



NEOM

ReadyAI

# BEGINNER AI

# Table of Contents

# Lesson Plans

---

- 07 Introduction to Cozmo and Computer Vision
- 15 The First Law of Calypso
- 21 Motion with Cozmo
- 27 The Second Law of Calypso
- 31 Fireman Challenge
- 37 Multiple Actions and Conflict Resolution
- 43 The Third Law of Calypso
- 49 Dependent Action and Rule Indentation
- 65 Introduction to State Machines
- 79 Appendix
  - Cozmo Intro Script - First Law Worksheet
  - Motion with Cozmo Script - Second Law Worksheet
  - Cube Tap Challenge Worksheet
  - Custom markers on adhesive stickers

## Standards and Guidelines Cited

### ISTE Standards

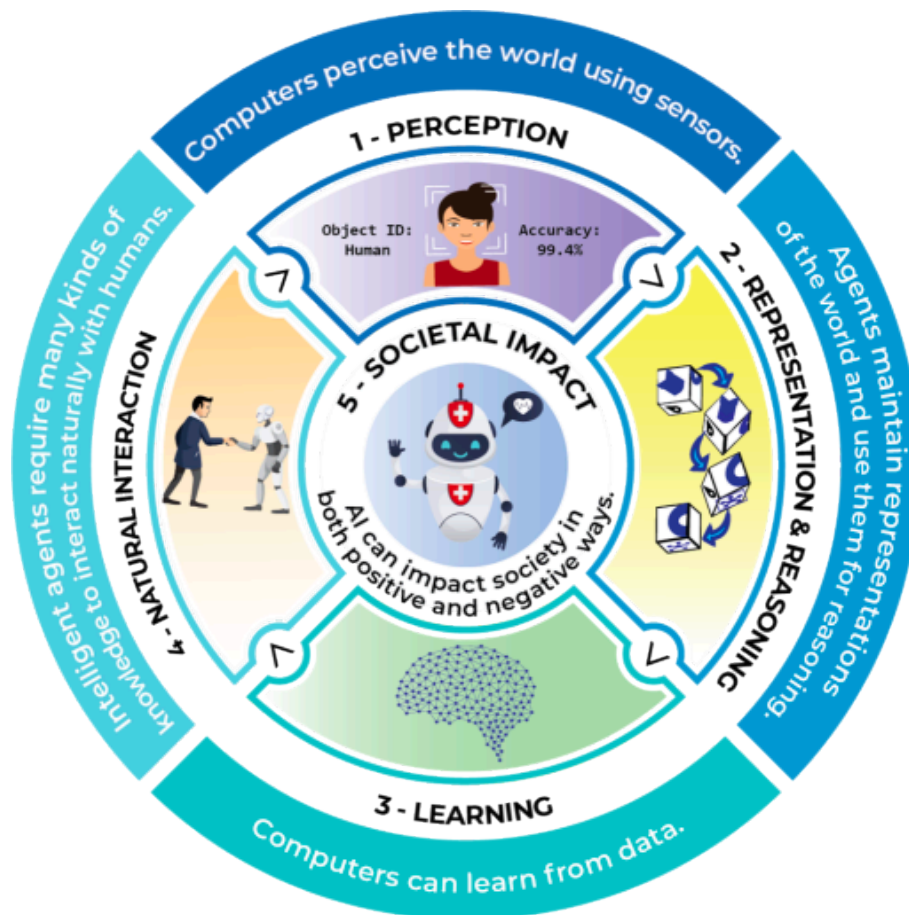
The International Society for Technology in Education (ISTE) produces a series of standards for those involved in education. The lessons in the following curriculum utilize the seven ISTE Student Standards.

- The ISTE Student Standards can be found at <https://www.iste.org/standards/iste-standards-for-students>

### AI4K12 Guidelines

AI4K12.org has released a list of “Five Big Ideas in AI” along with draft grade band progression charts for each of these ideas.

