



NEW YORK TUTORING CENTER

# Rover Lab Teacher Guide

## Scratch programming and robotics

A detailed elementary unit aligned to selected New York State K-12 Computer Science and Digital Fluency Learning Standards.

| Audience                    | Pacing               | Standards bands |
|-----------------------------|----------------------|-----------------|
| Grades 2-5, with extensions | 9 weeks x 50 minutes | 2-3 and 4-6     |

## Designed by Huanjiao Dong

Curriculum Created by Huanjiao Dong

Educator edition 4.0 | October 20, 2026

Open the nine-week course

This is an independently developed teaching resource. It is a focused standards alignment, not an official NYSED curriculum or coverage of every K-12 expectation.



## EDUCATOR GUIDE 01

# Start here: the unit at a glance

**A teacher-developed New York standards alignment**

This is a focused Scratch programming and robotics unit for grades 2-5, with grade 6 extension opportunities. It draws on NYSED grade bands 2-3 and 4-6. It is not an official NYSED curriculum, an endorsement, or coverage of all K-12 standards. The state sets learning expectations; curriculum and instructional choices are local. NYSED: Computer Science and Digital Fluency Guidance & Resources

| Planning decision   | Recommended starting point                                                                                                                                                                                                  |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Length              | Nine weekly 50-minute lessons (450 minutes): nine different 3D robotics games: delivery, harbor navigation, factory loops, cave sensing, cargo sorting, snow rescue, AI trials, model calibration, and Mars mission design. |
| Group               | One device per pair. Driver builds; navigator predicts and traces. Swap roles after every test or about 5-7 minutes.                                                                                                        |
| Prior experience    | No Scratch experience required. Begin with following directions and counting tiles.                                                                                                                                         |
| Core resources      | Rover Lab in a modern browser, teacher display, paper grid, arrow counter, pencil, and the reproducible sheets in this guide.                                                                                               |
| Optional resources  | A teacher-approved Scratch setup; a physical robot only if a suitable kit, extension, and supervision are available.                                                                                                        |
| Language and access | Instruction is in English. Programming and robotics are the subjects. Use home-language discussion, oral responses, diagrams, or tactile cards as appropriate.                                                              |
| Unit evidence       | One route plan, a decision trace, a debugging record, an individual explanation, and the final rubric.                                                                                                                      |

## What learners will understand

- Robots execute precise instructions in order; they do not infer the intended goal.
- Heading changes the meaning of forward. A turn does not travel to another tile.
- Loops can reduce written code while performing the same number of actions.
- Sensor data informs conditions; IF / ELSE chooses one branch for each decision.
- A useful test compares a prediction with observations and leads to a reasoned change.
- A model leaves things out. A real robot has imperfect motion and sensor readings.

## Nine-week course map



| Week | Lesson                        | Student activities            |
|------|-------------------------------|-------------------------------|
| 1    | Week 1: Parcel Post           | Open activities and worksheet |
| 2    | Week 2: Harbor Pilot          | Open activities and worksheet |
| 3    | Week 3: Loopworks Factory     | Open activities and worksheet |
| 4    | Week 4: Cave Sentinel         | Open activities and worksheet |
| 5    | Week 5: Color Cargo Sorter    | Open activities and worksheet |
| 6    | Week 6: Alpine Rescue         | Open activities and worksheet |
| 7    | Week 7: Orbital AI Trials     | Open activities and worksheet |
| 8    | Week 8: Robot Maker Lab       | Open activities and worksheet |
| 9    | Week 9: Mars Mission Designer | Open activities and worksheet |

## Choose the right pace

Use the nine-week sequence for classroom instruction and evidence collection. For a 45-60 minute introduction, sample the first three weekly games, demonstrate the sensor, and postpone independent assessment of nested decisions. For younger beginners, split each lesson into two 25-minute sessions. Give additional time for access needs rather than treating speed as achievement.

## Before the first lesson

- Open the course and run a short test on the teaching device. Confirm all nine game links, the robot scene, and the relevant program or lab controls are visible.
- Decide whether learners will use the public site or the offline HTML. Progress is stored in that browser on that device, not in a teacher gradebook; a different browser or site origin may have different progress.
- Print the relevant route map and worksheet. Provide an equivalent 2D or tactile route when a 3D view is not a useful access route for a learner.
- Teach the roles: the driver controls the device; the navigator predicts and traces. Both must be able to explain the program.
- Plan an individual check. A partner's successful program and game awards alone do not demonstrate each learner's understanding.

## A consistent teaching routine

| Step    | Teacher prompt                                              | Student evidence                                       |
|---------|-------------------------------------------------------------|--------------------------------------------------------|
| Predict | What will happen, and which instruction makes you think so? | An endpoint, heading, or chosen branch before running. |
| Build   | Where must this block go?                                   | A program whose nesting matches the plan.              |



| Step    | Teacher prompt                                        | Student evidence                                          |
|---------|-------------------------------------------------------|-----------------------------------------------------------|
| Test    | What did the rover actually do?                       | A recorded observation rather than a guess.               |
| Explain | Where did prediction and observation agree or differ? | A trace naming the relevant block and state.              |
| Revise  | What one change will you test, and why?               | A justified change and a comparison with the earlier run. |

## EDUCATOR GUIDE 02

# New York State standards and evidence

NYSED organizes these standards into K-1, 2-3, 4-6, 7-8, and 9-12 bands. Expectations are organized for achievement by the end of the relevant band. Grade 4-5 teaching can build toward the 4-6 expectations; this guide does not change the end-of-band expectation. NYSED: Computer Science & Digital Literacy FAQ

### How to read this crosswalk

Descriptions are brief paraphrases. Each code links to the official grade-band document and page. Core activities and teacher-added evidence are distinguished from extensions. A learning opportunity is not an automatic finding of mastery.

| Standard and focus                                                    | Activity and evidence                                                                 | Scope                        |
|-----------------------------------------------------------------------|---------------------------------------------------------------------------------------|------------------------------|
| <b>2-3.CT.4</b><br>Break a task into smaller steps in different ways. | Weeks 1, 2, 6<br>Plan travel, pickup, delivery, or rescue/return sub-tasks.           | Core + teacher discussion    |
| <b>2-3.CT.6</b><br>Create multiple algorithms for one task.           | Weeks 3<br>Compare four move/process pairs with a repeat-4 routine.                   | Core comparison activity     |
| <b>2-3.CT.7</b><br>Label fixed and changing information.              | Weeks 2, 4<br>Label a fixed 90-degree turn and changing heading or stored scan.       | Teacher-added state record   |
| <b>2-3.CT.8</b><br>Identify actions controlled by precise conditions. | Weeks 4, 5<br>Explain the blocked IF case and the chosen cargo-sorting branch.        | Core + oral explanation      |
| <b>2-3.CT.9</b><br>Debug sequencing and repetition.                   | Weeks 1-3, 6<br>Identify an order, repetition, or resource-planning error and retest. | Core debugging record        |
| <b>2-3.CT.10</b><br>Document a project plan with specific steps.      | Weeks 1, 6, 9<br>Document a delivery, rescue, or custom Mars mission plan.            | Teacher-added planning sheet |



| Standard and focus                                                           | Activity and evidence                                                                                                    | Scope                                                 |
|------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------|
| <b>2-3.NSD.1</b><br>Demonstrate program inputs and outputs.                  | Weeks 4, 7<br>Identify path-check inputs and movement/display outputs.                                                   | Teacher-added systems activity                        |
| <b>2-3.NSD.2</b><br>Explain what software does in a computing system.        | Weeks 1, 4<br>Explain how the program directs the robot's actions.                                                       | Core + systems explanation                            |
| <b>2-3.DL.2</b><br>Communicate ideas with others using digital tools.        | Weeks All<br>Use a shared program or annotated capture to explain a test.                                                | Teacher-facilitated collaboration                     |
| <b>4-6.CT.1</b><br>Build a computational model responsive to changed inputs. | Weeks 8 extension<br>Author a Scratch rover model and test changed inputs; guided module assembly alone is insufficient. | Extension; using this simulator alone is insufficient |
| <b>4-6.CT.4</b><br>Break a problem into named, smaller tasks.                | Weeks 2, 6, 9<br>Name navigation, rescue/return, or mission-design sub-tasks.                                            | Core + planning sheet                                 |
| <b>4-6.CT.6</b><br>Compare algorithms and explain their trade-offs.          | Weeks 3, 6<br>Compare code clarity, written block count, action count, and battery costs.                                | Core comparison activity                              |
| <b>4-6.CT.7</b><br>Identify information that changes during execution.       | Weeks 4, 5, 8<br>Track stored scan, cargo state, and controlled input changes.                                           | Teacher-added state record                            |
| <b>4-6.CT.8</b><br>Solve problems with repetition and conditionals.          | Weeks 3-5, 9<br>Use repetition and conditional actions and explain their roles.                                          | Core; nesting is an additional challenge              |
| <b>4-6.CT.9</b><br>Trace repetition and conditionals to debug a program.     | Weeks 4, 5, 9<br>Trace a condition or loop and locate the first unexpected action.                                       | Core + teacher-added trace sheet                      |
| <b>4-6.CT.10</b><br>Describe iterative design steps and design choices.      | Weeks 6, 9<br>Explain a plan, tests, evidence, revision, and limitation.                                                 | Teacher-added reflection                              |
| <b>4-6.NSD.2</b><br>Model hardware and software working as a system.         | Weeks 4 + Scratch extension<br>Create a four-part input, processing, storage, and output systems model.                  | Teacher-added four-part systems model                 |
| <b>4-6.DL.2</b><br>Choose digital tools for purposeful collaboration.        | Weeks 9 + extension<br>Justify a digital tool or presentation choice for partner collaboration.                          | Extension; include a justified tool choice            |
| <b>4-6.DL.4</b><br>Create and revise artifacts using multiple digital tools. | Weeks 8-9 extension<br>Create and revise a Scratch artifact as well as a browser-game design.                            | Extension; add the Scratch activity                   |



## Use the alignment accurately

- The 4-6.CT.8 clarification permits working with loops and conditionals separately. Combining them in the sorting and Mars games is an additional challenge, not a requirement inferred from that standard alone.
- Using a provided 3D simulation is not the same as developing a computational model. Use the Scratch bridge before claiming evidence toward 4-6.CT.1.
- For 4-6.NSD.2, include input, processing, storage, and output in the learner's systems model. Merely naming a front sensor is insufficient.
- The unit emphasizes Computational Thinking and selected Networks & System Design and Digital Literacy expectations. Other standards, including the wider Cybersecurity curriculum, need additional instruction.



