



# TanglIn

**Tangible Programming & Inclusion**

## TanglIn Toolbox Intro: Programming

6 -10 years old

Computational Thinking

Robotics and Algorithms

Itineraries



[www.tangin.eu](http://www.tangin.eu)



/tanginproject



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 to Computational Thinking, Programming, and Robotics by using commands and role play dynamics. Simulate inputs and outputs and predict outcomes. Give examples of programming and algorithms in everyday life.

Expected duration: **50 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:

- Break a task in simpler steps and commands;
- Plan routes and predict linear movements and rotations in space in the other's perspective;
- Give examples of a few programming principles and recognize the importance of giving clear and precise instructions and commands;
- Develop interest and commitment by using role-playing dynamics;
- Develop spatial awareness and rationalize from the perspective of others;
- Develop communication and team working 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
<b>Engineering</b>	<b>Mathematics</b> Geometry <ul style="list-style-type: none"> <li>• Length Measure</li> <li>• Rotation (quarter turn)</li> <li>• Itineraries</li> </ul>
	<b>Science</b> <ul style="list-style-type: none"> <li>• Uniform movement</li> </ul>
	<b>Technology</b> Algorithmics <ul style="list-style-type: none"> <li>• What are they and what're their practical applications?</li> </ul> Programming <ul style="list-style-type: none"> <li>• Programs with different complexity levels for the solution of specific problems</li> </ul>



## 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 quieter ones only observing.

In this initial stage, it is essential to explore the answers of the students and use them as a starting point to move forward into the topic







Try to emphasize not only the practical side of programming (developing computers, apps, etc.) but also the fun of teaching programs and what bots(robots) can do.

And the most important message, to stress that programming is not a complicated thing and that anyone can be a programmer.









For students that are already familiar with the concepts if programming and laterality it is possible to integrate this lesson plan with the "Introducing MI-GO" lesson plan. Joining both lesson plans can be a solution for adapting the lesson to the class level.

## Lesson Plan








---

				
Intro	10'	Class	<p>Discuss with the class the meaning of Programming and Robotics.</p> <p>Explore the answers until the concept has been understood.</p>	



				
			<p>Try to ask for examples of real-life use of programming and robotics.</p> <p>Example to use can be: Developing apps and computer software. The internet. Manufacturing robots in the car industry. Flying planes (autopilot) etc.</p>	
			<p>Tease the class to think about language as programming and of the alphabet as the code</p>	
Prep	15'	Class	<p>Give students the challenge to program a "human robot" in making specific itineraries in the classroom, using simple commands such as forward (and how many steps) turn right and left.</p> <p>Discuss/remember the concept of length measure, uniform movement, rotation, and itineraries.</p> <p>Use the board to exemplify to the class.</p> <p>Organize the class in pairs.</p>	



				
Play	15'	pairs	<p>Taking turns and not all at the same time, one student will assume the role of a programmer and the other of the robot.</p> <p>The programmer writes the instructions in the paper using arrows and numbers and gives it to the robot to execute it faithfully (he/she codes an instruction).</p> <p>Example of commands: try to go from the door to the board and/or going around a table/chair</p> <p>Note: they should try to be the most precise and reproducible possible in following the instructions</p>	
Review	10'	Class	<p>Students with the help of the teacher will correct if necessary, their colleagues (the robots) during the execution of the code.</p> <p>Together, discuss the activities made and the challenges faced.</p>	

## Resources List & Support Material

### Per each group:

- Pencil and paper.

