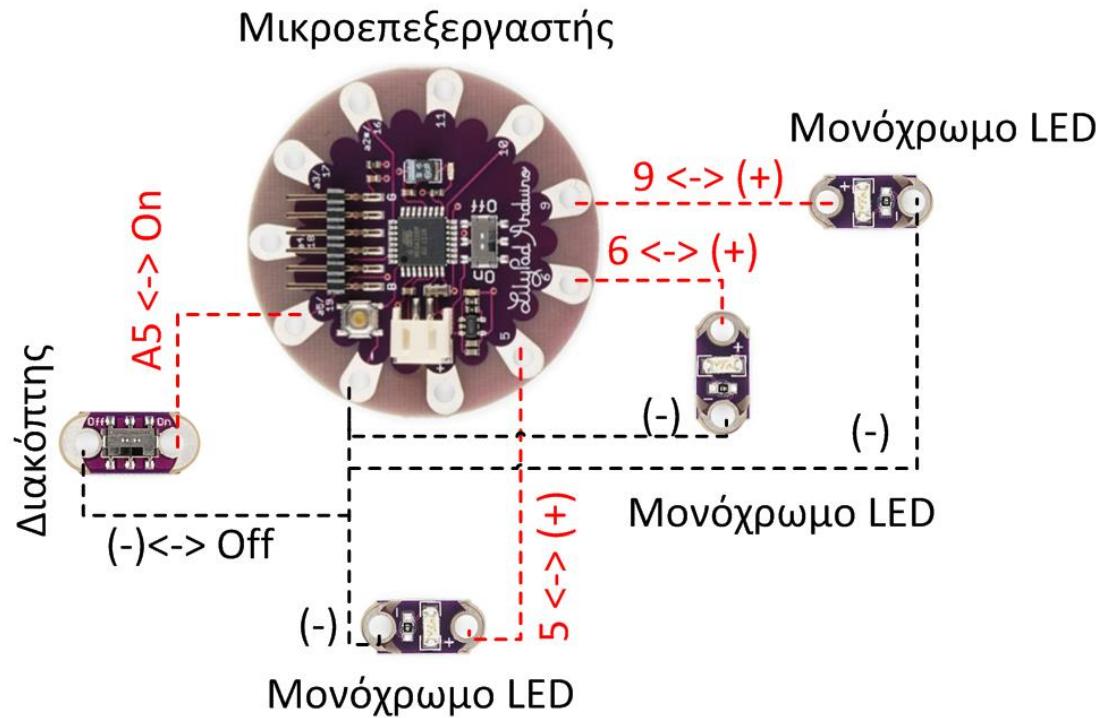


- Καλώδια (κροκοδιλάκια) για την σύνδεση των LED's και του διακόπτη με το μικροεπεξεργαστή.



