



# ROBOESL PROJECT

## ROBOTICS-BASED LEARNING INTERVENTIONS FOR PREVENTING SCHOOL FAILURE AND EARLY SCHOOL LEAVING

Erasmus+ 2015-1-IT02-KA201-015141

---

### Output 1: Curricula for 10 exemplary interdisciplinary robotics projects

#### Curriculum 9: Wall-e

---

**Lead Partner:** UNIPD (IT)

**Authors:** Michele Moro, Francesca Agatolio, Emanuele Menegatti (UNIPD)

#### **Contributions**

Dimitris Alimisis (EDUMOTIVA<sup>1</sup>) (Chapters 2 and 3)

Linda Daniela (University of Latvia) (Chapter 5)

**Circulation:** Public

**Version:** Final

**Stage:**

**Date:** March 31, 2017

---

<sup>1</sup> EDUMOTIVA stands for 'European Lab for Educational Technology

## Declaration

This report has been prepared in the context of the ROBOESL project. Where other published and unpublished source materials have been used, these have been acknowledged.

## Copyright

© Copyright 2015 - 2017 the ROBOESL Consortium

All rights reserved.

This document is licensed to the public under a **Creative Commons Attribution-NoDerivatives 4.0 International Public License**

## Funding Disclaimer

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

# Table of Contents

Chapter 1: Short description and scenario (O1.1) .....	3
1.1 The scenario .....	3
1.2 Connections with subjects .....	4
Chapter 2: Pedagogical objectives (O1.2) .....	5
2.1 General objectives .....	5
2.2 Specific objectives .....	5
Chapter 3: Suggestions for learning methodologies (O1.3) .....	6
Chapter 4: Technical guidelines (O1.4) .....	8
4.1 Building instructions .....	8
4.2 Illustrative solution .....	9
4.3 Implementation suggestions .....	11
4.4 Extensions and variants .....	12
4.4.1 Variant a: First scenario with a gripper [easy/medium] .....	12
4.4.2 Variant b: Second scenario with angles [medium] .....	13
4.4.3 Variant c: Second scenario with fixed distance [difficult] .....	15
4.4.4 Variant d: First scenario, a LF user block [easy] .....	16
Chapter 5: Evaluation tools (O1.5) .....	17

## Abstract

This document contains the description of the Curriculum n. 9, entitled ‘Wall-e’ which is part of the Intellectual Output 1 (Curricula for 10 exemplary interdisciplinary robotics projects) developed and tested in teacher training courses within the context of the ERASMUS+ ROBOESL project. With the excuse to refer to garbage collection, and more generally to environmental sensitiveness, in this Curriculum we introduce a smooth form of line following, based on feedback control, the use of an added gripper, more advanced detection techniques.

# Chapter 1: Short description and scenario (O1.1)



(source: lego.com)

Let us equip and instruct a robot to detect and move obstacles on its way.

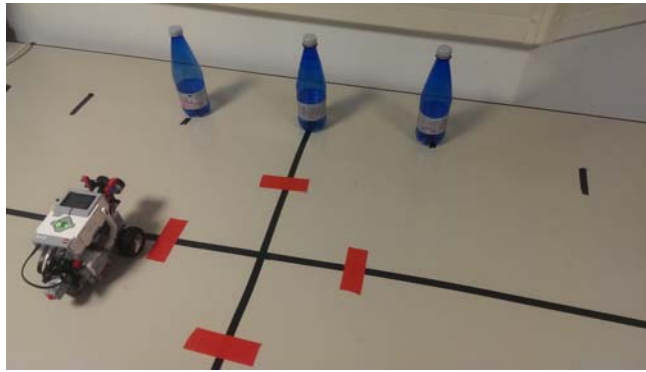
## 1.1 The scenario

In this curriculum we define two scenarios of different difficulty but with a common purpose: to make the robot recognize the presence of an obstacle and to move it in a different position. The first scenario is again based on a line follower but here we had two new challenges: to follow the line smoothly and, at the same time, to detect an obstacle possibly present on the way. Obstacles may be, for example, small bottles or cans which are easy to grip by the robot (fig. 1). In the second scenario some objects are put aligned in front of the robot at predefined distance; the robot has to move/turn in order to align its axis, and the ultrasonic sensor mounted in front of it, to each of the objects it has to reach and push ahead (fig. 2). In both scenarios a variant requires to add a motor to perform a more elaborated gripping feature.

These two scenarios recall the function of garbage collecting the robot Wall-e was in charge of daily doing in an abandoned, polluted earth (see <https://en.wikipedia.org/wiki/WALL-E>). The scenarios bring up real issues that the world, and especially densely populated areas, are dealing with today and even more so in the future. Americans, for example, produce nearly 400 million tons of solid waste per year but recycle less than a third of it, according to a recent Columbia University study. Landfills are filling up so quickly that the UK may run out of landfill space by the year 2017.



*Figure 1. First scenario*



*Figure 2. Second scenario*

## 1.2 Connections with subjects

1. Environmental Science: garbage collecting and disposing is today a big issue and it is connected to renewable energy production.
2. Physics: kinematics, motion profile.
3. Technology/Engineering: the feedback control principle.
4. Geometry/Trigonometry: (see variant c).

## Chapter 2: Pedagogical objectives (O1.2)

### 2.1 General objectives

- To show robotics relevance and usefulness to solve everyday life problems in a real-life context (waste management).
- To provide students with a stepwise approach for a step by step acquisition of technical skills in using robotic technologies (hardware and software) building on existing knowledge and skills.
- To offer the robotics benefits for all children, especially those at risk of school failure or early school leaving, putting robotics in real-life context.
- To engage students in STEM related subjects through interaction with the robotics technologies.
- To support self-directed action allowing learners to learn independently.
- To engage students in robotic constructions and problem solving through an interdisciplinary scenario that reflects aspects of real-life problems and situations.
- To align the robotics project to learners' needs and interests through tasks that derive from the initial activity but introduce new levels of complexity and difficulty.

