



ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΚΑΙ ΤΗ ΡΟΜΠΟΤΙΚΗ

Μάθημα 14^ο

Μεταβλητές και αισθητήρες

Στόχοι του κεφαλαίου

Με την ολοκλήρωση του κεφαλαίου οι μαθητές:

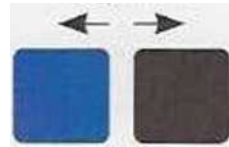
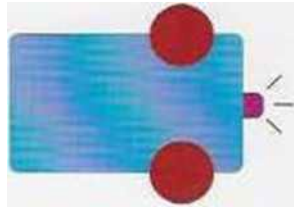
- Θα γνωρίζουν πώς να αποθηκεύουν σε μεταβλητές ή λίστες, τιμές που λαμβάνονται από αισθητήρες.
- Θα μπορούν να προγραμματίσουν το ρομπότ τους να εκτελεί απλές αποστολές, μέσω των προτεινόμενων δραστηριοτήτων.
- Θα είναι σε θέση να απαντήσουν σωστά τις ερωτήσεις του φύλλου αξιολόγησης, στο τέλος του κεφαλαίου.

Μην ξεχάσεις:

Πριν ξεκινήσεις να προγραμματίζεις, θα πρέπει να κατασκευάσεις το ρομπότ Driving Base 3, προσαρμόζοντας κατάλληλα τον αισθητήρα χρώματος να κοιτάει προς τα πλάγια. Το ρομπότ θα το βρεις στην αρχική σελίδα του προγράμματος του Spike Prime, στην κατηγορία Build.

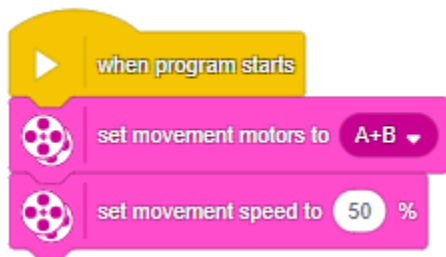
Το συγκεκριμένο κεφάλαιο θα σε βοηθήσει πολύ να κατανοήσεις πλήρως τις μεταβλητές και τις λίστες. Επίσης, είναι το κεφάλαιο που χρησιμοποιείται αρκετά σε διαγωνισμούς ρομποτικής. Σε πολλές κατηγορίες αγώνων απαιτείται από την εκάστοτε ομάδα, πριν ξεκινήσει το ρομπότ να εκτελεί την ετήσια αποστολή, να διαβάσει κάποια χρώματα από διάφορα δομικά υλικά (κύβοι) και έπειτα, ανάλογα με το χρώμα ή τη σειρά των χρωμάτων, να αλλάξει η σειρά εκτέλεσης των εντολών. Στο 1ο παράδειγμα που ακολουθεί, θα αναλυθεί βήμα προς βήμα ο τρόπος που θα αποθηκευτούν οι τιμές των χρωμάτων από τον αισθητήρα COLOR, σε 2 μεταβλητές. Στη συνέχεια, ανάλογα με τις τιμές των μεταβλητών, θα εκτελείται διαφορετικός κώδικας προγραμματισμού.

Στο 2ο παράδειγμα, θα αναλυθεί ακριβώς ο ίδιος κώδικας, με τη διαφορά πως οι τιμές θα αποθηκευτούν σε μία αριθμητική λίστα.

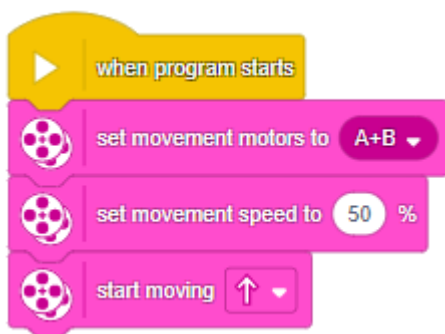
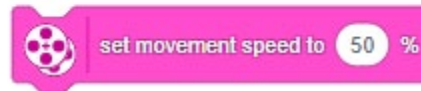
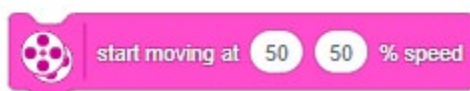
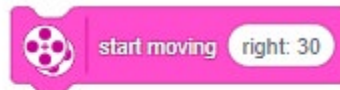
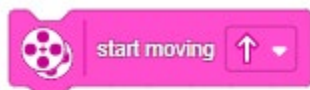


