ECT Lesson Plan: Water Water Everywhere!

Lesson plan at a glance...

Core subject(s)	Computer Science
Subject area(s)	Algorithms and Complexity
Suggested age	11 to 14 years old
Prerequisites	*None*
Time	**Preparation:** 22 to 37 minutes **Instruction:** 45 to 55 minutes
Standards	**Core Subject:** CCSS.MATH, NGSS **CS:** CSTA, UK, Australia

In this lesson plan...

- **Lesson Overview**
- **Materials and Equipment**
- **Preparation Tasks**
- **The Lesson**
- **Learning Objectives and Standards**
- **Additional Information and Resources**

Lesson Overview

In this lesson, students are presented with the difficult problem of measuring a volume of water using containers of the wrong measurement size. Students will **decompose** a complex problem into discrete steps, **design an algorithm** for solving the problem, and evaluate solution efficiencies and optimization in a **simulation**.

Materials and Equipment

- □ For the teacher:
    - ○ *Required:* Board and marker/chalk
    - ○ *Recommended:* Presentation set-up
        - ■ Internet-connected computer
            - ● Chrome browser (https://www.google.com/chrome/browser/desktop) recommended
        - ■ Projector and projection screen or other flat projection surface
        - ■ External speakers for audio playback
- □ For the student:
    - ○ *Required:* Paper and pencil
    - ○ *Recommended:* Internet-connected computers (one (1) computer per student recommended)
    - ○ *Recommended:* Flasks or containers with sand (or something pourable that isn't water)

Preparation Tasks

	Confirm that your computer is on and logged-in, if used	1 to 3 minutes
	Confirm that your projector is turned on and is projecting properly, if used	1 to 4 minutes
	Confirm that all students' computers are turned on and logged-in	15 to 25 minutes
	Draw picture or project picture of the flasks	5 minutes

The Lesson

|	
Warm-up Activity: The challenge	15 minutes
Activity 1: Pour the water	20 minutes
Wrap-up Activity: Assessment	10 to 20 minutes

Warm-up Activity: The challenge (15 minutes)

Activity Overview: In this activity, students will be faced with the <u>complex</u> problem of measuring 5 mL of water with only a 4 mL container and a 7 mL container. Students must use <u>decomposition</u> to break the problem into smaller problems and then to <u>design an algorithm</u> to solve it.

Activity:

Ask students to solve the following problem:

You've run into a predicament. You want to do a science experiment. Given two containers, one that can hold 4 millilitres and one that can hold 7 millilitres, how can you get exactly one of these containers to hold exactly 5 millilitres?
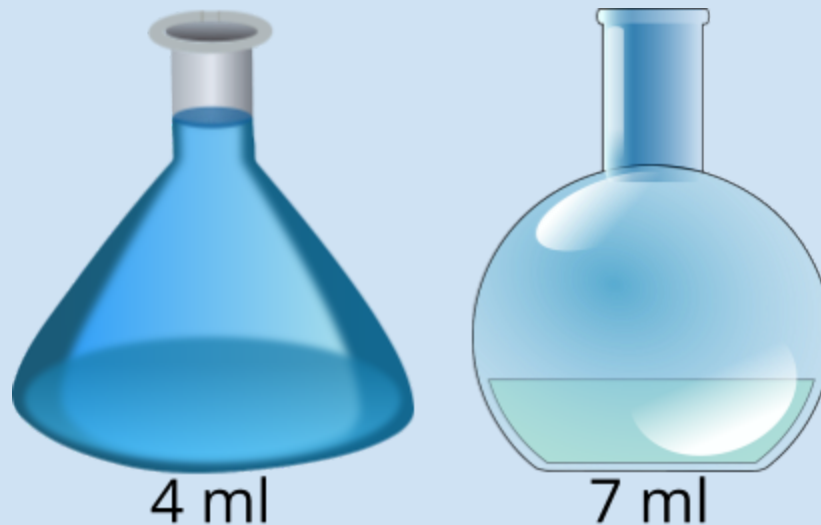
Notes to the teacher:

Tell the students the containers are blacked out and oddly shaped, so you can't tell by weight or watermark how much is in the container - you can only accurately tell if it is empty or if it is full. You can't look at the container to ascertain any other quantities.

You may notice that some students need help to get started. Stop their work and ask, "What do you know about the final desired volume?" "How can you express that value other than '5'?" "How do you think you can break this big problem into smaller problems?"

Assessment:

Ask students how they came up with the process for determining the correct solution. After the discussion, continue to Activity 1 in which you will suggest decomposition as a starting point.

Activity 1: Pour the water (20 minutes)

Activity Overview: In this activity, students will use <u>decomposition</u> to break down the problem into discrete steps and use the set of steps outlined by the instructor to design a method, or <u>algorithm</u>, to measure 5 mL of water. Finally, they will execute a <u>simulation</u> to test their algorithm for <u>optimization</u> and efficiency.

Activity:
Encourage students to describe the solution in discrete steps and to record the amount of water in each container (A and B) contains at each step.

We can do this by first noting that at any point there are only three actions we can do: completely fill a container, completely empty a container, or move the contents of one container to another (for this, we can use the shorthand A -> B to mean pouring the contents of A into B).

So an instruction to fill up the first container (A), pour it into the second (B) and then empty container B would look like:

Instruction	Quantity in A (max 4)	Quantity in B (max 7)
START	0	0
Fill A	4	0
A -> B	0	4
Empty B	0	0

Once students are comfortable with the notation, allow them to create their own table to describe the solution by interacting collaboratively in small groups.

