

LESSON 2

The Human Algorithm

40 minutes

Unplugged Adventures

WHAT THIS LESSON DOES

As the teacher, guide your students to understand the importance of clear, step-by-step instructions in everyday tasks. Facilitate activities that demonstrate how order and precision in sequencing are crucial, using interactive examples and group challenges to reinforce these concepts.

BEFORE THE LESSON

- This lesson is unplugged. No pupil devices are needed, but you will want the interactive whiteboard for the scrambled toast steps and the instruction rules.
- Print the arrow-box sequencing worksheet, one per pupil, and have pencils ready.
- Have your scrambled toast steps and the 'Spread Butter' example ready to display before you start.

The Human Algorithm

HOW THE LESSON RUNS

Introduction

The Scrambled Algorithm

What Makes a Good Instruction?

The Group Algorithm Challenge

Worksheet practice, Fill the sequence

Wrap Up and Debugging

TEACHING MOVES

- 2. The Scrambled Algorithm. Do not just reorder the steps. Ask what actually happens if butter goes on before toasting, or the lever is pressed before plugging in, so pupils feel why order matters.
- 3. What Makes a Good Instruction? Play the literal robot. When pupils say 'Spread Butter', mime spreading with no bread, no knife, on the wrong side. Let the failure make the case for precision.
- 4. The Group Algorithm Challenge. Give each group one task and hold them to the rules: one action per step, a strong verb to start each. Walk round pushing for the step they left out.
- 5. Worksheet practice, Fill the sequence. Read the completed Making Toast sequence down the arrows together first so pupils see the shape. Then set them loose on the three blanks and their own final task.
- 6. Wrap Up and Debugging. Name debugging as finding and fixing your own mistakes. Ask which step they had to go back and add, and treat that fix as the real work, not a failure.

WHAT TO WATCH FOR

- Pupils write vague instructions like 'clean your teeth' and assume the reader will fill in the obvious details. Have a partner follow the instruction word for word with no common sense. The gaps show up instantly and pupils rewrite with more precise steps.
- Pupils think a good set of instructions can be in any order as long as all the steps are there. Point back to the toast example. Ask them to follow their own steps out of order and see whether the task still works.
- Pupils see a missing or wrong step as getting it wrong rather than a normal part of the process. Frame it as debugging. Real programmers expect to find and fix mistakes, so celebrate the pupil who spots and repairs a broken step.

DIFFERENTIATION

- Emerging: Pair them with a partner for the group task and let them focus on saying each step aloud while the partner writes. Point to the completed Making Toast row as a model whenever they stall on the worksheet.
- Developing: Ask them to swap worksheets with a partner and follow each other's steps literally to catch missing actions. Prompt them to check every step starts with a strong verb before they call it finished.
- Proficient: For their own-choice final sequence, ask for a task with a decision or a repeated step, then have them explain it back to you. Challenge them to find and fix one weak step in another group's algorithm.