## EDUCATOR GUIDE 03

# Teach the nine weekly games accurately

| Feature                       | What to teach                                                                                                                                                                                                                                                                              |
|-------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nine robots and worlds        | Each week links to a different game with its own robot, scenery, actions, and completion rules. The older six-mission Rover simulator remains optional practice; its maps and controls differ.                                                                                             |
| Tile movement and directions  | Move is one tile by default; turns rotate 90 degrees without travel. Columns increase west to east; rows increase north to south. Camera rotation changes the view, not heading. Scratch transfer can use an agreed scale such as 40 steps per tile.                                       |
| Editing programs              | Select an insertion slot, then add a block. Add actions inside Repeat, THEN, or ELSE as required. Up, Down, and Remove edit the chosen list. Available blocks depend on the weekly game.                                                                                                   |
| Sensing and branches          | Clear means the next tile is permitted and unoccupied by a barrier. Week 4 requires a scan at the current pose. Week 5 scans cargo color. Exactly one IF / ELSE branch runs per check.                                                                                                     |
| Stopping and limits           | Reset robot cancels a run and preserves blocks. Invalid moves stop safely in the simulation. Weekly programs permit 45 written blocks, repeat counts 1-16, four nested groups, and a 150-evaluation guard. These are course limits, not Scratch limits.                                    |
| Goal behavior                 | The game ends a run when its full goal is met, even if a repeat has iterations left. Pickup, delivery, processing, rescue, and unload are explicit actions. The Mars goal also requires design edits and distinct test evidence.                                                           |
| Completion records            | Each game stores an award on this device after its conditions pass. The weekly reflection checklist is separate and self-reported. Neither record is a teacher gradebook. Replay does not erase a previous award.                                                                          |
| AI trials                     | The Week 7 decision tree learns from learner labels. Students predict and review before triggering a single tested 3D action. Three input combinations remain unseen in training. Relabeling requires retraining and repeating the trials.                                                 |
| Maker and final project       | Week 8 assembles a supplied model and compares repeat 3 with repeat 2 under controlled conditions. Add an authored Scratch project for model-development evidence. Week 9 permits trail and sample editing and records distinct program/map tests during the session.                      |
| Accessibility and offline use | All nine games have labeled click-to-add controls, keyboard focus, reduced motion, and written position/heading status. Use the per-lesson 2D references, counters, verbal tracing, or teacher-operated controls as needed. All assets are embedded; Scratch web extensions need internet. |
| Session and saved data        | Awards and worksheet responses save locally. Weekly block programs, AI labels, calibration stages, and Mars map/test history are session state and reset on reload. Download worksheet evidence before leaving.                                                                            |
| Scratch and hardware          | The games are independent Scratch-inspired simulations. They do not run Scratch projects or connect to physical hardware. Native Scratch has a 2D stage; these custom sensing and robotics actions need explicit representation when transferring.                                         |



## Troubleshooting

- A required branch or loop is empty: select its insertion slot and add an action.
- A turn seems wrong: check the written heading and use a physical arrow; camera view is independent.
- A game goal is incomplete: read the challenge and trace. Reaching a destination alone may omit pickup, docking, unloading, or required test evidence.
- A learner cannot interpret the 3D scene: provide the 2D map, accessible state text, counter, and spoken explanation. Record reasoning independently of input method.
- Work seems missing: check device, browser, and public/offline origin. Keep important learning evidence in downloaded worksheets or an approved teacher record.



## EDUCATOR GUIDE 04

# Week 1: Parcel Post

**Essential question**

How does instruction order make a delivery succeed?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                          |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can program a pickup and delivery in the correct order.                                                                                                                                                                                                                                                              |
| Evidence of learning | Pick up the parcel at C3 R3 and deliver it to C5 R3. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                                                                               |
| Standards focus      | 2-3.CT.4, 2-3.CT.9, 2-3.CT.10, 2-3.NSD.2.                                                                                                                                                                                                                                                                              |
| Vocabulary           | sequence, instruction, cargo, pickup, delivery, debug                                                                                                                                                                                                                                                                  |
| Readiness            | Follow a short sequence and use a grid. No previous Scratch experience is required.                                                                                                                                                                                                                                    |
| Preparation          | Open game-week-01.html (Parcel Post). Check the courier cart and Meadow Mail scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. Game awards and worksheet responses save locally on this device; use teacher records for assessment. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                                | Look for                                                                     |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: How does instruction order make a delivery succeed? Identify what must be observed to verify success.                                                                                                           | Learners restate the task and constraints.                                   |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                            | Each learner names an input, expected action, and observable result.         |
| 12-20   | <b>Activity 1: Find the parcel</b><br>Open Parcel Post. Locate C3 R3 and the destination at C5 R3. Run two moves and try delivery before pickup. Read the feedback; identify the missing action. Ask pairs to trace with a counter before retesting.                          | Collect the first worksheet prediction and observation.                      |
| 20-35   | <b>Activity 2: Complete a delivery</b><br>Build the route with a pickup at the parcel and a delivery at the house. Predict each cargo change, then Run. Swap roles and rebuild without copying. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result. |



| Minutes | Teaching moves                                                                                                                                                                                                          | Look for                                                                        |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 35-43   | <b>Activity 3: Debug the pickup</b><br>Move Pick up one instruction earlier. Predict and test the failure. Use the trace to put pickup back at C3 R3 and complete a new delivery.                                       | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: Which instruction changes the parcel from waiting to being carried? Allow a spoken answer, drawing, or written response without a partner supplying the explanation. | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                                    | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

move; move; pick up; move; move; deliver

Six actions: move twice, pick up, move twice, deliver. The parcel remains on the tile until pickup. Delivery requires both the correct location and carried cargo.

## 2D teaching reference

Columns run west to east; rows run north to south. S = start, G = goal, M = mission item, X = barrier, P = route. Start heading: east. Use counters and this grid as an alternative to interpreting the 3D view.

| Grid | C1 | C2 | C3 | C4 | C5 |
|------|----|----|----|----|----|
| R1   | .  | .  | .  | .  | .  |
| R2   | .  | .  | .  | .  | .  |
| R3   | S  | P  | PM | P  | G  |
| R4   | .  | .  | .  | .  | .  |
| R5   | .  | .  | .  | .  | .  |

## Misconceptions and a useful response

| Misconception                                      | Response                                                                                                        |
|----------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| Moving over a parcel collects it.                  | Test moving past it without Pick up. Ask the learner to name the missing state-changing action.                 |
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence. |



## Differentiate the task

**Support:** Use six illustrated action cards and a parcel counter. Ask learners to arrange the cards and explain the pickup position. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Compare a route plan grouped as travel / collect / travel / deliver with the same six individual actions. Explain why the order cannot be arbitrary.

## Independent exit check

**Prompt:** Which instruction changes the parcel from waiting to being carried?

**Teacher key:** Pick up at C3 R3. Moving across the parcel does not collect it automatically.

## Connect to Scratch or physical robotics

In Scratch, use a cargo variable and touching-destination condition. A physical gripper would need an explicit pickup action and a successful-grip check.



## EDUCATOR GUIDE 05

# Week 2: Harbor Pilot

## Essential question

Why do position and heading both matter when docking?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                                      |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can navigate two beacons and dock with the required heading.                                                                                                                                                                                                                                                                     |
| Evidence of learning | Visit the two beacons in order. Dock at C3 R1 facing north. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                                                                                    |
| Standards focus      | 2-3.CT.4, 2-3.CT.7, 2-3.CT.9; 4-6.CT.4.                                                                                                                                                                                                                                                                                            |
| Vocabulary           | position, heading, north, east, turn, beacon                                                                                                                                                                                                                                                                                       |
| Readiness            | Follow a short sequence and use a grid. Revisit the previous week's prediction and explanation before introducing this game.                                                                                                                                                                                                       |
| Preparation          | Open game-week-02.html (Harbor Pilot). Check the twin-hull robot boat and Lantern Harbor scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. Game awards and worksheet responses save locally on this device; use teacher records for assessment. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                     | Look for                                                                     |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: Why do position and heading both matter when docking? Identify what must be observed to verify success.                                                                                              | Learners restate the task and constraints.                                   |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                 | Each learner names an input, expected action, and observable result.         |
| 12-20   | <b>Activity 1: Pilot the first beacon</b><br>Open Harbor Pilot. The boat begins at C1 R5 facing NORTH. Predict and run two moves. Record the tile and heading at beacon 1. Ask pairs to trace with a counter before retesting.                                     | Collect the first worksheet prediction and observation.                      |
| 20-35   | <b>Activity 2: Navigate and dock</b><br>Build a route through beacon 2 at C3 R3, then the dock at C3 R1. Use the Dock facing north action. Complete the game and describe each turn. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result. |



| Minutes | Teaching moves                                                                                                                                                                                                      | Look for                                                                        |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 35-43   | <b>Activity 3: Test a wrong heading</b><br>Add a right turn immediately before Dock. Predict whether the position changes. Run, read the docking feedback, then correct the heading and retest.                     | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: Why can a boat reach the dock but still fail the docking check? Allow a spoken answer, drawing, or written response without a partner supplying the explanation. | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                                | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

move x2; right; move x2; left; move x2; dock

The boat travels north twice, turns right to east, travels twice, turns left to north, travels twice, and docks. Nine actions visit both beacons in order and satisfy the north-facing docking requirement.

