TangIn

Tangible Programming & Inclusion

TangIn Toolbox

Connecting Dots

8-12+ years old

Regularities and Patterns | Itineraries

Probotic | Loops

www.tangin.eu

/tanginproject

Co-funded by the Erasmus+ Programme of the European Union

This project has been funded with support from the European Commission. This publication [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. Project N°.: 2017-1-PT01-KA201-035975

Summary

Introduction of the *Loop* functions with increasing levels of complexity, in geometric contexts.

Expected duration: **multiple of 10 or 15 min** (the lesson plan duration is flexible, and teachers can adapt them accordingly to their needs and class duration).

Learning Outcomes

At the end of the session, students are expected to:

- Identify regularities in routes;
- Apply loop functions to solve problems;
- Program the robot adequately, taking advantage of the looping tool;
- Value STEM areas;
- Develop transversal competencies such as problem-solving, communication and reasoning;
- Develop group work skills, namely, to respect and favor the inclusion of all elements, regardless of gender, culture, etc.

Links With Curriculum Topics

Covered Curriculum Topics		
Subject		**Topics**
Engineering	**Mathematics**	Geometry • Localization and orientation - itineraries • Properties of geometric figures Algebra • Regularities and patterns • Pythagorean theorem (optional)
	Technology	Programming • Concepts of programming • Programs – Results, errors, and troubleshooting • Loops Robotics • Programming objects to solve challenges

Notes for Teachers

The teacher should prepare, in advance, all the materials needed and the classroom according to the activities to be developed.

The teams should be as heterogeneous as possible to foster the integration of all students. It's important that clear rules are established in terms of group work. This way, it avoids the most active children assuming the lead and the quitter ones only observing.

This lesson plan was thought as an introduction and application of loops for all group ages and can (should) be done partially in multiple sessions according to the students' evolution. In this plan, different levels of complexity are presented. In the same level, students can practice the same shapes but with different areas and lengths. It can be used resumed at different times in a way to assess students' progression after a few sessions.

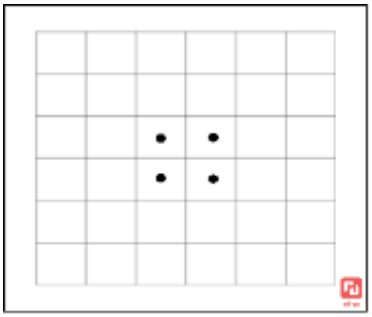
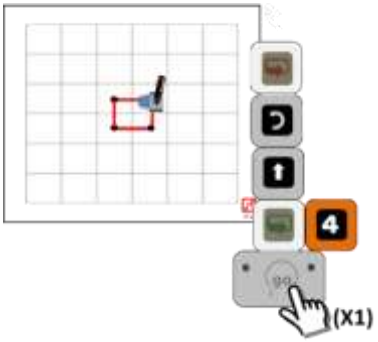
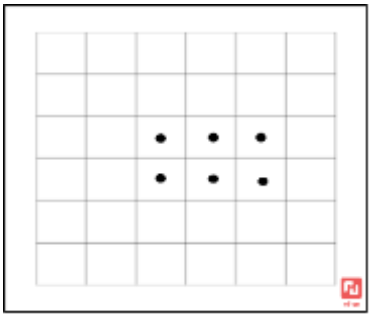
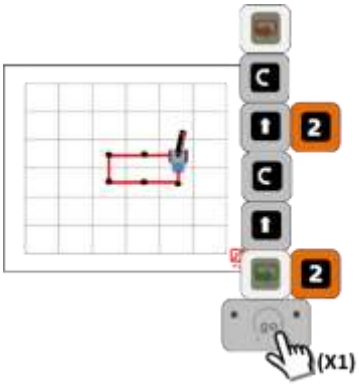
Alternatively, and according to the level of the students, after the first example/exercise is done, and for all the following tasks:

- students record their programming on paper. Subsequently, a discussion among all groups should be promoted. One of the groups that presented the correct programming (in turn) exemplifies with the robot;
- the teacher can distribute one of the different tasks per group. One group at a time presents his resolution to the class. It is discussed and corrected (if necessary).

