## MOTIVATION

We've discussed the importance of encoding and decoding, and explored these ideas by representing LEGO tower data using several codes, including binary.

Now, we switch to encoding a different kind of structure: a stack of cups. We will apply this new encoding system to learn about a new programming archetype: *functions*, or sets of repeated instructions.

## OBJECTIVES

Students will be able to:

* + Confidently describe the encoding and decoding process
	+ Understand functions and how they can clarify code

## RESOURCES

* + Lesson Plan ([Source](http://www.muddx.com/c4x/HMC/MyCS/asset/Cups_Lesson_Plan.docx)/[PDF](http://www.muddx.com/c4x/HMC/MyCS/asset/Cups_Lesson_Plan__1_.pdf) )
	+ Cup Methods Powerpoint ([Source](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_Robotics_1.pptx)/[PDF](http://www.muddx.com/c4x/HMC/MyCS/asset/Cupstack_Coding_Main.pdf))
	+ Video: Cup-Stacking Algorithms ([Source](http://www.cs.hmc.edu/~zdodds/MyCS/Cup%20Stacking%20Algorithms%20lq.mp4))
	+ Activity: Cup-Stacking Algorithms Worksheet ([Source](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_coding_worksheet_Part_1.pptx)/[PDF](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_coding_worksheet_Part_1.pdf))
	+ Video: Functions with Cups ([Source](http://www.cs.hmc.edu/~zdodds/MyCS/Functions%20with%20Cups%20lq%20copy.mp4))
	+ Activity: Functions with Cups Worksheet ([Source](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_Coding_Worksheet_Part_2.pptx)/[PDF](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_Coding_Worksheet_Part_2.pdf))

## OBJECTIVES

Today students will learn a new encoding system. We will use a language for stacking cups, and then we'll practice encoding and decoding these stacks using this new system.

While this language may feel a little clumsy at first, in the next section we'll use some new tricks to made our code easier to write and read.

## RESOURCES

* Cup Methods Powerpoint ([Source](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_Robotics_1.pptx)/[PDF](http://www.muddx.com/c4x/HMC/MyCS/asset/Cupstack_Coding_Main.pdf))
* Video: Cup-Stacking Algorithms ([Source](http://www.cs.hmc.edu/~zdodds/MyCS/Cup%20Stacking%20Algorithms%20lq.mp4))
* Activity: Cup-Stacking Algorithms Worksheet ([Source](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_coding_worksheet_Part_1.pptx)/[PDF](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_coding_worksheet_Part_1.pdf))

## VIDEO: CUP STACKING ALGORITHMS

## OBJECTIVES

In the last lesson, students learned a language for communicating information about cup stacks. In this section, we'll refine that language, and learn about how we can use functions to make our codes less repetitive.

## RESOURCES

* Cup Methods Powerpoint ([Source](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_Robotics_1.pptx)/[PDF](http://www.muddx.com/c4x/HMC/MyCS/asset/Cupstack_Coding_Main.pdf))
* Video: Functions with Cups ([Source](http://www.cs.hmc.edu/~zdodds/MyCS/Functions%20with%20Cups%20lq%20copy.mp4))
* Activity: Functions with Cups Worksheet ([Source](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_Coding_Worksheet_Part_2.pptx)/[PDF](http://www.muddx.com/c4x/HMC/MyCS/asset/Cup_Coding_Worksheet_Part_2.pdf))

## FUNCTIONS WITH CUPS

## VIDEO: WRAP-UP DISCUSSION

1. What were some challenges with the cup-stacking language? What problems arose, even after using functions?
2. Manufacturing often takes advantage of repeated processes in production. Take a look at the following GIFs of assembly line machinery in action:

|  |  |
| --- | --- |
| 1.
 | 1.
 |
| 1.
 | 1.
 |
| 1.
 | 1.
 |

1. How do you think functions are used in this type of factory work? Can you think of repeated steps that might be involved in making some your favorite things?
2. What are some repeated patterns in some of your personal routines? Can you think of functions for those patterns?

## DISCUSSION: ASSEMBLY LINES

Assembly lines are a manufacturing method which add small parts together in order to create a larger final product. Originally, people worked in assembly lines putting together products. However, modern assembly lines use machines.

**Are these machines computers?**



Human engineers create a set of instructions for these machines, which carry out the instructions to make the product. Machines on assembly lines need to know what *parts* they're working with and which *position* and *orientation* to put them in.

**How can we give a set of instructions to an assembly line?**

Should the instructions be given as a narrative or as a series of steps?

What kind of *encoding* should these instructions have?

Ultimately, machines "make decisions" based on the presence or absence of an electric signal, which can be represented with the binary digits 1 and 0. All instructions will eventually be translated into binary signals for the machine to carry out.