### 2.2 Specific objectives

More specifically, upon successful implementation of the activities described in this curriculum students will achieve the following objectives:

- Develop environmental awareness for current global problems of waste management and consequent global catastrophic risks
- Adopt critique against consumerism recognizing the human environmental impact and relevant concerns
- Build and use a gripper, programming the action of a medium motor
- Design a programming solution for the robot to detect and approach obstacles along its way
- Program the robot to grasp the obstacles and move them aside
- Learn how to program a smooth line follower using the feedback P control principle (advanced objective)
- Learn how to relate spin angles and distances in the second scenario with aligned obstacles

## Chapter 3: Suggestions for learning methodologies (O1.3)

This curriculum follows the methodology introduced in the previous curricula. However, it is tailored to the scenario introduced in the current curriculum. A special worksheet has been designed as a reference and supporting tool for the students. The students are encouraged to work in groups. The teacher acts as a scaffolder and facilitator of the learning process. S/he provides feedback without revealing solutions and probing students through key questions to overcome emerging problems and difficulties.

The activity starts with the delivery of the first scenario to the students. The worksheet brings up real issues that the world, and especially densely populated areas, are dealing with today and even more so in the future and poses questions for the students to discuss such as: how can robots help with the waste management? Can you devise a robot to collect garbage? What hardware and software solutions are needed? The students discuss these questions first in their group and then in the plenary of the class.

The students are then challenged to program their robot to detect an obstacle along its path, concurrently during its motion, to grasp the object (garbage) and to move it aside when it is detected. They design and experiment with their own programming solutions.

During students' experimentations with this task, an improved version of the line following function is introduced by the teacher. The students are first encouraged to discuss how to obtain a better, smoother motion of the robot when following the white-black border. Discussing and working in groups helps to form a general methodology for dealing with the problem, which regards the control theory, an advance topic introduced here only in intuitive terms.

In the context of this activity the students are encouraged to tune the line-follower parameters in order to obtain the desired performance and to set up the 'garbage collecting' in order to suitably clear the followed path. The teacher can suggest the students to reflect on the ability of the robot to detect objects when the path contains curves (for example, in a closed ring) and then to directly experiment the effect.

In the end students' groups present in the plenary of the class their solutions and reflect on them with critical mind. Feedback from peers and teacher is provided with a constructive spirit.

Once the first scenario has been fully implemented and the goal reached, the teacher can move to the second scenario, where some objects are put in-line, in front of the robot and sufficiently spaced apart so that the closest sensed by the ultrasonic sensor is the one which is orthogonally aligned with the robot. This second scenario can be also be afforded in a variant that requires further insights on angles. The students are challenged to estimate, first experimentally and then using trigonometric formulas, the angles to be used to align correctly the robot with respect to these objects. This advanced scenario is introduced only if the maths background of the students guarantees the related knowledge.

### *A. The role of the students*

Students first discuss a scenario through a free dialogue in their group and after that they devise an action plan to realise it. They work in groups following their ideas and the discrete feedback they receive from the teacher. Students may extend their initial scenario devising further stories to play with. First, they find solutions making their own experimentations. Then they are supported to find additional solutions and realise further ideas. The final creations of the groups are presented in the

class, are discussed and evaluated with students reflecting with critical mind on their work, expressing their views and recording their experiences in a diary or questionnaire.

*B. The role of the teacher*

The teacher in this constructivist learning framework acts as an organizer, coordinator and facilitator of learning for students. S/he organizes the learning environment, raises the task for making robotics theatre through a worksheet, introduces software tools when necessary for students' work, discreetly helps where and when necessary, encourages students to work with creativity, imagination and independence and finally organizes the presentation and evaluation of the activity in the plenary of the class.

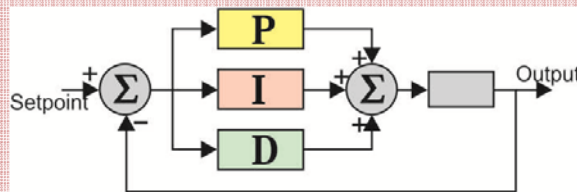
## Chapter 4: Technical guidelines (O1.4)

### 4.1 Building instructions

For the first scenario the usual *tribot* structure is still suitable but a static appendix must be added to the robot to facilitate the gripping of the object (fig. 1). It is not very important how this appendix is built, any solution able to move the object aside the path is acceptable.

In the actual gripping variants, we suggest to take the opportunity to add a further action and therefore a relatively complex substructure, equipped with a third medium motor, must be added (fig. 3). The purpose is to grip more stably the object in order to 'transport' it in a different, relatively far position. The experience must always be run having the grip fully open in a predefined initial position, then the students have to experimentally evaluate the angle to be performed by the motor to close the grip correctly, i.e. enough to capture the object but not too much to stress the motor.