## 2D teaching reference

Columns run west to east; rows run north to south. S = start, G = goal, M = mission item, X = barrier, P = route. Start heading: north. Use counters and this grid as an alternative to interpreting the 3D view.

| Grid | C1 | C2 | C3 | C4 | C5 |
|------|----|----|----|----|----|
| R1   | .  | .  | G  | .  | .  |
| R2   | .  | .  | P  | .  | .  |
| R3   | PM | P  | PM | .  | .  |
| R4   | P  | .  | .  | .  | .  |
| R5   | S  | .  | .  | .  | .  |

## Misconceptions and a useful response

| Misconception                                      | Response                                                                                                        |
|----------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| Turning moves the boat to the next tile.           | Trace the same coordinate before and after a turn. Rotate an arrow counter without sliding it.                  |
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence. |



## Differentiate the task

**Support:** Provide an arrow counter and mark the two numbered beacons. Use cardinal-direction cards independent of camera angle. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Turn the boat right then left at the start. Explain the same endpoint and the two extra actions.

## Independent exit check

**Prompt:** Why can a boat reach the dock but still fail the docking check?

**Teacher key:** It must also face north and have visited both beacons in order. Heading and position are different.

## Connect to Scratch or physical robotics

Scratch direction 0 points up; this simulator internally uses east/south/west/north indices. Use visual arrows when transferring rather than copying numeric heading values.



## EDUCATOR GUIDE 06

# Week 3: Loopworks Factory

**Essential question**

How can one routine process four stations?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                                        |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can use a loop to repeat a move-and-process routine.                                                                                                                                                                                                                                                                               |
| Evidence of learning | Process all four stations and reach the final platform using a repeat loop. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                                                                      |
| Standards focus      | 2-3.CT.6, 2-3.CT.9; 4-6.CT.6, 4-6.CT.8.                                                                                                                                                                                                                                                                                              |
| Vocabulary           | loop, iteration, routine, written block, executed action                                                                                                                                                                                                                                                                             |
| Readiness            | Follow a short sequence and use a grid. Revisit the previous week's prediction and explanation before introducing this game.                                                                                                                                                                                                         |
| Preparation          | Open game-week-03.html (Loopworks Factory). Check the mini forklift and Neon Assembly Line scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. Game awards and worksheet responses save locally on this device; use teacher records for assessment. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                                                | Look for                                                                     |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: How can one routine process four stations?<br>Identify what must be observed to verify success.                                                                                                                                 | Learners restate the task and constraints.                                   |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                                            | Each learner names an input, expected action, and observable result.         |
| 12-20   | <b>Activity 1: Process one station</b><br>Open Loopworks Factory. Build Move, then Process station. Predict the station result and Run. Explain why the remaining stations are unfinished. Ask pairs to trace with a counter before retesting.                                                | Collect the first worksheet prediction and observation.                      |
| 20-35   | <b>Activity 2: Automate four stations</b><br>Clear the program. Add Repeat and place both Move and Process inside it. Choose a count, predict the total actions, and run until all four stations are processed. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result. |



| Minutes | Teaching moves                                                                                                                                                                                                                                              | Look for                                                                        |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 35-43   | <b>Activity 3: Compare two algorithms</b><br>Build the long eight-action sequence and test it. It processes the stations but needs a repeat to earn game completion. Restore the loop version. Compare the same work, readability, and written block count. | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: How many actions run when two actions are inside repeat 4? Allow a spoken answer, drawing, or written response without a partner supplying the explanation.                                              | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                                                                        | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

```
repeat 4 { move; process station }
```

Repeat 4 containing Move and Process runs eight primitive actions and processes each of four stations exactly once. Three written blocks express the routine. Processing a station twice fails; moving without processing leaves it unfinished.

## 2D teaching reference

Columns run west to east; rows run north to south. S = start, G = goal, M = mission item, X = barrier, P = route. Start heading: east. Use counters and this grid as an alternative to interpreting the 3D view.

| Grid | C1 | C2 | C3 | C4 | C5 |
|------|----|----|----|----|----|
| R1   | .  | .  | .  | .  | .  |
| R2   | .  | .  | .  | .  | .  |
| R3   | S  | PM | PM | PM | GM |
| R4   | .  | .  | .  | .  | .  |
| R5   | .  | .  | .  | .  | .  |

## Misconceptions and a useful response

| Misconception                                      | Response                                                                                                          |
|----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| Only Move belongs inside Repeat.                   | Use a two-card routine: move then process. Trace the pair four times. Ask which action was omitted from the loop. |
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence.   |



## Differentiate the task

**Support:** Start with one station, then repeat a physical two-card pattern. Keep the loop count visible. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Compare repeat 3 plus one move/process pair with repeat 4. Explain the same executed actions and different written structure.

## Independent exit check

**Prompt:** How many actions run when two actions are inside repeat 4?

**Teacher key:** Eight actions: four moves and four processing actions. A shorter written program can perform the same work.

## Connect to Scratch or physical robotics

In Scratch, a repeat block can contain movement plus a costume, sound, or variable update. Preserve the order of the whole repeated routine.



## EDUCATOR GUIDE 07

# Week 4: Cave Sentinel

## Essential question

How does a fresh scan help a robot decide to stop?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                                                |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can scan, conditionally move, and confirm a safe stop at a barrier.                                                                                                                                                                                                                                                                        |
| Evidence of learning | Reach C3 R3, record a blocked scan, and confirm a safe stop. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                                                                                             |
| Standards focus      | 2-3.CT.7, 2-3.CT.8, 2-3.NSD.1; 4-6.CT.7, 4-6.NSD.2 with a four-part systems model.                                                                                                                                                                                                                                                           |
| Vocabulary           | sensor, input, condition, false, state, output                                                                                                                                                                                                                                                                                               |
| Readiness            | Follow a short sequence and use a grid. Revisit the previous week's prediction and explanation before introducing this game.                                                                                                                                                                                                                 |
| Preparation          | Open game-week-04.html (Cave Sentinel). Check the four-legged cave scout and Crystal Sensor Cavern scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. Game awards and worksheet responses save locally on this device; use teacher records for assessment. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                                         | Look for                                                                     |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: How does a fresh scan help a robot decide to stop? Identify what must be observed to verify success.                                                                                                                     | Learners restate the task and constraints.                                   |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                                     | Each learner names an input, expected action, and observable result.         |
| 12-20   | <b>Activity 1: Investigate a scan</b><br>Open Cave Sentinel. Try Move without Scan and read the feedback. Add Scan before Move. Predict and test how the current-position reading changes after moving. Ask pairs to trace with a counter before retesting.                            | Collect the first worksheet prediction and observation.                      |
| 20-35   | <b>Activity 2: Build the safe scout</b><br>Inside Repeat 4, add Scan, then IF path clear with Move in THEN. Add Confirm stop after the loop. Predict which checks skip movement, then complete the game. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result. |



| Minutes | Teaching moves                                                                                                                                                                                                                                          | Look for                                                                        |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 35-43   | <b>Activity 3: Explain the system</b><br>Reset and run again, tracing scan input, IF processing, current scan storage, and movement/display output. Use the blocked case to explain why skipping Move is correct. Record an individual systems diagram. | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: What should happen after the scout scans the barrier? Allow a spoken answer, drawing, or written response without a partner supplying the explanation.                                               | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                                                                    | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

```
repeat 4 { scan; if clear { move } }; confirm stop
```

The scout scans and moves twice to C3 R3. The next two scans are blocked by the barrier at C4 R3, so IF skips Move. Confirm stop records the decision using the blocked reading. A scan is required at the current pose before a move or IF check.

## 2D teaching reference

Columns run west to east; rows run north to south. S = start, G = goal, M = mission item, X = barrier, P = route. Start heading: east. Use counters and this grid as an alternative to interpreting the 3D view.

| Grid | C1 | C2 | C3 | C4 | C5 |
|------|----|----|----|----|----|
| R1   | .  | .  | .  | .  | .  |
| R2   | .  | .  | .  | .  | .  |
| R3   | S  | P  | G  | X  | P  |
| R4   | .  | .  | .  | .  | .  |
| R5   | .  | .  | .  | .  | .  |

## Misconceptions and a useful response

| Misconception                                      | Response                                                                                                                            |
|----------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| A blocked IF is an error.                          | Point to the safe stopping tile. Compare skipping Move with trying to cross the barrier. The false condition is useful information. |
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence.                     |



## Differentiate the task

**Support:** Use clear/blocked cards and a tangible barrier. Offer the outer Repeat frame while learners choose scan and branch placement. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Explain the stored reading and why a changed position or heading would make it stale. Recreate the probe in Scratch with explicit values.