- The five big ideas are summarized on a poster that can be found at <https://ai4k12.org/resources/big-ideas-poster/>.
- The grade band progression charts are available at <https://ai4k12.org/gradeband-progression-charts/>



**Big Idea 1 - Perception:** Computers perceive the world using sensors.

**Big Idea 2 - Representation and Reasoning:** Agents maintain representations of the world and use them for reasoning.

**Big Idea 3 - Learning:** Computers can learn from data.

**Big Idea 4 - Natural Interaction:** Intelligent agents require many kinds of knowledge to interact naturally with humans.

**Big Idea 5 - Societal Impact:** AI can impact society in both positive and negative ways.



NEOM

In partnership with



# 01

## Introduction to Cozmo and Computer Vision

---



# TEACHING GUIDE

## Lesson Objectives:

By the end of this lesson, students will be able to

- Articulate the abilities and limitations of Cozmo's vision
- Identify the meaning of the world map and what it represents

## Alignment with Big Ideas:

Big Idea #1: Computers perceive the world using sensors

Big Idea #2: Agents maintain representations of the world and use them for reasoning.

## ISTE Standards for Students:

1.1 Empowered Learner

1.3 Knowledge Constructor

1.5 Computational Thinker

## ISTE Computational Thinking Competencies:

5.1 Computational Thinking (Learner)

## Equipment

Robot and light cubes. [Custom markers on adhesive stickers](#). Tape measure or yardstick for measuring distances up to 600 mm. Protractor to measure angles.

## Preparation

Each robot should be on its charger, with its head level so it is looking out at the table. Each robot should be paired with its light cubes. Calypso should be running on the laptop, with a fresh empty program. The camera viewer should be exposed and set to 2X size. The cubes should be nearby, and right side up.



## Instructions

1. Point out that the camera viewer is showing the robot's view of the table. Students can verify this by waving their fingers in front of the robot.
  - a. Ask a question such as, "What do we know about Cozmo's ability to see?"
  - b. Answers may include
    - i. Limited distance - approximately 500 millimeters
    - ii Can identify objects including cubes, people, and the charger
    - iii Show the following video and ask how Cozmo's vision is different:

 [Besam Automatic Swing Doors at Super Target \(Grocery\) in Lewisville TX](#)

Highlight how automatic doors do not identify objects or people. They simply open based on a (ultrasonic) sensor. The sensor only tells the doors that some thing is there.

## Sensing without perception?

This is what you get when your automatic doors have sensing but not perception.



Cozmo has a sensor (camera), but AI helps him recognize some of what he is seeing.

Lesson Differentiation	
Early Learners	Older Learners
<p>Devices that can't see are not likely to have AI. Can you think of examples? (washing machine, toaster, coffee machine)</p> <p>Devices that can see are more likely to have AI. Can you think of examples? (Cozmo, self-driving car, some smartphone applications)</p> <p><b>Trick question:</b> Video cameras can record but may not have AI. It records without knowing what it sees. Some cameras use software that detect faces (focusing, face recognition, etc.)</p>	<p>Explain what the difference is between sensing and perception.</p> <p>Sensing can include a microphone or webcam. These record visual and audio signals. They don't tell anything about what is in that signal.</p> <p>Perception extracts meaning from those signals, i.e. what a face is, what language or words were said, and what objects are in the scene.</p>

2. Show the robot one of its cubes, and observe a yellow box drawn around the cube in the camera viewer. Note that an entire face of the cube must be visible for the robot to recognize it.
3. Move the cube around and verify that the yellow box stays with the cube.
4. **Experiment:** how much of the cube can be obscured before the robot can no longer detect it? (Answer: the square frame and everything within it must be visible.)
5. Show the robot a second cube and verify that it can track two cubes at once.
6. Gently tilt the robot's head up (or crouch down so your face is visible in the camera image) and see that the robot can track faces, displaying a green box in the camera viewer.
7. Tilt the head back down and show the robot a single cube. Gradually move the cube away from the camera until the robot can no longer detect it (the yellow box in the camera viewer disappears). Try moving the cube back and forth and observe the yellow box appear and disappear.
8. Experiment: how far can the cube be from the robot before the robot can no longer detect it? Use a yardstick or tape measure to measure this value. Compare your result with other students. (Answer: 500 mm.)