*PID is an abbreviation for Proportional-Integral-Derivative and it is a specific control algorithm used in feedback controlled systems. This subject is too complex to be analytically dealt here but you can have an overall intuition looking at this figure (source: [http://www.pcbheaven.com/wikipages/PID\\_Theory/](http://www.pcbheaven.com/wikipages/PID_Theory/)):*



*The Setpoint is the expected value for the Output, and in general it can be not constant in time. It is compared (difference) with the current value measured from the output so that the resulting 'error' can be transformed into a counteraction trying to reduce it. Thus the aim of a controlling algorithm is to obtain an output that follows as closely as possible the variable Setpoint. In this algorithm the controlling signal, which is applied to the actuator (or, more generally, to a plant), is proportional to the value, to the derivative and to the integral in time, of the error. It can be showed that the most relevant component is usually the proportional one and therefore a P controller can obtain good results also without the other two components, particularly when the variation of the Setpoint are not very rapid. In our specific case, the Setpoint is constant being represented by the intermediate value of reflected light that corresponds to an intermediate position of the red light spot on the black tape border.*



Figure 3. Robot with a gripper

## 4.2 Illustrative solution

In the simplest solution of the line follower, the robot turns a bit to the left or to the right according to the level of reflected light sensed by the color sensor is greater or a less a given threshold. Fortunately, such a sensor gives more than just the condition to be greater or less than this threshold. Because the sensor illuminates the ground with a circular spot of red light, the amount of reflected light is higher or lower depending on the relative position of this circle with respects to the black stripe (fig. 4). Therefore, we have the possibility to set the steering factor proportionally to the measured reflected light, so that the farer the robot is from the border, the more it turns towards it. This type of reactive behavior is typical of a so called ‘feedback controlled’ system and the type of control tends to minimize the error between the expected value of a function and its actual value through an action which is proportional to this error (a more general algorithm would include a component proportional to the derivative of the error and a component proportional to the its integral, and for this reason called PID, see the box above). The effect is to obtain a smoother motion and a greater precision in following the border.

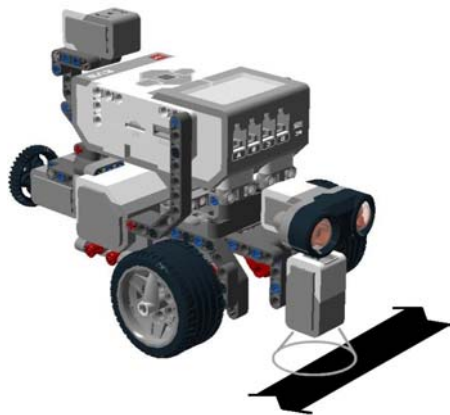


Figure 4. The illuminating light (source: legoengineering.com)

The simplest scenario is to put some obstacles along a straight line path made of a stripe of black tape (fig. 5). The overall program is given in fig. 6. The initial line following is provided by the first loop which ends when the robot, within that loop, detects with its ultrasonic sensor that an object is closer than a given threshold (12 cm in the example). The P control algorithm as a constant setpoint represented by the value in the `Level` variable, which is the intermediate value of reflected light as described above, while the `Factor` variable contains its constant of proportionality. Suggest the students to modify this couple of parameters to observe the effects.

