INTERMEDIATE PROGRAMMING LESSON

JALEXAK ADDED

INTRODUCTION TO MY BLOCKS

By Sanjay and Arvind Seshan

EV3 CLASSROOM LESSON
BY EV3LESSONS.COM

Lesson Objectives

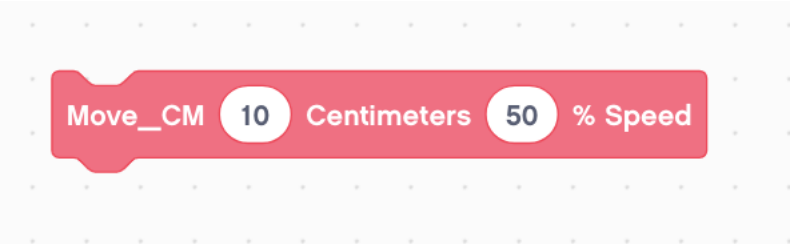
Learn how to make custom blocks in the EV3 Classroom Software (My Blocks)

Learn why a My Block is useful

Learn to construct a My Block with Inputs and Outputs (Parameters)

What is a My Block?

- A My Block is a combination of one or more blocks that you create that can be grouped into a single block
- My Blocks are basically your own custom blocks
- Once a My Block is created, you can use it in multiple programs
- Just like any other block in EV3, My Blocks can have both inputs and outputs (parameters)



The blocks above is an example of a My Block:

- Move_CM tells the robot to move the number of CM we input
- You will create this My Block in the next lesson

When do You Use a My Block?

- Whenever the robot is going to repeat an action inside your program
- When code is repeated in a different program
- Organize and simplify your code

Why Should You Bother?

Because of My Blocks, your missions will look like this…

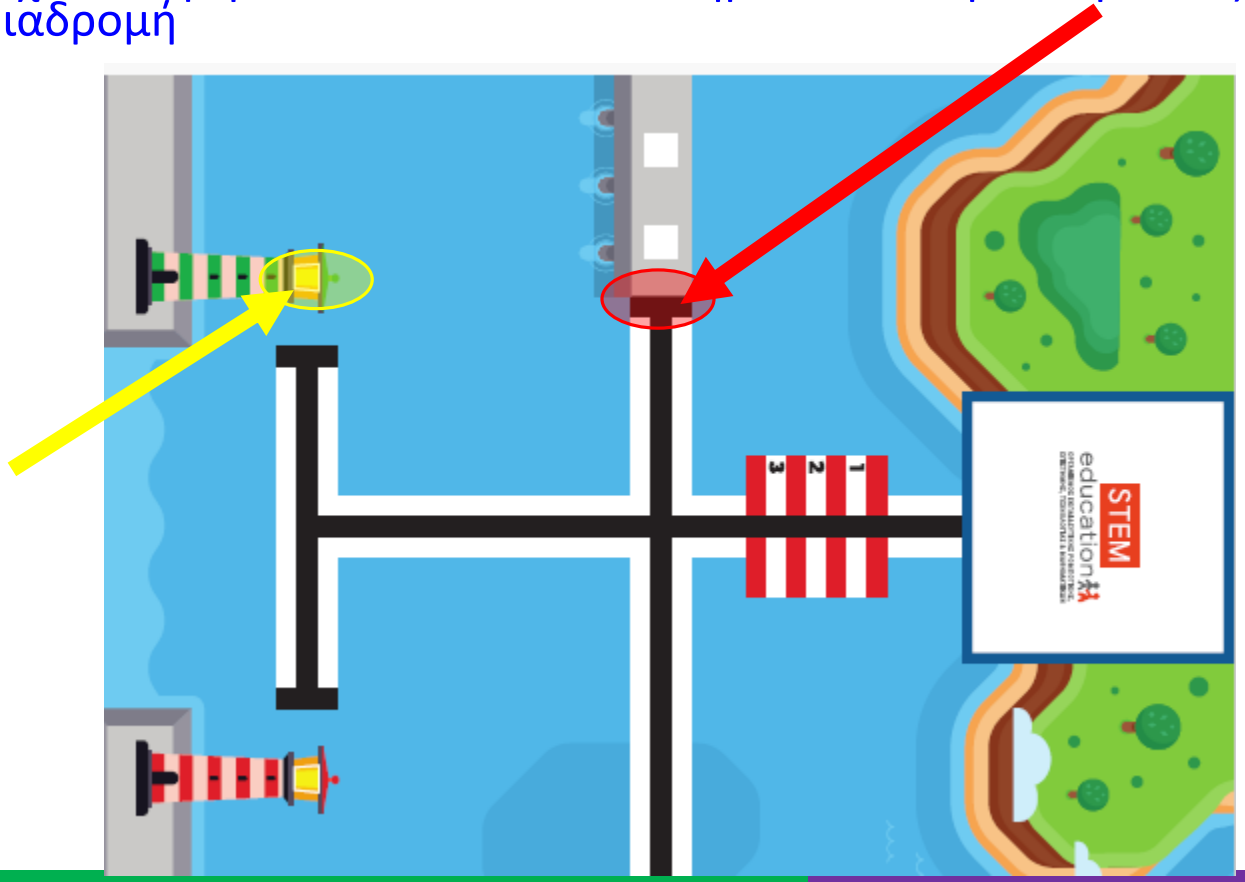
Παρακάτω το robot προχωράει σε 2 πλευρές ενός ορθογωνίου με κατανοητό κώδικα

This makes your code easier to read and easier to modify!