The teacher must circulate through the various groups to support the activities and the dynamics of each one. In the end, it should promote a collective discussion of the main issues focused and the constraints and difficulties experienced.

Lesson Plan

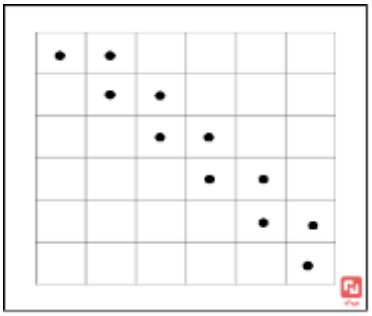
Intro	10'	Class	Introduce the loop blocks and the concept of repetition. Give examples of sequences with repeated patterns (example 4x forward). Ask them to write down the code of a figure (e.g.: square) and see if there are repeated sequences. The teacher will have a Set and be the only one with a BOT (and all the blocks) to confirm/give the solutions. The students can use the remaining *Sets* to draw on them.	 

Play	10'| 15' (each task)	Class	**Squares** 1) Draw little dots in the center of the grid squares as shown in the image. Ask how can the BOT cross over all of them in one go and with the minimum blocks possible. 2) Let them think, discuss and write down the solution. Confirm the solution by executing the code in the image. 3) Try another example. If this level is still not completely mastered, continue with more examples of different sizes of squares.	1)  2) 
			Rectangles (not squares) 1) Draw little dots in the center of the grid squares as shown in the image. Ask how can the BOT cross over all of them in one go and with the minimum blocks possible. 2) Let them think, discuss and write the solution. Confirm the solution by executing the code in the image 3) Try another example. If this level is still not completely mastered continue with more examples of different sizes of rectangles.	1)  2) 

Zigzag

1) Draw little dots in the center of the grid squares as shown in the image. Ask how can the BOT cross over all of them in one go and with the minimum blocks possible.

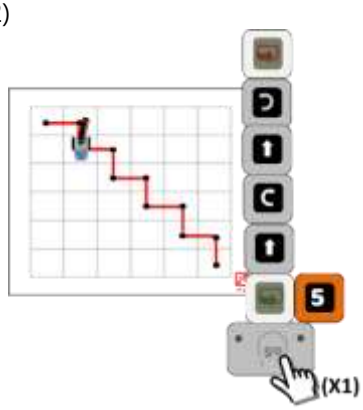
1)



2) Let them think, discuss and write the solution. Confirm the solution by executing the code in the image

3) Try another example. If this level is still not completely mastered continue with more examples of different paths and lengths in zigzag.

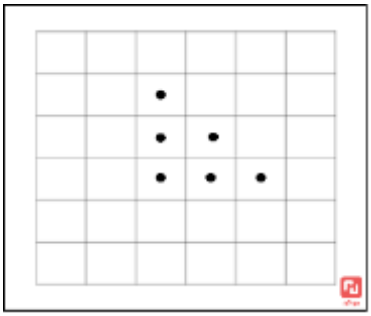
2)



Right Triangle (angles)

1) Draw little dots in the center of the grid squares as shown in the image. Ask how can the BOT cross over all of them in one go and with the minimum blocks possible.