## Independent exit check

**Prompt:** What should happen after the scout scans the barrier?

**Teacher key:** The scan is blocked, so IF clear skips its move. Confirm stop records the decision at the safe tile.

## Connect to Scratch or physical robotics

The simulation has ideal map-based sensing. Real distance sensors can be noisy, and a safe classroom robot needs a tested threshold and stop behavior.



## EDUCATOR GUIDE 08

# Week 5: Color Cargo Sorter

## Essential question

How does one condition choose the correct cargo bin?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                                                |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can scan cargo and use IF / ELSE to sort a mixed batch.                                                                                                                                                                                                                                                                                    |
| Evidence of learning | Sort four items: RED goes left; BLUE goes right. Use a loop and IF / ELSE. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                                                                               |
| Standards focus      | 2-3.CT.8; 4-6.CT.7, 4-6.CT.8, 4-6.CT.9.                                                                                                                                                                                                                                                                                                      |
| Vocabulary           | IF, ELSE, branch, scan, color input, queue                                                                                                                                                                                                                                                                                                   |
| Readiness            | Follow a short sequence and use a grid. Revisit the previous week's prediction and explanation before introducing this game.                                                                                                                                                                                                                 |
| Preparation          | Open game-week-05.html (Color Cargo Sorter). Check the sorting-arm robot and Candy Color Sortworks scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. Game awards and worksheet responses save locally on this device; use teacher records for assessment. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                                                                             | Look for                                                                     |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: How does one condition choose the correct cargo bin? Identify what must be observed to verify success.                                                                                                                                                       | Learners restate the task and constraints.                                   |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                                                                         | Each learner names an input, expected action, and observable result.         |
| 12-20   | <b>Activity 1: Inspect the first parcel</b><br>Open Color Cargo Sorter. The first cargo is RED. Run Scan, then Send BLUE right. Predict and observe the wrong-bin feedback. Ask pairs to trace with a counter before retesting.                                                                                            | Collect the first worksheet prediction and observation.                      |
| 20-35   | <b>Activity 2: Sort the whole queue</b><br>Use Repeat 4 containing Scan and IF cargo RED. Put Send RED left in THEN and Send BLUE right in ELSE. Trace RED, BLUE, BLUE, RED; predict exactly one branch per parcel, then complete the batch. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result. |



| Minutes | Teaching moves                                                                                                                                                                                                 | Look for                                                                        |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 35-43   | <b>Activity 3: Find a branch error</b><br>Swap the two sorting actions and predict the first failure. Run, locate the first wrong bin in the trace, restore the branches, and retest.                          | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: Can both sorting branches run for the same scanned parcel? Allow a spoken answer, drawing, or written response without a partner supplying the explanation. | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                           | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

```
repeat 4 { scan; if RED { send left } else { send right } }
```

Four scans and four bin actions sort RED, BLUE, BLUE, RED into left, right, right, left. A repeat loop and IF RED / ELSE are required. Every new parcel must be scanned; a previous parcel's label cannot substitute.

## 2D teaching reference

Columns run west to east; rows run north to south. S = start, G = goal, M = mission item, X = barrier, P = route. Start heading: east. Use counters and this grid as an alternative to interpreting the 3D view.

| Grid | C1 | C2 | C3 | C4 | C5 |
|------|----|----|----|----|----|
| R1   | .  | .  | .  | .  | .  |
| R2   | .  | .  | P  | .  | .  |
| R3   | .  | P  | SG | .  | .  |
| R4   | .  | .  | P  | .  | .  |
| R5   | .  | .  | .  | .  | .  |

## Misconceptions and a useful response

| Misconception                                      | Response                                                                                                        |
|----------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| Both branches run on each item.                    | Physically choose one of two bins for a single card. Trace only the branch selected by the scanned color.       |
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence. |



## Differentiate the task

**Support:** Provide colored paper parcels and two labeled bins before the digital run. Read labels aloud; color is also written in text. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Compare four explicit scan-and-sort pairs with the conditional routine. Explain why the conditional is reusable for a changed color order.

## Independent exit check

**Prompt:** Can both sorting branches run for the same scanned parcel?

**Teacher key:** No. IF / ELSE selects one branch. A RED label chooses left; a BLUE label chooses right.

## Connect to Scratch or physical robotics

In Scratch, represent color input with a variable or color sensor probe and use IF / ELSE. The simulator's labels are authored task inputs, not camera recognition or AI.



## EDUCATOR GUIDE 09

# Week 6: Alpine Rescue

## Essential question

How do we plan a rescue that includes the return journey?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                                            |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can decompose a rescue, reuse loops, and meet a battery constraint.                                                                                                                                                                                                                                                                    |
| Evidence of learning | Collect both pods and unload at base using no more than 22 battery units. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                                                                            |
| Standards focus      | 2-3.CT.4, 2-3.CT.9, 2-3.CT.10; 4-6.CT.4, 4-6.CT.6, 4-6.CT.10.                                                                                                                                                                                                                                                                            |
| Vocabulary           | decomposition, constraint, battery, return, unload                                                                                                                                                                                                                                                                                       |
| Readiness            | Follow a short sequence and use a grid. Revisit the previous week's prediction and explanation before introducing this game.                                                                                                                                                                                                             |
| Preparation          | Open game-week-06.html (Alpine Rescue). Check the tracked rescue rover and Snowcap Rescue Base scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. Game awards and worksheet responses save locally on this device; use teacher records for assessment. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                                          | Look for                                                                     |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: How do we plan a rescue that includes the return journey? Identify what must be observed to verify success.                                                                                                               | Learners restate the task and constraints.                                   |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                                      | Each learner names an input, expected action, and observable result.         |
| 12-20   | <b>Activity 1: Plan the rescue budget</b><br>Open Alpine Rescue. Locate two pods and the shared start/base. Plan travel, rescue, return, and unload. Moves/rescues/unload cost 1 unit each; turns cost 0.5. Budget from 22. Ask pairs to trace with a counter before retesting.         | Collect the first worksheet prediction and observation.                      |
| 20-35   | <b>Activity 2: Rescue and return</b><br>Build perimeter travel with Repeat 4 groups. Rescue at C5 R5 and C5 R1. Return to C1 R5, then Unload at base. Predict remaining battery and complete the mission. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result. |



| Minutes | Teaching moves                                                                                                                                                                                                                 | Look for                                                                        |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 35-43   | <b>Activity 3: Investigate a limited resource</b><br>Add three extra right turns and three left turns at the start. Predict the battery effect. Run and compare with the original route. Remove the wasted actions and retest. | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: Why is reaching the second pod not the end of the rescue? Allow a spoken answer, drawing, or written response without a partner supplying the explanation.                  | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                                           | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

```
repeat 4 { move }; rescue; left
repeat 4 { move }; rescue; left
repeat 4 { move }; left
repeat 4 { move }; left; unload
```

The model route uses 16 moves, 4 left turns, 2 rescue actions, and 1 unload: 23 actions costing 21 battery units. One unit remains. Merely collecting both pods does not complete the rescue; unload requires both pods at base.

## 2D teaching reference

Columns run west to east; rows run north to south. S = start, G = goal, M = mission item, X = barrier, P = route. Start heading: east. Use counters and this grid as an alternative to interpreting the 3D view.

| Grid | C1 | C2 | C3 | C4 | C5 |
|------|----|----|----|----|----|
| R1   | P  | P  | P  | P  | PM |
| R2   | P  | .  | .  | .  | P  |
| R3   | P  | .  | .  | .  | P  |
| R4   | P  | .  | .  | .  | P  |
| R5   | SG | P  | P  | P  | PM |

## Misconceptions and a useful response

| Misconception                      | Response                                                                                             |
|------------------------------------|------------------------------------------------------------------------------------------------------|
| The rescue ends at the second pod. | Ask who still needs transport and where the base is. Include return and unload in the sub-task plan. |



| Misconception                                      | Response                                                                                                        |
|----------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence. |

## Differentiate the task

**Support:** Give a perimeter map and a 22-unit token budget. Debit half-units with pairs of tokens or a number line. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Remove the final left turn before Unload: the rover can unload facing south, so 1.5 units remain. Compare a valid alternative without claiming it is globally optimal.

## Independent exit check

**Prompt:** Why is reaching the second pod not the end of the rescue?

**Teacher key:** The rover must carry both pods back and unload at base. Plan battery for the return trip.

## Connect to Scratch or physical robotics

A real robot's energy use is not a fixed per-action token cost. Treat the battery as a simplified planning model and identify factors it omits.