Παράδειγμα 1 - Αποθήκευση τιμών αισθητήρα σε μεταβλητές

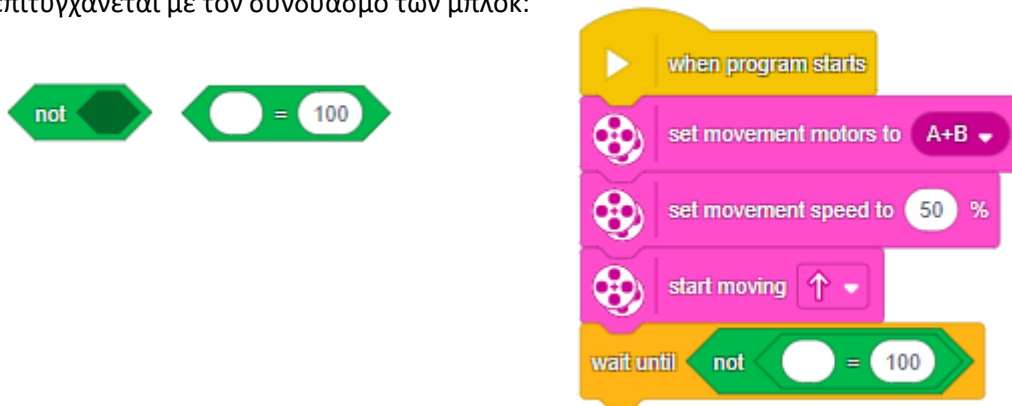
Βήμα 1: Ενεργοποίησε τους κινητήρες που θα χρησιμοποιήσεις και την ταχύτητά τους (απαιτείται μόνο αν στο επόμενο βήμα επιλέξεις μπλοκ, χωρίς δήλωση ταχύτητας).



Βήμα 2: Πρόσθεσε ένα μπλοκ που θα κινεί το ρομπότ ευθεία. Η απόσταση δεν θα πρέπει να βρίσκεται ενσωματωμένη στο μπλοκ, γιατί θα καθοριστεί μέσω του αισθητήρα. Επίλεξε ένα από τα παρακάτω μπλοκ :

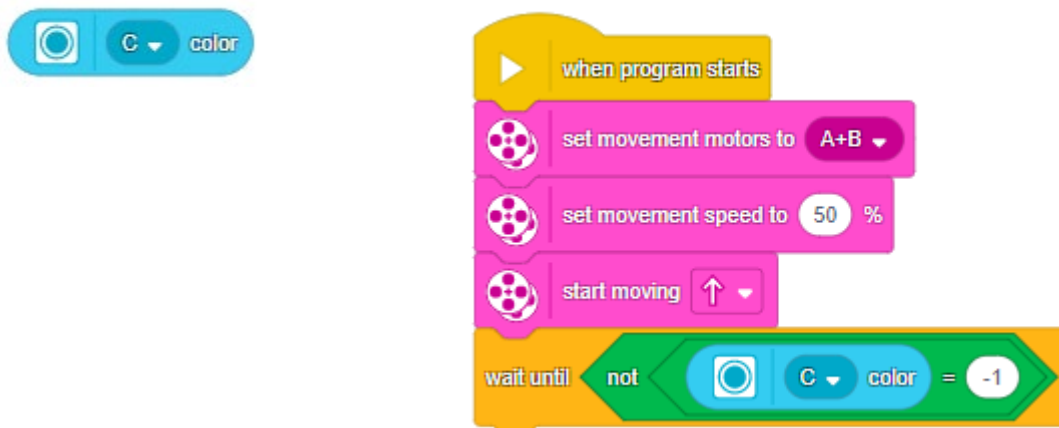


Βήμα 3: Το ρομπότ πρέπει να προχωρήσει μέχρι η τιμή του αισθητήρα χρώματος να μην είναι -1. δηλαδή μέχρι να «πέσει» πάνω στο αντικείμενο. Η συνθήκη ελέγχου τερματισμού επιτυγχάνεται με τον συνδυασμό των μπλοκ:

