U

UNPLUGGED

Relay Programming

Lesson time: 20 Minutes   Basic lesson time includes activity only. Introductory and Wrap-Up suggestions can be used to delve deeper when time allows.

LESSON OVERVIEW

This activity will begin with a short review of Graph Paper Programming, then will quickly move to a race against the clock, as students break into teams and work together to create a program, one instruction at a time.

TEACHING SUMMARY

Getting Started - 15 minutes

1) Review
2) Vocabulary
3) Refresh Graph Paper Programming

Activity: Relay Programming - 20 minutes

4) Relay Programming

Wrap-up - 10 minutes

5) Flash Chat: What did we learn?
6) Vocab Shmocab

Assessment - 10 minutes

7) Relay Programming Assessment:Debugging

LESSON OBJECTIVES

Students will:

- Practice communicating ideas through codes and symbols
- Use teamwork to complete a task
- Verify the work of their teammates to ensure a successful outcome

TEACHING GUIDE

MATERIALS, RESOURCES AND PREP

For the Student

- Blank Paper or Index Cards for programs
- Sheets of 4x4 paper grids for the students to use as practice (These are provided as part of the Relay Programming Activity Packet, but if you have the students create their own, you can include Common Core Math standard 2.G.2.)
- Markers, pens, or pencils

- Relay Programming Assessment

For the Teacher

- Lesson Video
- Locate a wide open space for this activity, such as the gym or outdoor field
- Print out one Relay Programming Activity Packet for each group
- Print one Relay Programming Assessment for each student
- Supply each group with plenty of paper and pens/pencils

GETTING STARTED (15 MIN)

1) Review

This is a great time to review the last lesson that you went through with your class. You can do this as one large group or have students discuss with an elbow partner.

Here are some questions that you can ask in review:

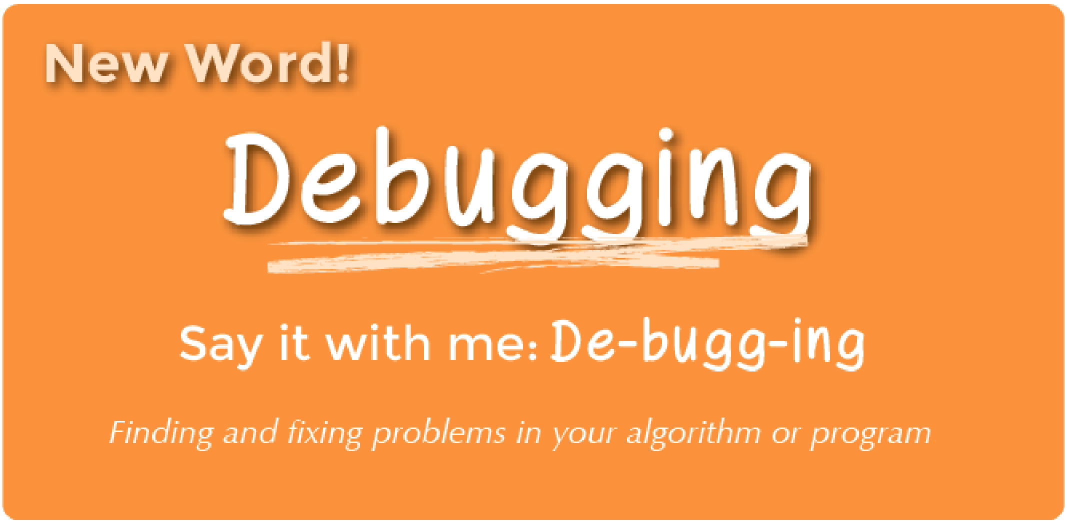
- What did we do last time?
- What do you wish we had had a chance to do?
- Did you think of any questions after the lesson that you want to ask?
- What was your favorite part of the last lesson?

LESSON TIP

Finishing the review by asking about the students' favorite things helps to leave a positive impression of the previous exercise, increasing excitement for the activity that you are about to introduce.

2) Vocabulary

This lesson has one new and important word:



New Word!

Debugging

Say it with me: De-bugg-ing

Finding and fixing problems in your algorithm or program

Debugging - Say it with me: De-bugg-ing
Finding and fixing problems in your algorithm or program

3) Refresh Graph Paper Programming

Recall that in "Graph Paper Programming" we guided our teammates' Automatic Realization Machine (ARM) using arrows. Take a moment to do a refresher Graph Paper Programming image -- either one that you have already covered or an entirely new one.

We are going to do the same kind of thing today, but instead of controlling each other, we are going to work together to create a program one symbol at a time.

ACTIVITY: RELAY PROGRAMMING (20 MIN)

4) Relay Programming

The practice lesson was easy enough; let's add some action! We're going to do the same type of thing (create a program describing an image) but now we're going to do it in relay teams, one symbol at a time.

The rules of this game are simple:

1. Divide students into groups of 3-5.
2. Have each group queue up relay-style.
3. Place an identical image at the other side of the room/gym/field from each team.
4. Have the first student in line dash over to the image, review it, and write down the first symbol in the program to reproduce that image.
5. The first student then runs back and tags the next person in line, then goes to the back of the queue.
6. The next person in line dashes to the image, reviews the image, reviews the program that has already been written, then either debugs the program by crossing out an incorrect symbol, or adds a new one.
7. That student then dashes back to tag the next person, and the process continues until one group has finished their program.
8. First group to finish is the winner!

Play through this several times, with images of increasing difficulty.

CLARIFICATIONS

Here are some clarifications that need to be shared from time to time:

- Only one person from each group can be at the image at one time.
- It *is* okay to discuss algorithms with the rest of the group in line, even up to the point of planning who is going to write what when they get to the image.
- When a student debugs a program by crossing out an incorrect instruction, they must also cross out the rest of the program after that. This counts as their entire turn. The next player is allowed to continue from the last *correct* instruction.

WRAP-UP (10 MIN)

5) Flash Chat: What did we learn?

- What did we learn today?
- What if we were each able to do five arrows at a time?
  - How important would it be to debug our own work and the work of the programmer before us?
  - How about with 10 arrows?
  - 10,000? Would it be more or less important?
- Is it easier or harder to have multiple people working on the same program?
- Do you think people make more or fewer mistakes when they're in a hurry?
- If you find a mistake, do you have to throw out the entire program and start over?

6) Vocab Shmocab

You can choose to do these as a class, or have the students discuss with an elbow partner and share.

- Which one of these definitions did we learn a word for today?

"To rub or scrape out letters or characters"
"Doing something repeatedly in a similar way"
"Finding and fixing problems in your algorithm or program"

...and what is the word that we learned?

- Which of these things could you debug?

*The wrong answer to a math problem
*A tunafish sandwich
*Two baskets of pine cones

Explain why you chose your answer.

ASSESSMENT (10 MIN)

7) Relay Programming Assessment: Debugging

EXTENDED LEARNING

Use these activities to enhance student learning. They can be used as outside of class activities or other enrichment.

Pass the paper

- If you don't have the time or room for a relay, you can have students pass the paper around their desk grouping, each writing one arrow before they move the paper along.

Fill It, Move It

- As the teacher, draw an image with as many filled squares as children in each group.
- Have the students write as many arrows in the program as it takes to get to a filled-in square (including actually filling that square in) before passing to the next person.

Debugging Together

- As the teacher, draw an image on the board.
- Have each student create a program for the image.
- Ask students to trade with their elbow partner and debug each other's code.
    - Circle the first incorrect step, then pass it back.
    - Give the students another chance to review and debug their own work.
    - Ask for a volunteer to share their program.
        - How many students had the same program?
        - Anyone have something different?