## The World Map

9. Show how to enter execution mode by pressing the Back button on the game controller, hitting the backspace key on the keyboard, or clicking the Run icon in the rule display.
10. Show the robot an upright cube and observe its appearance on the world map.
11. Move the cube and watch the world map update.
12. Rotate the cube and watch the world map update.
13. Try turning the cube on its side or upside-down, and watch the world map update.
14. Tap on the physical cube with a fingertip and observe the response on the world map.
15. Use a hand to block the robot's view of the cube and notice that the cube becomes grayed.

- Remove the hand and see the cube un-gray.
16. Move the cube out of the camera frame and observe that it goes into the dock in the top right corner of the screen; the robot no longer knows its location.
  17. Put the cube back in the robot's field of view and see it return to the world map.
  18. Place a second cube between the robot and the first cube, so that it occludes the first cube. Note that the occluded cube stays on the map but is now gray because the robot cannot currently see it.
  19. **Faces:** show your face to the robot and notice a face appears on the world map. Changing your facial expression results in the face changing color.
  20. **Custom markers:** show a custom marker to the robot and notice it appears on the world map. Rotating the custom marker is also reflected in the world map display.

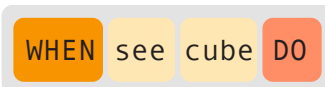
**To older students,** introduce the idea of lawfulness.

- Lawfulness is the idea of logical thinking, robots following well-defined laws of behavior. Ask students to give examples of laws that govern behavior.
- You might liken it to laws of physics, such as gravity. Humans have to obey certain laws such as gravity. Cozmo and Calypso follow core fixed laws that govern all actions as well.

## Line of Sight Indicator

This section introduces the rule editor as students write their first program fragment.

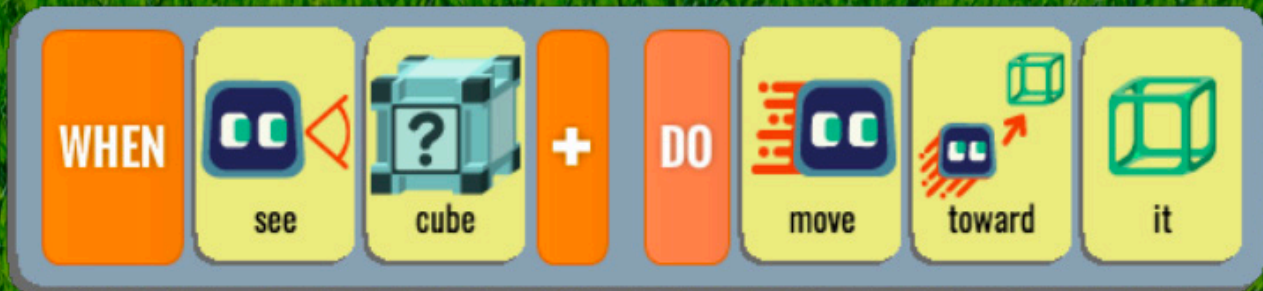
21. If still in execution mode, stop the program by pressing the Back button on the game controller, hitting the backspace key on the keyboard, or clicking on the Stop icon to the left of the world map.
22. Using either the game controller or the mouse, enter the following rule fragment:



