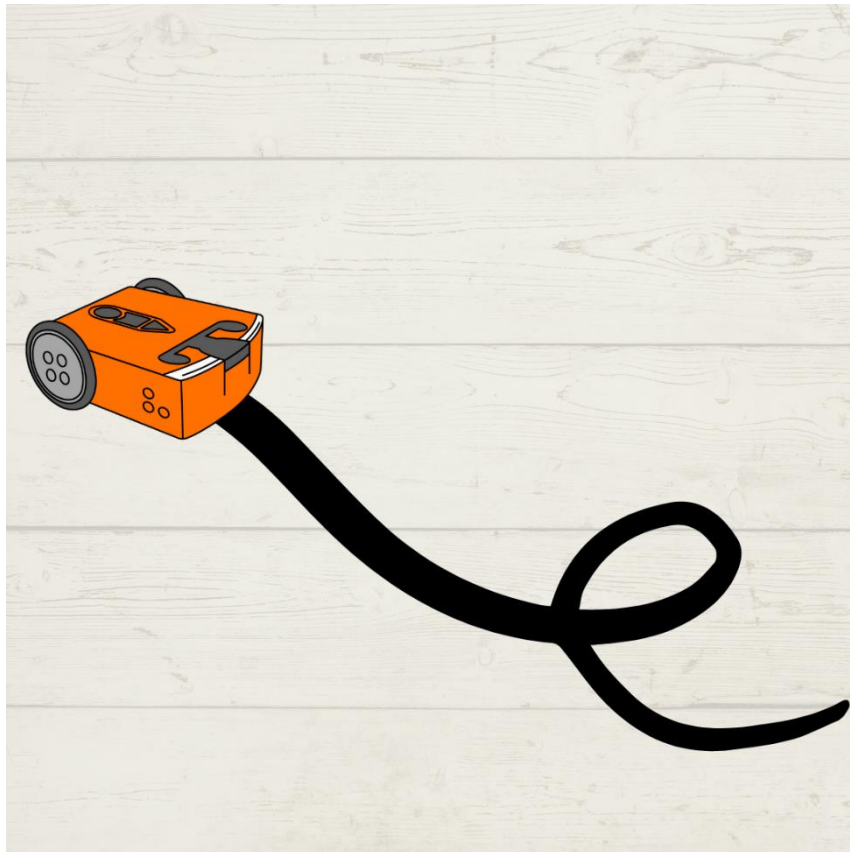




Edison and the mystery line

Teacher's notes



The *Edison's line sensor* lesson set by [Kat Kennewell](#) and [Jin Peng](#) was adapted by [Emma Grace](#) and is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#).

Contents

About this lesson and guide	3
Lesson overview	4
Part 1: How does Edison's line tracking sensor work?	5
Part 2: How do you master the mystery line?	6

Go ahead – show off!

We love seeing how classrooms use Edison! If you and your students want to share your Hour of Code Edison EdVenture, be sure to tag us into the fun!



[@meetedison](https://twitter.com/meetedison) twitter.com/meetedison



[@meet_edison](https://www.instagram.com/meet_edison) [instagram.com/meet_edison](https://www.instagram.com/meet_edison)



[@meetedison](https://www.facebook.com/meetedison) [facebook.com/meetedison](https://www.facebook.com/meetedison)

EdScratch is developed by Microbric Pty Ltd using open source software created and maintained by the Scratch Foundation. The Scratch Foundation does not sponsor, endorse, or authorize this content. See scratch.mit.edu for more information.

About this lesson and guide

This guide offers teachers and instructors overview information, facilitation recommendations and other supporting information for the *Edison and the mystery line* lesson available at <https://meetiedison.com/robotics-lesson-plans-edison-mystery-line/>

Do you need to read this whole guide to run the lesson? **Absolutely not!**

As long as the robots and programming devices are set up¹, you can start learning along with your students! The student sheets for this lesson have been designed to allow students to work through the different parts of the lesson independently, learning key computer science concepts and practicing skills along the way. This guide simply offers further information for teachers and instructors to help make using this lesson easy and fun.

Each part of the lesson is included in this guide along with any relevant supporting information for that part. Supporting information is divided into the following sections:

Overview

General information about the section and key learning objectives covered.

Delivery recommendations

Suggestions for how you can cover the lesson section if you want to run the lesson in a more facilitator-led capacity.

Tips and tricks

Helpful hints and ways to overcome common issues students may encounter.

Answer key

Solutions or examples to the included questions.

Creative Commons licence attribution details

The *Edison and the mystery line* lesson set is comprised of the student sheets and this guide. This set was developed using resources from the [EdScratch Lesson Plans Set](#)² and is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#)³.

Developed and written by: Kat Kennewell

Illustrations by: Jin Peng

Adapted by: Emma Grace

¹ The *Getting started with Edison and EdScratch* set available at <https://meetiedison.com/robotics-lesson-plans-edison-mystery-line/> has step-by-step help for setting up your robots and programming devices. If you are new to Edison or EdScratch, it is recommended you start with that guide.

² <https://meetiedison.com/robot-programming-software/edscratch/#EdScratch-resources>

³ <http://creativecommons.org/licenses/by-sa/4.0/>

LESSON OVERVIEW

Learn about algorithms, a key computer science skill, using the Edison robot’s line tracker and the [Scratch-based programming language EdScratch](#)⁴.

