

Graph Paper 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

By "programming" one another to draw pictures, students will begin to understand what programming is really about. The class will begin by having students instruct each other to color squares in on graph paper in an effort to reproduce an existing picture. If there's time, the lesson can conclude with images that the students create themselves.

TEACHING SUMMARY

Getting Started - 15 minutes

- 1) [Vocabulary](#)
- 2) [Introduce Graph Paper Programming](#)
- 3) [Practice Together](#)

Activity: Graph Paper Programming - 20 minutes

- 4) [Four-by-Fours](#)

Wrap-up - 5 minutes

- 5) [Flash Chat: What did we learn?](#)
- 6) [Vocab Shmocab](#)

Assessment - 10 minutes

- 7) [Graph Paper Programming Assessment](#)

LESSON OBJECTIVES

Students will:

- Understand the difficulty of translating real problems into programs
- Learn that ideas may feel clear and yet still be misinterpreted by a computer
- Practice communicating ideas through codes and symbols

TEACHING GUIDE

MATERIALS, RESOURCES AND PREP

For the Student

- [Four-by-Fours Activity Worksheet](#)
- [Graph Paper Programming Assessment](#)
- Sheets of 4x4 paper grids for the students to use as practice (These are provided as part of the [Four-by-Fours](#))

[Activity Worksheet](#), but if you have the students create their own, you can include Common Core Math standard 2.G.2.)

- Blank paper or index cards for programs
- Markers, pens, or pencils

For the Teacher

- [Lesson Video](#)
- Print out one [Four-by-Fours Activity Worksheet](#) for each group
- Print one [Graph Paper Programming Assessment](#) for each student
- Supply each group with several drawing grids, paper, and pens/pencils

GETTING STARTED (15 MIN)

1) Vocabulary

This lesson has two new and important words:

New Words!

Algorithm
Say it with me: Al-go-ri-thm

A list of steps that you can follow to finish a task

Program
Say it with me: Pro-gram

An algorithm that has been coded into something that can be run by a machine

Algorithm - Say it with me: Al-go-ri-thm
A list of steps that you can follow to finish a task

Program - Say it with me: Pro-gram
An algorithm that has been coded into something that can be run by a machine

2) Introduce Graph Paper Programming

In this activity, we are going to guide each other toward making drawings, without letting the other people in our group see the original image.

For this exercise, we will use sheets of 4x4 graph paper. Starting at the upper left-hand corner, we'll guide our teammates' Automatic Realization Machine (ARM) with simple instructions. Those instructions include:

- Move One Square Right
- Move One Square Left
- Move One Square Up
- Move One Square Down
- Fill-In Square with color

For example, here's how we would write an algorithm to instruct a friend (who is pretending to be a drawing machine) to color their blank grid so that it looks like the image below:

Start Here

- * Move One Square Right
- * Fill-In Square with color
- * Move One Square Right
- * Move One Square Down
- * Fill-In Square with color

That's simple enough, but it would take a lot of writing to provide instructions for a square like this:

Start Here

- * Move One Square Right
- * Fill-In Square with color
- * Move One Square Right
- * Move One Square Right
- * Fill-In Square with color
- * Move One Square Down
- * Move One Square Left
- * Fill-In Square with color
- * Move One Square Left
- * Move One Square Left
- * Fill-In Square with color
- * PLUS 12 MORE INSTRUCTIONS!

With one little substitution, we can do this much more easily! Instead of having to write out an entire phrase for each instruction, we can use arrows.