## EDUCATOR GUIDE 10

# Week 7: Orbital AI Trials

## Essential question

What evidence shows that an AI can handle a new input?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                                    |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can label examples, train a decision tree, and test three unseen inputs.                                                                                                                                                                                                                                                       |
| Evidence of learning | Train on at least four examples and pass all three unseen AI trials. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                                                                         |
| Standards focus      | Supporting investigation: 2-3.NSD.1; 4-6.CT.7, 4-6.CT.9 through explicit input/output tracing. AI is an enrichment context, not an additional claimed state standard.                                                                                                                                                            |
| Vocabulary           | training example, label, decision tree, held-out input, prediction, test                                                                                                                                                                                                                                                         |
| Readiness            | Follow a short sequence and use a grid. Revisit the previous week's prediction and explanation before introducing this game.                                                                                                                                                                                                     |
| Preparation          | Open game-week-07.html (Orbital AI Trials). Check the four-rotor ai drone and Orbital Learning Station scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. AI training labels and trial passes reset on reload; keep evidence in the worksheet. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                                                            | Look for                                                                     |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: What evidence shows that an AI can handle a new input? Identify what must be observed to verify success.                                                                                                                                    | Learners restate the task and constraints.                                   |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                                                        | Each learner names an input, expected action, and observable result.         |
| 12-20   | <b>Activity 1: Train your drone</b><br>Open Orbital AI Trials. Inspect training scenes in 3D. Label at least four varied scenes using at least two actions, then Train my AI. Explain a label before training. Ask pairs to trace with a counter before retesting.                                        | Collect the first worksheet prediction and observation.                      |
| 20-35   | <b>Activity 2: Review unseen predictions</b><br>For each of the three test chambers, choose your predicted action. Review AI prediction and its learned path, then Test AI action in 3D. Record the independent task check. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result. |



| Minutes | Teaching moves                                                                                                                                                                                                                                                                 | Look for                                                                        |
|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 35-43   | <b>Activity 3: Improve and verify</b><br>If a trial fails, review and correct a justified training label; retrain and test all three trials again. Complete the three unseen trials. Explain a learned split and why leaf label agreement is not a probability of correctness. | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: Does perfect agreement on training examples prove that a new prediction is right? Allow a spoken answer, drawing, or written response without a partner supplying the explanation.                                          | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                                                                                           | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

Label four varied training scenes. Train.

For each unseen trial: predict -> review AI -> test its action.

A useful four-scene set is 100>Move, 010>Right, 001>Left, 000>Stop (bits = ahead/right/left clear). A learned tree then passes held-out 110 and 101 with Move and 011 with an open-side turn. Trial inputs cannot be used as training scenes. The independent rule checks the prediction; it never replaces it. A label change clears the current trial passes and requires retraining.

## Misconceptions and a useful response

| Misconception                                      | Response                                                                                                                                       |
|----------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| The answer check is the AI.                        | Compare the learned tree prediction with the separate rule check. Deliberately mislabel a training example and observe a wrong learned result. |
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence.                                |

## Differentiate the task

**Support:** Use path cards and one label at a time. Keep the explicit task rule visible. Allow spoken prediction and teacher-assisted selection. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Add the optional 111 training scene or deliberately mislabel one scene, then explain changes in tree paths and test results. Identify the tiny dataset as a limitation.

## Independent exit check

**Prompt:** Does perfect agreement on training examples prove that a new prediction is right?

**Teacher key:** No. Test new inputs and independently check the action. Training-label agreement is not a probability of correctness.



## Connect to Scratch or physical robotics

A Scratch IF / ELSE is an explicitly authored rule. This local supervised model learns a rule from labels. Neither this activity nor the task check is an online conversational assistant.



## EDUCATOR GUIDE 11

# Week 8: Robot Maker Lab

## Essential question

How can a controlled comparison explain a model's output?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                                                    |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can assemble a robot model and predict outputs when only repeat count changes.                                                                                                                                                                                                                                                                 |
| Evidence of learning | Assemble three modules, then pass two calibration tests with correct endpoint predictions. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                                                                   |
| Standards focus      | Supporting controlled-test practice: 4-6.CT.7. Add the learner-authored Scratch extension before collecting evidence for 4-6.CT.1 or 4-6.DL.4.                                                                                                                                                                                                   |
| Vocabulary           | module, model, parameter, input, output, controlled comparison                                                                                                                                                                                                                                                                                   |
| Readiness            | Follow a short sequence and use a grid. Revisit the previous week's prediction and explanation before introducing this game.                                                                                                                                                                                                                     |
| Preparation          | Open game-week-08.html (Robot Maker Lab). Check the modular walking robot and Blueprint Builder Studio scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. Game awards and worksheet responses save locally on this device; use teacher records for assessment. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                         | Look for                                                                     |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: How can a controlled comparison explain a model's output? Identify what must be observed to verify success.                                                                                              | Learners restate the task and constraints.                                   |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                     | Each learner names an input, expected action, and observable result.         |
| 12-20   | <b>Activity 1: Assemble and inspect</b><br>Open Robot Maker Lab. Attach chassis, leg module, and sensor head; inspect the changing 3D robot. Identify each model part and one aspect it simplifies. Ask pairs to trace with a counter before retesting.                | Collect the first worksheet prediction and observation.                      |
| 20-35   | <b>Activity 2: Calibrate Test A</b><br>Keep movement at one tile per action and Repeat 3 from C1 R3. Predict the endpoint, then Run calibration. Record the observation and pass Test A. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result. |



| Minutes | Teaching moves                                                                                                                                                                                                             | Look for                                                                        |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 35-43   | <b>Activity 3: Compare Test B</b><br>Keep start, heading, and one-tile movement fixed. Change only Repeat from 3 to 2. Predict and run Test B. Complete both calibrations, then describe why this was a fair comparison.   | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: Which inputs must stay fixed to compare repeat 3 with repeat 2 fairly? Allow a spoken answer, drawing, or written response without a partner supplying the explanation. | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                                       | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

Attach a chassis, movement module, and sensor.

Test A: repeat 3 -> C4.

Test B: same start and module; repeat 2 -> C3.

All three module buttons change the robot assembly. Test A requires one tile per move, repeat 3, and a C4 prediction; Test B requires the same start and movement scale, repeat 2, and a C3 prediction. Wrong predictions or different parameters produce feedback and do not pass the calibration. The supplied model is not evidence of independently authoring a computational model.

## 2D teaching reference

Columns run west to east; rows run north to south. S = start, G = goal, M = mission item, X = barrier, P = route. Start heading: east. Use counters and this grid as an alternative to interpreting the 3D view.

| Grid | C1 | C2 | C3 | C4 | C5 |
|------|----|----|----|----|----|
| R1   | .  | .  | .  | .  | .  |
| R2   | .  | .  | .  | .  | .  |
| R3   | S  | P  | P  | G  | P  |
| R4   | .  | .  | .  | .  | .  |
| R5   | .  | .  | .  | .  | .  |

## Misconceptions and a useful response

| Misconception                               | Response                                                                                                                   |
|---------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| Changing several inputs explains the cause. | Keep a two-column input record. Circle repeat count as the only difference; keep start, heading, and movement scale fixed. |



| Misconception                                      | Response                                                                                                        |
|----------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence. |

## Differentiate the task

**Support:** Provide a five-cell strip and move a counter before each digital prediction. Let learners name modules orally. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Author a rover sprite and trail in Scratch, define move 40 as one tile, compare equivalent scripts, and test a changed repeat count. Save the authored project; this adds model-development evidence toward 4-6.CT.1.

## Independent exit check

**Prompt:** Which inputs must stay fixed to compare repeat 3 with repeat 2 fairly?

**Teacher key:** Keep the same starting position, heading, and one-tile movement module. Change only repeat count.

## Connect to Scratch or physical robotics

Scratch extension: create a sprite, start position and direction, movement scale, and repeat script. The browser module assembly alone is guided configuration; a learner-created Scratch model provides stronger authorship evidence.



## EDUCATOR GUIDE 12

# Week 9: Mars Mission Designer

**Essential question**

How do design changes and tests support a claim that a mission works?

| Planning element     | Teacher notes                                                                                                                                                                                                                                                                                                                               |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Learning target      | I can edit a mission map, compare three tests, and explain a successful final controller.                                                                                                                                                                                                                                                   |
| Evidence of learning | Edit at least two tiles, place two samples, run three distinct tests, and finish with a successful loop-and-decision program. Keep individual predictions, observations, and an explanation; a completion badge alone does not establish mastery.                                                                                           |
| Standards focus      | 2-3.CT.10; 4-6.CT.4, 4-6.CT.8, 4-6.CT.9, 4-6.CT.10; 4-6.DL.2 with a justified collaboration/tool choice.                                                                                                                                                                                                                                    |
| Vocabulary           | design brief, constraint, test case, revision, evidence, limitation                                                                                                                                                                                                                                                                         |
| Readiness            | Follow a short sequence and use a grid. Revisit the previous week's prediction and explanation before introducing this game.                                                                                                                                                                                                                |
| Preparation          | Open game-week-09.html (Mars Mission Designer). Check the solar hexapod and Red Planet Expedition scene. Prepare a 5-by-5 grid, arrow counter, and worksheet. Use one device per pair and review the goal before revealing a solution. Game awards and worksheet responses save locally on this device; use teacher records for assessment. |

## 50-minute lesson sequence

| Minutes | Teaching moves                                                                                                                                                                                                                                                                                                 | Look for                                                             |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| 0-5     | <b>Introduce the mission</b><br>Read the story and goal. Ask: How do design changes and tests support a claim that a mission works? Identify what must be observed to verify success.                                                                                                                          | Learners restate the task and constraints.                           |
| 5-12    | <b>Model one action or test</b><br>Demonstrate one control without solving the whole game. Explain the difference between robot heading and camera view. Ask for a prediction before clicking Run.                                                                                                             | Each learner names an input, expected action, and observable result. |
| 12-20   | <b>Activity 1: Design a Mars mission</b><br>Open Mars Mission Designer. Use Trail tool to change at least two trail tiles; keep a connected route to the goal. Use Sample tool to place at least two samples. State your goal, constraints, and sub-tasks. Ask pairs to trace with a counter before retesting. | Collect the first worksheet prediction and observation.              |



| Minutes | Teaching moves                                                                                                                                                                                                                                                                                                                                                 | Look for                                                                        |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| 20-35   | <b>Activity 2: Run three different tests</b><br>Build a loop with IF clear: Move; ELSE: Turn left. Plan short and longer repeat-count tests. Run at least three distinct program/map combinations. Record predicted and actual outcomes; identical reruns do not add a new test. Swap driver and navigator after one run. Pause at the first unexpected state. | Both partners connect a block, label, or parameter with the observed result.    |
| 35-43   | <b>Activity 3: Demonstrate and explain</b><br>Finish with a successful run that collects all samples and reaches the goal using a loop and IF / ELSE. Present the final map, program, test evidence, revision, and limitation. Each partner independently explains a design choice.                                                                            | A revision or comparison is supported by a recorded test.                       |
| 43-48   | <b>Independent exit check</b><br>Ask each learner: What evidence supports your claim that the final mission design works? Allow a spoken answer, drawing, or written response without a partner supplying the explanation.                                                                                                                                     | Use the model answer as a guide; accept other explanations grounded in the run. |
| 48-50   | <b>Save evidence and reflect</b><br>Record completion and a next teaching step. Ask learners to download their worksheet or use a teacher-approved classroom record.                                                                                                                                                                                           | A next test or remaining limitation is named.                                   |

## Model program and expected behavior

Example for the starter route:

```
repeat 12 { if clear { move } else { left } }
```

Design changes need their own prediction and tests.