Grade levels	Difficulty	Duration
Year 6+	Intermediate	45+ minutes

Prerequisite knowledge

To be successful with this activity, it is recommended that students:

- Have used block-based coding applications
- Familiarity with Edison and EdScratch⁵
- Understand sequence and sequential programming
- A basic understanding of the input-process-output cycle

Computer science and computational thinking topics

- Algorithms
- Pattern recognition
- Inputs
- Events

Tie-ins to other subjects

- Art: visual design

Supplies you need

- Full set of Edison robots and EdComm programming cables
- Full set of prepared programming devices (computers or tablets)
- 4x AAA batteries per robot
- Print or digital copies of the student sheets and print-outs of the activity sheets
- Space and supplies for test-spaces [such as large paper and markers or black tape on a white floor]
- **Optional** – EdMats

Some great advice from the Hour of Code team

It’s okay not to know! Respond to student questions and struggles with phrases like:

- “I don’t know. Let’s figure this out together.”
- “Technology doesn’t always work out the way we want.”
- “Learning to program is like learning a new language; you won’t be fluent right away.”

And don’t forget to have fun! (^_^)

⁴ <https://meetiedison.com/robot-programming-software/edscratch/>

⁵ The *Getting started with Edison and EdScratch* set available at <https://meetiedison.com/robotics-lesson-plans-edison-mystery-line/> has a step-by-step activity for introducing Edison and EdScratch. If this is your first time using Edison or EdScratch, start with that guide and activity.

Part 1 : How does Edison’s line tracking sensor work?

Overview

Kickstart student’s exploration of using Edison’s line tracker using Edison robots and the Scratch-based programming language EdScratch. Students first become familiar with the Edison robot’s line tracking sensor technology, learning the fundamentals of how the sensor works. By experimenting with the line tracker’s red LED on different coloured surfaces, students learn how the robot can use the sensor to detect whether it is driving on a reflective or non-reflective surface.

Delivery recommendations

- Recommended time: 20 minutes

Tips and tricks

- If you want to explore more about how Edison’s line tracking sensor works, consider having a breakout session to explore how light reflection and absorption works, including for coloured light.
- Edison’s line tracking sensor shines light on the surface and then measures the amount of light that is reflected back. White reflects a lot of light, giving a high light reading and black reflects very little, giving a low light reading.
- You can use EdMats or make your own test space to run the ‘drive until black’ program.
- Edison’s line tracking sensor shines light on the surface and then measures the amount of light that is reflected back. White reflects a lot of light, giving a high light reading and black reflects very little, giving a low light reading. To have your Edison successfully follow the line:
 - Place Edison slightly to the left of your line, so that the line tracking sensor (where the EdComm cable plugs in) is on the white surface, next to the line.
 - Now press the play (triangle) button and watch Edison follow the line.
 - Edison will spin to one side, once it finds the black (line) surface and it will start to follow the line.
- Encourage students to check the ‘bug box’ often in EdScratch.

Answer key

Question	Answer/Sample answer	Notes
1	<i>A white surface reflects more light back to Edison than a black surface reflects. I think this is because the spot from the red LED is brighter on a</i>	The light will appear brighter on a white surface. If students note black, consider reviewing this with them.

	<i>white surface (compared to black) and that means more of the light is being reflected (rather than absorbed).</i>	
2	Colour	Reflective or non-reflective?
	Red	reflective
	Blue	non-reflective
	Green	non-reflective

Part 2: How do you master the mystery line?

Overview

The concept of algorithms is introduced in this lesson along with the important but often misunderstood concept of how algorithms differ from programs. Students learn that algorithms are a tool for solving sets of problems and can guide the logic used in computer programs. They explore an algorithm for getting Edison to follow any black line using the robot's line tracking sensor and see how the logic of the algorithm translates into an EdScratch program.

Learning that algorithms can inform computer programs, but that not all programs are algorithms, is an important concept to enable students to be able to practice so-called 'algorithmic thinking'. Understanding that some problems need a specific, dedicated program to solve while others can be solved using an algorithm is an important skill for students to develop.

The final 'extra challenge' digs into applying algorithms further, challenging students to create two other programs that use the same algorithmic logic to follow any line.

Delivery recommendations

- Recommended time: 25+ minutes

Tips and tricks

- For the line tracking program to work, the difference between the dark and light surfaces needs to be easily understood by the robot, and there needs to be enough white space between any curves in the line. The lines that students make for Edison to follow need to be dark (e.g. black), approximately 1.5cm (0.6 inches) wide, and on a white background. Alternatively, they can use an EdMat.
- Make sure students start by placing Edison with the line tracking sensor on white. The robot can start near the black line but not on top of it.

- You may choose to have students run the barcode program 'follow a line' to see the same algorithm from the worksheet at work in that barcode program.
- For task 2, consider having students write their name (or similar) in cursive as the 'follow a different black line' activity.
- Keep in mind that if two black lines are too close, the robot might 'hop' to the other line and if a curve is too tight, the robot will be unable to follow it.

Answer key

Question	Answer/Sample answer	Notes
1	<i>The robot 'waddles' along the edge of the line. This is because the algorithm says that the robot should drive forwards to the left until the line tracking sensor is on a non-reflective (black) surface. That pushes the sensor along until it finds a black line to the left. Then the logic says that the robot should drive forwards to the right until the line tracking sensor is on a reflective (white) surface. That pushes the sensor off black, onto white. The robot alternates this behaviour, moving back and forth on the edge of the line. This makes the robot 'waddle' left and right as it goes along the line.</i>	Students should identify the robot's waddle is caused by the program logic going back and forth between the two 'until' blocks.
2	<i>Edison was able to follow my line except in one spot where the line curved very sharply. The robot kept missing the curve and catching the other part of the line. I think this happened because the two parts of the line were really close together, so the robot would just find a non-reflective spot and start following it, even though it wasn't the part of the line I wanted it to follow.</i>	

What should I do if a student finishes the lesson early?

- Encourage students to make another pattern and program Edison to drive it using the line tracking sensor.
- You can always ask students who finish early to help classmates who are having trouble with the activity.