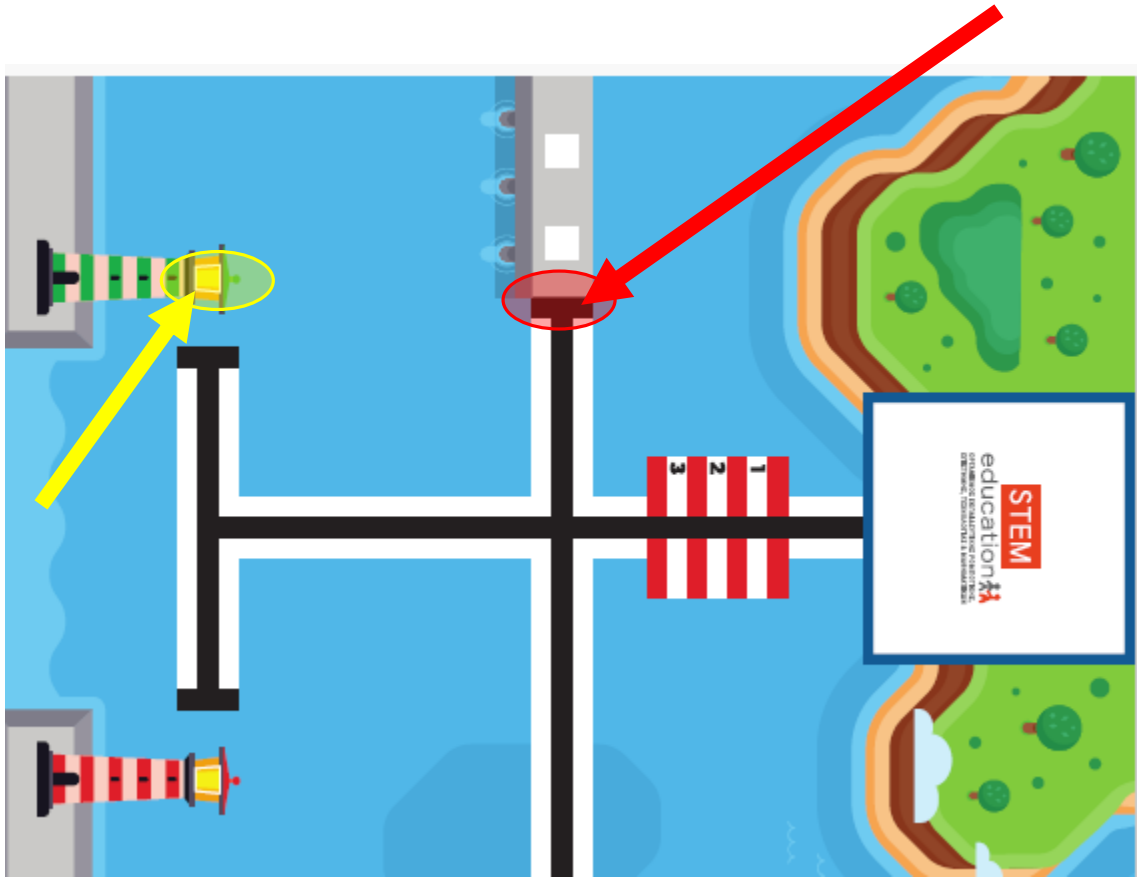
Ας φτιάξουμε block βήμα - βήμα

Στην παρακάτω πίστα (διαγωνισμός STEM 2023-2024) θέλουμε:

Το ρομπότ να ξεκινήσει από την αρχή να πάει στο σημείο του κόκκινου βέλους και στη συνέχεια να γυρίσει και να πάει στο σημείο του κίτρινου βέλους ακολουθώντας πάντα τη διαδρομή



Βήμα 1: Ας φτιάξω τον αλγόριθμο



Κινήσου ευθεία ως τη διασταύρωση

Pivot δεξιά 90

Follow line μέχρι διασταύρωση

Spin 180

Follow line till cross

Pivot δεξιά 90

Follow line till cross

Pivot δεξιά 90

Follow line till cross

Βήμα 2: αλγόριθμος σε κώδικα

Κινήσου ευθεία ως τη διασταύρωση

Pivot δεξιά 90

Follow line μέχρι διασταύρωση

Spin 180

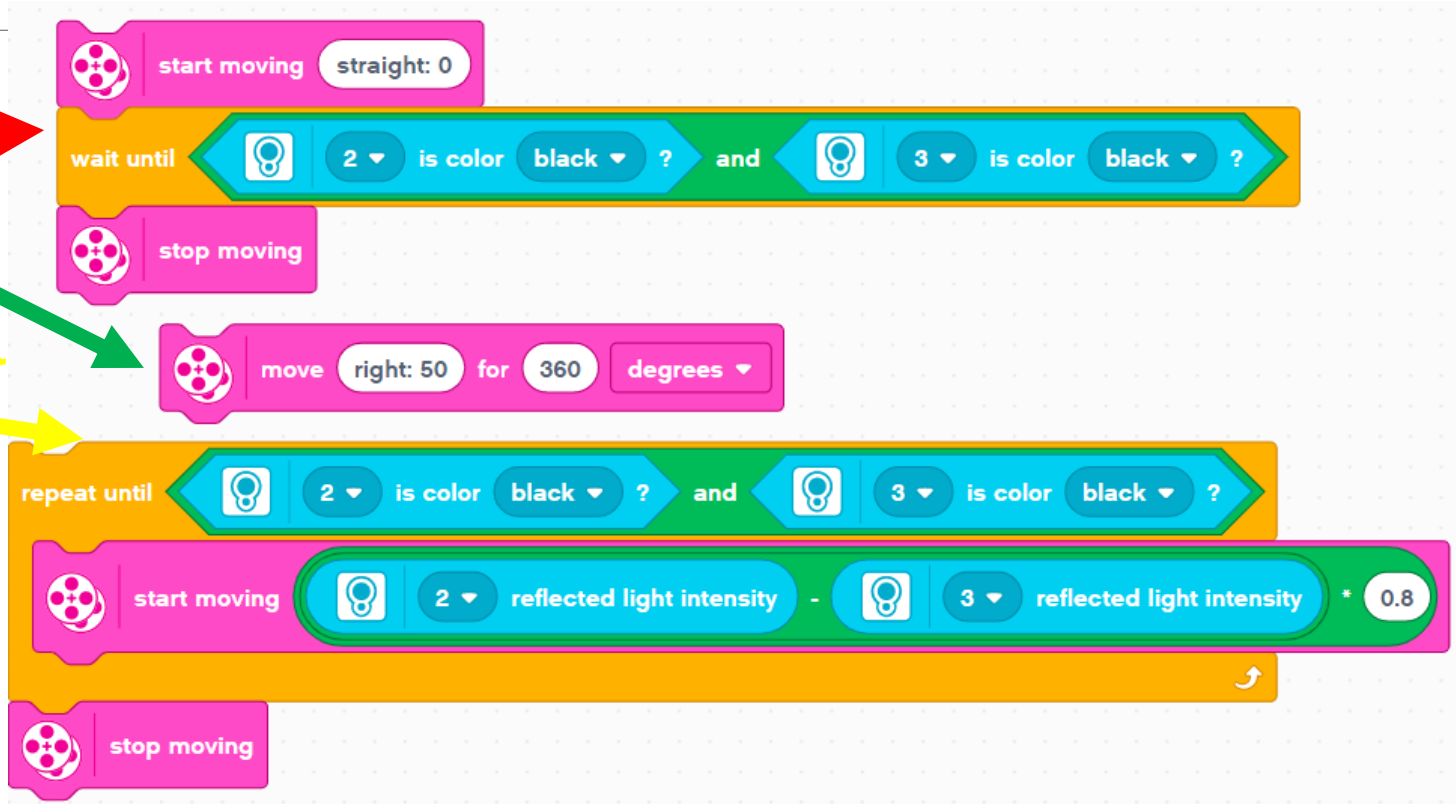
Follow line till cross

Pivot δεξιά 90

Follow line till cross

Pivot δεξιά 90

Follow line till cross



<u>Ερώτηση:</u> Μήπως είναι πολύ πιο απλός ο αλγόριθμος από το πρόγραμμα; Άρα μήπως καλύτερα να γράφαμε αλγόριθμο; Αν συμφωνείτε πάμε να φτιάξουμε τις δικές μας εντολές (block) για τον αλγόριθμο.