For an accessible starter design, add trail at C2 R4 and C3 R4, then place samples at C3 R5 and C5 R3. Test repeat counts 4, 8, and 12 with IF clear: Move / ELSE: Left. The final run reaches C5 R1 after eight moves and one left turn; goal checking ends the program early. Three distinct tests, two actual trail changes, at least two samples, and a successful loop-and-decision run are all required. Other maps may need other controllers.

## 2D teaching reference

Columns run west to east; rows run north to south. S = start, G = goal, M = mission item, X = barrier, P = route. Start heading: east. Week 9 shows example sample locations; learners place samples with the editor.

| Grid | C1 | C2 | C3 | C4 | C5 |
|------|----|----|----|----|----|
| R1   | .  | .  | .  | .  | G  |
| R2   | .  | .  | .  | .  | P  |
| R3   | .  | .  | .  | .  | PM |
| R4   | .  | .  | .  | .  | P  |
| R5   | S  | P  | PM | P  | P  |

## Misconceptions and a useful response



| Misconception                                      | Response                                                                                                                         |
|----------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| One successful run proves every map works.         | Keep a record of the specific map, start, program, and samples tested. Ask for a boundary or counterexample before generalizing. |
| Finishing the game proves I understand everything. | Ask for an individual prediction, trace, explanation, and a next test. Use completion as one piece of evidence.                  |

## Differentiate the task

**Support:** Offer two suggested safe side tiles, but require the learner to place samples, predict tests, and explain the final revision. Use enlarged grids and dictated notes. Provide the written position/heading status and a 2D or tactile map when the 3D view is not useful.

**Extend:** Create a route where the starter left-turn controller fails. Design and test a different algorithm, then transfer the custom map to a learner-authored Scratch project.

## Independent exit check

**Prompt:** What evidence supports your claim that the final mission design works?

**Teacher key:** A recorded map revision, three distinct tests, a successful final run, and an explanation of a limitation.

## Connect to Scratch or physical robotics

The editable Mars map is a design artifact. Extend it in Scratch with an explicit goal, sensor probe, step limit, and saved project. Physical robotics would require different motion and sensor calibration.

### EDUCATOR GUIDE 13

# Assess reasoning, not just a finished game

## A short diagnostic before Lesson 1

- Ask the learner to follow three spoken or pictured instructions in order. Look for sequence, not reading speed.
- Show an east-facing arrow turning left. Ask the learner to move or rotate a counter to show what changes.
- Show one clap inside repeat 3. Ask for a demonstration or prediction.
- Ask what the word if changes in a familiar direction such as "if it rains, take an umbrella."
- Present an incorrect two-step route. Ask what information would help locate the mistake.

Use the diagnostic to select supports, not to rank learners or exclude them from programming. Record what they can explain with and without prompts.

## Evidence collection across the unit



| Checkpoint        | Keep                                                                              | Instructional response                                                             |
|-------------------|-----------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
| After Lessons 1-2 | One independent endpoint/heading prediction and a route plan.                     | If heading and movement are confused, return to a physical arrow counter.          |
| After Lessons 3-4 | Flat-versus-loop comparison and a true/false trace.                               | If only true cases are understood, deliberately test a blocked false case.         |
| After Lesson 5    | A trace containing both branches.                                                 | If both branches are executed, cover the unselected branch and enact one decision. |
| After Lesson 6    | Plan, program, test record, revision explanation, and final independent response. | Use the rubric and identify one next learning goal for each learner.               |

## Four-level analytic rubric

Score each dimension separately from 1 to 4. Levels: 1 = needs substantial modeling; 2 = succeeds with prompts; 3 = explains and completes independently in this task; 4 = justifies and transfers to a meaningful new case. This is a teacher-created rubric, not a NYSED state scoring scale.

| Dimension                                  | Evidence at level 3                                                                               | What distinguishes level 4                                                            |
|--------------------------------------------|---------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------|
| Sequence and direction                     | Traces the order and distinguishes a turn from a move.                                            | Explains a changed starting heading or an alternative route.                          |
| Loops and decisions                        | Explains the repeated body and selects the correct IF/ELSE branch.                                | Predicts a changed repeat count or identifies where this controller will fail.        |
| Debugging and design                       | Uses an observation to locate a wrong step and justify a revision.                                | Designs a useful additional test and explains what it does or does not establish.     |
| Systems and model limits                   | Explains a sensor input, a decision, an output, and stored information with the systems activity. | Compares the simulated safe-path reading with limitations of physical sensing.        |
| Communication and individual understanding | Explains a program using a trace, demonstration, or equivalent accessible response.               | Chooses an effective representation for the audience and answers a transfer question. |

A suggested next step for a learner at level 2 is a smaller worked example followed by a new independent check. Do not turn a summed rubric score into a state proficiency designation. Pair work should include individual evidence.

## Optional six-mission simulator: worked Mission 5 trace

C means column and R means row, both numbered 1-5. Energy is the total after that action. Turning stays on the same tile. Eight decisions produce six moves and two turns.

| Loop | Before | Heading | Sensor | Action | After | Energy |
|------|--------|---------|--------|--------|-------|--------|
| 1    | C1 R2  | East    | Clear  | Move   | C2 R2 | 0      |



| Loop | Before | Heading | Sensor  | Action | After | Energy |
|------|--------|---------|---------|--------|-------|--------|
| 2    | C2 R2  | East    | Clear   | Move   | C3 R2 | 1      |
| 3    | C3 R2  | East    | Blocked | Right  | C3 R2 | 1      |
| 4    | C3 R2  | South   | Clear   | Move   | C3 R3 | 1      |
| 5    | C3 R3  | South   | Clear   | Move   | C3 R4 | 2      |
| 6    | C3 R4  | South   | Blocked | Right  | C3 R4 | 2      |
| 7    | C3 R4  | West    | Clear   | Move   | C2 R4 | 2      |
| 8    | C2 R4  | West    | Clear   | Move   | C1 R4 | 3      |

## A useful final conference

- Point to one block and explain what happens before and after it.
- Choose one blocked case and explain the branch taken.
- Show one prediction that changed after testing.
- Explain one limitation of the controller or simulator.
- Name the next test you would run and what it would tell you.

## EDUCATOR GUIDE 14

# Scratch and robotics transfer

### Optional learner-authored Scratch model

Week 8 provides guided assembly and controlled tests. Add the Scratch work below to collect evidence of learner model development. Allow an additional session if learners need more time. The safe-path condition is not a native Scratch block. A teacher must build and test the path-probe mechanism before asking learners to reason about it. Scratch: official ideas and tutorials

| Stage                | Teacher setup and learner task                                                                                                                          | Evidence                                                         |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------|
| 1. Represent (8 min) | Create a top-down trail and a rover sprite. Agree that one tile represents 40 Scratch steps. Learners label the model's essential parts.                | Explain how the representation simplifies a robot and its route. |
| 2. Control (10 min)  | Set a known starting position and direction when the flag is clicked. Build a short move/turn script, then replace repeated actions with a repeat loop. | Compare two programs from the same initial state.                |



| Stage                       | Teacher setup and learner task                                                                                                                                                                                                                                         | Evidence                                                                                     |
|-----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|
| 3. Sense (12 min)           | Use a separate probe sprite positioned one tile ahead. The probe tests the chosen trail/obstacle representation and records a clear/blocked value. Coordinate sensing and movement with a completed broadcast-and-wait cycle or another tested synchronization design. | Explain how the custom Boolean value is produced and why it must be fresh.                   |
| 4. Change and test (10 min) | Have learners deliberately change an input: start heading, path layout, or repeat count. Predict and record an output change.                                                                                                                                          | A student-created or modified computational model with a documented input/output comparison. |
| 5. Explain (5 min)          | Compare the model with physical sensing and motor control.                                                                                                                                                                                                             | Name one useful simplification and one limitation.                                           |

A probe that samples only one point can miss narrow edges; sprite costumes, rotation centers, and color matching affect results. A single Scratch sprite cannot simply ask an arbitrary offset point for a color using the ordinary touching-color test. Test the probe implementation at corners and boundaries before use. Learners may first transfer only sequence, loops, and turns if a sensing model is not yet ready.