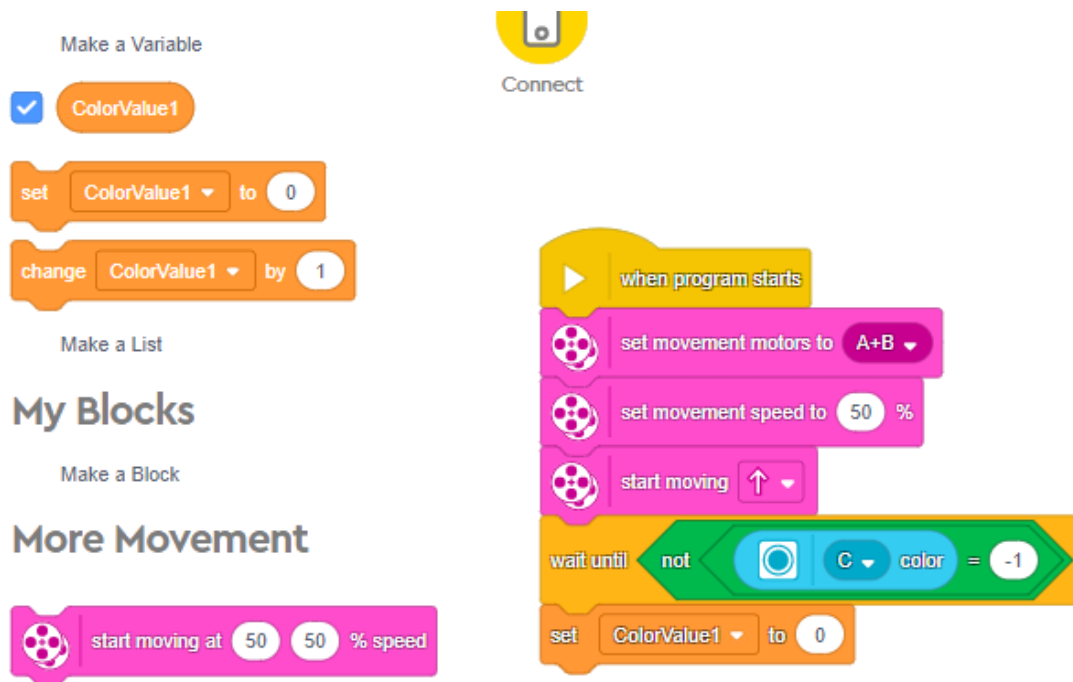


ΑΠΛΑ ΘΥΜΗΣΟΥ : Το -1 είναι η τιμή που έχει ο αισθητήρας όταν δεν διαβάζει κανένα χρώμα.

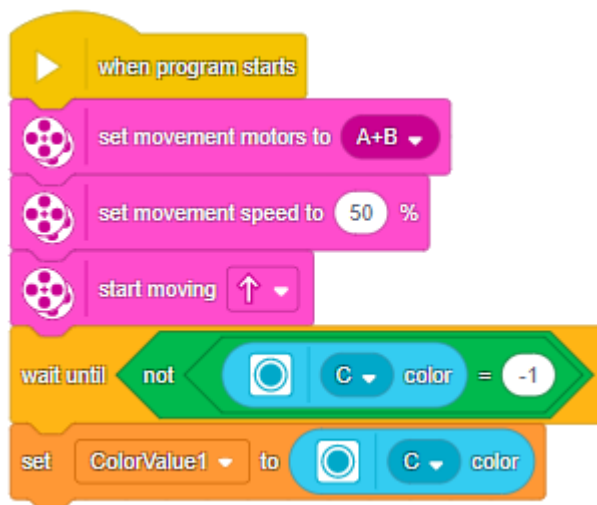
Βήμα 4 : Άλλαξε το 100 σε -1 και στο άδειο κυκλάκι βάλε το μπλοκ του αισθητήρα **COLOR**.