Βήμα 3: Make blocks

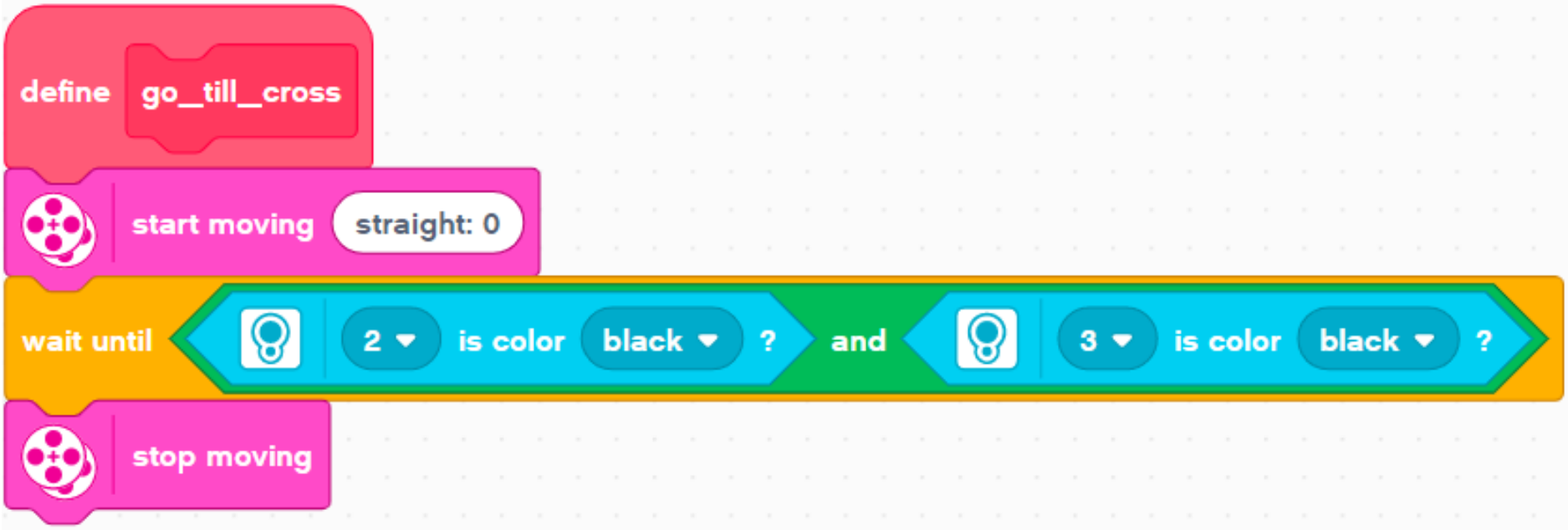
Πήγαινε στο MyBlocks

Πάτα makeblock

Δώσε όνομα «go_till_cross» και πάτα save

Βάλε κάτω από το block που φτιάχτηκε τις εντολές

Πρέπει να έχεις το block της παρακάτω εικόνας



Βήμα 4: Πίσω στον κώδικα

Κινήσου ευθεία ως τη διασταύρωση

Pivot δεξιά 90

Follow line μέχρι διασταύρωση

Spin 180

Follow line till cross

Pivot δεξιά 90

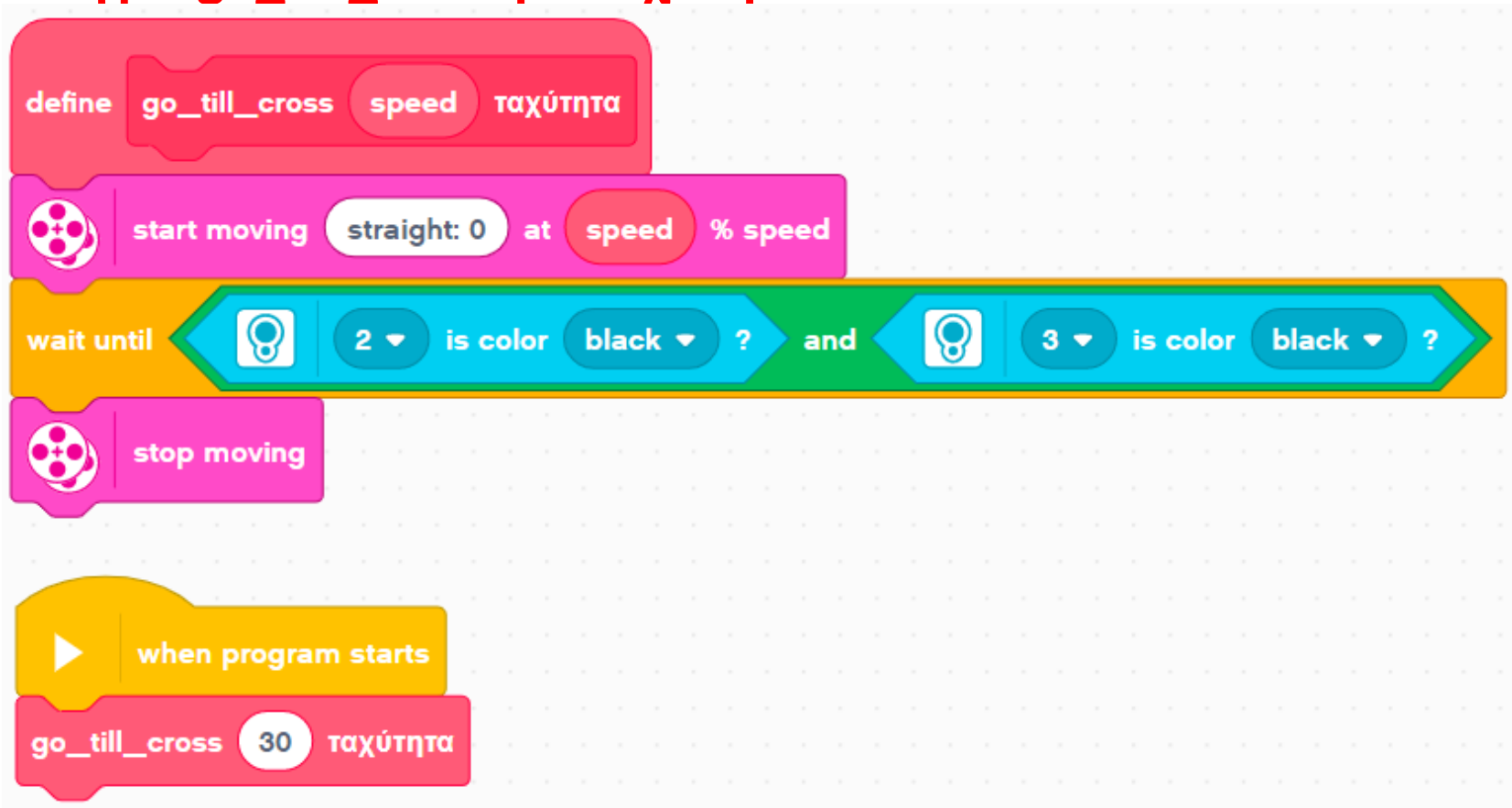
Follow line till cross

Pivot δεξιά 90

Follow line till cross

Βήμα 5a: Make blocks - παράμετροι

Πολλές φορές όμως τα block θέλουμε να έχουν παραμέτρους. Για παράδειγμα go_till_cross με ταχύτητα 30

