

## technology

# Algorithms and debugging: unplugged to sequenced

## Lesson at a glance

Open with everyday fixed-order examples (traffic lights, a school-dinner queue) and ask what fails if a step vanishes. Model a live plant-watering algorithm, hide one step as a bug, then debug it. Groups write 5–8 steps for a classroom job card, break one step, swap packs and hunt the bug. They record algorithm, bug, fix and one left-out detail on paper, then stack forward/turn commands on the algorithm-sequencer interactive to reach an apple in seven steps or fewer.

## Learning objectives

- Write precise step-by-step instructions for a real task and identify unnecessary detail
- Find and fix a deliberate bug in another group's algorithm
- Build and debug a sequenced robot route within a step limit

## Before the bell – prep

Print one set of the four task cards and assign each group its card; cut 8–10 blank step strips per group. Gather props: three library books, plant with watering can and tray, four sealed paint pots and art tray. Clear floor space near the door for PE-line groups. Pair tables for the swap before the bell. Load the algorithm-sequencer interactive in challenge mode on the IWB.

**Materials**

| Item                                                                 | Qty | Per                       | Source             | Low-cost substitute                                                         |
|----------------------------------------------------------------------|-----|---------------------------|--------------------|-----------------------------------------------------------------------------|
| task prompt cards (library sort, PE line, plant care, pack away art) | 1   | group                     | provided printable | write the four task titles on sticky notes                                  |
| blank step cards or paper strips                                     | 10  | group                     | classroom          | half-sheets of scrap paper                                                  |
| pencils                                                              | 1   | pupil                     | classroom          | any writing tool                                                            |
| Investigation Journal page                                           | 1   | pupil                     | provided printable | a blank page with four short note areas ruled by the pupil                  |
| classroom plant, watering can and tray                               | 1   | class                     | classroom          | empty cup and a simple plant sketch on the board; mime the pour over a tray |
| story books for library-sort groups                                  | 3   | group doing library sort  | classroom          | three scrap-paper cards labelled with different author surnames             |
| empty or sealed paint pots and art tray for pack-away-art groups     | 4   | group doing pack away art | classroom          | four pencil pots or clean yoghurt pots and a shallow box                    |

**Safety watch-point**

Water and paint-pot groups: pour over the tray only, seal pots before carrying, wipe spills at once. PE-line groups walk—no running to the door.

### Teaching moves

- **Precise steps, bugs and fixing them:** Model the full cycle live with the plant, can and tray before any group writes. Board only algorithm and bug at the start; name debugging the moment you restore the missing 'walk to the plant' step. Treat abstraction as a spoken phrase only—cross out jumper/smile details aloud without adding a fourth board word.
- **What details matter?:** Display one shared Library-sort prompt only—no group cards yet. Take two minutes of keep-or-leave-out talk, then three or four voices. Revoice: keep what changes the result, leave out what does not. Stress that two different clear algorithms can both be right.
- **Write, break and fix:** Run the 18-minute cycle tightly: class bug-type prediction, 5 min write, 3 min robot-test, 2 min break exactly one step, 5 min swap-and-debug, 2 min share. Walk the room so the change is a real hidden mistake, not announced. When a pair acts at the front, ask watchers where the bug is so idle pupils still read the steps.
- **Record the algorithm and the fix:** Show the four-note worked example once, then pupils record on paper only: algorithm as commands, bug in one line, the fix, and one left-out detail. Prompt 'commands, not a story.' Early finishers sketch a shorter algorithm for the same job—nothing longer.
- **Robot route challenge:** Drive the algorithm-sequencer on the IWB; pupils have no devices. Whole-class thumbs-up/down before Run, one volunteer adds each command. Core challenge only: 4×4 grid, apple in ≤7 steps (three forward, turn left, three forward works). After a fail ask which single command to change, then Run again so debugging is visible.

### What it should show

Groups produce 5–8 ordered commands that a silent 'robot' can follow; a deliberate one-step break (wrong order, missing move, reversed direction) is found and fixed so a re-test works. On the sequencer, the class reaches the apple in seven steps or fewer; a common miss is turning before the third forward or facing the wrong way after a turn—re-run from the start cell facing right.

### Misconceptions & interventions

- **'The bug means the person following the steps was careless.'** — Replay the plant demo: a careful volunteer still pours on the floor when step 3 is missing. Say clearly—we debug the steps, we do not blame the follower.
- **'More detail always makes better instructions.'** — Point back to the jumper and smile lines crossed out in the model. Ask: if we leave this out, does the job still work? Keep only what changes the result.
- **'If the steps are wrong, trying harder will still get the right result.'** — Return to the key question from the open: wrong steps fail no matter how hard you try. Fix one command, re-test, and show the job only works when the algorithm is right.

**Differentiation**

| Emerging                                                                                                                                                                                                                                                                                      | Developing                                                                                                                                                                                                                                                                      | Proficient                                                                                                                                                                                                                                           |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"><li>• Offer the board stem First... Next... Then... Last... and sit the group at the teacher table for the first write-and-test.</li><li>• During the robot challenge, let them suggest only the next single command while you hold the sequence.</li></ul> | <ul style="list-style-type: none"><li>• After a successful debug, ask why that one changed step was enough—and whether a different clear algorithm could also work.</li><li>• On the sequencer, predict thumbs-up/down before each Run and name the likely bug aloud.</li></ul> | <ul style="list-style-type: none"><li>• Write a second, shorter algorithm for the same job that still passes a silent robot test.</li><li>• If time remains, tackle the optional 4×3 star route in six steps and justify each cut command.</li></ul> |

**Cross-curricular hook**

Link to English procedural writing and Maths alphabet order—library-sort groups order authors by surname.