

Robots and rules: programming the floor robot

Lesson at a glance

Open by holding the floor robot and arrow cards: the arrows in order are the program, a wrong arrow is a bug, and three forwards is a short repeat. Model one deliberate wrong-turn on the taped mat so the class sees predict → run → notice → fix. Drive the algorithm-sequencer interactive for Challenge 1 (around the corner), then groups plan a turn-and-repeat route with cards, predict, run on a robot path or child-robot lane, and fix one step at a time. They record the working program, bug and fix in the Investigation Journal, then talk about everyday machines that follow exact steps.

Learning objectives

- Program a floor robot to follow a longer route with a turn and a repeat
- Predict where it stops, then find and fix a bug in the program

Before the bell – prep

Night before: tape a ~4×4 grid mat (one square = one robot step) with a clear start and a finish that needs a turn plus repeated forwards; add 2–3 short taped paths and/or child-robot floor/desk grids so every group can run. Cut arrow cards (forwards, left, right) per group. Charge or clear the floor robot(s).

**Materials**

Item	Qty	Per	Source	Low-cost substitute
Bee-Bot or similar floor robot	1	class (or 1 per group if the school has several)	school kit	a child acting as the robot, moving only when shown arrow cards (expected parallel setup when only one programmable robot is available)
taped floor path or small grid with start and finish squares	2–3 short paths, or 1 class mat plus child-robot lanes	class (plan parallel run space so every group gets a turn)	classroom	masking-tape path of large floor squares in the hall, or desk-top grids for child-robot groups
arrow step cards (forward, turn left, turn right)	about 8–10 cards	group	classroom	arrows drawn on sticky notes or scrap card (this is the main prep, not a fallback)
Investigation Journal page	1	pupil	classroom	plain paper with three headings: Program in order, Bug we found, What we changed

Safety watch-point

Clear lanes before Go; robots stay on the mat/path only — no chasing across the room. Child-robots walk carefully on marked squares; do not crowd several robots on one small mat.



Teaching moves

- **Getting Started:** Hold up the robot, then land one idea at a time with the cards: fan a short line (program), swap one wrong turn (bug), fan three forwards (short repeat). Ask: if the robot only does what we press, whose job is it when it goes wrong? Draw out that the bug is in our program. Show the mat shape briefly — do not run a full program yet.
- **Watch a program run:** On the real mat, model one deliberate bug story aloud: plan forward, forward, turn right, forward, forward, but enter turn left by mistake. Press Go in silence, then freeze: which step made it go wrong? Change left to right and re-run so it lands on the finish. Revoice any 'robot is broken' talk immediately.
- **Plan the route on the board:** Drive the algorithm-sequencer interactive in challenge mode — pupils call the next step; you or one helper drag and Run. Complete Challenge 1 only (around the corner, ≤ 6 steps) through a full predict → run → freeze-and-debug cycle. If it misses, ask the whole class where the bug is. Only if Challenge 1 is solid and time remains, briefly model Challenge 2's repeated forwards. If the class stalls, walk the route with your finger on the grid first.
- **Program the floor robot:** Agree start, finish, and that every program needs at least one turn and a short repeat. Groups of 3–4 plan with arrow cards, say the sequence, and predict the stop before Go. Run parallel: several robots on separate short paths, or child-robot lanes on marked squares — do not queue six groups on one mat. Fix one step at a time after a miss. Waiting groups refine cards at desks. Every group must complete one predict-run-fix cycle.
- **Record our program:** Paper only on the Investigation Journal: program in order, the bug found, what they changed — from the real run just done. Circulate with those three prompts. First-time successes may add a near-miss 'what wrong step would have sent it wrong?' Groups who never ran get a quick child-robot cycle before writing.
- **Programs in the real world:** Lead with the lift example, then take two or three pupil ideas (washing machine, traffic lights, automatic doors). Revoice: real machines need exact programs; a bug is almost always in the instructions. Keep it oral and short — lift plus one pupil example is enough if time is tight.

What it should show

A successful run lands on the finish with a spoken sequence that includes at least one turn and a short repeat (e.g. several forwards). Groups should predict before Go and either succeed first time or fix one wrong step (often the turn direction or a missing/extra forward). Odd paths usually mean a card was entered out of order, the turn was left instead of right, or the child-robot stepped off the marked squares — re-say the cards in order and re-run from start.



Misconceptions & interventions

- **‘The robot is broken / being bold’ when it misses the finish.** — Point at the cards still on the mat: the robot followed every press perfectly. Ask which single arrow you would swap, change it, and re-run so they see the fix land on the finish.
- **Pupils rebuild the whole program after one miss instead of changing one step.** — Freeze after the failed run and ask: ‘Which one step made it go wrong?’ Allow only that card or button to change before the next Go.
- **A ‘repeat’ is treated as a special button rather than the same step said more than once.** — Hold three forward arrow cards together and say them aloud — forward, forward, forward — then have the group lay the same pattern in their own plan.

Differentiation

Emerging	Developing	Proficient
• Walk the route with a finger on the grid or mat, saying each step before the group lays cards; offer a partly ordered card line they complete with the turn and one repeat. • Pair at the teacher path for the first predict-run; use child-robot on desk squares if button entry is too much.	• After one successful fix, ask them to name aloud which step was the bug and why the new step works. • If first run succeeds, try one deliberate near-miss on the cards and fix it before journaling.	• On the journal page, add one improvement they would try next (shorter route or a second turn) using only the same arrow cards. • Watch another group’s run and whisper one precise tip about which step to change.

Cross-curricular hook

Oral language: groups must say the full program in order and justify the one-step fix before they press Go.

