



Encoding & Decoding with Cups

Instructional Days: 1-2 days

Topic Description: Students will practice their encoding and decoding techniques with a different coding language than before. They will then learn about functions and the importance of functions in coding.

Objectives:

The student will:

- Review the concepts of encoding and decoding from the previous lesson
- Learn to use a new encoding/decoding system specialized to stacks of cups
- Learn about functions and practice implementing sample functions as well as some functions of their own

Outline of the Lesson and Teaching Strategies:

- Introduce the cup-stacking code
 - Present the cup-stacking language to students after watching the first video (Cup-Stacking Algorithms) and/or reviewing the alternative powerpoint (Cup Methods Powerpoint). Alternatively, review these materials with students.
 - Have students complete the first companion worksheet (Cup-Stacking Algorithms Worksheet)
- Expand the cup-stacking code by implementing functions
 - Present functions to students after watching the second video (Functions with Cups) and/or reviewing the alternative powerpoint (Cup Methods Powerpoint). Alternatively, review these materials with students.
 - Have students complete the second companion worksheet (Functions with Cups Worksheet)

Lesson Plan: Encoding & Decoding with Cups

- Wrap up the day: regroup the class and consider the following discussion questions.
 - What were some challenges with the cup-stacking language? What problems arose, even after using functions?
 - Given a set of assembly line GIFs, How do you think functions (finding repeated patterns in a process) play into this type of factory work? How might you design an assembly line machine that stacks cups?
 - What are some repeated patterns in some of your personal routines? Can you define functions for these patterns?

Resources :

- All the resources including the worksheets are available on the Encoding & Decoding with Cups Overview page of the muddX website (<http://goo.gl/nKAK6u>)