In this instance, the arrow symbols are the “program” code and the words are the “algorithm” piece. This means that we could write the algorithm:

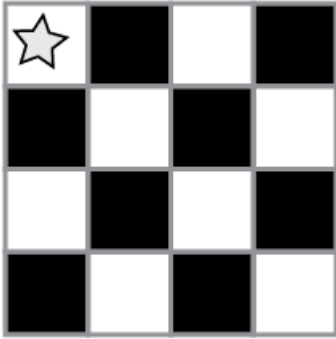
“Move one square right, Move one square right, Fill-in square with color”

and that would correspond to the program:



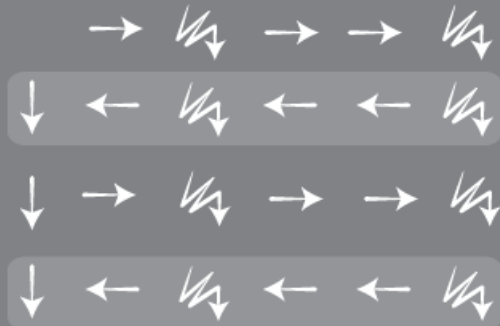
Using arrows, we can redo the code from the previous image much more easily!

Start Here



Follow along with your finger and see if you can figure out how to get this image from the program to the right.

Now, our entire program looks like this:

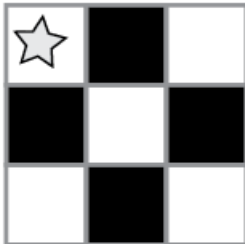


3) Practice Together

Start your class off in the world of programming by drawing or projecting the provided key onto the board.

Select a simple drawing, such as this one to use as an example.

Start Here



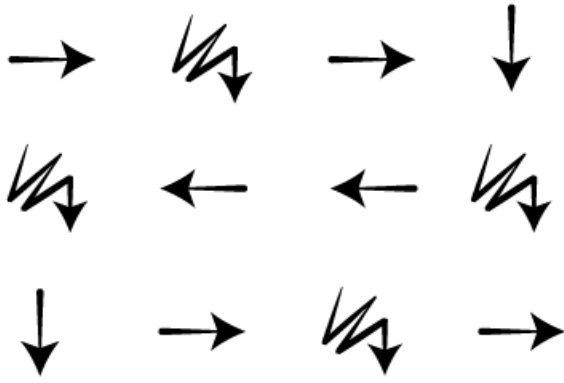
This is a good way to introduce all of the symbols in the key. To begin, fill in the graph for the class -- square by square -- then ask them to help describe what you've just done. First, you can speak the algorithm out loud, then you can turn your verbal instructions into a program.

A sample algorithm:

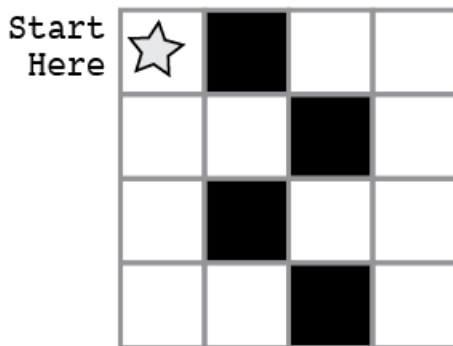
"Move Right, Fill-In Square, Move Right, Move Down
 Fill-In Square, Move Left, Move Left, Fill-In Square
 Move Down, Move Right, Fill-In Square, Move Right"

Some of your class may notice that there is an unnecessary step, but hold them off until after the programming stage.

Walk the class through translating the algorithm into the program:



The classroom may be buzzing with suggestions by this point. If the class gets the gist of the exercise, this is a good place to discuss alternate ways of filling out the same grid. If there is still confusion, save that piece for another day and work with another example.



If the class can shout out the algorithm and define the correct symbols to use for each step, they're ready to move on. Depending on your class and their age, you can either try doing a more complicated grid together or skip straight to having them work in groups on their [Four-by-Fours Activity Worksheet](#).

LESSON TIP

Have the class imagine that your arm is an Automatic Realization Machine (ARM). The idea of "algorithms" and "programs" will be brought to life even further if students feel like they're actually in control of your movements.

ACTIVITY: GRAPH PAPER PROGRAMMING (20 MIN)

4) [Four-by-Fours Activity Worksheet](#)

1. Divide students into pairs.
2. Have each pair choose an image from the worksheet.
3. Discuss the algorithm to draw that image with partner.
4. Convert algorithm into a program using symbols.
5. Trade programs with another pair and draw one another's image.
6. Choose another image and go again!