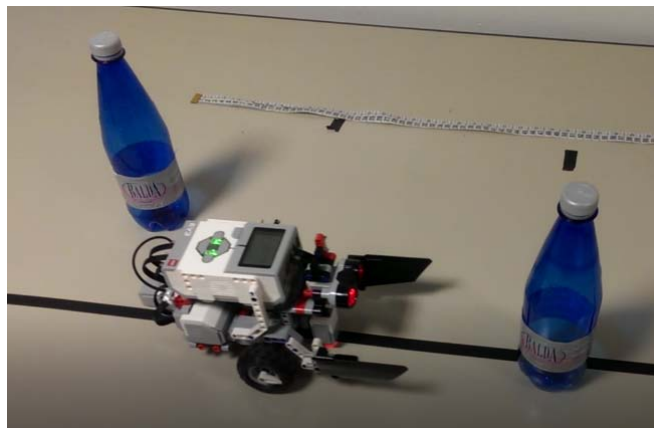


Figure 5. Simple garbage collector

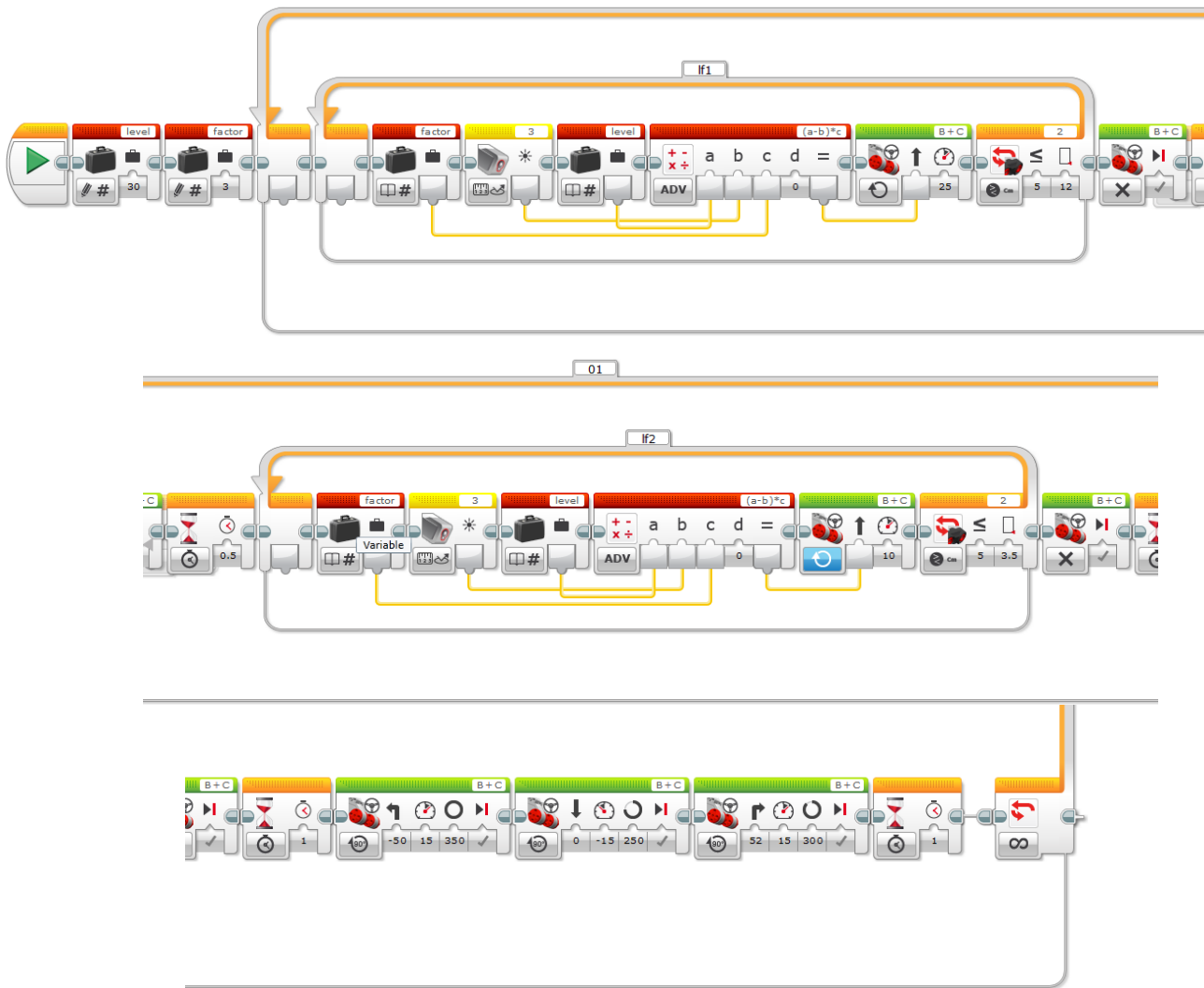


Figure 6. Simple garbage collector, the program

Once exited the first internal loop, a second loop is entered to make the robot approach the obstacle more slowly and its ‘arms’ embrace the object. Then the robot turns to put the object aside, retracts a bit and turns back to be able to realign with the black stripe, repeating the external loop. Constant parameters (speeds, distances) have been set experimentally to make the experience correct and acceptable. Added *Wait* commands help to have a clearer idea of the successive phases but they can be removed without problems.

It can be noticed that the section of program controlling the approaching of the obstacle and the following operation to remove it, represents the implementation of a specific motion profile. This is a very general principle which suggests first, to analyze the sequence of steps the robot must perform (you can ask the students to use again a state diagram to describe this sequence) and then, for every step, to specify the kind of motion requested in terms of speed, steering and ending condition(s).

In the second scenario, some objects are put in-line in front of the robot; this latter is initially put in front of the first object. The distance  $D$  between two adjacent objects and the number  $N$  of objects are known. The task consists in making the robot estimate the distance of the object in front of it, move ahead a bit more than this distance to reach and push the object down. Then the robot must go back and move parallel to the line where the objects are put to realign in front of the successive object and repeat the process (fig. 7).

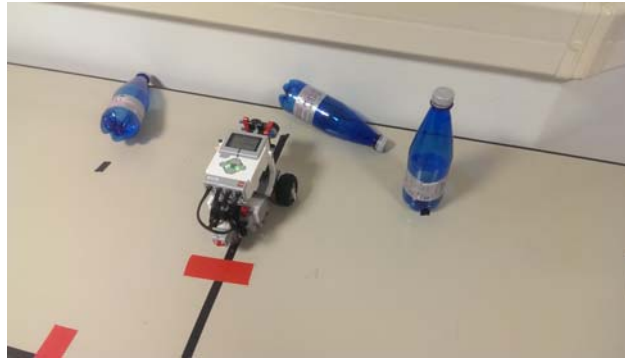


Figure 7. A second garbage collector

The task is similar to ‘Go to park’ but here the robot must measure the distance with respect to the closest object and move accordingly. The code is shown in fig. 8.

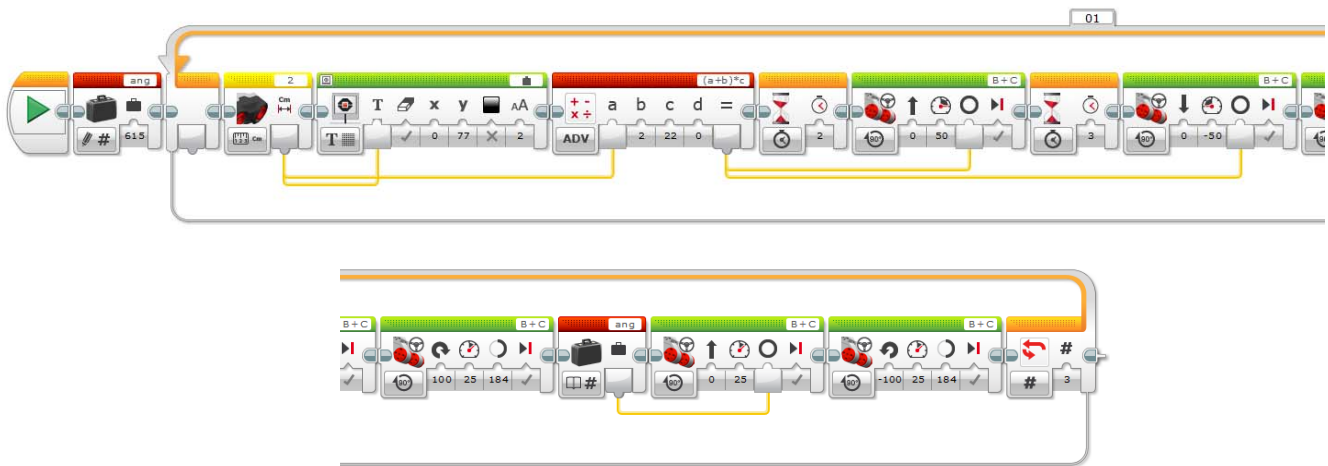


Figure 8. Second scenario

The **ang** variable stores the number of degrees to rotate both wheels to span the distance between two objects (this amount can be determined experimentally). The ultrasonic sensor is used to measure the distance to the object, also displayed on the brick screen. After a short wait, the robot moves for an amount of degrees (of the wheel) which is properly calculated correcting the distance with a certain offset (the **b** parameter of the **Math** command), experimentally estimated, to move beyond the position of the object. The calculated amount of space is then converted into an angle for the wheels as usual (through the **c** parameter of the **Math** command). Forward and backward motions are made with the same angle of wheel rotation, so that the robot comes back to its initial position. Then it spins clockwise 90 degrees, moves ahead the amount given by the **ang** variable so that the last spin of 90 degrees puts the robot in front of the following object, and the process can be repeated a total of N times. Notice that, with this solution, the code works also if the objects are not perfectly aligned because the measure of the distance is repeated every time.