23. Run the program and show the robot a cube. The red line running from the robot to the cube on the world map is called a Line of Sight (LoS) indicator.
24. Use a hand to block the robot's view of the cube. Although the cube is grayed, it maintains its position on the world map, the LoS indicator still points to it, and the rule is still firing (the rule is un-grayed in the rule display in the left half of the screen.)
25. Pick up the cube and remove it from the robot's view, so it goes back into the dock and is no longer in the world map. If no cubes are present in the world map, the rule cannot match any cube and so it will be grayed.
26. Show the robot two cubes at once. Both should have yellow outlines in the camera viewer, and both should be visible in the world map, but only one will have an LoS indicator pointing to it.
27. Ask students to figure out which cube gets the LoS indicator. They may come up with several hypotheses, e.g., the LoS indicator attaches to the first cube the robot sees, or to the cube closest to the center of the camera image, or to the cube closest to the robot.
28. Ask students how they could test their hypotheses, and guide them to test their ideas by moving the cubes around. They should eventually discover that the LoS indicator points to the cube that is closest to the robot.
29. Introduce students to the First Law of Calypso: "Each rule picks the closest matching object." Show them the graphic for the first law, and point out how it illustrates the law.

# First Law of *Calypso*

Each rule picks the closest matching object.



## *Time Permitting:*

Students may dramatize the First Law of Calypso.

One student stands in the room and plays the role of Cozmo. Another student plays the role of the cube. The Cozmo students should point to the student playing the role of the cube. This reflects how Cozmo uses a line of sight indicator.

Next, ask another student to volunteer as the second cube student. This student stands further away from the first cube student. Ask the class: *Which cube should Cozmo point to?* (Answer: First cube student).

Third, slowly move the second cube student towards the Cozmo student. Ask the students at what point the Cozmo student should point at second Cozmo student. (Answer: When second cube student is closer)

Ask students how this illustrates the First Law of Calypso. Explain that these laws govern how Calypso and Cozmo work. Add that AI and all computer science operates on strict laws.

30. Have students move the cubes around and verify that the First Law explains which cube gets the LoS indicator.
31. Go back to the rule editor and add a second rule, so the program looks like:

WHEN see cube DO

WHEN see red cube DO

32. Run the new program and note that only the first rule is able to run, since none of the cubes are red.
33. Right click on a cube in the world map and set its color to red. Now both rules can fire.
34. Each rule has its own LoS indicator. Under what circumstances will both indicators point to the same cube? (Answer: when the red cube is the closest cube.)
35. How can you ensure that the two LoS indicators point to different cubes? (Answer: keep the red cube further away from the robot than the non-red cube.)
36. What happens when both cubes are red? (Answer: both LoS indicators will point to whichever cube is closest.)

*Time Permitting:*

Teachers might use the following worksheet to gauge student understanding of the First Law of Calypso. Explain that Calypso and Kodu are both programming tools, and they both run on laws. Ask students to apply their understanding of the First Law of Calypso to the following three scenarios:

[1. Cozmo Intro Script - First Law Worksheet](#)

For additional experiments, ask students to dramatize the First Law in small groups using pointing. For added complications, ask students to hold objects and dramatize according to the First Law (i.e. “Point at Red Student Cube” versus “Point at Student Cube”).

For additional experiments, try the [First Law Activity](#).

## Enrichment

Advanced students may be encouraged to begin programming with Calypso. Here are some tasks to ask them to try:

**Program “WHEN see cube DO glow it {Insert Color}”**

Ask students to demonstrate how the First Law is applicable  
Activity to test: What happens if the “it” tile is left out?

**Program “WHEN see cube inverted”**

Again, ask students to demonstrate how the First Law is applicable  
Activity to test: What happens if they turn the cubes upside down?

**Program “WHEN see” and mix inversions and color usage**

Ask students to demonstrate how the First Law is applicable.  
Activity to test: What happens if they have a cube blue and sideways / red and inverted / etc.?

**Program “WHEN see cube upright DO glow it green” and “WHEN see cube inverted DO glow it red”**

Ask students to test what happens.  
Ask students what happens if Cozmo cannot see the cube.