## Physical robotics: adapt to the selected kit

- Choose the hardware and its supported programming extension before planning motor or sensor blocks. This guide deliberately avoids pretending one device-specific block set works for every kit.
- Have learners identify input, processor/controller, storage, and motor/display outputs. Map these to the four-part systems model.
- Calibrate forward travel and quarter-turns on a floor-level test area. Compare measured results with the simulator's exact tile motion.
- For a distance sensor, collect readings at several distances and angles, then choose and justify a threshold. A range sensor does not, by itself, prove a floor edge is safe.
- Use low-speed, supervised tests with a clear stop procedure. Keep moving parts and cables clear, and follow the kit's handling and setup instructions.

## Optional data investigation

Use a teacher-approved spreadsheet to record decisions, forward moves, and turns across Missions 4-6. The supplied model programs produce 1/0, 6/2, and 10/4 forward-move/right-turn totals, respectively. Record the sensor-check count separately. Build a chart and explain what it shows. This provides additional work toward 4-6.CT.2 and 4-6.CT.3 when learners collect and organize digital data for an explicit question. See NYSED standards, grades 4-6 (p. 4).

## K-12 progression: keep the task developmentally appropriate

| NYSED band | Adaptation or next step                                                                                                                                | Verified connection          |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------|
| K-1        | Use three arrow cards, a physical rover counter, and a short repeated pattern. Collaboratively find one error. Do not require independent nested code. | K-1.CT.6, K-1.CT.8, K-1.CT.9 |



| NYSED band | Adaptation or next step                                                                                                                                                                                    | Verified connection           |
|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|
| 2-3        | Prioritize precise sequences, two ways to describe the same task, conditional meaning, and a documented plan.                                                                                              | 2-3.CT.6, 2-3.CT.8, 2-3.CT.10 |
| 4-6        | Compare algorithms; work with loops and conditions; trace and explain design choices. Use model-building and systems extensions when assessing those standards.                                            | 4-6.CT.6, 4-6.CT.8-10         |
| 7-8        | Move to a richer Scratch or robotics project that combines control structures and predicts outcomes under varied inputs. Add a learner-controlled state variable.                                          | 7-8.CT.7-9                    |
| 9-12       | Use a more advanced environment, complex conditions, and a planned test suite for sensor faults, different starts, and user behavior. The beginner simulator alone does not establish high-school mastery. | 9-12.CT.8, 9-12.CT.9          |

## EDUCATOR GUIDE 15

# Differentiation and classroom access

| Learner need                          | Keep the learning goal; change the access route                                                                                                                                                            |
|---------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Early readers / multilingual learners | Preteach a small vocabulary set with pictures and actions. Read blocks aloud. Allow home-language planning, pointing, arrows, or dictated explanations before an English technical label is added.         |
| Motor or input access                 | Use click-to-add and labeled controls instead of dragging. Offer a partner acting as a scribe while the learner gives precise instructions and independently explains the logic.                           |
| Visual access                         | Enlarge text and controls. Provide a high-contrast 2D route map, tactile counter, and a spoken description of position, heading, and sensor state. Do not rely on color or camera interpretation alone.    |
| Working memory / executive function   | Present one goal at a time; use a visible plan-test-record checklist; provide a partially completed trace; offer a predictable role-change cue.                                                            |
| Learners ready to extend              | Require a counterexample, alternative algorithm, new test case, or Scratch model. Extra speed or more repetitive levels is not the only extension.                                                         |
| Assessment accommodations             | Accept equivalent oral, physical, drawn, or recorded evidence consistent with the learner's IEP/504 plan and local practices. Preserve the reasoning target rather than requiring the same written format. |

## Collaboration and digital responsibility

Use respectful feedback: "I predicted..., I noticed..., could we test...?" Separate the role of typing from the role of thinking. Use first names or classroom identifiers on teacher-held records. Students do not need an account to use Rover Lab; any Scratch accounts, online sharing, recordings, or third-party tools should follow the school's established procedures. The certificate name field is optional and should not be treated as a student information system.



## Cross-curricular opportunities

Connect quarter-turns, counting, repeated addition, and route diagrams to mathematics; use claim-evidence explanations in speaking and writing; and connect sensor observations with scientific modeling. These are suggested connections rather than an additional formal mathematics or ELA standards crosswalk.



## EDUCATOR GUIDE 16

# AI teaching notes: examples, predictions, and evidence

**Week 7: Orbital AI Trials**

Open Orbital AI Trials and use the full 50-minute Week 7 plan. Learners label training scenes, train a local decision tree, predict and review an unseen action, then explicitly trigger its 3D test. This is a supervised classifier implemented in the browser.

**Four teaching methods used in the course**

| Method                     | Application                                                                                                                            | Evidence                                               |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| Contextual instruction     | The robot and world provide a reason to program: delivery, docking, sorting, rescue, or investigation. Tie the story to the task rule. | A learner explanation of the goal and constraints.     |
| Task-driven learning       | Break the game into planning, testable actions, observations, and a final challenge. Name sub-tasks before Run.                        | A plan and trace connecting actions to results.        |
| Game-based learning        | Use an observable goal, feedback, retry, and a new test. Value explanation and useful debugging.                                       | A prediction, correction, and independent explanation. |
| AI-supported investigation | Train on labeled examples, inspect the learned prediction, independently check it, and improve labels when justified.                  | Training data, new-case tests, and a limitation.       |

**Explain the mechanism accurately**

- Inputs are three Boolean map checks: ahead, right, and left. Clear indicates an available neighboring path tile; these are ideal map checks, not measured distances.
- Learners supply Move, Right, Left, or Stop labels. At least four different labeled training scenes and at least two action labels are required.
- Training fits a small decision tree by choosing feature splits that reduce label impurity. Each leaf predicts its most common action label; ties use a fixed ordering.
- The panel displays the learned decision path and supporting label count. Agreement is not a calibrated confidence or a probability of correctness.
- The independent task rule checks the output; it cannot replace a wrong prediction with a correct answer. Wrong labels can produce wrong model actions.
- The task is small: move if ahead is clear; otherwise turn toward an open side; stop if all three paths are blocked. Either turn is acceptable when both sides are open. This is not whole-route planning.
- Training and prediction run locally without an account, API key, voice recording, or external AI service. AI labels and trial passes reset on reload. Completion awards and worksheet responses can persist locally.



## Training and held-out tests

| Ahead / right / left        | Useful label  | Purpose                        |
|-----------------------------|---------------|--------------------------------|
| Clear / Blocked / Blocked   | Move          | Training: open front.          |
| Blocked / Clear / Blocked   | Right         | Training: only right is open.  |
| Blocked / Blocked / Clear   | Left          | Training: only left is open.   |
| Blocked / Blocked / Blocked | Stop          | Training: all three blocked.   |
| Clear / Clear / Clear       | Move          | Optional fifth training scene. |
| Clear / Clear / Blocked     | Move          | Held-out trial 1.              |
| Clear / Blocked / Clear     | Move          | Held-out trial 2.              |
| Blocked / Clear / Clear     | Right or Left | Held-out trial 3.              |

The three trial combinations cannot be labeled in training. Passing these trials is evidence about this small test set, not a general guarantee. Changing a label clears the current model and trial passes; learners retrain and repeat the trials.

## A useful counterexample

Label front-clear 100 as Right instead of Move, while labeling 010 Right, 001 Left, and 000 Stop. Train and test the held-out front-clear scenes. The model can suggest a turn that fails the separate task check. Correct the justified label, retrain, and compare the same tests. Adding arbitrary data does not guarantee improvement.

## Prompts for discussion

- Before training: Which information is an input, and which is the action label? Why does the label fit?
- Before review: What do you predict for this new combination?
- After review: Which learned conditions led to this output? What does the label count mean?
- After testing: Did the learned action meet the task rule? What is your evidence?
- After revision: What changed in the training set, and what happened on the same test?
- Transfer: How is this learned rule different from writing a Scratch IF / ELSE yourself?

## AI test record

| Trial input | My prediction | AI prediction / path | Task check and revision |
|-------------|---------------|----------------------|-------------------------|
|             |               |                      |                         |
|             |               |                      |                         |
|             |               |                      |                         |



Exit answer: training performance does not prove new-case accuracy. Test held-out inputs, compare against the task goal, and identify the tiny dataset and limited action task as limitations.

## Supplemental sandbox

The optional older six-mission simulator retains a separate AI Rover Lab with freely chosen inputs and reviewed block suggestions. Its controls and saved examples differ. Use Orbital AI Trials for the three held-out 3D challenges and the Week 7 assessment.



## CLASSROOM REPRODUCIBLE

# Reproducible A: plan before Run

Learner / classroom identifier: \_\_\_\_\_ Date: \_\_\_\_\_ Mission: \_\_\_\_\_

## 1. Define the problem

My rover must: \_\_\_\_\_

A constraint I must remember: \_\_\_\_\_

## 2. Plan and predict

Starting position / heading: \_\_\_\_\_

My predicted endpoint / energy total: \_\_\_\_\_

My planned instructions or diagram:

---

---

---

---

## 3. Explain before testing

The block or condition I need to watch is: \_\_\_\_\_

I predict this because: \_\_\_\_\_

## 4. Compare after testing

What actually happened: \_\_\_\_\_

One next test or revision, and why: \_\_\_\_\_



## CLASSROOM REPRODUCIBLE

# Reproducible B: trace a decision loop

Learner: \_\_\_\_\_ Mission: \_\_\_\_\_ Starting heading: \_\_\_\_\_

Record the state before the decision, choose one action, then record what changed. A turn changes heading but not position. For an IF-only false case, write "skip."

| Loop | Position before | Heading before | Clear / blocked | Selected action | Position after |
|------|-----------------|----------------|-----------------|-----------------|----------------|
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |
|      |                 |                |                 |                 |                |

## Explain a branch

At iteration \_\_\_\_\_, the condition was \_\_\_\_\_, so the program \_\_\_\_\_.