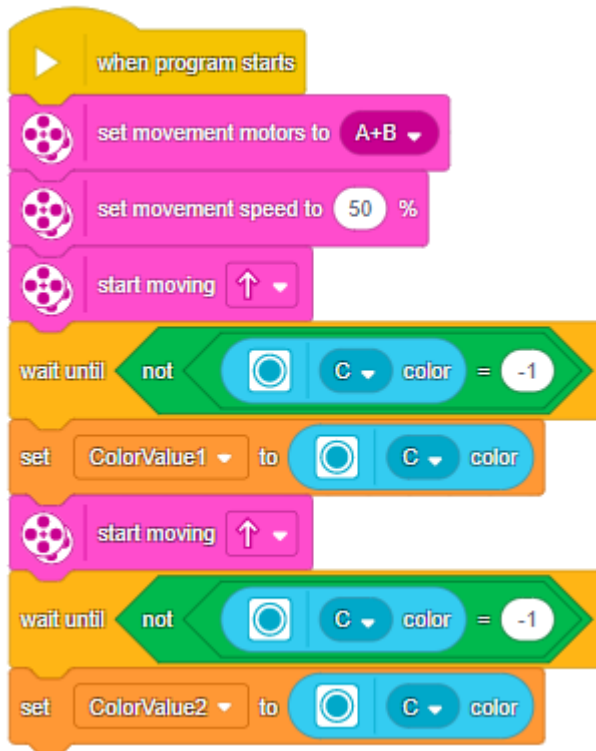
Βήμα 5 : Όταν ο αισθητήρας αποκτήσει τιμή διαφορετική από το -1, τότε θα πρέπει αυτή η τιμή να αποθηκευτεί σε μία μεταβλητή. Για τη δημιουργία της μεταβλητής, επέλεξε από την κατηγορία **Variables** → **make a variable** → **name** . Στη συνέχεια, πρόσθεσε το μπλοκ **Set name to 0**.



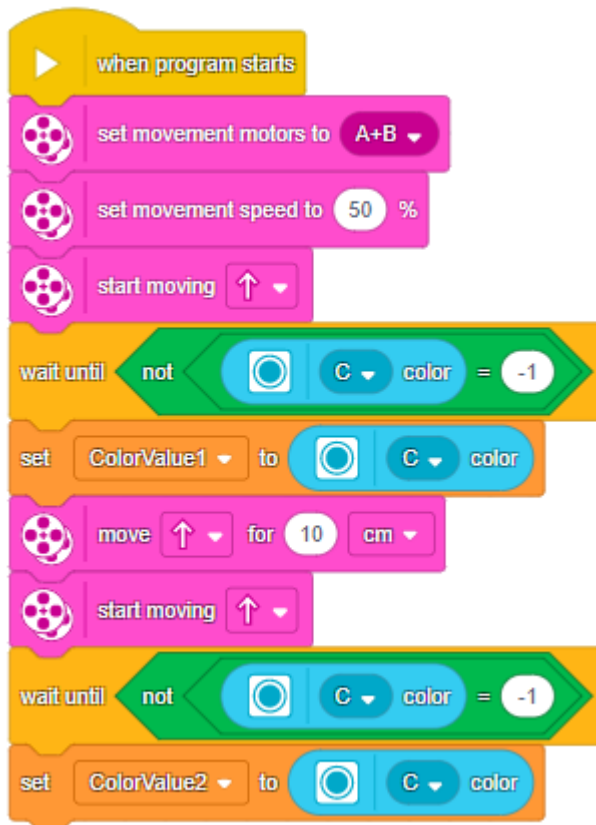
Βήμα 6: Στη θέση του 0 θα πρέπει να μπει το μπλοκ του αισθητήρα, όπως μπήκε και στη συνθήκη ελέγχου.



Βήμα 7: Επανάλαβε τα βήματα 2-6, όσες φορές θέλεις ανάλογα με το πλήθος των αντικειμένων που έχεις. Σύμφωνα με την εκφώνηση του παραδείγματος, θα χρειαστεί να επαναλάβεις τα βήματα άλλη μία φορά, όπως φαίνεται παρακάτω.

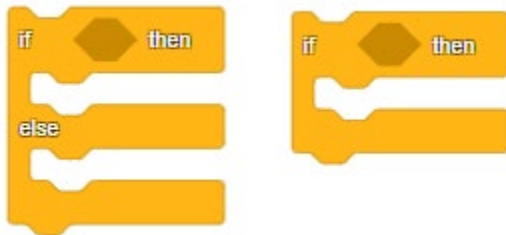


Βήμα 8: Ενδιάμεσα στα 2 διαβάσματα των μεταβλητών, θα πρέπει να προστεθεί ένα μπλοκ κίνησης, ώστε το ρομπότ να απομακρυνθεί από το χρωματιστό αντικείμενο όσο απαιτείται, ώστε ο αισθητήρας χρώματος να ξαναπάρει την τιμή -1.