### 4.3 Implementation suggestions

Again the setup is rather simple and poor materials (plastic glasses or bottles, or cans) can be used as garbage objects and obstacles. In the first scenario, a straight path simplify the experience but also other more elaborated paths may be experienced, having the foresight to put obstacles after a sufficiently long semi-straight portion to assure the detection by the ultrasonic sensor in front of the

robot. In the second scenario, as already highlighted, the success of the experience depends on a correct alignment robot-objects: it is suggested to use a (flexible) meter to help to set correct distances between objects according to the angles imposed in the program.

## 4.4 Extensions and variants

### 4.4.1 Variant a: First scenario with a gripper [easy/medium]

In this variant, we suggest to add a gripping system in the higher part of the front of the robot. This system is based on a medium axial motor which is provided by the EV3 kit for purposes of this kind (fig. 9). The grip is opened and closed by an endless screw, an interesting mechanical component that, connected with a cogwheel, can transmit the rotation of one axis to the rotation of a usually orthogonal axis with reduction and irreversibility (it is almost impossible to act on the cogwheel to transmit a (multiplied) rotation to the screw, see fig. 10).

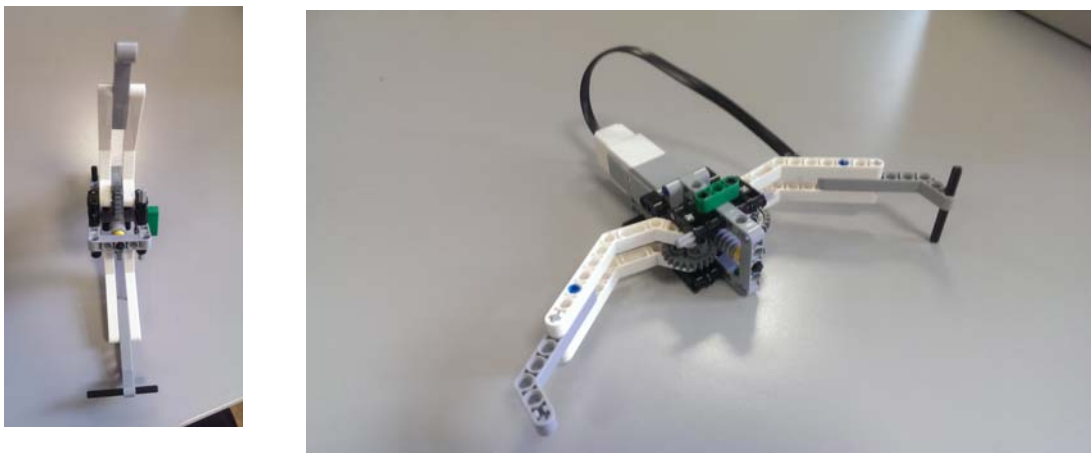


Figure 9. The gripper



Figure 10. Endless screw and cogwheel (source: atti.it)

With respect to the first example, in this variant the robot grips the obstacle and moves it farther, for example beyond a certain distance on one side (fig. 11). The program is showed in fig. 12, apart from the first part which is equal to the first part, including the two internal *lf1* and *lf2* loop, of fig. 6 but respectively with distance thresholds 15 and 6 cm. After having approached enough the obstacle, the robot acts on the medium motor, assumed connected to port *A*, to close the gripper around the object. Then the robot turns a bit on one side, moves ahead until it reaches a certain distance, then it relinquishes the garbage opening the gripper, comes back, and realigns itself for travelling towards the next obstacle.