## ΕΤΟΙΜΑΣΟΥ

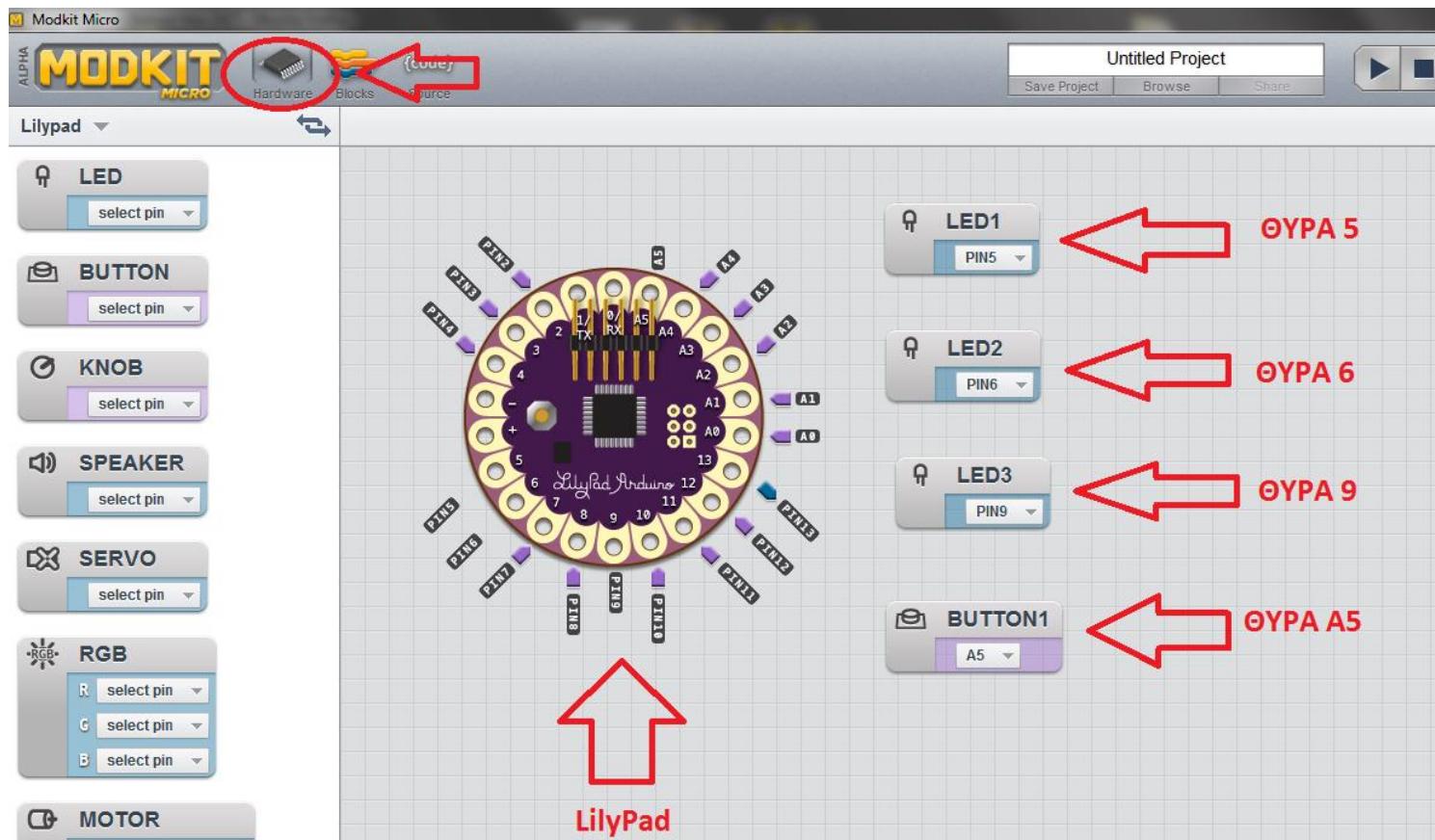
10. Φτιάξε το παρακάτω κύκλωμα:



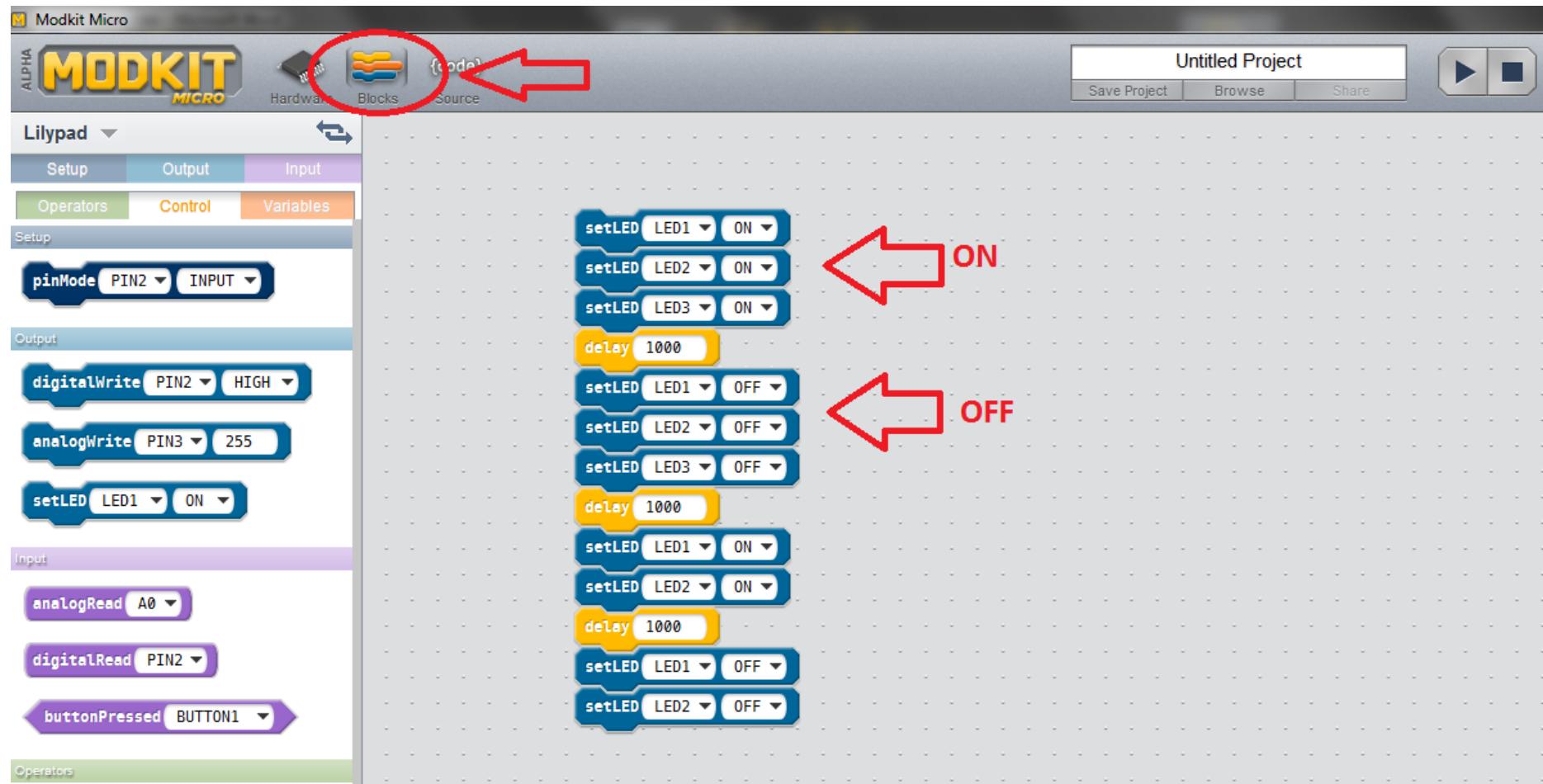
- Ένωσε το πρώτο LED με το μικροεπεξεργαστή. Χρησιμοποιώντας ένα καλώδιο (κροκοδιλάκι) ένωσε τη θύρα (+) του LED με τη θύρα 5 του μικροεπεξεργαστή και στην συνέχεια με ένα άλλο καλώδιο την θύρα (-) του LED με την θύρα (-) του μικροεπεξεργαστή.
- Ένωσε και τα αλλά δυο LED με το μικροεπεξεργαστή με ανάλογο τρόπο χρησιμοποιώντας τα καλώδια. Προσοχή οι συνδέσεις να γίνουν όπως φαίνονται στο σχήμα. Ένα LED στη θύρα 6 και στη (-) και ένα LED στη θύρα 9 και στη θύρα (-).
- Ένωσε το Διακόπτη με το μικροεπεξεργαστή. Χρησιμοποιώντας ένα καλώδιο (κροκοδιλάκι) ένωσε τη θύρα On του διακόπτη με τη θύρα A5 του μικροεπεξεργαστή και με ένα άλλο καλώδιο την θύρα Off του διακόπτη με τη θύρα (-) του μικροεπεξεργαστή.

11. Κάνε διπλό κλικ στο εικονίδιο  στην επιφάνεια εργασία για να ανοίξεις το πρόγραμμα Modkit.

12. Επέλεξε το  και πρόσθεσε τα 3 μονόχρωμα LED's (PIN5, PIN6, PIN9 ) και τον διακόπτη στην θύρα A5.



## ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



## ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



Κάνε κλικ στο εικονίδιο ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω στο LilyPad.

Τι συμβαίνει; Πόσες φορές αναβοσβήνουν τα λαμπάκια; Κάθε πότε σβήνουν και ανοίγουν τα λαμπάκια;

## ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ

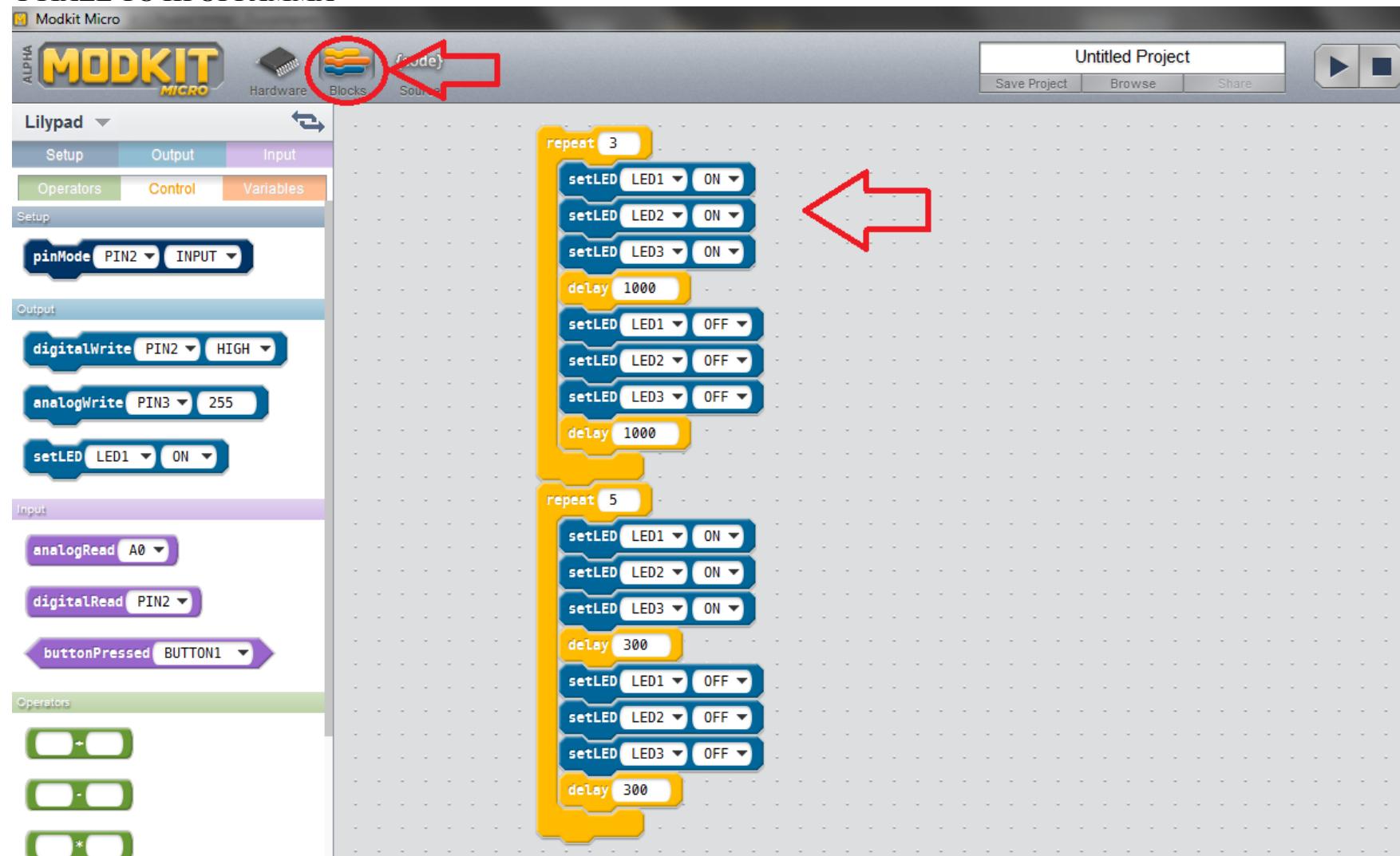
Φτιάξε το πρόγραμμα ώστε:

- πρώτα να ανάβει το LED1
- μετά από 1 δευτερόλεπτο να σβήνει το LED1 και να ανάβει το LED2
- μετά από 1 δευτερόλεπτο να σβήνει το LED2 και να ανάβει το LED3
- μετά από 1 δευτερόλεπτο να σβήνει και το LED3

Που θα μπορούσες να χρησιμοποιήσεις την εντολή τυχαίου αριθμού :

randomNumber 1000 3000

## ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



## ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



Κάνε κλικ στο εικονίδιο ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω στο LilyPad.

Πόσες φορές αναβοσβήνουν τα λαμπάκια αργά και πόσες φορές γρήγορα;