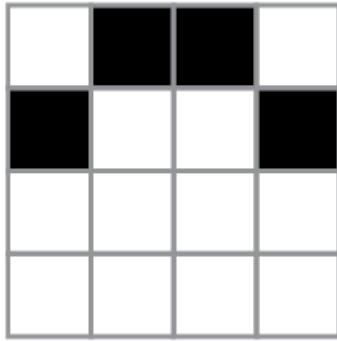


Image 1

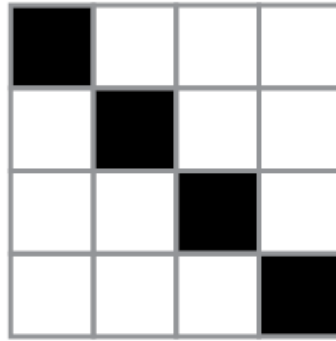


Image 2

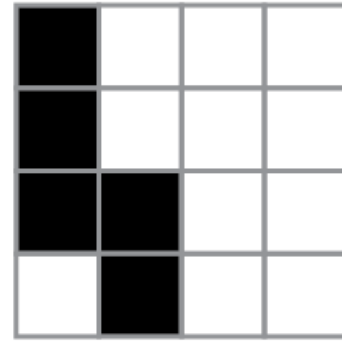


Image 3

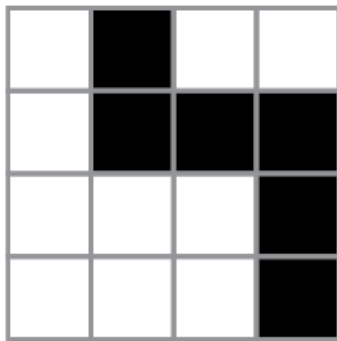


Image 4

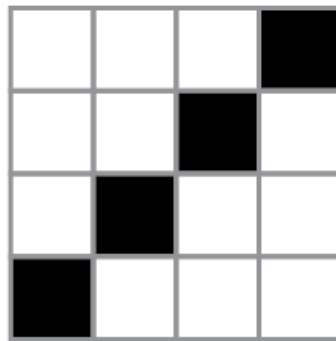


Image 5

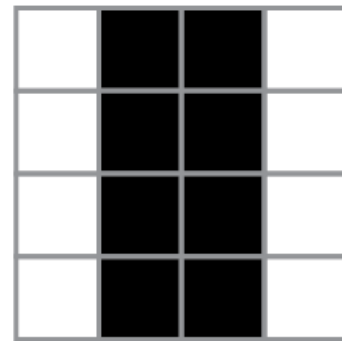


Image 6

WRAP-UP (5 MIN)

5) Flash Chat: What did we learn?

- What did we learn today?
- What if we used the same arrows, but replaced "Fill-In Square" with "Lay Brick"? What might we be able to do?
- What else could we program if we just changed what the arrows meant?

6) Vocab Shmocab

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

"A large tropical parrot with a very long tail and beautiful feathers"

"A list of steps that you can follow to finish a task"

"An incredibly stinky flower that blooms only once a year"

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

- Which one of these is the *most* like a "program"?

*A shoebox full of pretty rocks

*Twelve pink flowers in a vase

*Sheet music for your favorite song

Explain why you chose your answer.

ASSESSMENT (10 MIN)

7) [Graph Paper Programming Assessment](#)

EXTENDED LEARNING

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

Better and Better

- Have your class try making up their own images.
- Can they figure out how to program the images that they create?

Class Challenge

- As the teacher, draw an image on a 5x5 grid.
- Can the class code that up along with you?



This curriculum is available under a [Creative Commons License \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/)

If you are interested in licensing [Code.org](https://code.org) materials for commercial purposes, contact us: <https://code.org/contact>