Modified with permission from codeSpark. Written by Joe France.

Program: The Foos	Video	Worksheet
Cross-Curricular Ties: Design Thinking	**bit.ly/FooGator**	**bit.ly/FooGator Worksheet**
Age Range: 5–9		
Duration: 45–60 minutes		
Scan the QR Code or copy the link to see additional curriculum elements.		

Screengrab, Coding With the Foos: Build an Alligator Challenge!

Source: Modified with permission from Code Spark. Written by Joe France.

Overview

This lesson challenges students to apply their programming knowledge to build a simple timing game. Students will sequence different commands together in a loop and learn how small changes in code can result in big changes in the computer's behavior and how fun the game is. Students may then adapt this pattern in new contexts.

Vocabulary

Sequence: The step-by-step order of a set of instructions

Loop: A command that repeats a set of instructions over and over again

Parameters: Settings that change the behavior of commands

Lesson Objectives	Materials and Resources
Students will be able to:	• Tablets with *The Foos*
• Sequence commands for a specific behavior	• Whiteboard or projector
• Use a loop to repeat behaviors	
• Explain outcomes of code variations	
• Explore and explain design choices	

Prerequisites

Players must have access to the loop, hide, and wait commands.

Preparation

1. Read the lesson.
2. Watch the video at http://bit.ly/FooGator in preparation to show it to your class.
3. Create a new level in Foo Studio and follow along with the instructions in the video.
4. Download or print out the Worksheet at http://bit.ly/FooGatorWorksheet to use with your whiteboard or projector.
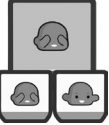
Activity

Step 1. Introduction

Begin with a group discussion of what players have learned about sequences, loops, and parameters.

Show them the icons for the **hide** and **wait** commands. Ask them what they think these commands do.

TABLE 15.1 Some of the Computing Concepts Introduced in "Coding Withe the Foos"

	The **wait command** creates a pause before the next command is performed.
	The **hide command** can make programmed game elements (spikes in this example) appear and disappear.
	Many commands have a set of parameters. Press on a parameter to change it. For the hide command, "hide" and "show" are the two parameters.

Step 2. Building—Show your class the video at http://bit.ly/FooGator, and then have them follow the instructions for creating the alligator.

- Students will need to pause the video when placing the blocks of their alligator—the video is sped up.
- Encourage students to play their level when the video does. Remind them that coding is a process consisting of the following steps:
  - The coder thinks of something she wants to do.
  - The coder tries to figure out how to get the computer to do it.
  - The coder writes code that he thinks will work.
  - The coder tests the code to see if it works.
  - If the code doesn't work, the coder investigates to find out why.
- Have students publish their level when they are finished.

Step 3. Discuss together—Get the class's attention and ask:

- What was the most challenging part of the activity?
- Why does the video use two wait commands?

Put the worksheet on the projector. Discuss:

- What would each of the following sequences do? *Answers:*
  - *Sequence 1—The spikes would appear and disappear quickly.*
  - *Sequence 2—The spikes would disappear and never come back.*

- ○ *Sequence 3—The spikes would disappear quickly.*
- ○ *Sequence 4—The spikes would disappear and reappear three times and then stop.*
- ○ *Sequence 5—The spikes would appear for longer.*

- *Design Thinking—*Which of these would be fun for the player? Could any of these sequences lead to an unfair game for the player? *(Different answers are okay. Get students to explain why they think one way or another.)*

Step 4. Remixing—Have your students make a new level, and have them attempt one of the following challenges:

- **Different Animal** *(for students who struggled with the exercise)*—Get them to repeat it again, but make a different toothy animal.
- **Coding Challenge—**Code a second type of spike so it alternates appearing and disappearing with the first one.
- **Design Thinking Challenge—**Use the same code sequence, but apply it to blocks, hearts, gems, and enemies. Have your students think about whether it makes the game more fun or more frustrating.

Step 5. Real World Wrap Up—Show, hide, and wait are used in many places in computer science. Whenever a video is paused, a coder had to program the video player so it will wait until the play button is pressed. Even when typing, most cursors blink on and off—show, wait, hide, wait, and repeat—just like the spikes.

- **Challenge** students to look for times when a wait command is being used in an app or a game.