After a turn, I checked again because: \_\_\_\_\_

## Count carefully

Decision iterations: \_\_\_\_\_ Forward moves: \_\_\_\_\_ Turns: \_\_\_\_\_

Written blocks (exclude the flag): \_\_\_\_\_ Explain why these counts can differ: \_\_\_\_\_



## CLASSROOM REPRODUCIBLE

# Reproducible C: debugging and test log

Learner: \_\_\_\_\_ Mission / program: \_\_\_\_\_

| Test | Prediction | Observed behavior | One change / next test | Reason |
|------|------------|-------------------|------------------------|--------|
| 1    |            |                   |                        |        |
| 2    |            |                   |                        |        |
| 3    |            |                   |                        |        |

## Locate the first divergence

The first step where my prediction and observation differed was: \_\_\_\_\_

The position, heading, sensor, and block at that moment were: \_\_\_\_\_

## Use evidence

I changed \_\_\_\_\_ because I observed \_\_\_\_\_.

The next run showed \_\_\_\_\_.

## Explain the limit of the test

This successful test shows \_\_\_\_\_.

It does not yet show \_\_\_\_\_.

## Partner check

Driver: \_\_\_\_\_ Navigator: \_\_\_\_\_ Role change time: \_\_\_\_\_

Each partner's individual explanation: \_\_\_\_\_



## CLASSROOM REPRODUCIBLE

# Reproducible D: independent exit cards

## 1. Week 1

Which instruction changes the parcel from waiting to being carried?

Name: \_\_\_\_\_ My response: \_\_\_\_\_

## 2. Week 2

Why can a boat reach the dock but still fail the docking check?

Name: \_\_\_\_\_ My response: \_\_\_\_\_

## 3. Week 3

How many actions run when two actions are inside repeat 4?

Name: \_\_\_\_\_ My response: \_\_\_\_\_

## 4. Week 4

What should happen after the scout scans the barrier?

Name: \_\_\_\_\_ My response: \_\_\_\_\_

## 5. Week 5

Can both sorting branches run for the same scanned parcel?

Name: \_\_\_\_\_ My response: \_\_\_\_\_

## 6. Week 6

Why is reaching the second pod not the end of the rescue?

Name: \_\_\_\_\_ My response: \_\_\_\_\_



## 7. Week 7

Does perfect agreement on training examples prove that a new prediction is right?

Name: \_\_\_\_\_ My response: \_\_\_\_\_

## 8. Week 8

Which inputs must stay fixed to compare repeat 3 with repeat 2 fairly?

Name: \_\_\_\_\_ My response: \_\_\_\_\_

## 9. Week 9

What evidence supports your claim that the final mission design works?

Name: \_\_\_\_\_ My response: \_\_\_\_\_



## CLASSROOM REPRODUCIBLE

# Reproducible E: systems model and assessment record

Learner: \_\_\_\_\_ Date: \_\_\_\_\_ Evidence reviewed: \_\_\_\_\_

## Draw the connections

| Input              | Processing         | Storage            | Output             |
|--------------------|--------------------|--------------------|--------------------|
| Examples / arrows: | Examples / arrows: | Examples / arrows: | Examples / arrows: |

What is real hardware here, and what is a simulation? \_\_\_\_\_

One difference from a physical robot: \_\_\_\_\_

## Teacher assessment record

| Dimension              | Level 1-4 | Observed evidence | Next teaching step |
|------------------------|-----------|-------------------|--------------------|
| Sequence / direction   |           |                   |                    |
| Loops / decisions      |           |                   |                    |
| Debugging / design     |           |                   |                    |
| Systems / model limits |           |                   |                    |
| Communication          |           |                   |                    |

Independently demonstrated: \_\_\_\_\_

Support or accommodation used: \_\_\_\_\_

Next check and date: \_\_\_\_\_



## EDUCATOR GUIDE 22

# Optional practice: six-mission simulator maps

These maps belong to the optional older Rover simulator, not the nine weekly games. The weekly lesson plans contain their own 2D references. Columns run west to east; rows run north to south. S = start, G = goal, E = energy, X = crate, P = permitted trail, and a dot = no permitted trail. GE means a crystal is on the goal; PX is a trail tile occupied by a crate. All starts face east. The numbered grid is a teaching aid.

## 1. First steps

| N  | C1 | C2 | C3 | C4 | C5 |
|----|----|----|----|----|----|
| R1 | .  | .  | .  | .  | .  |
| R2 | .  | .  | .  | .  | .  |
| R3 | S  | P  | P  | GE | .  |
| R4 | .  | .  | .  | .  | .  |
| R5 | .  | .  | .  | .  | .  |

## 2. Take a turn

| N  | C1 | C2 | C3 | C4 | C5 |
|----|----|----|----|----|----|
| R1 | .  | .  | .  | .  | .  |
| R2 | .  | .  | GE | .  | .  |
| R3 | .  | .  | P  | .  | .  |
| R4 | S  | P  | E  | .  | .  |
| R5 | .  | .  | .  | .  | .  |

## 3. Loop the loop

| N  | C1 | C2 | C3 | C4 | C5 |
|----|----|----|----|----|----|
| R1 | .  | .  | .  | .  | .  |
| R2 | .  | .  | .  | .  | .  |
| R3 | S  | P  | E  | P  | GE |
| R4 | .  | .  | .  | .  | .  |
| R5 | .  | .  | .  | .  | .  |

## 4. Robot senses

| N  | C1 | C2 | C3 | C4 | C5 |
|----|----|----|----|----|----|
| R1 | .  | .  | .  | .  | .  |
| R2 | .  | .  | .  | .  | .  |
| R3 | S  | GE | PX | P  | .  |
| R4 | .  | .  | .  | .  | .  |
| R5 | .  | .  | .  | .  | .  |



## 5. Choose a path

| N  | C1 | C2 | C3 | C4 | C5 |
|----|----|----|----|----|----|
| R1 | .  | .  | .  | .  | .  |
| R2 | S  | P  | E  | .  | .  |
| R3 | .  | .  | P  | .  | .  |
| R4 | GE | P  | E  | .  | .  |
| R5 | .  | .  | .  | .  | .  |

## 6. Rover rescue

| N  | C1 | C2 | C3 | C4 | C5 |
|----|----|----|----|----|----|
| R1 | S  | P  | P  | E  | .  |
| R2 | .  | .  | .  | P  | .  |
| R3 | .  | P  | GE | P  | .  |
| R4 | .  | E  | P  | E  | .  |
| R5 | .  | .  | .  | .  | .  |



## EDUCATOR GUIDE 23

# Vocabulary, answer keys, and official references

| Term                | Working definition                                                           |
|---------------------|------------------------------------------------------------------------------|
| Algorithm           | A precise set of steps for accomplishing a task.                             |
| Program             | Instructions expressed so a computing system can execute them.               |
| Sequence            | The order in which instructions run.                                         |
| Event               | Something that starts or triggers a script, such as clicking the green flag. |
| Heading             | The direction the rover currently faces; independent of the viewing camera.  |
| Loop / iteration    | A loop repeats instructions; one iteration is one pass through its body.     |
| Condition / Boolean | A question evaluated as true or false.                                       |
| Branch              | One possible route through a decision in a program.                          |
| Sensor              | A source of information about a system or its environment.                   |
| Input / output      | Information supplied to a system / results produced by it.                   |
| State               | Current information, such as position, heading, and collected energy.        |
| Debugging           | Finding a cause of unexpected behavior and testing a correction.             |
| Model               | A simplified representation used to explore or explain a system.             |
| Constraint          | A limit or requirement within which a solution must work.                    |

## Exit-card answer key

1. Pick up at C3 R3. Moving across the parcel does not collect it automatically.
2. It must also face north and have visited both beacons in order. Heading and position are different.
3. Eight actions: four moves and four processing actions. A shorter written program can perform the same work.
4. The scan is blocked, so IF clear skips its move. Confirm stop records the decision at the safe tile.
5. No. IF / ELSE selects one branch. A RED label chooses left; a BLUE label chooses right.
6. The rover must carry both pods back and unload at base. Plan battery for the return trip.
7. No. Test new inputs and independently check the action. Training-label agreement is not a probability of correctness.
8. Keep the same starting position, heading, and one-tile movement module. Change only repeat count.



9. A recorded map revision, three distinct tests, a successful final run, and an explanation of a limitation.

## Official source documents

Standards documents were retrieved and the cited codes checked on October 18, 2026. Computational Thinking is on page 4 of each cited grade-band PDF; elementary Networks & System Design is on page 5 and Digital Literacy on page 7. Codes and brief summaries link to the original documents. Activities, timings, worksheets, and the rubric are original teacher-planning materials.

- NYSED: Computer Science and Digital Fluency Guidance & Resources
- NYSED: Computer Science & Digital Literacy FAQ
- NYSED standards, grades K-1
- NYSED standards, grades 2-3
- NYSED standards, grades 4-6
- NYSED standards, grades 7-8
- NYSED standards, grades 9-12
- Scratch: official ideas and tutorials
- Scratch: getting started guide

Prepared for New York Tutoring Center. Designed by Huanjiao Dong. Curriculum Created by Huanjiao Dong. Educator edition 4.0. This unit uses the New York State standards framework with a focused elementary scope and an extension roadmap across K-12.