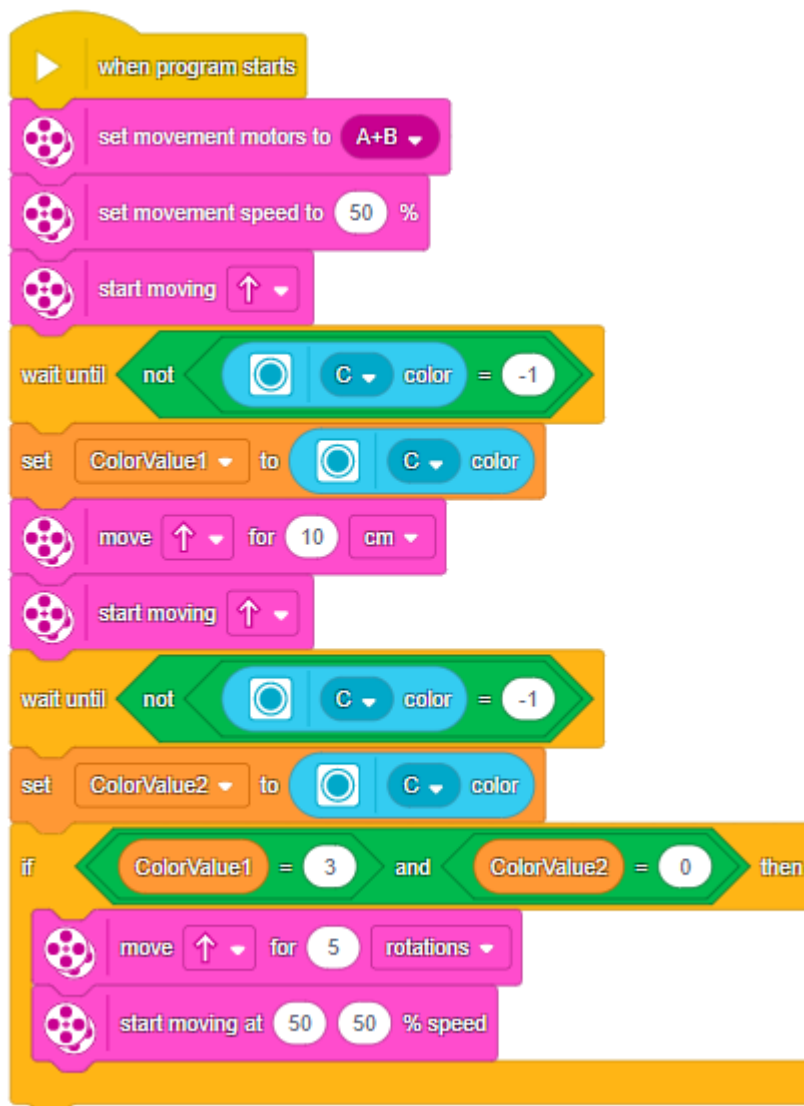
Σημείωση: Η τιμή (10cm) είναι ενδεικτική. Άλλαξε την τιμή ανάλογα με το μέγεθος του αντικειμένου σου.

Βήμα 9: Πλέον, στις μεταβλητές σου βρίσκονται αποθηκευμένες οι τιμές των χρωμάτων αντικειμένων. Χρησιμοποιώντας κάποιο/α από τα παρακάτω μπλοκ επιλογής, μπορείς πολύ εύκολα να προγραμματίσεις το ρομπότ να εκτελεί διαφορετική ακολουθία μπλοκ κάθε φορά.



Ενδεικτική εκφώνηση: Να γραφτεί πρόγραμμα στο οποίο το ρομπότ θα κινείται ευθεία για 5

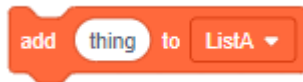
rotations μόνο στην περίπτωση που η 1η μεταβλητή έχει την τιμή 3 (μπλε χρώμα) και η 2η μεταβλητή την τιμή 0 (μαύρο χρώμα).