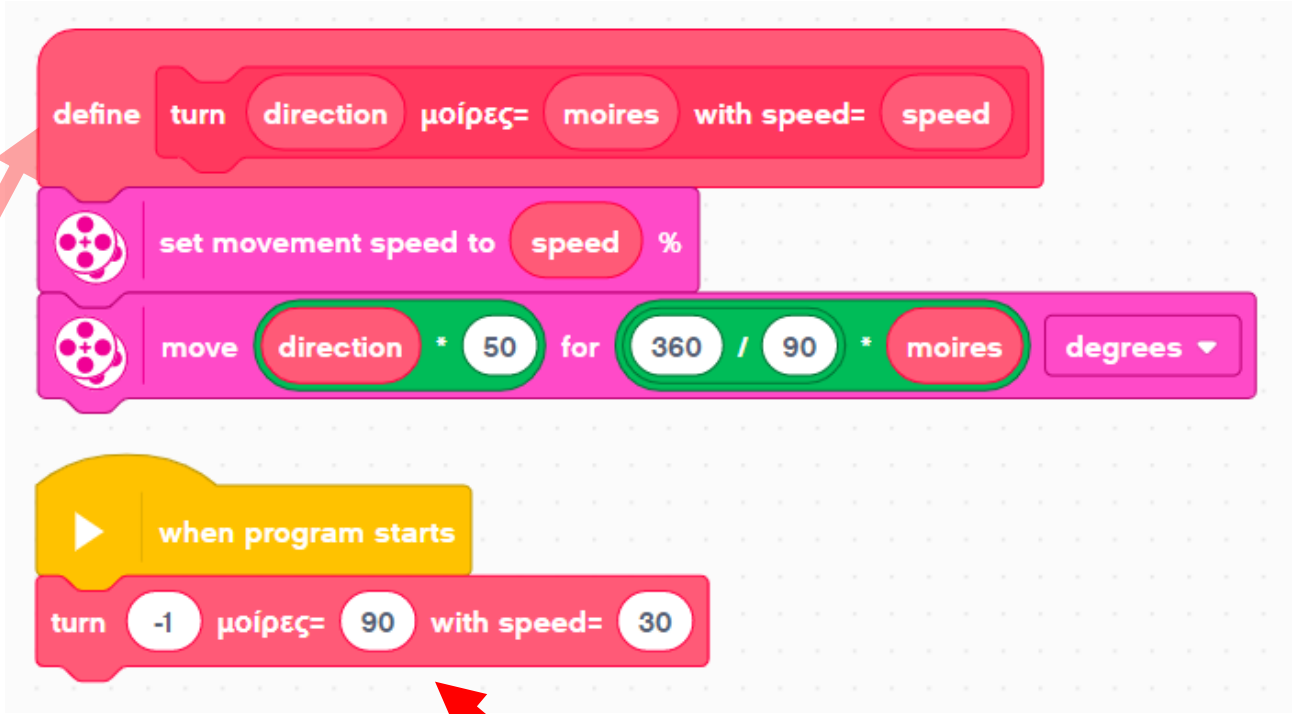


Βήμα 5b: Παράμετροι στα block

Pivot90:

- **Πόσες μοίρες να στρίψω**
- **Προς ποια κατεύθυνση**
- **Με ποια ταχύτητα;**

Φτιάχνω μια εντολή (block) που θα μπορεί να δεχτεί αυτές τις παραμέτρους



Τη χρησιμοποιώ για να στρίψω pivot αριστερά 90 μοίρες με ταχύτητα 30

Create Blocks

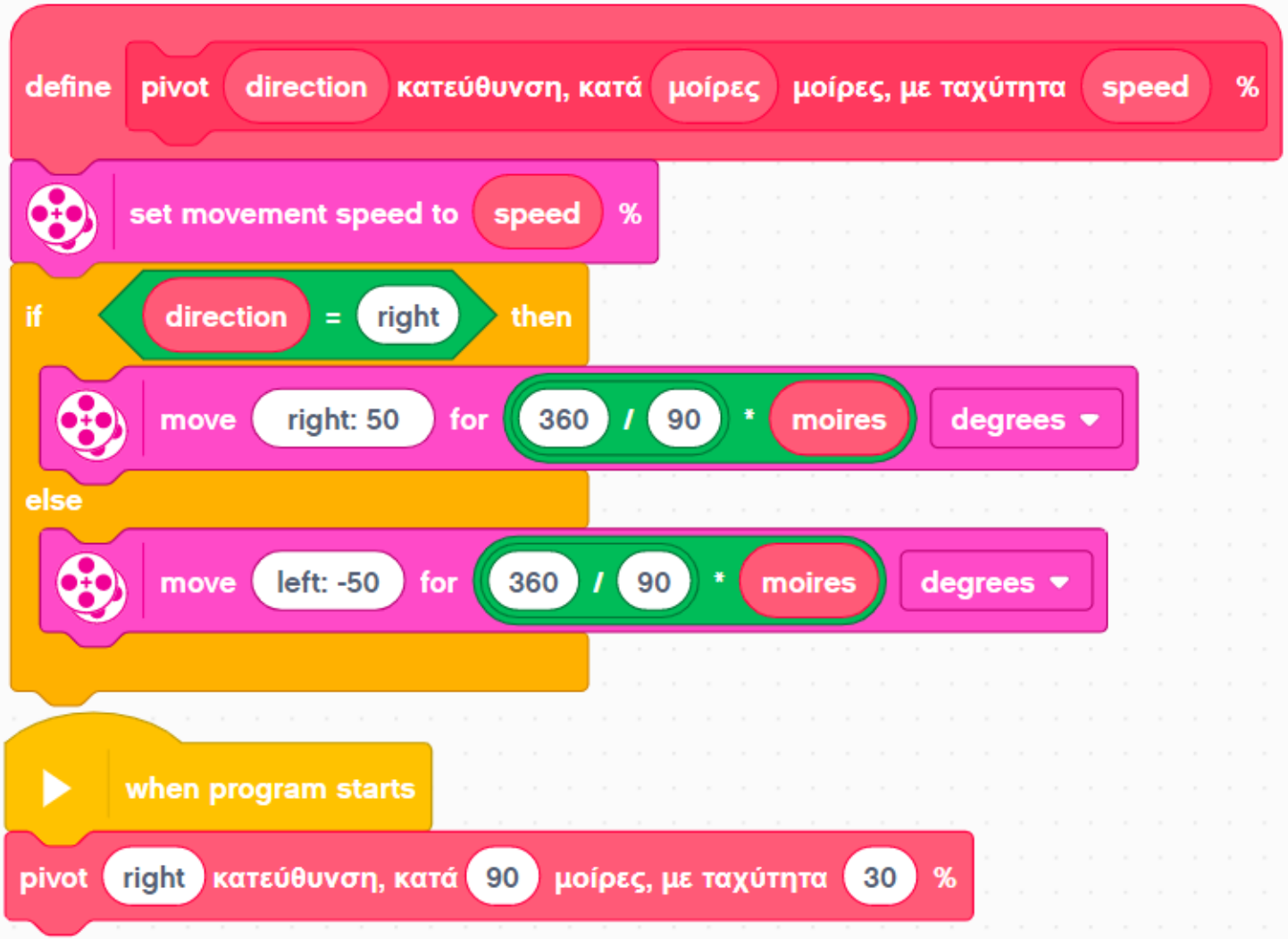
Παρακάτω θα δούμε πως μπορούμε να φτιάξουμε blocks

Μπορείτε να συνεχίσετε με την παρουσίαση όπου δείχνει τη δημιουργία blocks με video.

Εναλλακτικά μπορείτε να δείτε τον τρόπο δημιουργίας blocks από το https://ev3lessons.com/en/Lessons.html?tab=intermediate και πιο συγκεκριμένα στο pdf https://ev3lessons.com/en/ProgrammingLessons/intermediate/scratch-MyBlocksUpdated.pdf