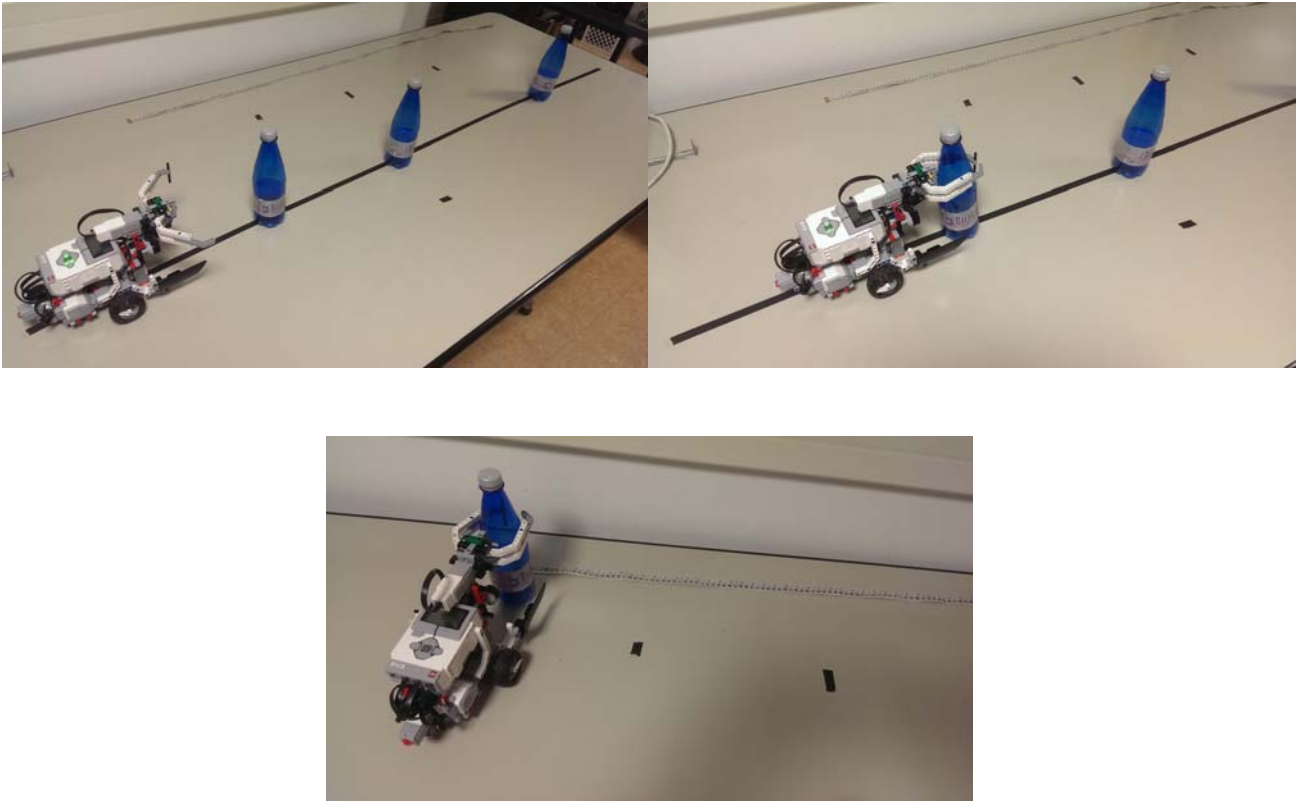


Figure 11. A third garbage collector with gripper

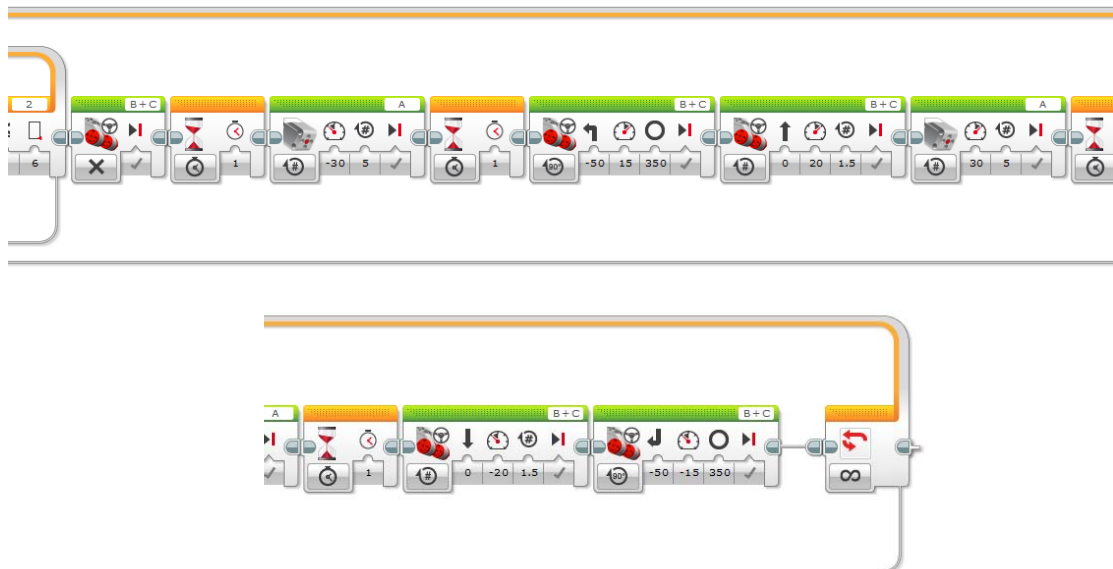


Figure 12. A third garbage collector with gripper, the program

#### 4.4.2 Variant b: Second scenario with angles [medium]

In this variant of the second scenario, three objects are put in-line in front of the robot, the middle one aligned with the initial axis of the robot, the other two on each side at a predefined D distance (fig. 13).

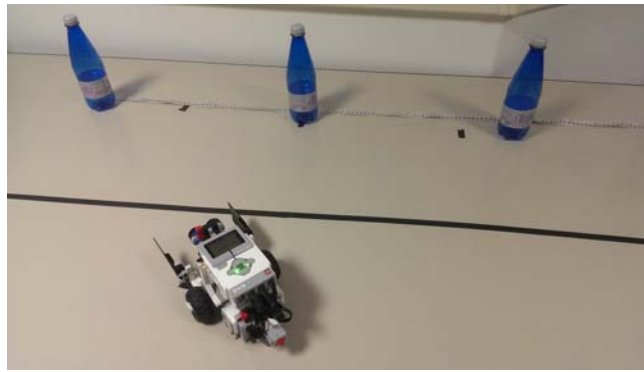


Figure 13. A second garbage collector

Aim of this variant is the same of the basic one, to reach and push ahead each object after having tried to align the robot with the target object from the initial position, and measured the distance to the object. The difference here is that the robot is requested to spin a correct amount of degrees to align from the same initial position. Fig. 14 shows the proposed program: for simplicity, the angle of rotation of the wheels for making the robot aligned with the object on the left, is set initially to the `ang` variable. So the first spinning in the main loop turns the robot counterclockwise for an angle which could be estimated using the geometric model we introduced in the third curriculum. Once established this angle, simple trigonometry would give you the distance of the object on the left from the object in the middle: all this calculations can be substituted by executing a first time the program with some initial (negative) value and then putting manually the object in an acceptably aligned position.