1)

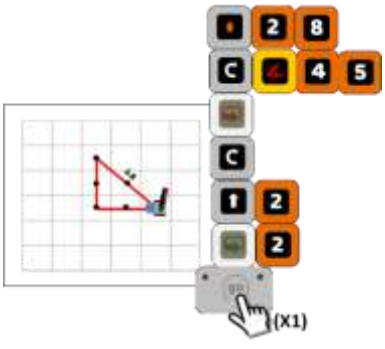


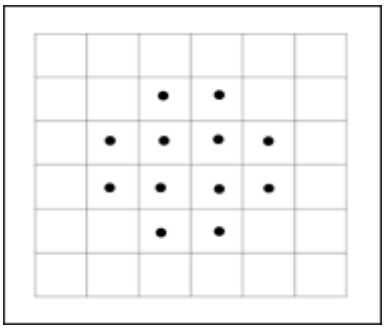
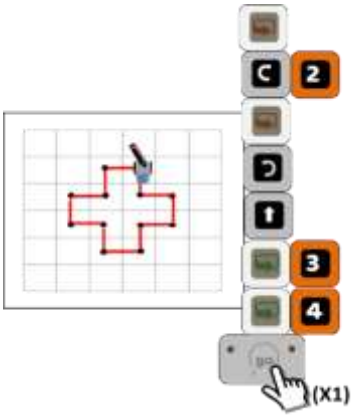
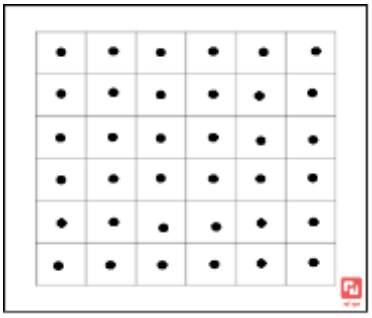
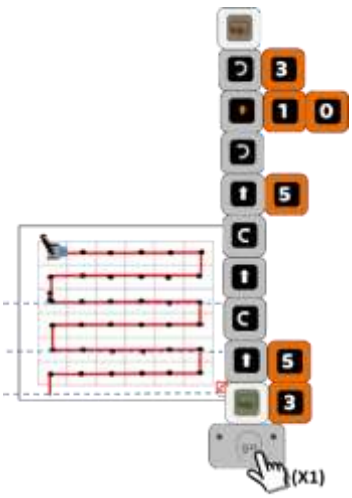
2) Let them think, discuss and write the solution. Confirm the solution by executing the code in the image.

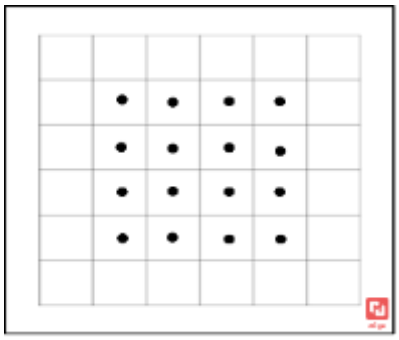
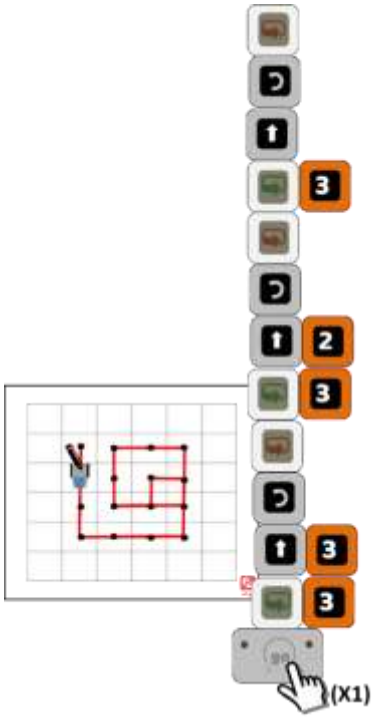
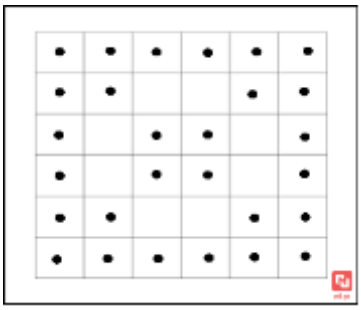
3) Try another example. If this level is still not completely mastered continue, with more examples of different sizes of right triangles.

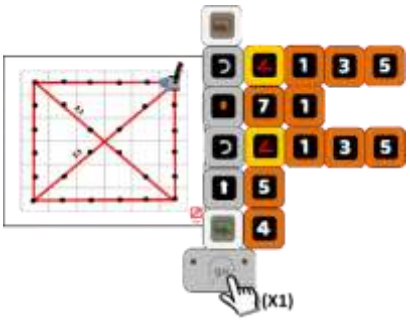
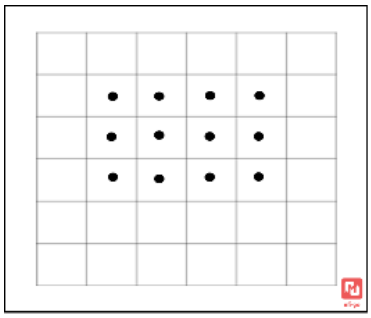
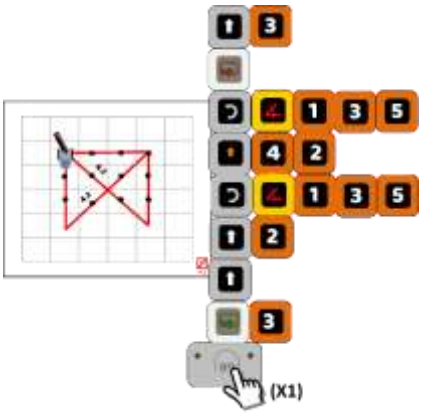
Note: to find the hypotenuse measure, one can use a ruler to directly measure the distance between the points or calculate using the Pythagoras Theorem.

2)



			Cross (*loop* inside the *loop*) 1) Draw little dots in the center of the grid squares as shown in the image. Ask how can the BOT cross over all of them in one go and with the minimum blocks possible.	1) 
			2) Let them think, discuss and write the solution. Confirm the solution by executing the code in the image. 3) Try another example. If this level is still not completely mastered, continue with more examples of different sizes of crosses.	2) 
			Full *Set* (symmetry, find the smallest fragment of repeated sequence) 1) Draw little dots in the center of the grid squares as shown in the image. Ask how can the BOT cross over all of them in one go and with the minimum blocks possible.	1) 
			2) Let them think, discuss and write the solution. Confirm the solution by executing the code in the image. 3) Try repeat one fragment at the time. Try reproducing in a smaller area with fewer dots. Tip: tell students to write on paper all the bot moves to connect the dots and try to see if it is composed of identical fragments.	2) 

			Serpentine (succeeding loops) 1) Draw little dots in the center of the grid squares as shown in the image. Ask how can the BOT cross over all of them in one go and with the minimum blocks possible.	1) 
			2) Let them think, discuss and write the solution. Confirm the solution by executing the code in the image. 3) Try another example. If this level is still not completely mastered continue with one loop at a time, ask what happens next.	2) 
			Envelope (angles & intersection) 1) Draw little dots in the center of the grid squares as shown in the image. Ask how could the BOT cross over all of them in one go and with the minimum blocks possible.	1) 

			2) Let them think, discuss and write the solution. Confirm the solution by executing the code in the image. 3) Try another example. If this level is still not completely mastered, try to one segment at a time and let them measure lengths and angles. Note: to find the Hypotenuse measure length one can use a ruler to directly measure the distance between the points or calculate using the Pythagoras Theorem.	2) 
			Think **out of the box**! 1) Connect all the dots but this time only with 5 straight lines in total and only passing one time per dot. 2) Mission complete! No more levels! Everyone is now a Loop Master! Still, many more examples can be done, but now you can create your own!	1)  2) 

Resources List & Support Material

For the teacher or per each group:

- A robot Kit with drawing capabilities;
- Black markers for each group (easy to erase/clean);
- Alcohol for cleaning the scenarios (for teacher use only);
- Transparent scenario with a 6x6 grid; 2X Shape cards (Annex).