Βήμα 5c Make and use pivot block

Ας φτιάξουμε το παρακάτω pivot block με παραμέτρους

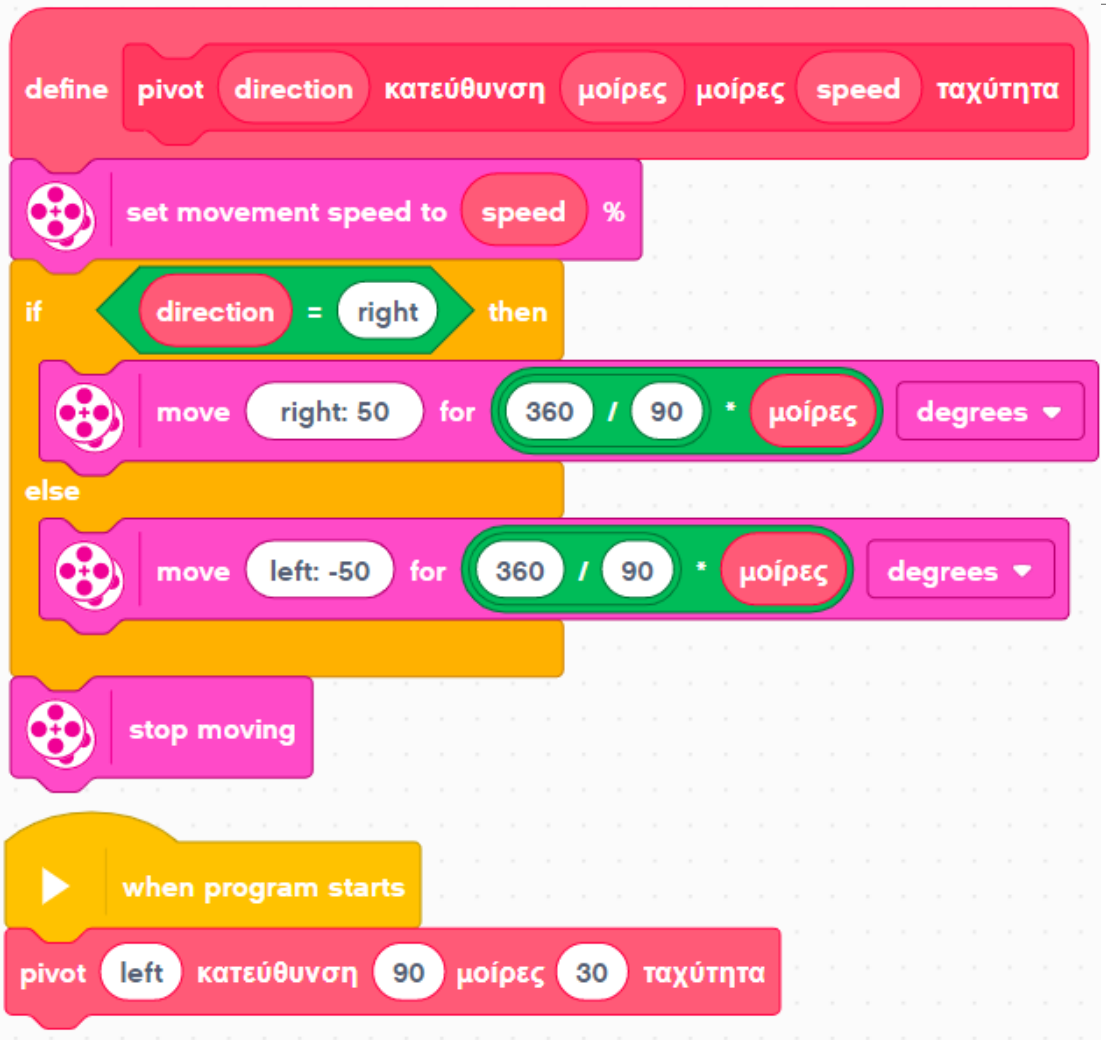


Βήμα 5c: Create and use pivot block

Video: https://www.youtube.com/watch?v=UE1gZnimvok

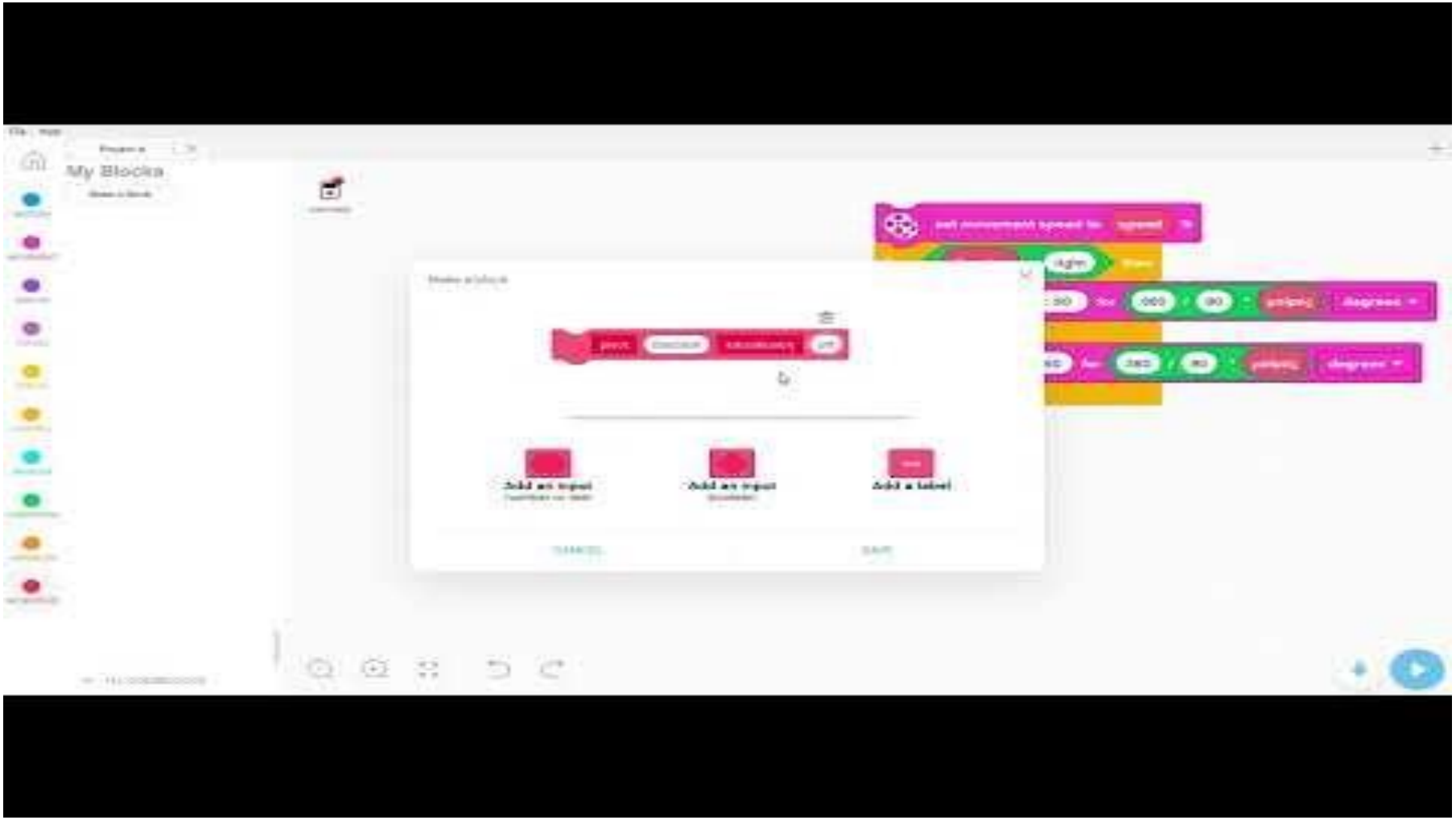
Βήμα 5c: Make and use pivot block

Το ίδιο block με ένα label για κάθε παράμετρο



Βήμα 5c: Make and use pivot block

Video: https://youtu.be/X9RNSGlf0I0



Βήμα 6: Ο κώδικας με blocks

Κινήσου ευθεία ως τη διασταύρωση

Pivot δεξιά 90

Follow line μέχρι διασταύρωση

Spin 180

Follow line till cross

Pivot δεξιά 90

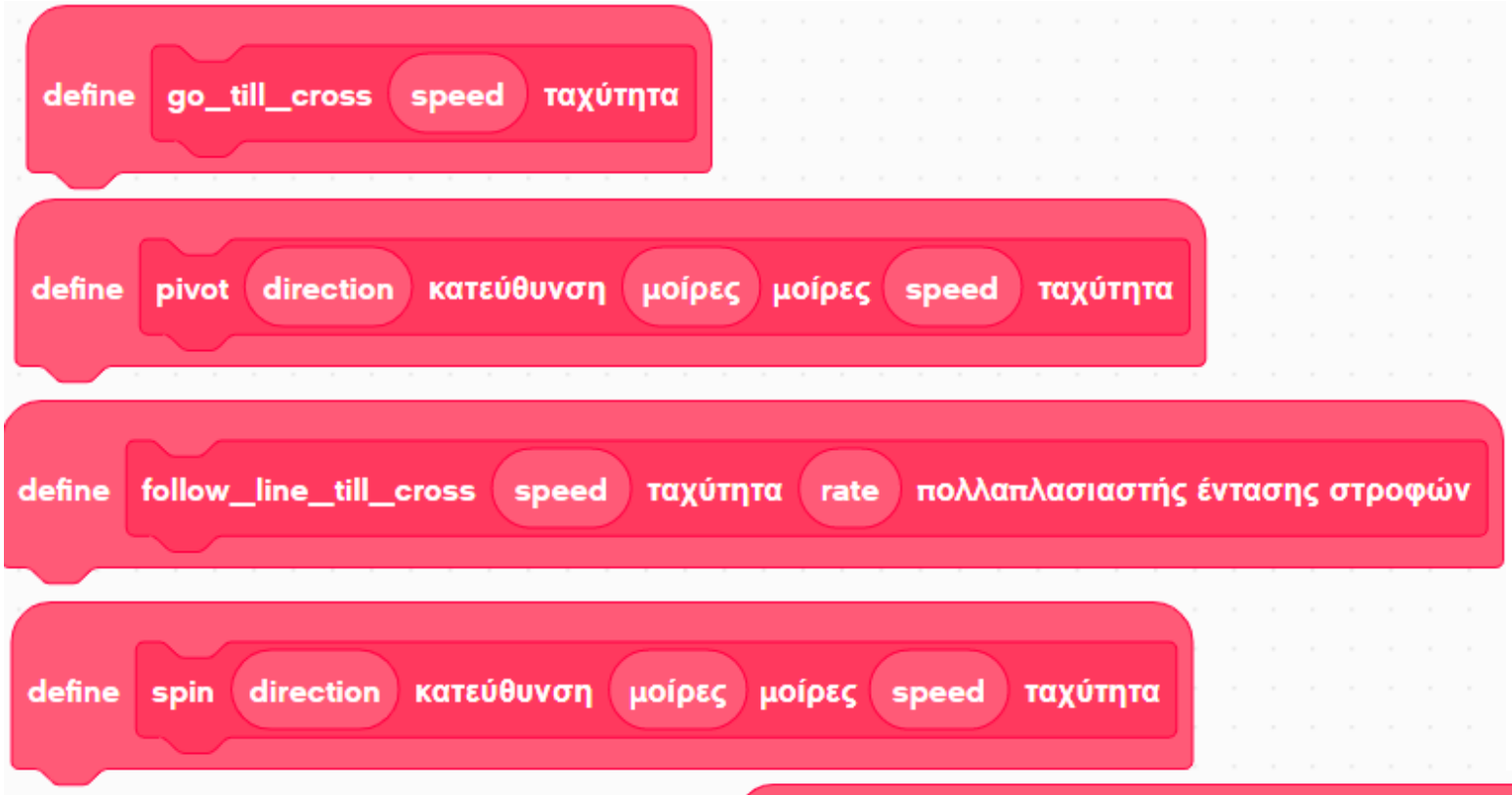
Follow line till cross

Pivot δεξιά 90

Follow line till cross

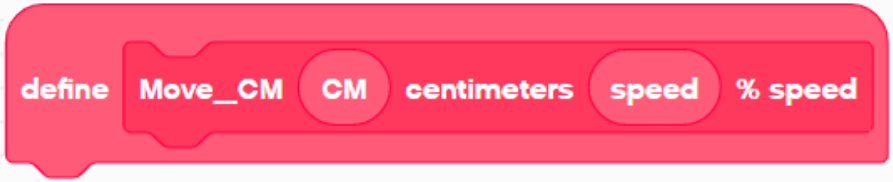
<u>Άσκηση</u>: Να φτιαχτούν τα αντίστοιχα blocks και να χρησιμοποιηθούν για την αποστολή

Χρήσιμα blocks:



Επίσης πολύ χρήσιμο το : →

Οδηγίες στις παρακάτω διαφάνειεσ



Lesson Objectives

1. Create a useful My Block
2. Learn why creating a My Block that takes measurements made with a ruler can be useful
3. Make a Move_CM My Block

Prerequisites: Moving Straight, Port View, My Blocks with Inputs and Outputs, Math Blocks, Variables

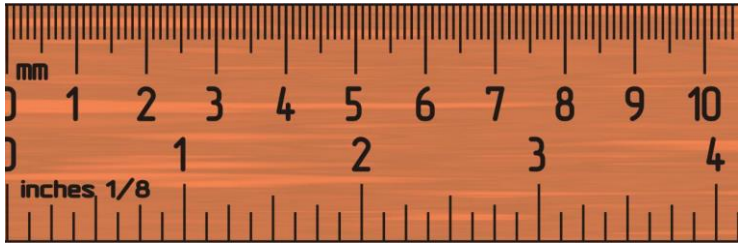
Why a Move Distance My Block?

Built-in move blocks will not take inputs (values) in centimeters or inches.