Παράδειγμα 2 - Αποθήκευση τιμών αισθητήρα σε αριθμητική λίστα

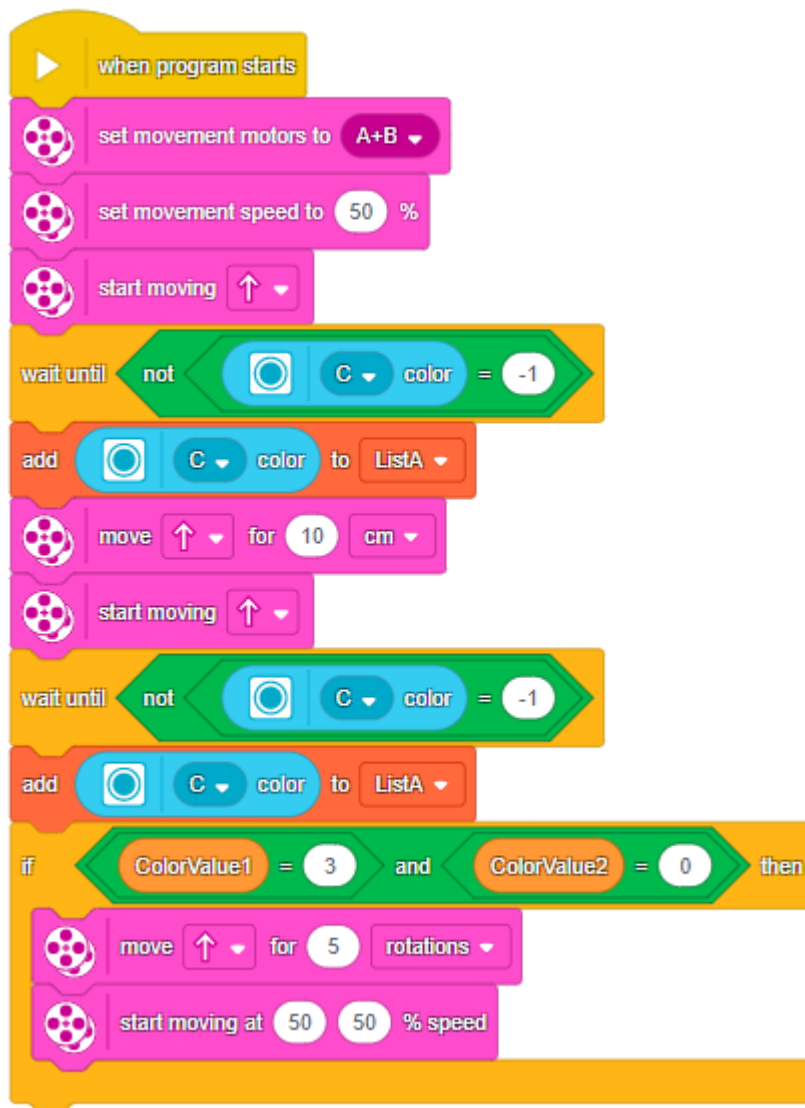
Βήμα 1: Για την αποθήκευση των τιμών του αισθητήρα σε μία αριθμητική λίστα, δεν χρειάζεται να γράψεις τον κώδικά σου από την αρχή. Οι αλλαγές που θα γίνουν θα αφορούν μόνο τα μπλοκ των μεταβλητών και την αντικατάστασή τους με τα απαραίτητα μπλοκ της λίστας.

Για να δημιουργήσεις μία λίστα, επίλεξε από την κατηγορία **Variables** → **make a List** → **name**. Στη συνέχεια, αντικατάστησε το μπλοκ «Set name to 0» που είχες προηγουμένως, με το μπλοκ

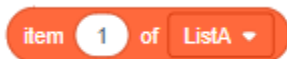


ΥΠΕΝΟΥΜΙΣΗ : Το συγκεκριμένο μπλοκ προσθέτει την τιμή του **thing** στο τέλος της λίστας.