Then the robot measures its distance to the object, shows the value on the display and, after a short wait, moves straight towards the object and pushes it down. Forward and backward motions are made with the same angle of wheel rotation, so that the robot comes back to its initial position and, through another spin turn, to its initial direction. The next cycle is executed after having updated the `ang` variable: the amount added is such that, with three cycles, the performed rotations are -100, 0, +100. Ideally, changing the initial value in the `ang` variable, the updating amount and, consequently, the number of cycles, you could make the robot reach 5, 7, 9 or more objects, but unfortunately, when the objects are rather close one another, the ultrasonic sensor can no more discriminate two successive objects, and fails to correctly measure the distance with the target object, producing incorrect results. Again the waiting commands are added for a better understanding of the various phases and can be removed.

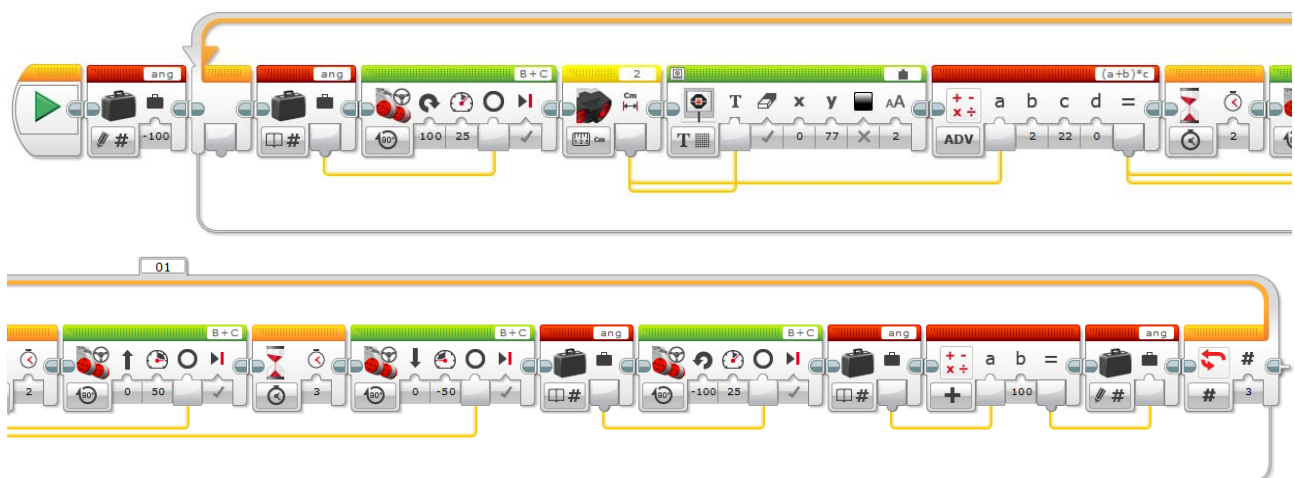


Figure 14. Second variant of the garbage collector, the program

#### 4.4.3 Variant c: Second scenario with fixed distance [difficult]

This variant requires some trigonometry. As shown in fig. 15, we assume to have an odd number of objects in front of the robot, the middle one aligned with its initial direction, put at regular distance  $d=30$  cm (in the example).

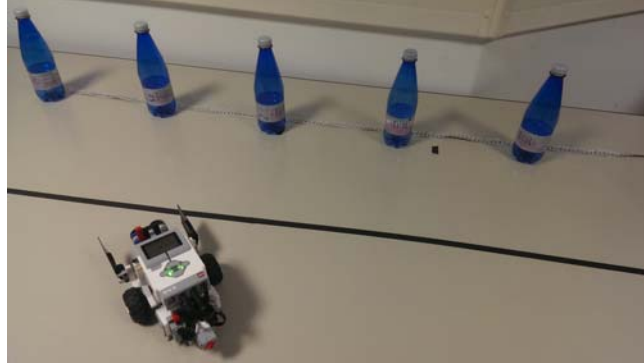


Figure 15. A fourth garbage collector

In order to understand how the travel parameters, angle of spin from the initial direction and distance, are calculated, see fig. 16.

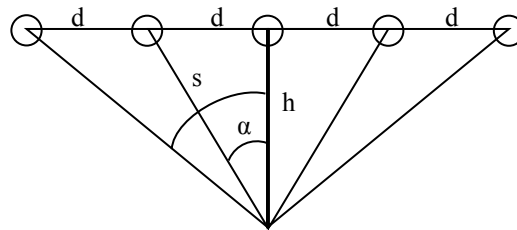


Figure 16. The trigonometric model

The formulas to be used are the following:

$$\alpha = \text{atan}(k \cdot d / h) \quad k = -2, -1, 0, 1, 2 \text{ with 5 objects}$$

$$s = \sqrt{h^2 + (k \cdot d)^2}$$

The distance  $h$  is not known but initially measured by the robot which is aligned with the object in the middle. This in theory: practically some corrections must be applied. One regards the fact that usually there is an offset between the position of the ultrasonic sensor and the center of spinning, which is the ideal starting point of the robot. So the measure of  $h$  must be augmented by this offset to be applied in the formulas. Moreover, for the sake of simplicity, in this experiment we want to make the robot only approach an object without touching it, thus  $s$  must be suitably reduced.

The overall program is showed in fig. 17: observe first that the `pos` variable contains the current relative position of the object to be reached in the main loop, and it is initialized to  $-2 \cdot 30 = -60$  because the total number of objects in the example is 5. The measure of  $h$  is then augmented by 6 which is the experimentally estimated offset.