An example solution might look like:

Instruction	Quantity in A (max 4)	Quantity in B (max 7)
START	0	0
Fill B	0	7
B -> A	4	3
Empty A	0	3
B -> A	3	0
Fill B	3	7
B -> A	4	6
Empty A	0	6
B -> A	4	2
Empty A	0	2
B -> A	2	0
Fill B	2	7
B -> A	4	<u>5</u>

If students find a valid solution, encourage them to find other solutions to the same problem. Ask them to count the number of 'instructions' used by their solution (for example, the number of instructions for the above would be 12).

Explain that the number of instructions used can be taken as a measure of the **optimization** or **efficiency** of their solution - the lower the number, the more efficient the solution because it takes fewer operations to achieve the goal.

A more efficient solution (because it only uses 9 instructions) would be:

Instruction	Quantity in A (max 4)	Quantity in B (max 7)
START	0	0
Fill A	4	0
A -> B	0	4
Fill A	4	4
A -> B	1	7
Empty B	1	0
A -> B	0	1
Fill A	4	1
A -> B	0	**5**

Assessment:
Assess students on their ability to reach the correct answer. Assess higher-level students on finding the most efficient/optimal method (the fewest number of steps).

Wrap-up Activity: Assessment (10 to 20 minutes)

Activity Overview: In this activity, students analyze their method for measuring 5 mL to determine: a) how they figured out how to start - **decomposition**, b) how they worked out the solution - **algorithm** - to solve the problem, and c) if their method is **efficient** based upon testing a **simulation**. Finally, they solve a similar problem and their solution is assessed.

Activity:
Give students the task of optimizing the algorithm in Activity 1 (if not done already).

Assign a similar problem with different sized flasks of water and a different goal. If they have learned a process for solving this type of problem, they can apply the process to a different scenario.

Q1: How do you know how to start?

Q2: How do you know if you should use the method that pours A into B or the method that pours B into A?

Assessment:
Instructors should assess student answers to a new problem or assess optimization strategies.

A1: You know that you do not have a 5 mL container, so the only way you can get 5 is by adding two smaller components like 4 and 1 or 2 and 3, for instance. The only way you can get 1 mL, 2 mL, or 3 mL of water is to mix and

pour water from one container into another.

A2: Generally speaking pouring the smaller container (4) into the larger container (7) gets you to the amount of 1 mL much faster which is helpful if you want to add it to 4 mL in order to get 5 mL, the goal.

Learning Objectives and Standards

Learning Objectives	Standards
LO1: Students will be able to use decomposition to break the water measurement problem into smaller more manageable problems.	*Common Core* CCSS MATH.PRACTICE.MP7: Look for and make use of structure. NGSS HS-ETS1-2: Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering. *Computer Science* CSTA L2.CT.12: Use abstraction to decompose a problem into subproblems.
LO2: Students will be able to develop an algorithm to solve similar measurement problems.	*Common Core* CCSS MATH.PRACTICE.MP2: Reason abstractly and quantitatively. CCSS MATH.PRACTICE.MP1: Make sense of problems and persevere in solving them. *Computer Science* AUSTRALIA 6.4 (Creating digital solutions by: defining): Define problems in terms of data and functional requirements, and identify features similar to previously solved problems. CSTA L2.CT.1: Use the basic steps in algorithmic problem-solving to design solutions (e.g., problem statement and exploration, examination of sample instances, design, implementing a solution, testing and evaluation). UK 3.2: Understand several key algorithms that reflect computational thinking [for example, ones for sorting and searching]; use logical reasoning to compare the utility of alternative algorithms for the same problem.
LO3: Students will be able to record the data from the simulation for efficiency assessment.	*Common Core* CCSS MATH.PRACTICE.MP5:Use appropriate tools strategically. *Computer Science* AUSTRALIA 6.4 (Creating digital solutions by: defining) CSTA L2.CT.1 UK 3.2
LO4: Students will be able to analyze and evaluate the algorithm to determine its efficiency.	*Common Core* NGSS HS-ETS1-3: Evaluate a solution to a complex real-world problem based on prioritized criteria and trade-offs that account for a range of constraints, including cost, safety, reliability, and aesthetics as well as possible social, cultural, and environmental impacts.

	NGSS MS-ETS1-2: Evaluate competing design solutions using a systematic process to determine how well they meet the criteria and constraints of the problem.
	CCSS MATH.PRACTICE.MP2: Reason abstractly and quantitatively.
	Computer Science AUSTRALIA 6.4 (Creating digital solutions by: defining)
	CSTA L2.CT.1
	UK 3.2

Additional Information and Resources

Lesson Vocabulary

Term	Definition	For Additional Information
Optimization	The process of modifying a software system to make some aspect of it work more efficiently or use fewer resources.	http://en.wikipedia.org/wiki/Program_optimization] http://en.wikipedia.org/wiki/Mathematical_optimization
Complexity	Characterization of something with many parts where those parts interact with each other in multiple ways, or the degree to which something can be characterized in this way.	http://en.wikipedia.org/wiki/Complexity

Computational Thinking Concepts

Concept	Definition
Decomposition	Breaking down tasks into smaller, manageable parts
Algorithm Design	Creating an ordered series of instructions for solving similar problems
Simulation	Representing or modeling a process

Administrative Details

Contact info	For more info about Exploring Computational Thinking (ECT), visit the ECT website (g.co/exploringCT)
Credits	Developed by the Exploring Computational Thinking team at Google and reviewed by K-12 educators from around the world.
Last updated on	07/02/2015
Copyright info	Except as otherwise noted, the content of this document is licensed under the Creative Commons Attribution 4.0 International License, and code samples are licensed under the Apache 2.0 License.