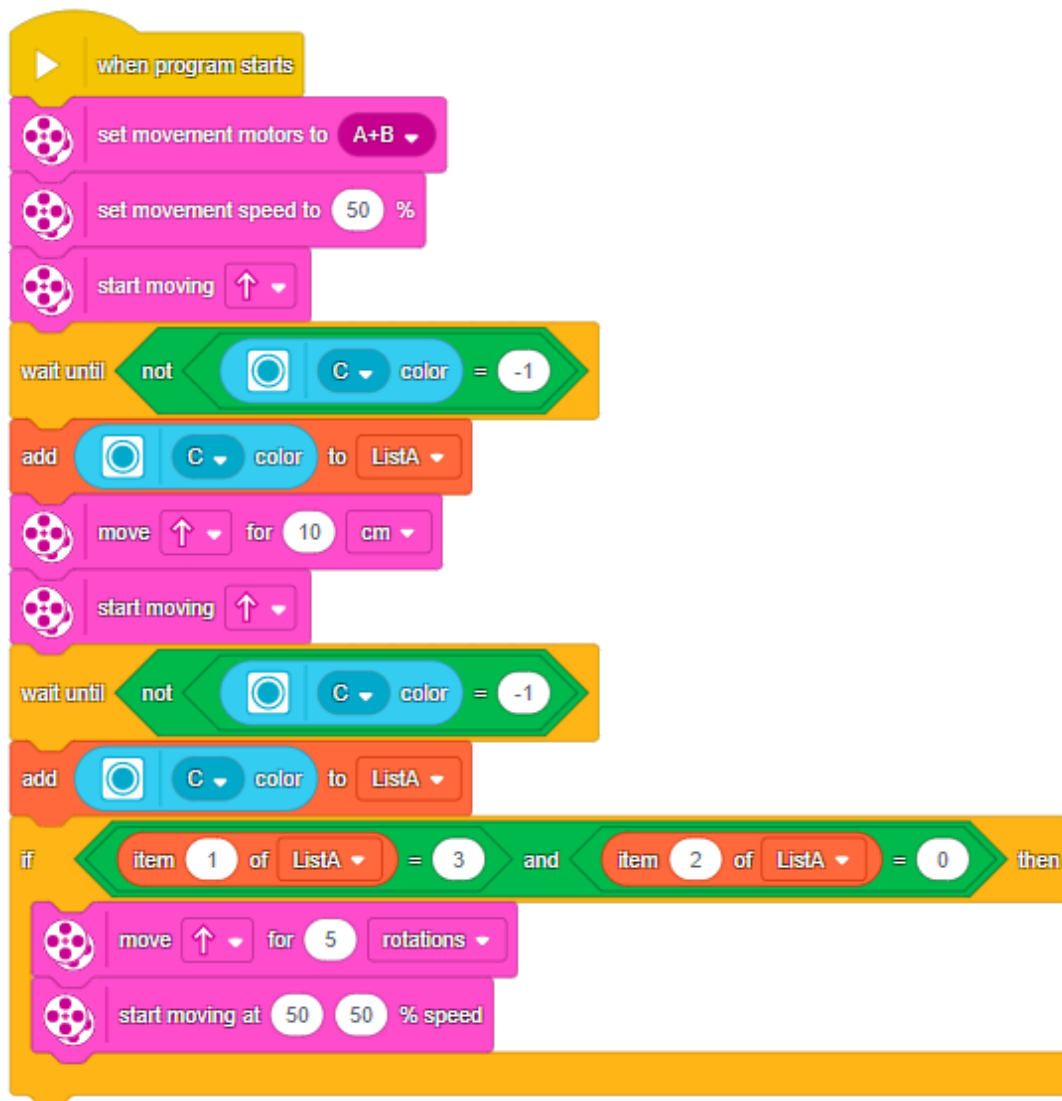
Έπειτα πρόσθεσε το μπλοκ του αισθητήρα χρώματος  στη θέση του **thing**, όπως φαίνεται παρακάτω.



Βήμα 2: Στη συνέχεια, άλλαξε τα μπλοκ των μεταβλητών που υπάρχουν μέσα στο **if** με το μπλοκ.



Το συγκεκριμένο μπλοκ μπαίνει σε συνθήκες ελέγχου. Το «item», δηλώνει τη θέση του στοιχείου που θα ελεγχθεί μέσα στη λίστα .



Σημείωση: Στο 2ο μέρος της συνθήκης, το 1 άλλαξε σε 2 γιατί πρέπει να ελεγχθεί η τιμή του 2ου αντικειμένου. Αντιστοιχεί, δηλαδή, στη 2η μεταβλητή που είχαμε προηγουμένως.

Σημείωση: Αν χρειαστεί να τρέξεις τον κώδικα σου πολλές φορές για δοκιμές, θα πρέπει να

A screenshot of a red button with the text "delete all of ListA" and a small downward-pointing triangle icon.

προσθέσεις σαν πρώτο μπλοκ το . Μπορείς να σκεφτείς γιατί χρειάζε-
ται;

ΑΠΑΝΤΗΣΗ: Το συγκεκριμένο μπλοκ διαγράφει τη λίστα, οπότε σε κάθε δοκιμή που θα εκτελείς, θα πρέπει η λίστα σου να αδειάζει.