

INSTRUCT A ROBOT: PRACTICING CODING

OBJECTIVE: Students will learn how to translate a complicated task into an algorithm that has clear and simple steps.

TIME: 45 minutes

MATERIALS: Pencil or pen, “Instruct a Robot” student worksheet

LESSON PLAN

1. Evaluate a Set of Instructions: Tell students that you are going to give them instructions to bake a batch of cookies. Then write the following steps on the classroom board.

- Gather eggs, butter, sugar, flour, baking soda, and chocolate chips.
- Mix ingredients in large bowl.
- Place cookie dough on a pan.
- Bake until done.

Guide students to evaluate your instructions. Ask the class: *Do they think a person could successfully bake cookies by following these steps? If 10 people followed these steps, would they all make the exact same cookies? Why or why not?* Prompt students to be specific when they describe the limitations of the instructions. (For example: The instructions don’t say how much of each ingredient is needed. They don’t explain the order in which the ingredients should be added. They don’t specify how the dough should be placed on the pan—in balls or as one large layer. The temperature that the cookies should be baked at is not given. There are no instructions explaining how a person knows when the cookies are done.)

2. Write a Recipe: Explain to students that their task is to write an algorithm for baking cookies. An **algorithm** is a set of steps that can be followed from start to finish to complete a task. For example, scientists might write an algorithm that instructs a robot on how to bake cookies. As a class, try to write an algorithm for baking cookies. Make sure that each step is very specific. (For example: Break open 1 egg. Add 1 teaspoon of baking soda. Mix until there are no more lumps. Scoop out a 1-inch ball of dough.)

STANDARDS FOCUS:

Science (NGSS)

Science and Engineering Practices: Asking Questions and Defining Problems, Developing and Using Models, Using Mathematics and Computational Thinking

ETS1.A: Defining and Delimiting an Engineering Problem

Language Arts (CCSS)

R7: Integrate content presented in diverse formats

RSci2: Follow precisely a multistep procedure when performing technical tasks

Art (National Core Arts Standards)

VA—Cr1: Generate and conceptualize artistic ideas and work

Math (CCSS)

MP5: Use appropriate tools strategically

MP7: Look for and make use of structure

3. Introduce Conditional Statements: Computer codes are algorithms that give the computer instructions for how to complete an action. Explain that codes often include a specific type of language structure. They may include loops (See “Step-by-Step” lesson) or conditional statements. Also called if/then statements, a conditional statement instructs the computer to do a particular task if another condition is met. For example, a conditional statement might be “If the oven is at a temperature of 375°F, then place the cookie pan in the oven.” Challenge the students to revise your code to include if/then statements. (For example, If the cookies are light brown in color, then remove them from the oven.)

4. Conduct the Activity: Hand out the “Instruct a Robot” student worksheet. In the exercise, students will write an algorithm that can be used by a fellow student—their “robot”—to complete a complicated task. Have them test their algorithms with another student. Discuss what went wrong if they are unsuccessful in their tests. Was there a step missing? Was the order of the steps accurate? Should the steps have been simpler actions?

DIVE DEEPER WITH KOOV

The KOOV platform makes it easy for students to challenge themselves to write complicated codes for their robots. Have them hone their skills by following the My First Robot Coding lessons in the Learning Course. Once they’ve mastered the basics, you can challenge the class to brainstorm a complex action for a KOOV robot and write the code to make it work. Experiment with using loops and conditional statements.

INSTRUCT A ROBOT

Today, many robots perform complicated tasks—from building cars to sorting items in warehouses. Suppose you are an engineer who is designing a robot that can perform a task a human does today. Think about a task that can be completed in your classroom, such as sharpening your teacher's pencils or sorting recyclables. Your job is to write the instructions the robot needs to follow, called an **algorithm (al-guh-ri-them)**.



ASK

Identify the problem you want to solve. What job will your robot do?

What benefit would there be to having a robot do this task instead of a human?

PLAN

What actions does your robot need to do to complete this task? Write a list of the steps involved.

Does your robot need any materials to complete the task? If so, what is needed?

CREATE

On separate paper, write an algorithm for your robot to complete the task. Remember that an algorithm breaks complicated actions into very simple steps. For example, if your task is to sharpen a pencil, your instructions need to describe how to pick up the pencil, exactly how to use the sharpener, and how to know when the task is complete.

To write your own algorithm, follow these steps:

1. Plot out your work.
2. Write a first draft.
3. Review your draft and revise as necessary.

Try to use the following features in your algorithm:

A **loop**, which is a group of actions that is repeated a certain number of times.
(**Example:** Repeat 10 times:
Turn sharpener clockwise around the pencil.)

An **if/then statement**, which is an instruction to perform a task if a certain condition is met. (**Example:** If the end of the pencil is a sharp point, then place the pencil on the desk.)

TEST

Give your algorithm to another classmate to follow. Are they successful in completing the task? _____

IMPROVE

Revise your algorithm if needed.