

Giving Directions (Grid)

40 minutes

Hands-On Logic Challenges

WHAT THIS LESSON DOES

Guide your students to think like programmers by teaching them to give precise commands to a robot for drawing shapes on a grid. Introduce decomposition and debugging concepts through interactive activities, acting as the Art Bot yourself.

BEFORE THE LESSON

- This lesson is unplugged: no devices needed. Have a whiteboard and marker ready and be prepared to act as the robot, doing only exactly what pupils tell you.
- Draw a 3x3 grid on the board and write the four directional commands (MOVE_UP, MOVE_DOWN, MOVE_LEFT, MOVE_RIGHT) where all can see.
- Prepare grid template pictures (house, capital letter, stick figure), blank grids and stacks of blank command cards for group work.
- Sort groups in advance so swapping programs between groups runs smoothly.

Giving Directions (Grid)

HOW THE LESSON RUNS

Introduction
Giving Precise Instructions
Introduce the Art Bot
Collaborative Programming
What Is Decomposition?
Group Challenge: The Programmer
The Debugger
Debugging & Reflection
Connect to the Real World

TEACHING MOVES

- 2. Giving Precise Instructions. Commit to the bit. Obey vague instructions literally: "draw a line" gets a tiny squiggle. Let pupils feel the frustration so they push toward precision themselves.
- 3. Introduce the Art Bot. Stress that these four commands are the only words you understand, and that directions are the grid's, not yours. Refuse anything outside the command list.
- 4. Collaborative Programming. State the start position (top-left, pen down) clearly before they begin. Write each command on the board in order so the growing sequence is visible to all.
- 5. What Is Decomposition? Name the skill: breaking a big task into small steps is decomposition. Draw out real examples like a LEGO set or getting ready in the morning.
- 6. Group Challenge: The Programmer. Give each group a target image and have them write the command sequence on cards. Remind them the receiving group will see only the cards, not the picture.
- 7. The Debugger. Insist the Debugger group follows the cards exactly, with no peeking at the target. This is the point: a computer follows the program, not the intention.
- 8. Debugging & Reflection. When results differ from the target, guide the group to find which single card caused it rather than rewriting the lot. Then let the original group re-run their fix.
- 9. Connect to the Real World. Tie it back: one wrong command breaks the whole program, which is why programmers spend so long testing and debugging.

WHAT TO WATCH FOR

- Pupils assume the robot will fill in the obvious steps, so their sequence skips moves they think are implied. Keep playing the literal robot. If a move is missing, stop dead or do the wrong thing so they see the gap and add the step.
- Pupils confuse the grid's directions with their own left and right when facing the board. Fix directions to the grid itself: point to up, down, left and right on the drawn grid and check a couple of commands together before groups start.
- When debugging, a group scraps the whole sequence rather than locating the one faulty command. Have them execute the program one card at a time and mark the exact card where the drawing first goes wrong.

DIFFERENTIATION

- Emerging: Give a smaller or simpler target, such as a single letter L, so the sequence stays short. Pair them to execute cards one move at a time, one reading, one drawing.
- Developing: Ask them to talk through each command as they place it before the group executes it. Have them predict the outcome of a short sequence before drawing it.