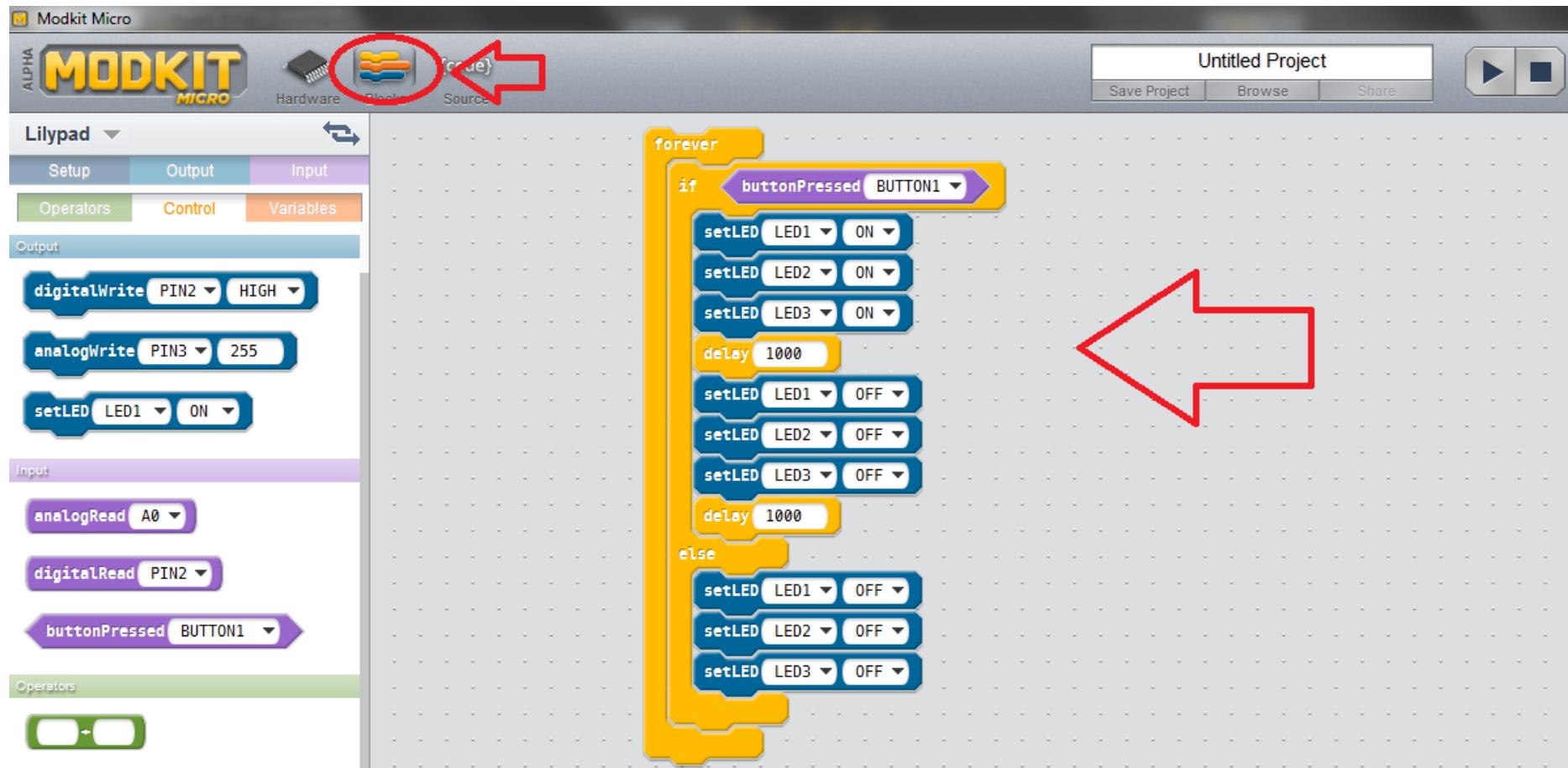
## ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ

Κάνε τα λαμπάκια να αναβοσβήσουν 4 φορές αργά (με 1 δευτερόλεπτο καθυστέρηση ) και μετά 6 φορές γρήγορα (0.3 δευτερόλεπτα καθυστέρηση ).

Κάνε τα λαμπάκια να αναβοσβήνουν **συνέχεια** με τον παραπάνω ρυθμό (4 αργά και 6 γρήγορα) χρησιμοποιώντας κατάλληλα την εντολή για πάντα:



## ΦΤΙΑΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ



## **ΤΡΕΞΕ ΤΟ ΠΡΟΓΡΑΜΜΑ**



Κάνε κλικ στο εικονίδιο ώστε να τρέξει το πρόγραμμα που έφτιαξες παραπάνω στο LilyPad.

Άνοιξε (On) και κλείσε (Off) τον διακόπτη.

Τι συμβαίνει όταν ο διακόπτης είναι στο On και τι συμβαίνει όταν ο διακόπτης είναι στο Off;

## **ΠΗΓΑΙΝΕ ΠΑΡΑΠΕΡΑ**

Άλλαξε το πρόγραμμα ώστε:

- όταν ο διακόπτης είναι στο On τα φωτάκια να αναβοσβήνουν αργά (ανά 1 δευτερόλεπτο)
- ενώ όταν ο διακόπτης είναι στο Off τα φωτάκια να αναβοσβήνουν γρήγορα (ανά 0.3 δευτερόλεπτα)

«Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα εργασία αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον»