

Introduction to Ozobot and Colour Codes

Year level band: F-2 (can also be adapted for 3-4 and 5-6 as a first lesson with Ozobot)

Description: Students are introduced to Ozobot and how drawing lines and colour codes can control it. This lesson allows students to experiment with different lines and codes to create a path for Ozobot to follow.

Resources:

- Ozobots
- Blank white paper
- Markers/textas in colours black, red, light blue and light green (recommended: Ozobot pens, Sharpie wide chisel tip or Crayola markers), one set per group
- Colour codes downloaded and printed (<http://files.ozobot.com/stem-education/ozobot-ozocodes-reference.pdf>)
- Ozobot activity sheets (pages 8-10 of <https://storage.googleapis.com/ozobot-lesson-library/programming-with-colors/programming-with-colors.pdf>)

(Instead of paper, you can also code Ozobot by drawing on tablets, using apps such as Explain Everything or OneNote.)

Prior Student Learning:

Maths: Students have done some work on navigational language (left, right, forward, backward).

Digital Technologies: It may be that students have done some prior unplugged algorithms (simple following and providing instructions).

By the end of Year 2, students will have had opportunities to create a range of digital solutions through guided play and integrated learning, such as using robotic toys to navigate a map.

Students use the concept of abstraction when defining problems, to identify the most important information, such as the significant steps involved in navigating a robot. They begin to develop their design skills by conceptualising algorithms as a sequence of steps for carrying out instructions, such as identifying steps in a process or controlling robotic devices. Students are able to use data as an input for their robotic device.

Year	Content Descriptors
F-2	Follow, describe and represent a sequence of steps and decisions (algorithms) needed to solve simple problems (ACTDIP004)



Element	Summary of tasks
Learning hook	<p>Introduce Ozobot and explain how the robot works.</p> <p>With a black marker, draw a line and show how Ozobot follows it.</p> <p>Repeat for other colours.</p> <p>Show Ozobots sensors on the bottom and explain that these are its eyes – it can follow lines and can see different colours.</p> <p>Draw crossing lines (or use print out) and ask students to predict what will happen when Ozobot reaches a junction.</p> <div data-bbox="469 674 1305 1267" style="text-align: center;"> </div> <p>Place Ozobot at the start, and repeat a few times. The turn is random, so run enough times to demonstrate this.</p> <p>Were your predictions correct?</p>
Achievement Standards	<p>By the end of Year 2, students design solutions to simple problems using a sequence of steps and decisions.</p>
Learning Map (Sequence)	<ul style="list-style-type: none"> • Students describe the sequence of turns that Ozobot needs to make. • Students work in teams to design their algorithm using a sequence of colour codes, that navigate Ozobot along a path. • Students can draw paths, including colour codes and control Ozobot along the paths. • Students can debug their algorithms and troubleshoot (line thickness, calibration etc.)
Learning input	<p>Show how colour codes work. Demonstrate “Go right” and ‘Fast’ as examples.</p>



	<p>Explain that students are going to construct their own paths and make some paths for Ozobot to follow.</p> <p>Ozobot is quite fussy, so you'll need to work out how thick the lines need to be and what size to draw the colour codes.</p> <p>Suggest checking that ink is dry before putting Ozobot on the line.</p>
Learning construction	<p>Students understand that Ozobots have sensors and follow lines and colour code instructions.</p> <p>They work in small groups or pairs to construct paths using paper and coloured pens.</p> <p>This is an opportunity for students to play and find out how to control the Ozobots.</p> <p>Have codes printed and available or displayed at the front of the class. Example codes to start with:</p> <p style="text-align: center;">Go Left, Go Straight, Go Right, Slow, Fast, U-turn</p> <p>As students draw lines, teacher asks questions:</p> <ul style="list-style-type: none"> ● Why did you draw this bit? ● What would happen if you added a line here? ● Would Ozobot always go that way? <p>(Perhaps have available some copies of activity sheet 3 from https://storage.googleapis.com/ozobot-lesson-library/programming-with-colors/programming-with-colors.pdf)</p>
Learning demo	<p>Once students have got working drawings and solved any problems, choose a couple of groups to present what they have created with the class:</p> <ul style="list-style-type: none"> ● What does Ozobot do? ● What did you discover about the lines or codes? ● What worked well? ● What did not work well? ● Did you change anything?
Learning reflection	<p>Bring together the observations from the learning demo and, with the students, come up with a list of rules, e.g.</p> <ul style="list-style-type: none"> ● Thickness of the lines ● What size the codes need to be ● What works and doesn't work with the lines (e.g. curves, turns etc.)



Assessment:

Formative Assessment:

- Teachers observe students using the Ozobots, creating their algorithms and debugging.
- Use questioning to elicit student understanding of the functions of the Ozobot and their algorithmic thinking.
- You might take photos of the students' work to document their progress, or record the Ozobot in their final demonstration.

Criteria	Quantity of knowledge			Quality of understanding	
	Pre-structural	Uni-structural	Multi-structural	Relational	Extended abstract
Algorithms Codes	No algorithm or colour codes shown	Algorithm only shows a limited number of instructions which are not linked – possibly use of different colour lines	Algorithm has enough instructions to complete the task but not linked or not linked in the correct sequence – or there are codes that do not work	Algorithm has instructions linked in the correct sequence to achieve the task – Ozobot can follow a path as designed using colour codes	Algorithm brings in prior learning and/or independent learning beyond the task and possibly includes additional colour codes
Vocabulary	When describing algorithm, no specific vocabulary is used	The terms instruction or code may be used as a general description	The term algorithm is used as a general description	The terms algorithm is used confidently with specific reference to learner's work	Specific vocabulary like decisions and repetition is used, going beyond the set language



Teacher/Student Instructions:

It is useful for teachers to have read the Teacher Guide prior to using Ozobots with students:
<http://files.ozobot.com/stem-education/ozobot-teachers-guide.pdf>

Teachers may want to explain calibration with the students, or you could make sure each Ozobot is calibrated at the start of the lesson.

CSER Professional Learning:

This lesson plan corresponds to professional learning in the following CSER Digital Technologies MOOCs:

F-6 Digital Technologies: Foundations

- Unit 7: Algorithms and Programming

F-6 Digital Technologies: Extended

- Unit 2: Algorithms & Programming

See: <http://csermoocs.adelaide.edu.au/moocs>

Further Resources:

Ozobot Lesson Library (new lessons added monthly): <http://portal.ozobot.com/lessons>



Author: Steven Payne

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/). Computer Science Education Research (CSER) Group, The University of Adelaide.



CSER Group, The University of Adelaide