SOUTH SANPETE
SCHOOL DISTRICT

Let's Help Debug Blue-Bots

Students will learn how to read other algorithms, find bugs and correct them! Lesson adapted from the San Francisco School District's CS Program

CURRICULUM STANDARDS

1. **Math Standard 2.NBT.6** Add up to four two-digit numbers using strategies based on place value and properties of operations.
2. **CS Standard 2.AP.4** Debug and solve simple problems within an algorithm or program that includes sequences and simple loops.
3. **CS Core Practice** Algorithms and Programming

INTENDED LEARNING OUTCOMES

At the completion of this lesson, students will be able to:

- Debug already created algorithms
- Debug already created math algorithms, errors when adding two-digit numbers

MATERIALS NEEDED

1. Blue-Bots (1 per group of 4-5 students)
2. Debugging Worksheets (1 per pair of students) or work together as a class
3. Pencils (1 per pair of students)
4. Math Debugging Worksheet (that works for your class and what you are covering; example worksheet linked here)

COMPUTER SCIENCE WORDS TO INTRODUCE/REVIEW

- **Debug** The process of finding and correcting errors (bugs) in programs.
- **Perseverance** Continued effort to do or achieve something despite difficulties, failure, or opposition

INSTRUCTIONAL PROCEDURES

1. Attention Getter: Teacher says, "I have a friend who teaches computer science at another school. Their class did the same activity you did last week! The thing is, some of the students did their work a little too quickly and made some mistakes. I thought about all of you, and how you are so helpful. I told my friend that we can help **debug** the

mistakes! We are going to find and fix the errors. Let's try one together, then you and your groups will help debug the rest."

- Show the kids the problematic algorithm.
- Practice the track that the Blue-Bot will go, and talk aloud showing the way that the Blue-Bot will go (and how it will end up in the wrong place).
- Problem solve and debug with the students a way that the Blue-Bot could travel along the right track.

2. In order to make a connection to the real world, write a math problem on the board with an incorrect answer. Have students look at the algorithm and find where the error (bug) is and have students solve the problem!
3. Now it's time to debug! (pass out Debugging worksheets). Teach students to engage in paired programming, working together to fix the buggy algorithms. One student will debug the code using their pencil, and the other will help give instructions on how to change the code. Then, once they've solved a puzzle, the other student will edit the codes with their pencil and the first student will help them debug the code. Students will switch back and forth.
    a. Remind students that debugging takes perseverance, continued effort despite failure, and that they will succeed if they learn from their mistakes!
4. Give students a new task, in which they debug a math problem. Have students look at already existing math work, and find where the error (bug) is and have students solve the problem!

EXTENSIONS/ADAPTATIONS

- Extensions
    - Students can create their own buggy code and trade with a partner to change the code!
- Adaptations
    - Students may join a small group to change the code in order that they can do it in guided practice (if it is too challenging to do it by themselves or with a partner).

WRAP UP/REFLECTION

Students will discuss with the whole group the following questions: (5 min)

- What was challenging about debugging someone else's work?
- How did you feel when you found a bug?
- Who do you want to appreciate for helping you do this challenging work?

MODIFICATIONS FOR EACH GRADE LEVEL

- **Math Standard 1.NBT.4:** Add within 100, including adding a two-digit number and a one-digit number, and adding a two-digit number and a multiple of 10, using concrete models or drawings and strategies based on place value, properties of operations, and/or the relationship between addition and subtraction; relate the strategy to a written method and explain the reasoning used. Understand that in adding two-digit numbers, one adds tens to tens and ones to ones, and that it is sometimes necessary to compose a ten.
- **Math Standard 3.NBT.2:** Fluently add and subtract within 1,000 using strategies and algorithms based on place value, properties of operations, and/or the relationship between addition and subtraction.
- **Math Standard 4.NBT.4:** Fluently add and subtract multi-digit whole numbers using the standard algorithm.
- **Math Standard 5.NBT.5** Fluently multiply multi-digit whole numbers using the standard algorithm.
- This lesson is a great introduction to debugging in any skill level of computer science, for any grade. It could be modified for older students by allowing them to create their own buggy codes and create challenges for the class.