Code A Friend

Summary:
Students will instruct (program) another person (teacher or friend) to complete a simple task, such as turning on the light switch or throwing away a piece of paper using a set of explicit instructions (an algorithm). The teacher will need to follow instructions just as the student lays out. Hopefully there are no hiccups, or “bugs”, in the student’s instructions, or “code.” Students will practice skills in providing explicit instruction and sequencing, skills necessary for building programs that accurately complete the task they are meant to complete. This activity reinforces the ECC areas of social skills, orientation and mobility, and compensatory access.

Audience: Grades K-8
Plugged or unplugged: Unplugged

California State Standards for Language Arts:

- CCSS.ELA-LITERACY.SL.K.4: Describe familiar people, places, things, and events and, with prompting and support, provide additional detail.
- CCSS.ELA-LITERACY.SL.1.4: Describe people, places, things, and events with relevant details, expressing ideas and feelings clearly.
- CCSS.ELA-LITERACY.SL.2.4: Tell a story or recount an experience with appropriate facts and relevant, descriptive details, speaking audibly in coherent sentences.
- CCSS.ELA-LITERACY.SL.3.4: Report on a topic or text, tell a story, or recount an experience with appropriate facts and relevant, descriptive details, speaking clearly at an understandable pace.
- CCSS.ELA-LITERACY.W.4.4: Produce clear and coherent writing in which the development and organization are appropriate to task, purpose, and audience.
- CCSS.ELA-LITERACY.W.5.4: Produce clear and coherent writing in which the development and organization are appropriate to task, purpose, and audience.
- CCSS.ELA-LITERACY.W.6.4: Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.
- CCSS.ELA-LITERACY.W.7.4: Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.
- CCSS.ELA-LITERACY.W.8.4: Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.
Materials needed:
- Writing tool: braille writer, 20/20 pen, markers, artist pencils
- Braille paper or dark-lined paper

Prerequisite skills:
- Understanding of the concepts 1st, next, last
- Familiarity with following step by step instructions
- Familiarity with breaking down a big task into smaller steps

Time Estimate:
- 30 to 45 minutes

Objectives:
1. By the end of this lesson, students will understand the purpose of providing precise step by step instructions for completing a process or task.
2. Students will understand the concept of an “algorithm.”
3. Students will understand the concept of a “bug.”
4. Students will learn how to analyze code to find bugs, and how to rewrite the algorithm to correct for these errors.

Key Terms:
1. Algorithm: a set of instructions
2. Code: programming languages used to communicate instructions to computers
3. Bug: an error in an algorithm that does not allow successful completion of the task

Instructional Sequence:

Step 1: Introduce the activity
“Today we will be programming each other like robots. We will be giving each other instructions to complete simple tasks, like flipping a light switch. But we need to be careful, because any mistakes in our instructions could cause mayhem. The task could be incomplete. We could accidently complete a completely unrelated task. Any questions? Are you ready?”

Step 2: Introduce concept of algorithms
“Algorithms are instructions given to computers or programs by programmers. They are telling the piece of technology what
to do. Today you are the programmer, and your friend (or the instructor) is the robot or program. The algorithm, or instructions, are written in something called code."

Step 3: Introduce concept of code
“Coding is a language. It is the language of computers. The language in which computers interpret information.”
*check for understanding: can you think of other types of languages

Step 4: Introduce purpose of algorithms
“Algorithms are instructions a computer receives to complete a task. The task could be to turn on, update, open a program, analyze data, anything really.”

Step 5: Check for understanding
Ask student for situations they have received instructions. Discuss why they were important. Discuss what happened or would happen if you didn’t follow the instructions.

Step 6: Introduce concept of bugs
“Bugs are errors in the instructions (algorithms) which are written in the computer’s language (code). Bugs cause the algorithm not to work. The instructions cannot be accurately completed, therefore the task the computer is being asked to complete, cannot be properly completed.”

Step 7: Check for understanding
*Talk about a time the student felt they received unclear instructions. What happened? How did the situation resolve?

Step 8: Choose a task
Turning on a light switch, walking across the room, or throwing away a piece of paper. Be creative. It can be anything really.

Step 9: Model the Activity
Begin with a practice round in which the student is the robot, and the teacher is the programmer

Step 10: Model Writing Instructions
Provide explicit instructions to student for completing the selected action. Model the process of writing down steps so your student is prepared when it is their turn to be the programmer. Make sure to add an ambiguous step or incorrect direction in your instructions. This will serve as your bug. You can also come to the lesson with an action in mind, so

Author: Allison Mello, California School for the Blind
### Examples of explicit instructions:
- Take 4 large steps forward
- Extend your right arm forward at shoulder height
- Turn so that your back is parallel to the door

### Step 11: Get Moving
Instruct your student. Remind your student that they must follow your instructions explicitly. They will be unable to complete the task due to your bug. When you reach this point, talk about it. What happened? What should have happened? How do we correct the error?

### Step 12: Correction
Together, student and teacher correct the ambiguous or incorrect step of the algorithm. Task completed successfully.

### Step 13: Student’s Turn
Have student choose a new task. They can use examples from step 8 as inspiration, but students are encouraged to be creative. Next, student will write out steps to complete task using preferred medium. You can work together to create the instructions, but don’t forget the point of this exercise is to explore testing and redesigning instructions.

### Step 14: Student Instructs
Student instructs you in task. Follow their instructions explicitly. If you hit a bug, stop, and work together to solve the issue. Rework the algorithm until you work out the correct sequence of steps.

### Step 15: Rinse and Repeat
Repeat this activity with any/all of the provided actions or come up with an action of your own. An action can be as simple or as complicated as you wish. Have fun with it. Maybe include a task that you have been working on in a goal, such as inserting braille paper into the Perkins, placing the iPad on the charger at the end of the day, or opening a new Google Doc.

### Step 16: Review
Review the meanings of algorithms, code, and bugs. Discuss their purpose in designing processes to complete tasks.