It is much easier to measure distance with a ruler than degrees or rotations.

If you change your robot design to have bigger or smaller wheels later on, you don't have to re-measure every movement of your robot

- Instead of changing distances in every single block you place, just go into your new Move Distance Block and change the value for how many inches/cm one motor rotation would take.



MOVE_CM IN THREE EASY STEPS

STEP 1: Determine how many motor degrees your robot moves in 1cm

STEP 2: Create a Move_CM My Block with 2 inputs – distance (CM) and speed (%)

STEP 3: Define the Move_CM My Block

Step 1: How Many Degrees Does The Robot Move in 1 CM?

Method 1:

1. Look up the wheel size in mm printed on your tire and divide by 10 to convert to cm (because 1cm=10mm)
2. Multiply the answer in step 1 by π (3.1415…) to compute circumference
3. Divide 360 degrees by value from step 2. This computes degrees in 1cm since you travel one circumference in 1 rotation and 1 rotation is 360 degrees

Example calculation using the standard EV3 Edu 45544 set wheels:

1. EV3 EDU (45544) wheels are 56mm = 5.6cm in diameter
2. 5.6cm × π = 17.6cm per rotation
3. 360 degrees ÷ 17.6cm = 20.5 motor degrees per cm

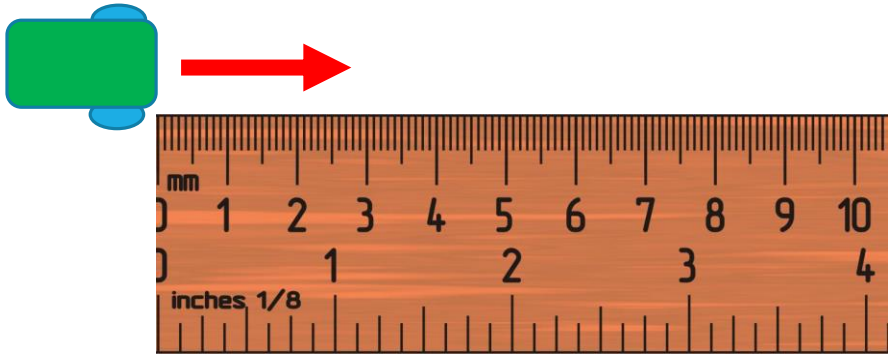
Helpful chart with common LEGO wheels and their diameters.