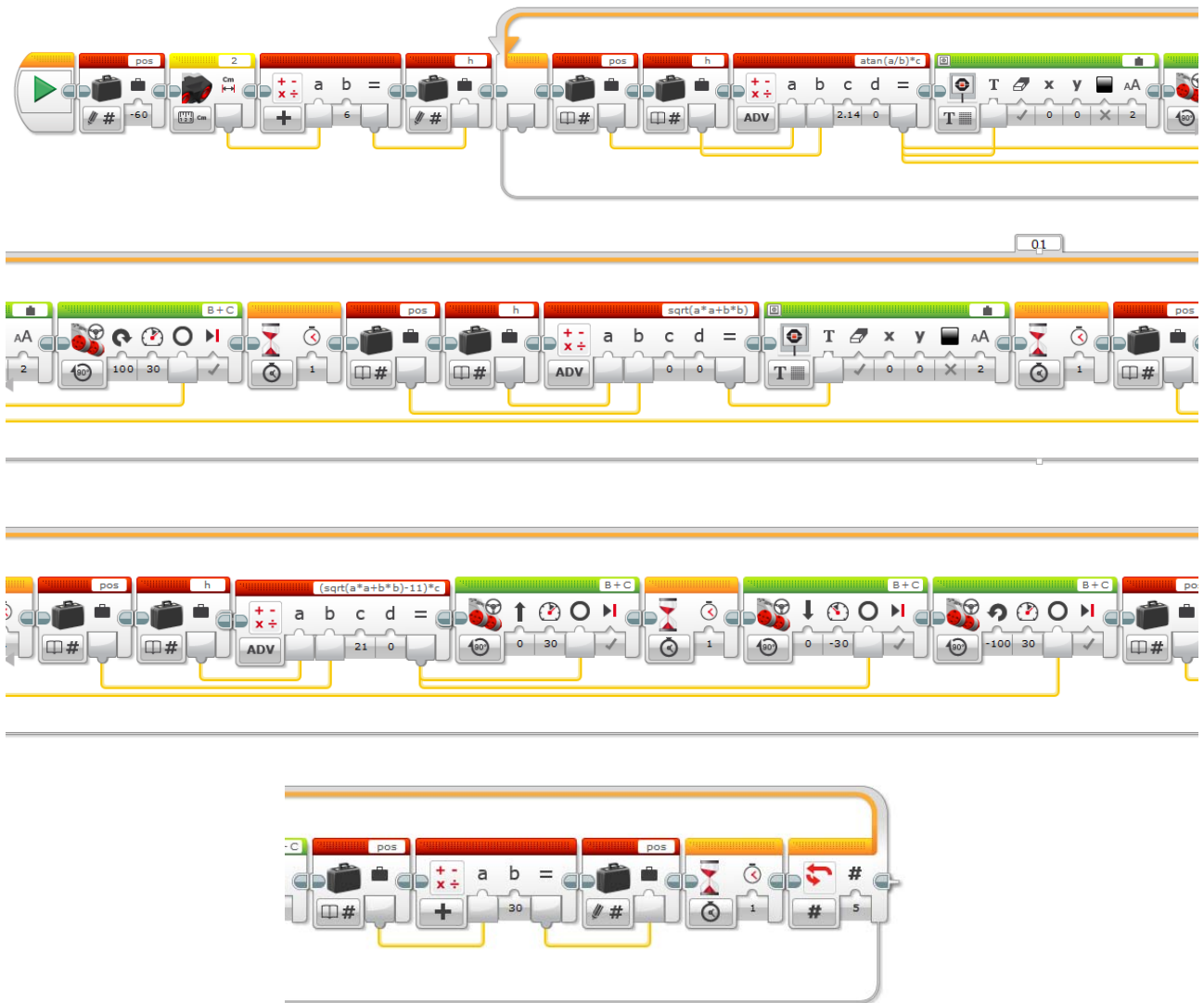


Figure 17. A fourth garbage collector, the program

In the main loop we calculate the spinning angle, transformed into a wheel rotational angle multiplying it by the 2.14 magic number. Notice that during the first cycle this amount is negative and therefore the robot actually turns counterclockwise. For diagnostic purposes the theoretical value of  $s$  is calculated and displayed but the value used for travelling towards the object is diminished by a certain amount of cm (11 in the example) as explained above. 21 is the magic number to transform travel space into wheel rotations measured in degrees. These two motions are reversely performed to put the robot again in its initial position. Finally the  $pos$  variable is updated adding the  $d$  amount. Notice that there is a strict relationship between the initial value of  $pos$  and the constant put in this last **Math** adding command: if  $n$  is the (odd) number of objects they must be respectively  $-int(n/2) \cdot d$  and  $d$ .

#### 4.4.4 Variant d: First scenario, a LF user block [easy]

Modify the program of fig. 6 encapsulating the line follower loop in a user block called *LF*. Maintain the meaning of the two *level* and *factor* variable and define the motor power and the ultrasonic sensor threshold as block parameters.

## Chapter 5: Evaluation tools (O1.5)

Use the rubric below to evaluate your students' achievement in each specific objective of this curriculum.

Name of student (or group of students): .....

<b>upon completion of the activities described in this curriculum students achieved the following objectives</b>	<b>Evaluation score</b> <b>0 = not attempted</b> <b>1 = attempted without success</b> <b>2 = partial success</b> <b>3 = completed with teacher's help</b> <b>4 = completed without teacher's help</b>
Mounted and used correctly the ultrasonic sensor	
Mounted the tribot extension (gripper) to 'embrace' objects	
Prepared carefully the first scenario	
Instructed the robot to smoothly follow the line by using the P feedback control principle	
Calibrate critical parameters	
Sensed the presence of an obstacle using the ultrasonic sensor	
Instructed the robot to approach the obstacle and to move it aside	
Prepared carefully the second scenario	
Instructed the robot to spin and align with each object to be reached	
Complete the task by pushing each object ahead	
Mounted correctly the gripping system	
Instructed the robot to approach the obstacle and to grip it	
Instructed the robot to move the gripped object far away	