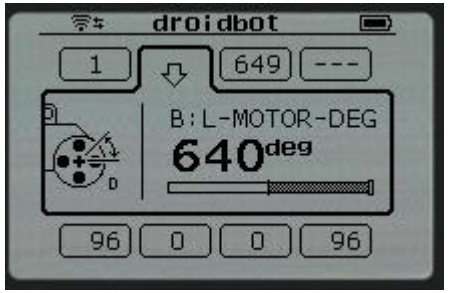
http://wheels.sariel.pl/

Step 1: Alternative Method

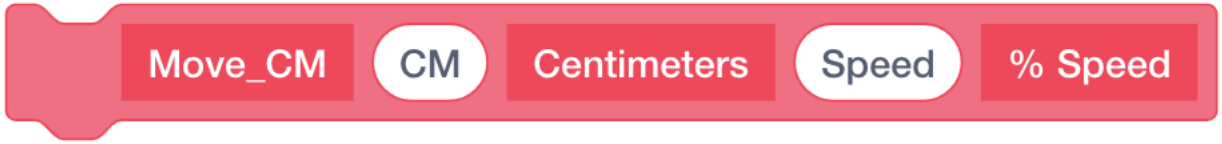
Alternate Method: Use Port View to find the Motor Degrees value. Use this method if you cannot find the diameter value printed on your wheel.

1. Put your ruler next to your wheel/robot at 0 centimeters (whatever part of the robot you use to align with 0, you should use to use to measure distance in step 2)
2. Roll your robot forward any amount of centimeters, making sure your robot does not slip.
3. Take the degree reading you see on the screen for the motor sensor and divide by the number of centimeters you moved (i.e. degrees measured/distance travelled)
4. The answer will be the number of degrees your robot's wheels turn in 1 centimeter.





Step 2: Create a My Block with 2 Inputs



Add an input
number or text

Add an input
boolean

Add a label

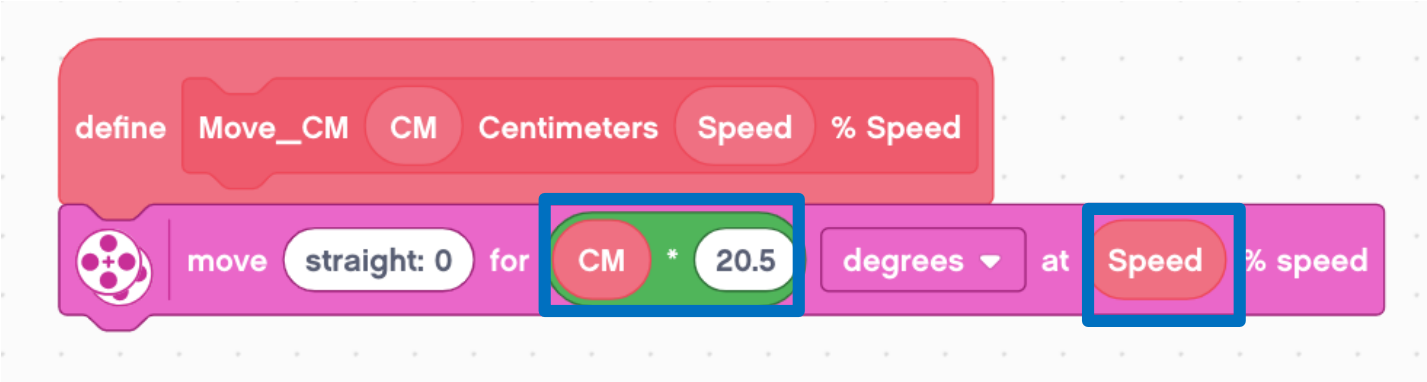
Step 3: Define the My Block

Use a Multiplication Math Block to Calculate the number of degrees the robot will move in 1CM

- Drag the CM input into the first parameter of the math block
- In the second parameter of the math block, enter the number of degrees your robot moves in 1 CM. (For Droidbot, this is 20.5)

Add a Moving Block under the define block

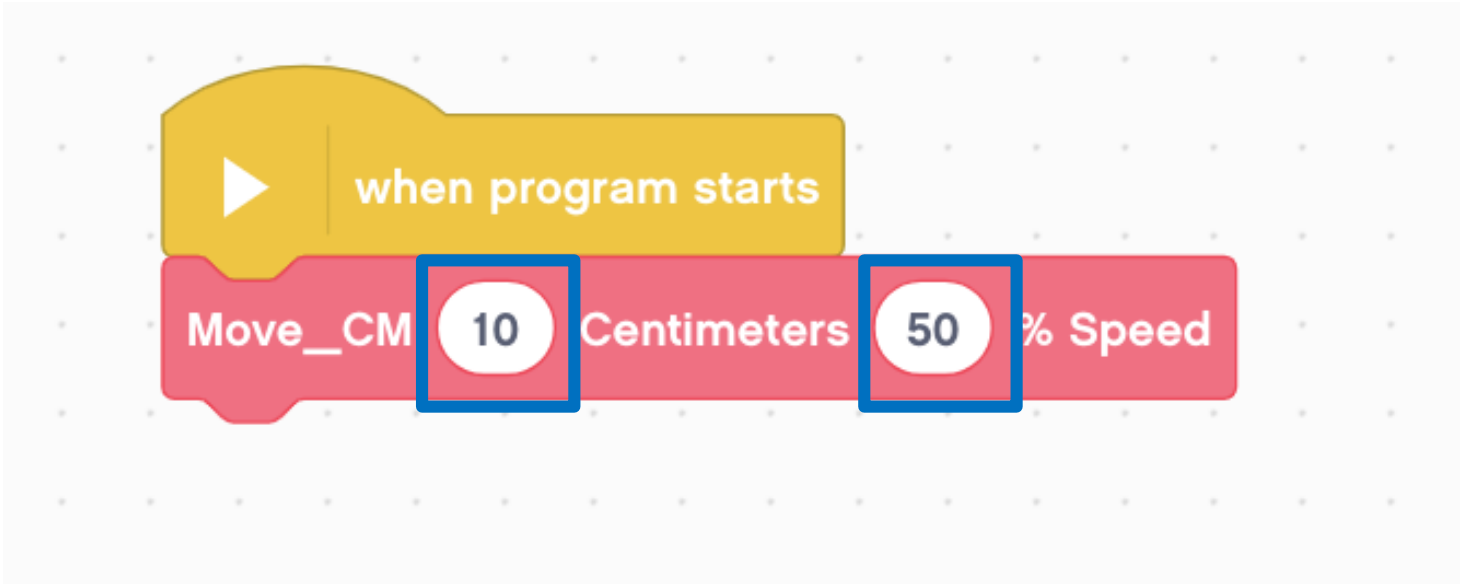
Place the Math Block in the distance parameter and the Speed input in the % Speed Parameter



Step 4: Use the My Block

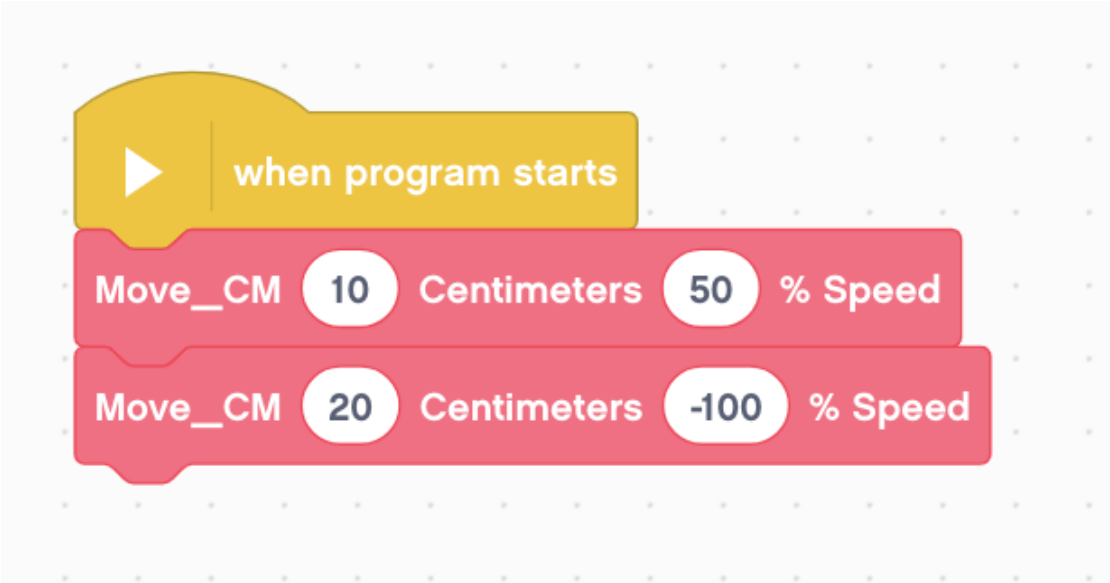
Now, when you drag the block into your programming canvas, you just need to enter the number of CM you want the robot to move and the speed it should move at.

In the example below, the robot will move 10CM at 50% speed



Reusable Move_CM Block

The same Move_CM My Block is used for two different moves. One moves forward 10cm at 50 % speed and the other moves backwards for 20cm at 100 speed. By changing the inputs, we can reuse the My Block.



Discussion

Why is a Move_CM My Block useful?

- You can measure distances in centimeters and input this number into your block instead of programming in degrees or rotations

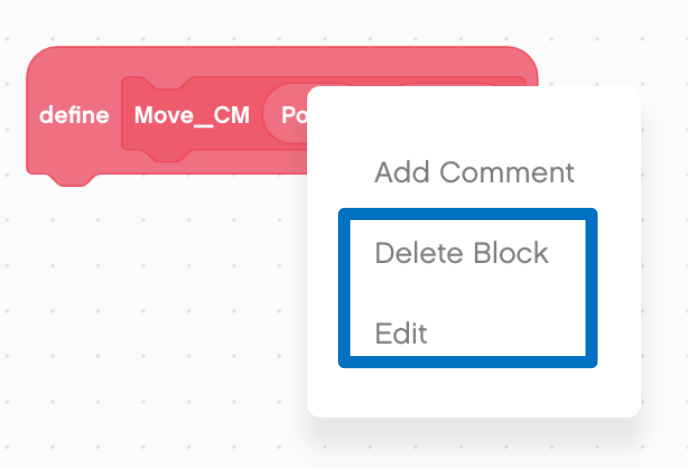
Will changing the inputs in one copy of Move_CM impact another copy of it?

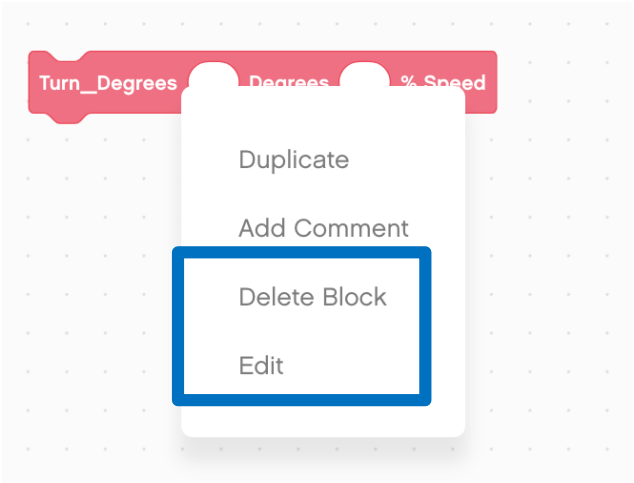
- No. That is exactly why a My Block is useful. You can use the same block multiple times, each time using a different number for power and centimeters (or any other parameter you set up).

Can you alter a My Block after it is made?

- Yes. Right Click on the My Block and click Edit.

How to Edit or Delete a My Block





- Right Click on a My Block in the Programming Canvas and select “Edit” to edit the My Block.
- This will take you back to the My Block creation screen where you can edit the name, add inputs, or delete inputs.
- To delete, you must first right click and press delete on all uses of the My Block in your program. Then, you can press delete on the definition of the My Block.

What Makes a Useful My Block

Note: Making My Blocks with inputs and outputs can make them far more useful. However, you need to be careful not to make the My Block too complicated.

Question: Look at the list of three My Blocks below. Which ones do you think are useful for to use?

- Move5CM (Moves the robot five centimeters)
- MoveCM with a centimeter and power input
- MoveCM with centimeter, power, angle, coast/brake, etc. inputs

Answer:

- Move5CM may be used often, but you will be forced to make other My Blocks for other distances. This will not be fixable later.
- MoveCM with centimeters and power as inputs is probably the best choice.
- MoveCM with centimeters, power, angle, coast/brake, etc. might be most customizable, but some of the inputs might never be used.

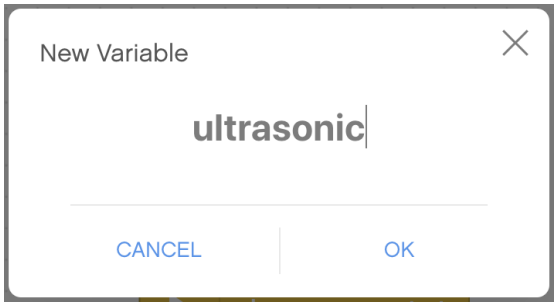
Adding Outputs

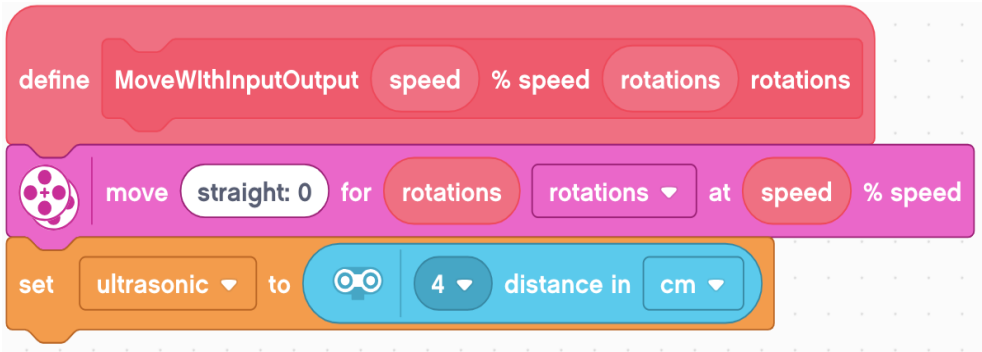
Unlike My Blocks in EV3-G, you cannot define Outputs. Here is a work-around.

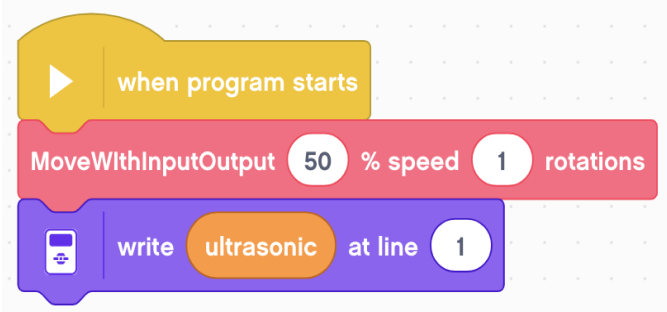
1. Define a variable to store the value of your output.
2. Write the data you want to input to the variable inside the My Block.
3. Use the variable in your main code

In the code on the right, the My Block reads the ultrasonic sensor, sets it to a variable.

The value can be used later in the program such as print to the screen.







Credits

This tutorial was created by Sanjay Seshan and Arvind Seshan

More lessons are available at www.ev3lessons.com